



HAL
open science

Mécanismes d'accès multiple dans les réseaux sans fil large bande

Alexandre Ragaleux

► **To cite this version:**

Alexandre Ragaleux. Mécanismes d'accès multiple dans les réseaux sans fil large bande. Réseaux et télécommunications [cs.NI]. Université Pierre et Marie Curie - Paris VI, 2016. Français. NNT : 2016PA066407 . tel-01492844

HAL Id: tel-01492844

<https://theses.hal.science/tel-01492844>

Submitted on 20 Mar 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.

**THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PIERRE ET MARIE CURIE**

Spécialité : Informatique

École doctorale : « EDITE de Paris »

réalisée

au Laboratoire Informatique de Paris 6

présentée par

Alexandre RAGALEUX

pour obtenir le grade de :

DOCTEUR DE L'UNIVERSITÉ PIERRE ET MARIE CURIE

Sujet de la thèse :

**Mécanismes D'Accès Multiple dans les Réseaux Sans Fil
Large Bande**

soutenue le 22/09/2016

devant le jury composé de :

M^{me}	Véronique Vèque	Rapporteur
M.	Nadjib Achir	Rapporteur
M^{me}	Maria Potop-Butucaru	Examineur
M.	Steven Martin	Examineur
M.	Sébastien Baey	Encadrant
M^{me}	Anne Fladenmuller	Directeur de thèse

Remerciements

Je dois bien l'avouer, une thèse est une aventure très riche. Pendant ces trois années, j'ai eu le plaisir de vivre de nombreux moments inoubliables et de traverser une succession d'émotions, parfois négatives mais bien plus souvent positives. Avant tout, cette thèse m'a permis de resserrer les liens avec mon entourage et m'a donné la possibilité de rencontrer de nouvelles personnes. Cette première section est donc l'occasion pour moi d'exprimer mes remerciements les plus sincères à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail. En plus, il paraît que la section « Remerciements » est la plus lue d'une thèse !

Je tiens à remercier tout particulièrement Sébastien pour m'avoir fait confiance, pour ses conseils précieux et ses relectures minutieuses. Il fut un pilier indispensable à la concrétisation de mes idées et à la réussite de cette thèse.

Mes remerciements les plus sincères à Anne pour sa gentillesse, et pour s'être assurée que je puisse travailler sur ma thèse dans les meilleures conditions possibles. Merci énormément pour m'avoir donné l'opportunité de voyager.

Je tiens également à remercier Mme Véronique Vèque, M. Nadjib Achir, M. Steven Martin et Maria qui m'ont fait l'honneur d'accepter de faire partie de mon jury de thèse.

J'adresse toute ma gratitude à l'équipe NPA. J'ai passé trois ans dans ce laboratoire dont l'ambiance est propice aux échanges, aux débats et à la création. Merci à Marcelo pour avoir su conserver et maintenir une telle cohésion. Merci à Sébastien Tixeuil pour m'avoir permis de débiter par un stage et ainsi de commencer cette formidable aventure. Merci à Prométhée pour sa personnalité atypique rafraîchissante et ses histoires aussi insolites qu'hilarantes. Enfin, merci à Kim, Olivier ainsi que toute l'équipe pédagogique de la spécialité Réseaux, qui m'ont beaucoup apporté en Master.

Un grand merci aux doctorants de l'équipe pour leurs bonnes humeurs et l'intérêt qu'ils ont porté à mes recherches. Ce fut un honneur de partager ces moments avec eux. Un merci tout particulier à Benjamin avec qui j'ai eu le plaisir d'effectuer toutes mes études universitaires. Cette thèse marque ainsi la fin de notre longue collaboration (8 ans !), mais je sais que notre amitié perdurera.

Je remercie Björn pour m'avoir accueilli trois mois dans son laboratoire, en Suède, ainsi que toute son équipe pour m'avoir intégré très chaleureusement, ce qui a été nécessaire pour affronter la température extérieure...

Merci à mes parents pour leur soutien inestimable. Merci à ma mère pour sa disponibilité, sa lucidité et son ouverture d'esprit. Merci à mon père, pour m'avoir poussé à aller toujours plus loin et qui s'est toujours montré disponible pour m'écouter autour

d'une bonne table parisienne.

Bien évidemment, merci à Charline pour m'avoir supporté pendant ces trois ans. Cohabiter avec un doctorant n'est pas chose aisée, et pourtant elle a été là pour moi. Je ne pourrai quantifier rationnellement le plaisir que j'ai eu à la retrouver chaque soir. Sa présence à mes côtés fut essentielle à bien des égards.

Table des matières

1	Introduction	1
1.1	Contexte	1
1.2	Résumé des contributions	2
1.3	Description du système	3
1.3.1	Modèle et hypothèses de travail	4
1.3.2	La contrainte du <i>Single MCS</i>	5
1.3.3	Non-linéarité du calcul du débit	7
1.3.4	Spécificités relatives au sens de transmission	7
1.3.5	Objectif	8
1.4	Organisation de la thèse	8
2	État de l’art	11
2.1	Un mot sur la théorie de la complexité	11
2.2	Stratégies d’allocation de ressources issues des réseaux filaires	13
2.2.1	Stratégies d’allocation de ressources opportunistes	14
2.3	Allocation de ressources dans les réseaux LTE	17
2.3.1	Sens descendant	18
2.3.2	Sens montant	20
2.4	Conclusion	23
3	Allocation de ressources dans le sens descendant	25
3.1	Formalisation du problème	25
3.1.1	Le problème LTE DL FDPS	26
3.2	Sous-problème et résultats fondamentaux	28
3.3	Algorithme d’approximation	31
3.3.1	Première étape : vers la résolution de P2	31
3.4	Deuxième étape : prise en compte du single MCS	33
3.5	Troisième étape : extension	35
3.5.1	Un mot sur la complexité	36
3.6	Évaluation des performances	36
3.6.1	Modélisation de la couche physique	37
3.7	Conclusion	44

4	Allocation de ressources avec Trafics Multimédias	45
4.1	Vers la quantification de la Qualité de Service	46
4.1.1	Objectif	48
4.2	AGSS : Schéma d’ordonnancement générique et adaptatif	50
4.2.1	Préliminaires : un point sur les stratégies d’ordonnancement	50
4.2.2	Présentation d’AGSS	51
4.2.3	Un mot sur la complexité	53
4.3	Stratégie d’ordonnancement	54
4.4	Évaluation des performances	57
4.4.1	Algorithme classique vs AGSS	58
4.4.2	Comparaison des stratégies d’ordonnancement	59
4.5	Conclusion	63
5	Allocation de ressources dans le sens montant	65
5.1	Formulation du problème	66
5.1.1	Problème du LTE UL TFDPS.	67
5.1.2	Expression des stratégies d’ordonnancement et sous-problèmes	67
5.1.3	Difficulté du problème	68
5.2	APASS : algorithme d’allocation de ressources dans le sens montant	69
5.2.1	Allocation de ressources pour une relaxation du problème	69
5.2.2	Sélection des utilisateurs	73
5.3	Améliorations et finalisation de l’allocation	74
5.3.1	Préliminaires	74
5.3.2	Dernière étape de l’allocation de ressources	75
5.3.3	Un mot sur la complexité	77
5.4	Évaluation des performances	78
5.4.1	Comparaison des stratégies d’ordonnancement	79
5.4.2	Comparaison des algorithmes	80
5.5	Conclusion	81
6	Conclusion	83
6.1	Synthèse	83
6.2	Perspectives.	84
A	Description du RAN de LTE	87
A.1	L’architecture des protocoles RAN	87
A.2	Structure de la trame LTE	88
A.3	Mode de duplexage	89
A.4	Détermination de la taille d’un <i>transport block</i>	90
A.5	Estimation de la qualité du canal	91
A.5.1	Allocation dans le sens descendant	92
A.5.2	Allocation dans le sens montant	92
A.5.3	Taille des Transport Blocks	93

TABLE DES MATIÈRES

v

B Publications	95
B.1 Articles publiés	95
B.2 Articles en cours de soumission	95
B.3 Rapports techniques	95
Bibliographie	97

Chapitre 1

Introduction

1.1 Contexte

Afin de toujours faire évoluer les débits ainsi que l'expérience des utilisateurs, les chercheurs et ingénieurs améliorent constamment l'architecture des réseaux mobiles. Les réseaux Long Term Evolution (LTE), commercialisés comme de la 4G, ont été largement déployés à travers le monde [1]. Son successeur, la 4G+ ou "la vraie 4G" (LTE-A, Releases 10-11), commence à être déployée dans les grandes villes [2]. Cette technologie offre le support de la voix et triple le débit théorique. La dernière génération, la pré-5G¹ (LTE-B, Release 12-13), a été conçue pour ouvrir la route de la 5G en offrant des débits encore améliorés afin de supporter les nouvelles utilisations des réseaux cellulaires (transfert massif de données).

Parallèlement, de nouveaux types d'applications mobiles ont émergé. Celles-ci bénéficient de l'augmentation des débits en utilisant massivement la bande passante et nécessitent une gestion efficace de la *Qualité de Service*. Pour s'adapter à ces nouvelles utilisations des réseaux cellulaires, les procédures d'allocation de ressources doivent évoluer afin de répondre aux attentes des applications.

Les réseaux LTE reposent sur une méthode d'accès au canal très performante appelée OFDMA (*Orthogonal Frequency Division Multiple Access*) qui permet de tirer pleinement parti de la diversité temporelle et fréquentielle du canal. Cependant, même une méthode d'accès efficace utilisée conjointement avec un schéma de modulation et de codage (ou MCS pour *Modulation and Coding Scheme*) adapté n'est généralement pas suffisante pour aboutir à des performances convenables. En effet, la façon dont les ressources sont allouées a une influence très importante sur les performances obtenues [3]. Par conséquent, l'allocation de ressources est devenue un domaine de recherche fondamental.

Dans cette thèse, nous considérons une station de base qui communique avec plusieurs équipements utilisateurs (Fig. 1.1 pour le cas du sens descendant). La cellule considérée peut être indifféremment une *small cell* ou une *macro cell*. Les sens descendant et montant sont distingués. Dans le sens descendant, les paquets sont originaires de flux de données provenant du réseau coeur et sont mis en mémoire dans des *buffers*.

1. Comme cette technologie n'est pas encore disponible, le nom commercial de LTE-B n'est pas définitif.

Nous considérons que chaque buffer est associé à un seul et unique flux. L'ordonnancier a pour rôle de partager les ressources radios entre les flux afin que les paquets puissent être transmis vers leur destinataire. Le sens montant est organisé de façon analogue, i.e. les flux sont originaires des applications mobiles et les buffers se situent dans les équipements utilisateurs.

Une particularité des réseaux sans fil est qu'ils sont soumis à de nombreuses déficiences du canal (atténuation du signal, évanouissement sélectif en fréquence, multi-trajets ...). Dans le système LTE, les ressources radios sont à la fois divisées dans le domaine temporel et dans le domaine fréquentiel. L'exploitation des variations temporelles et fréquentielles de la qualité des ressources est essentielle afin de fournir une connexion haut débit aux utilisateurs. En effet, à un instant donné, une ressource peut être de bonne qualité pour un utilisateur (i.e. l'utilisateur bénéficie d'un haut SNR (*Signal to Noise Ratio*) sur cette ressource), et de mauvaise qualité pour un autre. Dans ce contexte, les ordonnanceurs opportunistes sont apparus [4, 5, 6, 7, 8]. Ces derniers prennent en compte la qualité des liens radio dans leur procédure d'allocation de ressources. Cependant, une large majorité de ces travaux prend pour hypothèse que les buffers sont *toujours pleins* (*infinitely backlogged buffers model*). Or, la variabilité des trafics multimédias en termes de débit instantané peut être très importante. Ces types de caractéristiques doivent être pris en compte à courte échelle de temps afin d'assurer aux applications un niveau de qualité de service satisfaisant.

Nous avons étudié de très près la norme 3GPP LTE afin d'identifier les contraintes qui s'appliquent au problème de l'allocation de ressources. La plus importante d'entre elles, que nous avons appelée la contrainte du *Single MCS*, est peu prise en compte dans la littérature mais a des conséquences notables sur la manière d'allouer les ressources. En effet, la norme autorise uniquement l'utilisation d'un seul MCS par utilisateur sur l'ensemble des ressources qui lui sont allouées. Cette contrainte réduit considérablement l'exploitation de la diversité fréquentielle, et donc les performances du système. De plus, cette contrainte rend obsolètes les procédures d'allocation de ressources qui ne la prennent pas en compte. Notons que les solutions proposées dans cette thèse visent les toutes dernières versions de LTE, y compris LTE-B (*LTE Release 13*) qui a été publié au début de l'année 2016.

1.2 Résumé des contributions

Tout d'abord, nous avons modélisé un système LTE en utilisant directement la norme comme support d'information. De plus, nous proposons une formulation générique du problème de l'allocation de ressources. Ainsi, nos solutions sont également génériques et couvrent donc un grand nombre d'objectifs (maximisation de la capacité, équité entre les utilisateurs, etc...).

Nous avons fait le choix d'approcher les différents problèmes à la fois de manière théorique et pratique. Pour cela, pour chaque problème considéré, la méthodologie suivante a été appliquée :

- Étude de la complexité du problème et proposition d'une solution avec garantie par rapport à la solution optimale. Afin d'analyser le problème, l'hypothèse des

*utilisateurs gloutons*² a été considérée.

- Nous relâchons ensuite l’hypothèse des utilisateurs gloutons et nous considérons des trafics multimédia réalistes. Ainsi, nous proposons une solution rapide et efficace dans ce contexte multimédia (en se basant sur l’analyse théorique).

Toutes nos solutions sont évaluées à l’aide d’un simulateur à événements discrets. Pour cela, nous avons implémenté notre propre module LTE sur le logiciel *Riverbed Modeler* [9] (précédemment connu sous le nom d’*Opnet Model*) en nous basant sur l’étude de la norme (Annexe A).

Comme déjà indiqué, nous avons différencié le sens descendant (Chapitre 3) du sens montant (Chapitres 4 et 5) car les contraintes qui s’appliquent sont très différentes. Par conséquent, la thèse est composée de deux grandes parties. Les contributions de chacune de ces parties sont résumées ci-après.

Sens descendant. Nous montrons que le problème de l’allocation de ressources dans les réseaux LTE fait partie de la classe des problèmes difficiles et ne peut donc pas être résolu de façon optimale avec un algorithme efficace. Nous proposons ensuite un *algorithme d’approximation* conjointement avec son analyse de performances. Nous analysons par la suite une métrique de qualité de service, le Packet Delay Outage Ratio (PDOR), qui permet de quantifier le niveau de satisfaction d’un utilisateur. Nous utilisons cette métrique pour construire AGSS (*Adaptive and Generic Scheduling Scheme*), une heuristique d’allocation de ressources adaptée à la prise en charge des flux multimédias. Une analyse de performances est également réalisée.

Sens montant. De même que pour le sens descendant, nous donnons des éléments expliquant la difficulté du problème considéré³. Nous proposons ensuite un algorithme d’approximation qui sera suivi d’une analyse de performances. Dans un second temps, nous considérons des trafics multimédias. Nous définissons une métrique permettant de quantifier la qualité d’une allocation et nous l’utilisons pour construire APASS (*Adaptive and Potential Aware Scheduling Scheme*), une heuristique adaptée aux flux multimédias. Une analyse de performances est alors conduite.

1.3 Description du système

Cette section a pour but de donner une vue d’ensemble de l’accès à un réseau LTE, d’énoncer nos hypothèses de travail et de décrire brièvement le modèle qui sera utilisé. De plus, nous profitons de cette section pour expliquer plus en détail le problème de l’allocation de ressources dans un contexte LTE.

Afin de faciliter la lecture, la description du RAN (*Radio Access Network*) de LTE a été placée en Annexe A. Cette annexe présente précisément la structure d’une trame LTE ainsi que l’architecture des protocoles utilisés. Ces éléments justifient le modèle que nous utilisons ainsi que les paramètres des simulations.

Pour rappel, dans l’ensemble de cette thèse, nous considérons une seule cellule. Par conséquent, une seule station de base (appelée *eNodeB* dans LTE) communique

2. I.e. nous considérons que les buffers des utilisateurs sont toujours pleins (*infinitely backlogged buffers model*).

3. La preuve sur la difficulté du problème étant déjà établie dans [10].

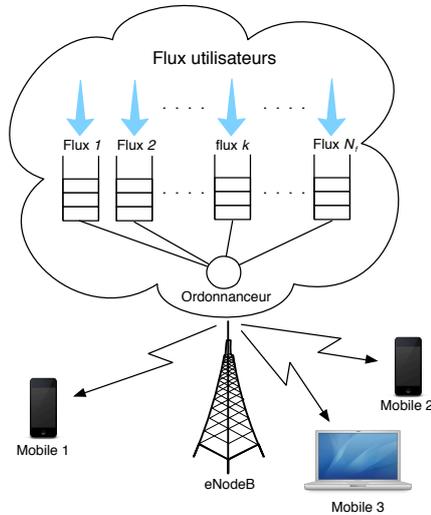


Fig. 1.1 – Contexte de travail.

avec plusieurs mobiles (équipements utilisateurs) qui sont actifs au sein de la zone de couverture de la cellule.

Dans le sens descendant, les paquets issus du cœur du réseau sont mis en mémoire dans des *buffers* au niveau de la station de base. Cette dernière est chargée d’acheminer les paquets vers les équipements utilisateurs. Chaque buffer, ou buffer virtuel, est associé à un flux utilisateur. Dans le sens montant, les paquets sont issus des flux générés par les mobiles. De même que pour le sens descendant, les paquets sont mis en mémoire dans des buffers avant d’être envoyés au eNodeB en utilisant l’interface radio.

C’est au niveau du eNodeB que l’on trouve l’**ordonnanceur** qui est chargé de faire le lien entre les buffers et l’interface radio. L’ordonnanceur est responsable de l’**allocation des ressources** radios. En d’autres termes, un algorithme d’allocation de ressources doit être implémenté au niveau de la station de base dont l’objectif est de partager les ressources entre les utilisateurs. Ces algorithmes doivent se baser sur la qualité des ressources (SNR) de chaque utilisateur afin d’aboutir à une allocation *opportuniste*.

1.3.1 Modèle et hypothèses de travail

Nous présentons dans cette section le modèle qui sera utilisé dans l’ensemble de la thèse. Ce modèle est directement dérivé de l’Annexe A. Le Tableau 1.1 fournit quelques notations essentielles à cette thèse.

Notations. Comme indiqué précédemment, le processus d’allocation de ressources est périodique de période égale à la durée d’un *TTI* (*Transmission Time Interval*) de valeur 1 ms [11]. Nous appelons SU (*Scheduling Unit*) la plus petite unité de ressource qui peut être allouée à un utilisateur. De manière générale, nous utiliserons les termes “SU” et “ressource” indifféremment. L’ensemble de tous les SUs est noté

Symbole	Signification
SU	<i>Scheduling Unit</i> , la plus petite unité de ressource allouable par l'ordonnanceur
U	Nombre d'utilisateurs
N	Nombre de SUs
M	Nombre de MCSs
\mathcal{U}	Ensemble des utilisateurs
\mathcal{N}	Ensemble des SUs
\mathcal{M}	Ensemble des MCSs
\mathcal{N}_u	Sous-ensemble de SUs alloués à l'utilisateur u
$m_{u,k}$	MCS le plus efficace qui peut être utilisé sur le SU k par l'utilisateur $u \in \mathcal{U}$
m_u	<i>Single MCS</i> de l'utilisateur $u \in \mathcal{U}$
$R(\mathcal{N}_u, m_u)$	Débit obtenu avec l'ensemble \mathcal{N}_u et le MCS m_u

Tableau 1.1 – Principales notations.

$\mathcal{N} = \{k_1, k_2, \dots, k_N\}$, où N est le nombre de SUs du système. De même, nous notons $\mathcal{U} = \{u_1, u_2, \dots, u_U\}$ l'ensemble des utilisateurs avec U le nombre d'utilisateurs actifs dans la cellule. Ainsi, à chaque TTI, l'ordonnanceur va allouer à chaque utilisateur u un sous-ensemble de \mathcal{N} noté \mathcal{N}_u (sous la contrainte que tous les ensembles \mathcal{N}_u soient disjoints). Enfin, nous notons $\mathcal{M} = \{m_1, m_2, \dots, m_M\}$ l'ensemble des MCS utilisables, et M le nombre total de MCS. Nous munissons l'ensemble \mathcal{M} d'un relation d'ordre " \leq " telle que $m_i \leq m_j$ ($i, j \in \{1, \dots, M\}$) si et seulement si m_j est spectralement plus efficace que m_i (ou $m_i = m_j$).

Le MCS le plus efficace qui peut être utilisé par la station de base sur le SU $k \in \mathcal{N}$ pour l'utilisateur u (sans enfreindre la contrainte sur le taux d'erreur binaire) est noté $m_{u,k}$. En effet, le taux d'erreur binaire ne doit pas dépasser un seuil fixé.

Pour le sens descendant, les valeurs des $\{m_{u,k}\}_{u,k}$ sont déterminées grâce aux CQIs contenus dans les *channel state reports* (voir Annexe A). Dans le sens montant, ces valeurs sont directement estimées au niveau de la station de base.

Instance du problème. En dehors des ensembles précédents, les données du problème peuvent être décrites par une matrice \mathcal{I} représentant les valeurs des MCS $m_{u,k}$ pour chaque couple (*utilisateur, SU*) :

$$\mathcal{I} = (m_{u,k})_{u \in \mathcal{U}, k \in \mathcal{N}} = \begin{pmatrix} m_{u_1, k_1} & \cdots & m_{u_1, k_N} \\ \vdots & \ddots & \vdots \\ m_{u_U, k_1} & \cdots & m_{u_U, k_N} \end{pmatrix}.$$

La Figure 1.2 illustre une instance du problème en représentant la qualité des liens radio pour chaque couple (*utilisateur, SU*), i.e. pour chaque couple $(u, k) \in \mathcal{U} \times \mathcal{N}$. Dans cette figure, le nombre indiqué dans chaque rectangle représente l'indice du meilleur MCS pour le couple (u, k) correspondant.

1.3.2 La contrainte du *Single MCS*

Comparé à un système OFDMA purement théorique, il n'est pas possible dans LTE d'utiliser une modulation différente pour chaque sous-porteuse et à chaque *time slot* (période d'un symbole OFDM). La granularité dans l'affectation des MCS est donc

Utilisateur	RB 1	RB 2	RB 3	RB 4	RB 5	RB 6	RB 7
Utilisateur 4	3	1	4	2	5	0	1
Utilisateur 3	5	0	4	2	2	5	2
Utilisateur 2	1	1	2	3	4	5	1
Utilisateur 1	5	0	5	0	3	2	1

Fig. 1.2 – Exemple d’instance de l’allocation de ressources. Pour chaque utilisateur, la qualité d’une ressource est représentée par un entier.

beaucoup moins fine. En réalité, la contrainte sur l’affectation des MCS est même encore plus restrictive.

Contrainte 1. Contrainte du Single MCS : *chaque utilisateur ne peut utiliser qu’un seul MCS sur tous les SUs qui lui sont alloués pour la totalité d’un TTI.*

Cette contrainte a des conséquences importantes sur l’allocation de ressources. À titre d’exemple, considérons un système classique dans lequel une ressource de bonne qualité et une ressource de mauvaise qualité sont allouées à un même utilisateur. Celui-ci peut théoriquement utiliser un MCS efficace sur la ressource de bonne qualité et un MCS peu efficace sur celle de moindre qualité. Cependant, si on considère la Contrainte 1, l’utilisateur ne peut mettre en œuvre qu’un seul et même MCS sur les deux ressources qui lui sont allouées. Par conséquent, la contrainte suivante permet de s’assurer que chaque utilisateur n’excède pas le taux d’erreur binaire sur chacune des ressources qui lui sont allouées.

Contrainte 2. Contrainte sur le taux d’erreur binaire.

Soit u un utilisateur, et \mathcal{N}_u l’ensemble des ressources qui lui sont allouées. Le Single MCS $m_u \in \mathcal{M}$ affecté à u doit être tel que $m_u \leq m_{u,k}$ pour tout $k \in \mathcal{N}_u$. Nous disons alors que m_u est supporté par u .

Afin de respecter la Contrainte 2, chaque utilisateur u va donc devoir utiliser le MCS le moins performant sur l’ensemble des SUs qui lui sont alloués. Ainsi, la Contrainte 2 peut également s’exprimer ainsi :

$$m_u = \min \{m_{u,k}, \text{ tel que le SU } k \text{ est alloué à l'utilisateur } u\},$$

avec m_u l’unique MCS qui sera mis en œuvre par l’utilisateur u . En réalité, l’opération de minimum effectuée ci-dessus n’est pas la seule technique possible pour déterminer une valeur de m_u . Plus précisément, la Contrainte 2 peut être exprimée de façon moins restrictive selon le codage utilisé dans le système. Cependant, il est important de constater que, quelle que soit la technique utilisée pour prendre en compte le taux d’erreur binaire avec la Contrainte 1, l’allocation d’un SU de moindre qualité à un utilisateur va nécessairement diminuer la quantité de données qui pourra être transmise sur tous ses autres SUs (i.e. son débit par SU est diminué). Bien que la contrainte du *Single MCS* ait une conséquence importante sur la manière d’allouer les ressources, cette difficulté n’a été que peu étudiée par la communauté [12, 13].

1.3.3 Non-linéarité du calcul du débit

Dans toutes les versions de LTE, le débit utile d'un utilisateur est donné par un tableau indexé par deux paramètres [14] (voir Annexe A) :

- l'ensemble des SUs alloués à l'utilisateur (\mathcal{N}_u pour l'utilisateur u),
- la qualité globale des ressources allouées à cet utilisateur (MCS m_u pour l'utilisateur u).

Par conséquent, nous notons $R_u(\mathcal{N}_u, m_u)$ le débit utile (et instantané) obtenu par un utilisateur en lui allouant le sous-ensemble $\mathcal{N}_u \subseteq \mathcal{N}$ et le MCS m_u .

Selon le tableau des débits, la fonction $R_u(\cdot, m)$ n'est pas totalement linéaire avec le nombre de SUs. Cette propriété spécifique à LTE rend le problème plus difficile car non-linéaire. Plus précisément, la contrainte du *single MCS* et le calcul non-linéaire des débits rendent la formulation du problème sous forme de programme linéaire en nombres entiers non-exploitable.

1.3.4 Spécificités relatives au sens de transmission

La contrainte du *Single MCS* est valable à la fois pour les sens descendant et montant. Cependant, d'autres contraintes/spécificités s'appliquent également selon le sens de transmission.

Un mot sur la nature des SUs. Tout d'abord, un SU dans le sens montant est différent d'un SU dans le sens descendant. En effet, il est communément admis que la plus petite unité de ressource qui peut être allouée par l'ordonnanceur est en réalité appelée un *Resource Block* (RB). Cependant, ce n'est pas tout à fait vrai dans le sens descendant comme nous l'expliquons ci-dessous.

Définition 1. *Allocation par groupe de ressources (sens descendant) : un SU correspond à un groupe de RBs adjacents dans le domaine fréquentiel.*

Le nombre de RBs qui constituent un SU peut varier de 2 à 8 selon la bande passante totale du système. Par exemple, pour une bande passante de 20 MHz, les groupes seront composés de 8 RBs. Un SU correspond donc à un groupe de RBs et non à un simple RB. Cette spécificité exclusive au sens descendant rend la granularité de l'allocation de ressource encore moins fine. Cela diminue donc davantage la diversité fréquentielle et affecte négativement les performances du système.

Dans le sens montant, un SU correspond bien à un RB. Nous avons choisi d'introduire le terme SU afin d'utiliser des conventions de nommage identiques et génériques quel que soit le sens de transmission.

Allocations contiguës. Dans le sens montant, la méthode d'accès utilisée est appelée SC-FDMA (*Single Carrier - Frequency Division Multiple Access*). Cette méthode d'accès dérivée de l'OFDMA permet d'obtenir un PAPR (*Peak to Average Power Ratio*) plus faible, et donc une consommation énergétique moins importante. Ceci est particulièrement important dans le sens montant où les mobiles ont une batterie de durée vie limitée. En contrepartie, cette technique d'accès entraîne une nouvelle contrainte sur l'allocation de ressources.

Contrainte 3. *Allocations contiguës (sens montant) : une allocation est valide si les SUs alloués à chaque utilisateur sont adjacents dans le domaine fréquentiel.*

Par conséquent, si l’ordonnanceur souhaite allouer les SUs d’indices i et j à un utilisateur ($i < j$), alors il est également contraint d’allouer tous les SUs d’indice l tel que $i < l < j$. En effet, il faut toujours que les SUs affectés à un utilisateur soient contigus. La Contrainte 3 combinée à la Contrainte 1 peut amener à un gaspillage important des ressources. Dans l’exemple précédent, si un seul des SUs situés entre le SU i et le SU j est de mauvaise qualité alors la contrainte du *Single MCS* impliquera une diminution de la quantité de données qui pourra être transmise sur l’ensemble des SUs alloués. L’allocation de ressources dans le sens montant de LTE est donc particulièrement délicate.

Limitation du nombre d’utilisateurs. Pour finir, la contrainte suivante s’applique uniquement dans le sens montant.

Contrainte 4. *Pour chaque TTI, le nombre d’utilisateurs qui peuvent être ordonnancés (i.e. qui peuvent recevoir des ressources) est borné.*

Cette contrainte réduit la diversité utilisateurs qui pouvait être exploitée en profitant d’un large panel d’utilisateurs afin d’associer les meilleures ressources aux meilleurs candidats. D’un point de vue pratique, un algorithme d’allocation de ressources devra souvent intégrer un processus de sélection des utilisateurs qui seront ordonnancés lors du prochain TTI.

1.3.5 Objectif

Le but de l’allocation de ressources est de maximiser une fonction objectif en partageant les ressources \mathcal{N} aux utilisateurs \mathcal{U} sous les contraintes de LTE.

Exemple d’objectif et de solution. Considérons un système avec 3 SUs $\mathcal{N} = \{k_1, k_2, k_3\}$ et deux utilisateurs $\mathcal{U} = \{u_1, u_2\}$. Supposons de plus que l’on souhaite maximiser la capacité du système.

Nous supposons également que l’ordonnanceur alloue les deux premières ressources à u_1 et la dernière ressource à u_2 (i.e. $\mathcal{N}_{u_1} = \{k_1, k_2\}$ et $\mathcal{N}_{u_2} = \{k_3\}$). Par conséquent, les *Single MCS* de chaque utilisateur sont choisis de cette façon : $m_{u_1} = \min(m_{u_1, k_1}, m_{u_1, k_2})$ et $m_{u_2} = m_{u_2, k_3}$. Le débit total du système de cette solution (possiblement sous-optimale) est donc égal à $R(\mathcal{N}_{u_1}, m_{u_1}) + R(\mathcal{N}_{u_2}, m_{u_2})$.

De manière générale, notre but est de pouvoir traiter de façon générique une large variété d’objectifs. Pour cela, nous reviendrons plus formellement sur l’expression des objectifs dans les chapitres suivants.

1.4 Organisation de la thèse

Cette thèse est structurée en six chapitres, suivant le fil conducteur explicité dans la Section 1.2. Le Chapitre 2 présente un état de l’art tandis que les chapitres 3, 4 et 5 décrivent nos travaux. Le Chapitre 6 conclut la thèse.

Le chapitre 2 est un état de l’art. Nous commençons par définir les outils et concepts utilisés dans la thèse. Nous présentons les principaux objectifs d’allocation de ressources ainsi que les stratégies d’allocation qui s’y rapportent. Nous terminons par introduire le problème du FDPS (*Frequency Division Packet Scheduling*) avant de présenter les principaux algorithmes FDPS conçus pour LTE.

Le chapitre 3 présente une étude sur l’allocation de ressources dans le sens descendant sous l’hypothèse des utilisateurs gloutons. Nous définissons formellement le problème “LTE DL FDPS” et nous montrons qu’il est NP-Difficile. De plus, nous montrons qu’une relaxation du problème est équivalente à un problème bien connu de la littérature. Nous utilisons ce résultat afin de construire un algorithme d’approximation calculant une solution au problème principal. Nos simulations montrent que notre algorithme permet d’obtenir une très haute équité entre les utilisateurs tout en maintenant un débit proche de l’optimal.

Le chapitre 4 présente l’allocation de ressources dans le sens descendant avec des trafics multimédias. Nous étudions un outil de mesure de la qualité de service dont bénéficie un utilisateur : le PDOR (*Packet Delay Outage Ratio*). Nous montrons que le PDOR couvre les métriques de qualité de service usuelles. Nous présentons ensuite AGSS (*Adaptive and Generic Scheduling Scheme*) une heuristique d’allocation de ressources rapide et performante. Enfin, nous montrons comment les stratégies d’allocation de ressources de l’état de l’art peuvent être adaptées afin d’être utilisées au sein d’AGSS avant de proposer notre propre stratégie (qui s’appuie sur le PDOR). Les résultats de simulations montrent qu’AGSS utilisé avec notre stratégie obtient le plus haut taux d’utilisateurs satisfaits comparé aux autres solutions.

Le chapitre 5 présente une formulation du problème d’allocation de ressources dans le sens montant. De nouveau, nous établissons qu’une solution optimale ne peut être calculée efficacement. Ensuite, nous présentons APASS (*Adaptive and Potential Aware Scheduling Scheme*), une procédure d’allocation de ressources adaptée aux flux multimédias. APASS est composé de trois étapes successives qui fonctionnent ensemble pour calculer une solution efficace au problème. Par simulation, nous montrons que notre algorithme permet d’obtenir, en moyenne, un gain en termes d’utilisateurs satisfaits de 16 % par rapport au deuxième algorithme le plus performant.

Le chapitre 6 apporte une conclusion à la thèse. Nous donnons également des extensions possibles de nos travaux ainsi que des pistes de recherche supplémentaires.

Chapitre 2

État de l'art

Nous présentons dans ce chapitre l'état de l'art ainsi que des définitions nécessaires à la compréhension des chapitres suivants.

À l'origine, l'allocation de ressources est une branche de la théorie de l'ordonnancement. Celle-ci s'attache à planifier dans le temps l'exécution de *tâches* afin de maximiser ou de minimiser un objectif. L'application directe de cette théorie est l'ordonnancement des *processus* s'exécutant sur un processeur (*job-shop problem* [15]). De nombreuses variantes de ce problème ont vu le jour et la théorie de l'ordonnancement est toujours un domaine de recherche actif.

Dans les réseaux filaires [16], les interfaces de sorties des équipements réseaux sont d'abord partagées entre les flux de paquets grâce à des *buffers d'émission* de type FIFO (*First In First Out*) [17]. C'est la stratégie d'ordonnancement la plus simple possible : le premier paquet arrivé dans la file est également le premier à être transmis sur l'interface destination de l'équipement. La notion d'*ordonnancement équitable* (ou *Fair Queuing*) est pour la première fois apparue en 1985 avec John Nagle [18]. L'article propose le *Round Robin* qui constitue un premier pas vers l'ordonnancement avec distinctions des flux.

Avec l'arrivée du GSM (la 2G), l'allocation de ressources dans les réseaux sans fil a commencé à devenir un sujet de recherche important [19]. En particulier, la différenciation de la Qualité de Service pour les appels vocaux commence à être étudiée [20]. Comme déjà indiqué, l'allocation de ressources est différente de l'ordonnancement classique car la qualité des ressources varie en temps et en fréquence, et ce pour chaque utilisateur.

2.1 Un mot sur la théorie de la complexité

Cette section introduit des notions issues de la théorie de la complexité qui seront utilisées par la suite. Par souci de concision et pour faciliter la lecture, de nombreux concepts sont volontairement vulgarisés au profit de l'intuition. Pour davantage de précisions, nous conseillons la lecture de [21].

De manière générale, les problèmes de recherche opérationnelle peuvent être regroupés en classes de complexités :

- La classe **P**, qui regroupe tous les problèmes qui peuvent être résolus (par une machine de Turing) en temps polynomial. Ce sont les problèmes dits “faciles”, i.e. dont une solution optimale peut être trouvée de façon efficace.
- La classe **NP**, qui regroupe les problèmes dont une solution peut être vérifiée en temps polynomial.
- La classe **NP-Complet**, qui regroupe les problèmes dans **NP** qui **ne peuvent pas** être résolus par un algorithme en temps polynomial, à moins que $P = NP$.
- La classe **NP-Difficile**, qui regroupe les problèmes qui sont *au moins* aussi difficiles que tout problème **NP-Complet**.

Comme les problèmes **NP-Complet** sont également **NP-Difficile**, nous utiliserons le terme **NP-Difficile** même lorsque le problème est “seulement” **NP-Complet**.

En fonction de la classe à laquelle appartient le problème, la démarche pour trouver une solution est différente :

- Si le problème est dans **P**, il peut être résolu avec un algorithme polynomial. Trouver un algorithme optimal et polynomial est généralement suffisant.
- Si le problème est **NP-Difficile** (ou **NP-Complet**), alors il n'existe pas d'algorithme polynomial et optimal pour trouver une solution systématique au problème (à moins que $P = NP$). On s'attachera donc à trouver un algorithme sous-optimal mais efficace en complexité temporelle.

Cette thèse traite de **NP-Difficile**, et de l'étude d'algorithmes sous-optimaux. Plus précisément, les algorithmes qui fournissent une solution rapidement à un problème **NP-Difficile** sont appelés des *heuristiques*.

Parmi les différents types d'heuristiques, on distingue les algorithmes d'approximation. Ces algorithmes fournissent une garantie sur la solution obtenue par rapport à la solution optimale. Comme nous allons construire plusieurs algorithmes d'approximation, nous introduisons une définition formelle d'un tel algorithme. Cette définition est inspirée de [21].

Définition 2. *Supposons que P soit un problème de maximisation. Pour chaque instance x du problème P nous pouvons lui associer un ensemble de solutions valides $F(x)$. Pour chaque solution $s \in F(x)$ nous avons un nombre réel positif $c(s)$ qui représente le coût de cette solution. Dans le cas d'un problème de maximisation, l'objectif est de trouver la solution s qui maximise $c(s)$. Le coût optimal est donc défini par $OPT(x) = \max_{s \in F(x)} c(s)$.*

Soit A un algorithme qui, étant donné une instance x , retourne une solution valide notée $A(x)$. On dit que A est une α -approximation ($0 \leq \alpha \leq 1$) si pour toute instance x on a :

$$\alpha \cdot c(OPT(x)) \leq c(A(x)) \tag{2.1.1}$$

En d'autres termes, un algorithme est une α -approximation si le coût de la solution obtenue n'est jamais plus petit que α fois le coût de la solution optimale.

2.2 Stratégies d'allocation de ressources issues des réseaux filaires

Dans cette section, nous passons brièvement en revue les principaux algorithmes d'ordonnancement non-opportunistes. Un ordonnanceur non-opportuniste est un ordonnanceur qui ne prend pas en compte la qualité du canal. Ce type de solution a été construit pour ordonnancer plusieurs flux utilisateurs sur un même lien filaire (par exemple, au niveau d'un routeur). En effet, dans les réseaux filaires, la bande passante disponible est constante et la qualité de la connexion ne varie pas entre les utilisateurs. Cependant, les concepts de ces algorithmes sont fondamentaux et constituent une base d'outils indispensable pour la conception de procédures d'allocation de ressources dans les réseaux sans fil. Nous considérons dans cette section que les paquets de chaque flux sont mis en mémoire dans des buffers différents.

Le Round Robin (RR) [18, 22]. C'est le plus simple et le plus connu des algorithmes d'ordonnancement qui différencient les flux. Les ressources (ou *time slots*) sont allouées une par une, tour à tour, aux utilisateurs. Par conséquent, cette stratégie est équitable en bande passante consommée par chacun des utilisateurs actifs. Cependant, dans un contexte radio, il n'y a aucune équité en termes de débit ou tout autre métrique de qualité de service. L'avantage de cet algorithme est sa facilité d'implémentation, même dans le cadre des réseaux cellulaires OFDMA (LTE, WiMax).

Le Fair Queuing (FQ) [23, 24]. Le *Fair Queuing* est une première approche pour apporter de l'équité en matière de débit entre les utilisateurs. L'idée est de partager les ressources de façon à obtenir un débit équitable entre tous les utilisateurs. Si par exemple le débit maximum du système est égal à R alors le débit de chaque utilisateur obtenu avec le FQ tend vers R/U . Dans le cadre de l'allocation de ressources, il n'est pas facile de définir le débit maximum du système car la qualité de canal de chaque utilisateur est différente. Ainsi, l'implémentation de cet algorithme pour un système OFDMA n'est pas aisée si l'on souhaite maximiser R tout en assurant l'équité.

De plus, la différenciation de services est complètement ignorée. En effet, si un utilisateur justifie un besoin en matière de débit supérieur au débit moyen R/U , sa demande ne pourra pas être satisfaite.

Le Weighted Fair Queuing (WFQ) [25, 26]. Le WFQ est une variante du FQ qui permet de résoudre le problème précédent. Chaque flux utilisateur est associé à un poids α_u (pour l'utilisateur u). L'algorithme WFQ calcule un ordonnancement et permet d'aboutir à une allocation dans laquelle chaque utilisateur u obtient un débit égal à :

$$\frac{\alpha_u}{\sum_{u' \in \mathcal{U}} \alpha_{u'}} R.$$

Le débit est ainsi partagé en fonction du poids de chaque utilisateur. Le WFQ permet donc de différencier les services et adapte les débits en conséquence. Cependant, comme pour le FQ, il n'est pas facile d'implémenter cet algorithme dans le cadre de l'allocation de ressources OFDMA. De plus, la configuration des poids est une limitation du WFQ. Enfin, hormis l'équité, d'autres métriques de qualité de service ne peuvent pas être

prises en compte avec cet algorithme. L'algorithme WFQ est également connu sous le nom de GPS (*Generalized Processor Sharing*) [27].

Le Max-Min Fair (MMF) [28, 29, 30]. La plupart des applications demandent seulement un débit minimum pour leur bon fonctionnement. Il n'est donc pas absolument nécessaire de partager équitablement la bande passante alors qu'il suffit de satisfaire la contrainte sur le débit minimum de chacun des utilisateurs (*worst case fairness*). Le MMF permet de répondre à cet objectif. Le principe de l'algorithme est le suivant :

1. Augmenter le débit de tous les utilisateurs *parallèlement* et à la même vitesse jusqu'à ce qu'un utilisateur ait atteint le débit minimum souhaité.
2. Retirer l'utilisateur satisfait des utilisateurs à ordonnancer.
3. Retourner à la seconde étape.

L'algorithme se termine lorsque la capacité du lien est atteinte.

Avec le MMF, dans le pire des cas, les utilisateurs se partagent la bande passante de façon équitable¹. Dans le cas contraire, MMF alloue davantage de ressources aux utilisateurs qui justifient l'utilisation d'un débit important. Cependant, de manière générale, les flux utilisateurs qui demandent peu de ressources seront toujours satisfaits alors qu'il sera plus difficile de satisfaire un flux qui demande beaucoup. De même que pour les algorithmes précédents, le MMF ne peut pas être facilement adapté à l'allocation de ressources OFDMA.

Bilan. Le RR est simple mais peu équitable. Le FQ permet d'uniformiser les débits tandis que le WFQ et le MMF sont construits pour différencier les services. Ces algorithmes d'ordonnancement sont difficilement adaptables quand le taux de service varie en fonction du temps et de la fréquence.

2.2.1 Stratégies d'allocation de ressources opportunistes

Objectif FDPS. Nous considérons la formulation d'origine du problème FDPS (*Frequency Division Packet Scheduling*), dans lequel les ressources sont divisées dans le domaine fréquentiel. Le problème FDPS est relativement nouveau et a été introduit par [3] avec l'essor de OFDMA. L'allocation des ressources en fréquence permet d'obtenir un gain de performance significatif comparé aux ordonnanceurs s'appliquant uniquement dans le domaine temporel [31].

Soit $\varphi : \mathcal{U} \times \mathcal{N} \rightarrow \mathbb{R}^+$ une fonction d'utilité. Le problème du FDPS consiste, sur un TTI fixé, à calculer l'allocation de ressources qui maximise la somme des fonctions d'utilité :

$$\text{Maximiser } \sum_{u \in \mathcal{U}} \sum_{k \in \mathcal{N}} b_k^u \varphi(u, k) \quad (2.2.1)$$

soumis à $\sum_u b_k^u \leq 1, \forall k$ et $b_k^u \in \{0, 1\}$. Le but est donc de trouver des sous-ensembles disjoints $\mathcal{N}_{u_1}, \mathcal{N}_{u_2}, \dots, \mathcal{N}_{u_U}$ tels que $\sum_{k \in \mathcal{N}} b_k^u \varphi(u, k)$ soit maximisée.

1. En effet, si les utilisateurs sont gourmands (*buffers* toujours pleins), le MMF se comporte comme le FQ.

Algorithme classique. Afin de résoudre ce type de problème, nous introduisons l'*algorithme classique* (Algorithme 1). Ce dernier alloue simplement les ressources une par une à l'utilisateur qui maximise la métrique $W(u, k)$. Sans aucune autre forme de contrainte, nous pouvons constater que l'Objectif 2.2.1 peut être résolu de façon optimale en prenant $W(u, k) = \varphi(u, k)$. Cependant, si l'objectif n'est pas aussi clair que la maximisation de $\sum_u \sum_k b_k^u \varphi(u, k)$ ou si le système est sujet à davantage de contrainte, l'Algorithme 1 peut tout de même être utilisé comme heuristique pour le problème considéré. La difficulté est alors de définir les poids $\{W(u, k)\}_{u,k}$ de façon adaptée. Une définition particulière des poids est appelée dans toute la suite *stratégie d'ordonnancement*. Nous présentons dans cette sous-section les stratégies d'allocation de ressources les plus réputées. Les stratégies qui suivent ont été construites pour des réseaux sans fil de type TDMA/FDMA ou encore CDMA mais peuvent être adaptée pour OFDMA [32]. Nous les présentons donc ici dans le contexte d'un réseau d'accès OFDMA. Notons que dans le cas du MaxSNR et du PF, l'expression des poids peut être identique à l'expression de la fonction d'utilité associée. Pour les stratégies "sensibles aux délais", les fonctions d'utilité et les stratégies d'ordonnancement sont des concepts dissociés.

```

1  $b_k^u \leftarrow 0, \forall (u, k) \in \mathcal{U} \times \mathcal{N}$ .
2 pour chaque ressource  $k \in \mathcal{N}$  faire
3    $u^* \leftarrow \arg \max W(u, k), \forall u \in \mathcal{U}$ .
4   Allocation de  $k$  à l'utilisateur  $u^*$  ( $b_k^{u^*} \leftarrow 1$ ).
5 fin
6 retourner  $\{b_k^u\}_{u,k}$ 

```

Algorithme 1 : Algorithme classique [4].

Le MaxSNR [33, 34] (ou MaxRate). Les ressources sont allouées une par une à l'utilisateur qui a la plus grande valeur de SNR sur la ressource considérée. Cette stratégie est également connue sous le nom de *MaxRate* ou encore *MaxCQI*. Dans un système de type LTE, le MaxSNR s'exprime avec la fonction d'utilité suivante $W(u, k) = \varphi(u, k) = m_{u,k}$. Ainsi, à chaque étape de l'algorithme 1, la ressource k est allouée à l'utilisateur u^* qui vérifie :

$$u^* = \arg \max_{u \in \mathcal{U}} m_{u,k}.$$

Contrairement aux cas précédents, l'algorithme classique avec le MaxSNR prend en compte la qualité du canal pour allouer les ressources. Si on ne considère pas les contraintes spécifiques à LTE, on note que cette stratégie maximise la capacité du système si les utilisateurs sont gourmands (i.e. si les buffers des utilisateurs sont *toujours pleins*). Cependant, cette stratégie souffre d'un manque d'équité entre les utilisateurs. En effet, les ressources seront toujours allouées en priorité à l'utilisateur qui a la meilleure qualité de canal (par exemple, l'utilisateur le plus proche de la station de base). Par conséquent, les autres mobiles ne pourront pas accéder au canal. Du point de vue d'un opérateur de téléphonie mobile, le service offert par le MaxSNR n'est donc pas acceptable.

Le Proportional Fair (PF) [4, 35, 36]. Le PF a été conçu pour résoudre le problème mis en avant avec le MaxSNR. L'objectif du PF est de maximiser $\sum_u \log(\bar{R}_u)$, avec \bar{R}_u le débit moyen à long terme de l'utilisateur u calculé sur une fenêtre glissante. Comme la fonction logarithme est concave, on remarque que plus le débit moyen d'un utilisateur est important alors moins le gain apporté par cet utilisateur sera grand. De cette façon, il devient souvent préférable d'allouer des ressources à tous les utilisateurs plutôt qu'à un groupe privilégié. Il a été montré dans [36], que maximiser $\sum_u \log(\bar{R}_u)$ sur le long terme peut être obtenu en utilisant l'algorithme 1 avec $W(u, k) = \varphi(u, k) = r_{u,k}/\bar{R}_k$, avec $r_{u,k}$ le débit instantané de l'utilisateur u lorsqu'on lui alloue la ressource k . Les allocations sont donc effectuées en fonction de la qualité de canal par rapport au débit moyen de chaque utilisateur. Ainsi, un utilisateur est choisi par l'ordonnanceur si le débit instantané obtenu est bon par rapport à son débit moyen. De cette façon, chaque utilisateur a une chance d'être élu et de se voir allouer des ressources. L'efficacité reconnue du PF a donné lieu à de nombreuses variantes [37, 38, 39].

Remarque. Si l'hypothèse des utilisateurs gloutons n'est pas considérée, notons que le PF maximise mieux la capacité sur le long terme que le MaxSNR. En effet, le MaxSNR présente le désavantage de vider les buffers des utilisateurs ayant une bonne qualité de canal. Ces utilisateurs ne seront donc pas considérés dans les TTIs suivants, diminuant ainsi la diversité multi-utilisateurs [40, 41].

La stratégie M-LWDF (Modified Largest Weighted Delay First) [42]. Cette stratégie est une version modifiée du PF et permet de prendre en compte le délai des paquets. Le type d'objectif implicitement traité par M-LWDF est de maximiser $\sum_u \log(\bar{R}_u)$ sous une contrainte de délai des paquets : " $d_{u,i} < D$, pour tout paquet i ". En d'autres termes, le délai de chaque paquet i de l'utilisateur u (désigné par $d_{u,i}$) doit être inférieur à un temps d'attente maximal D .

Ainsi, l'hypothèse des utilisateurs gourmands est relâchée, la notion de délai des paquets n'ayant pas de sens autrement. Plus précisément, M-LWDF prend en compte le temps d'attente du paquet en tête du buffer (*Head-of-Line packet delay*), qui sera noté w_u pour l'utilisateur u . L'ajout de cette métrique de QoS permet d'éviter de longs temps d'attente des paquets tout en conservant une partie des bénéfices du PF :

$$W(u, k) = w_u \times r_{u,k}/\bar{R}_u.$$

Par conséquent, M-LWDF favorise les paquets qui ont déjà un temps d'attente important dans le buffer. Cette stratégie constitue donc une première approche pour le support des applications en temps réel ou plus généralement des applications sensibles aux délais.

La stratégie exponentielle (EXP ou EWP-W) [43, 44]. Cette politique d'allocation de ressources prend en compte à la fois le délai des paquets et l'état du réseau. Cette stratégie se matérialise de la façon suivante :

$$W(u, k) = \beta_u \exp\left(\frac{a_u w_u}{1 + \sqrt{1/U} \sum_u w_u}\right) \times \bar{R}_u,$$

avec w_u toujours défini comme le temps d’attente du paquet en tête du *buffer*. Selon [43], les paramètres a_u et β_u sont calculés comme suit :

$$\begin{cases} a_u = 6/(0.99 \times \bar{R}_u); \\ \beta_u = 1/\bar{R}_u. \end{cases}$$

Intuitivement, on remarque que la stratégie exponentielle compare w_u à la moyenne des délais courants ce qui permet une gestion des priorités plus lisse (par rapport au M-LWDF). En pratique, EXP réagit rapidement aux variations rapides de la qualité du canal. Cette stratégie permet donc de réduire le délai maximum des paquets.

Il a été montré dans [44] que la stratégie EXP permet de rendre les files d’attente stables si la qualité du canal est modélisée par une chaîne de Markov.

Remarque. L’algorithme classique associé à une stratégie d’ordonnement permet de réaliser une allocation fine des ressources, i.e. une seule ressource est allouée à chaque itération de la boucle principale de l’algorithme. De cette façon, les ressources sont évaluées une par une et l’allocation évolue en fonction des nouveaux calculs de poids. Ce type d’allocation amène à des performances élevées comparé aux allocations “par blocs” de ressources [45]. Dans cette thèse, nous chercherons à utiliser les stratégies d’ordonnement tout en allouant finement les ressources.

Bilan. Une stratégie d’ordonnement permet de répondre à un objectif. Parmi les stratégies présentées, le MaxSNR et le PF favorisent les utilisateurs qui ont une bonne qualité de liens radio avec la station de base. Les stratégies M-LWDF et EXP permettent d’apporter de la qualité de service aux flux utilisateurs en prenant en compte le délai des paquets. Nous renvoyons à [46] pour un exposé plus complet sur les stratégies d’ordonnement dans le cadre d’un environnement LTE non-restreint. De plus, dans le cadre général des réseaux OFDMA, Shams et. al [47] exposent des stratégies équitables plus avancées.

2.3 Allocation de ressources dans les réseaux LTE

Dans la section précédente nous avons présenté les *stratégies d’ordonnement*. Comme il le sera montré dans les chapitres suivants, l’algorithme classique n’obtient pas de très bonnes performances dans les réseaux LTE-A/LTE-B. En effet, les contraintes qui s’appliquent sur le système rendent l’évaluation “ressource par ressource” parfois inadaptée.

L’objectif de cette section est de présenter les algorithmes d’allocation de ressources construits dans le cadre des réseaux LTE. Cependant, de nombreux travaux ne prennent pas en compte l’intégralité des contraintes de la norme. Par conséquent, la plupart des résultats ci-dessous ne répondent pas exactement au problème présenté dans l’introduction (Chapitre 1).

2.3.1 Sens descendant

Nous rappelons que dans le sens descendant de LTE, les ressources allouables sont des *groupes de RBs* que nous désignons par SU (*Scheduling Units*)². De plus, nous insistons sur le fait que la plupart des travaux sur LTE ne considèrent pas intégralement les spécificités de LTE comme la contrainte du *Single MCS* et le calcul non-linéaire des débits. Sans la prise en compte de ces spécificités, il est possible de construire un ordonnanceur optimal en débit (*throughput optimal*) [48], même dans le cas de buffers de taille finie. Dans ce contexte, l'algorithme classique offre également de très bonnes performances et les stratégies d'ordonnement présentées dans la section précédente peuvent être aisément implémentées [46, 49]. Le relâchement de ces contraintes permet également de traiter des problèmes plus ambitieux. Par exemple, Xiao et. al [50] ont traité le problème de la minimisation énergétique tout en assurant un débit minimum à chacun des utilisateurs. Pour cela, les auteurs ont utilisé la *fonction duale de Lagrange* [51] afin de réduire la complexité combinatoire et ainsi proposer un algorithme proche de l'algorithme optimal.

Algorithme MIMO. L'allocation de ressources avec la technologie MIMO est intéressante car elle offre un degré de liberté supplémentaire pour l'ordonnement. En effet, l'ordonnanceur doit également choisir un mode de transmission (*multiplexage spatial* ou *diversité de transmission*) par utilisateur et par TTI [52]. La sélection d'un mode de transmission modifie la qualité de chacune des ressources, il faut donc choisir le mode le plus bénéfique à l'utilisateur selon l'allocation considérée. Lee et al. [53, 54] ont proposé un algorithme d'allocation de ressources pour le mode MIMO de LTE. Il est également montré que le problème est NP-Difficile. Cependant, les spécificités de LTE citées ci-dessus ne sont pas prises en compte. Malgré cela, l'algorithme proposé est un premier pas vers la prise en compte de la contrainte du *Single MCS* puisque l'allocation avec deux modes de transmission est proche d'un système avec deux MCSs (sous la contrainte du *Single MCS*).

Prise en compte de la contrainte du *Single MCS*. Cette contrainte a été traitée pour la première fois par Kwan et. al [12]. Leur solution est composée de deux étapes. La première étape affecte chaque SU à l'utilisateur qui peut recevoir le meilleur débit sur cette ressource. Dans la deuxième étape, le meilleur MCS pour chaque utilisateur est sélectionné (par rapport aux allocations effectuées). L'idée de cet algorithme est d'assigner les SUs aux utilisateurs de façon à réduire le problème multi-utilisateurs en U sous-problèmes d'optimisation avec un simple utilisateur.

Ainsi, cet algorithme ressemble beaucoup à l'algorithme classique (Algorithme 1) avec une étape supplémentaire pour sélectionner le MCS. Par conséquent, dans la suite de la thèse, quand nous faisons référence à l'algorithme classique nous nous référons en fait à l'algorithme proposé dans cet article [12].

Algorithme USA (*Unified Scheduling Algorithm*). Un progrès important a été réalisé dans [55]. Les auteurs ont montré que le problème FDPS sous la contrainte du *Single MCS* peut être réduit à un problème de maximisation de *fonctions ensemble*

2. En réalité, la plupart des travaux considèrent que la plus petite unité de ressource allouable est un RB. Cependant, cela n'a pas de conséquence au niveau théorique.

sous-modulaires. En utilisant ce résultat, un algorithme obtenant au moins la moitié des performances de l'algorithme optimal a été proposé. L'algorithme fonctionne uniquement avec des fonctions d'utilité qui s'expriment par $\varphi(u, k) = \alpha_u r_{u,k}$, avec $0 < \alpha_u \leq 1$. Ce type de fonction d'utilité couvre la stratégie MaxSNR ($\alpha_u = 1$) et PF ($0 < \alpha_u < 1$). La Figure 2.1 résume les étapes de cet algorithme³. A chaque itération, l'algorithme sélectionne l'utilisateur qui maximise un gain noté $\Delta_{u,m}$ (pour l'utilisateur u et le MCS m). Les ressources qui sont bénéfiques à l'utilisateur sélectionné lui sont ensuite allouées.

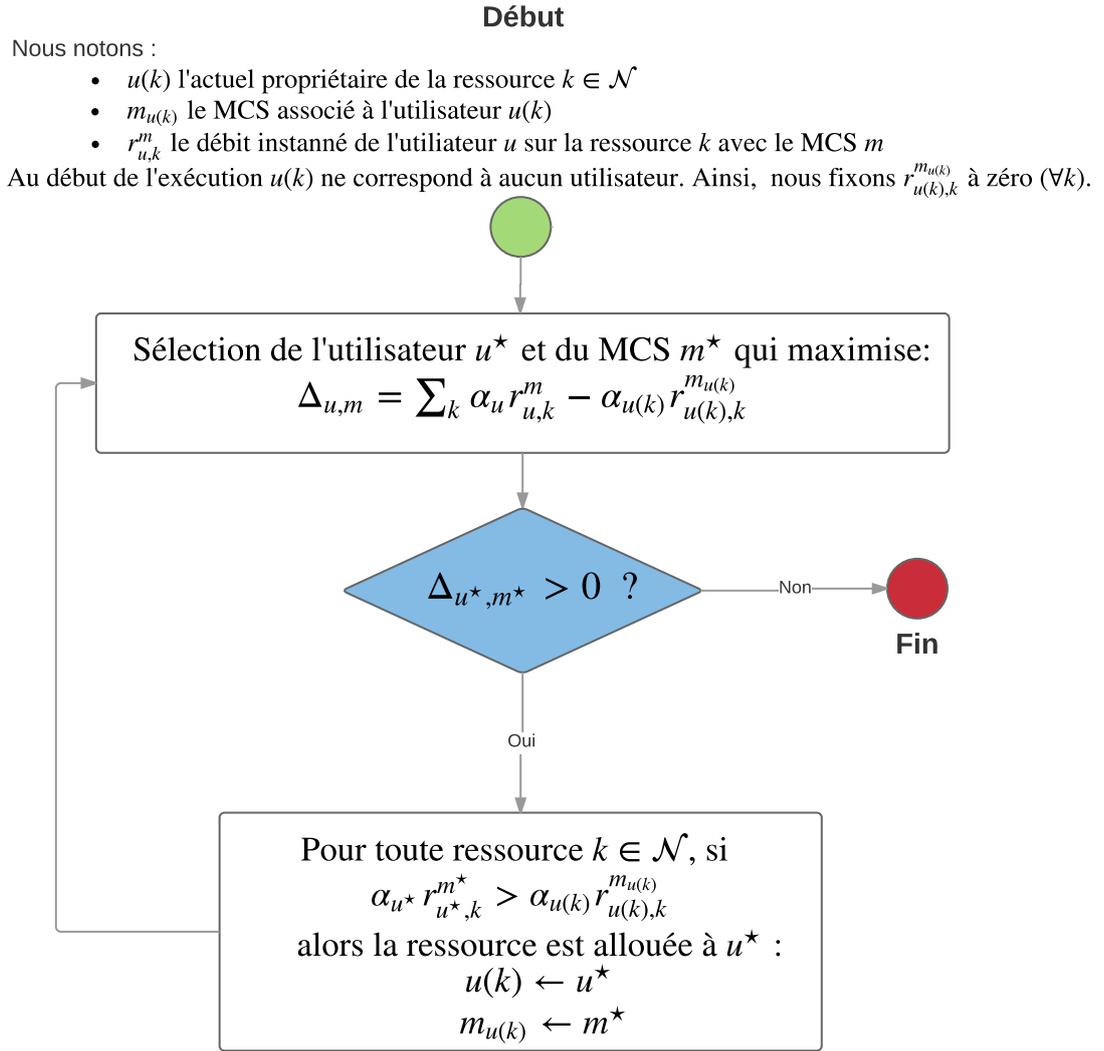


Fig. 2.1 – Détail de l'algorithme USA [55, 13].

Cependant, la borne de performances suppose que le calcul du débit est linéaire. Par exemple, si l'ordonnanceur alloue les ressources k_1 et k_2 à l'utilisateur u avec le

³. Nous avons détaillé cet algorithme car la troisième routine de l'algorithme proposé dans le Chapitre 3 s'en inspire.

MCS m , le débit de l'utilisateur sera égal à $r_{u,k_1}^m + r_{u,k_2}^m$. Nous l'avons cependant adapté pour le rendre compatible avec le calcul non-linéaire des débits (Section 3.6.1). De cette façon, nous avons pu comparer notre solution à USA (Chapitre 3).

Notons également que ce travail a été étendu à l'agrégation de porteuses dans [13] mais sans réelle nouveauté algorithmique.

Par résolution d'un programme linéaire. Une approche plus classique peut également être considérée pour prendre en compte la contrainte du *Single MCS* [56]. Les auteurs expriment tout d'abord le problème comme un programme linéaire en nombres entiers. La contrainte impliquant que les variables sont entières est ensuite relâchée. Le programme linéaire relâché est ensuite résolu avec l'algorithme du simplexe classique [57]⁴. La solution entière arrondie est donc une approximation de la solution optimale. Cependant, avec le calcul non-linéaire des débits, le problème devient également non-linéaire et il n'est plus possible d'utiliser cette méthode de résolution.

Bilan. Seulement quelques approches ont été proposées pour prendre en compte la contrainte du *Single MCS* au sein d'un algorithme d'allocation de ressources. En particulier, l'algorithme classique peut être adapté de façon à considérer cette contrainte [12]. L'avantage de cette solution est qu'elle permet l'implémentation des stratégies d'ordonnancement de façon naturelle. L'algorithme USA [55, 13] prend également pleinement en compte cette contrainte et offre de bonnes performances pour le MaxSNR et le PF.

2.3.2 Sens montant

Nous rappelons que dans le sens montant, toutes les ressources allouées à un utilisateur donné doivent être adjacentes dans le domaine fréquentiel. Par conséquent, les approches proposées pour ce sens de transmission sont totalement différentes.

De plus, l'allocation de ressources dans le sens montant (UL) a été un peu moins étudiée que dans le sens descendant. Une grande partie des travaux calcule une métrique PF pour chaque paire (utilisateur, SU) et alloue les ressources en fonction de ces métriques. Cette métrique, ou poids, est déterminée comme le débit obtenu par l'utilisateur sur le SU divisé par le débit moyen de l'utilisateur (voir Section 2.2.1). Cependant, pour rendre ces algorithmes plus généraux, nous avons remplacé la métrique PF par une métrique générique, notée $W_{u,k}$ pour l'utilisateur u et la ressource k . L'objectif revient alors généralement à maximiser la somme de ces métriques. De même que pour le sens descendant, de nombreux travaux considèrent des objectifs alternatifs tels que la minimisation énergétique [58], ou bien l'allocation d'une même ressource à plusieurs utilisateurs [59]. Dans cette thèse, nous ne considérons pas de tels objectifs.

Premiers algorithmes. L'un des premiers ordonnanceurs pour SC-FDMA a été présenté dans [60]. Cette heuristique, appelée FME (*First Maximum Expansion*), est

4. Cependant, cet article comporte quelques failles. Les auteurs tentent de montrer que l'allocation retournée par l'algorithme simplexe est garantie d'être entière, ce qui rentre en contradiction avec le fait que le problème est NP-Difficile. En effet, par définition, il n'est donc pas possible de trouver un algorithme polynomial qui résout un problème NP-Difficile. Par conséquent, l'algorithme n'est pas optimal, comme cela est indiqué par les auteurs.

construite pour répondre au problème de l'allocation de ressources dans le sens montant (sans la considération des spécificités de LTE). FME commence par allouer la ressource k à l'utilisateur u dont la valeur du poids $W_{u,k}$ est la plus grande. Puis, FME étend l'allocation progressivement à gauche puis à droite⁵ de la ressource k (i.e. la ressource allouée à l'utilisateur u). Plus précisément, les ressources à gauche de k sont évaluées une par une en partant de la ressource située juste avant k . Chacune des ressources évaluée est affectée à l'utilisateur qui bénéficie de la métrique la plus élevée sur cette ressource mais telle que la contrainte de contiguïté (Contrainte 3) est respectée. La même procédure est appliquée pour les ressources situées à droite de k . Comme la solution calculée dépend énormément de la ressource allouée au premier utilisateur, la diversité multi-utilisateurs n'est pas pleinement exploitée.

Une recherche récursive de la métrique la plus élevée a également été proposée dans [60]. Cette version est appelée RME (*Recursive Maximum Extension*) et diffère de FME en permettant l'allocation d'une ressource qui n'est pas située directement à droite ou à gauche des ressources déjà allouées. Ainsi, l'heuristique RME permet d'obtenir de meilleures performances qu'avec FME. Ce résultat a été confirmé par l'analyse de performances réalisée dans [61]. Une version améliorée de RME, appelée IRME (*Improved RME*), a été construite dans [62]. Les auteurs ont introduit un nouveau critère pour l'affectation des ressources : le *seuil de classement*, noté T_r . IRME étend l'allocation sur un des SUs adjacents aux ressources déjà allouées seulement si la métrique du couple (utilisateur, ressource) considéré fait partie des $T_r^{\text{ème}}$ plus importantes métriques. Cette astuce permet d'éviter que des ressources de trop mauvaise qualité soient allouées aux utilisateurs. Ainsi, IRME s'adapte parfaitement bien à la contrainte du Single MCS. Cet algorithme est en moyenne 15 % plus efficace que l'algorithme RME conventionnel et offre de bonnes performances générales [62].

Difficulté du problème. Lee et al. [63] apportent une nouvelle approche au problème de l'allocation de ressources dans le sens montant de LTE. Tout d'abord, les auteurs montrent que le problème est NP-Difficile. Même si l'article ne considère pas la contrainte du *Single MCS*, il suit que le problème que nous considérons dans le Chapitre 1 est également NP-Difficile⁶. Plusieurs heuristiques sont également proposées. L'une d'entre elles consiste à reprendre le schéma de l'algorithme classique en allouant la meilleure ressource au meilleur utilisateur, i.e. à chaque itération l'algorithme alloue la ressource k à l'utilisateur u dont la métrique $W_{u,k}$ est la plus grande. Toutes les ressources situées entre la ressource k et le groupe de SUs déjà alloués à l'utilisateur u (s'il existe) seront également affectées à cet utilisateur. Cependant, si au moins une de ces ressources est déjà allouée à un autre utilisateur alors aucune affectation n'est effectuée et la procédure se répète avec le deuxième couple (utilisateur, ressource) dont la métrique est la plus grande, etc.

Riding Peaks et ses variantes. Le même article [63] propose également *Riding Peaks*. Cette heuristique cherche la métrique la plus importante et l'alloue à l'utilisateur associé uniquement si cette ressource est adjacente aux SUs qui lui sont déjà

5. Nous considérons qu'une ressource k_i est "à gauche" d'une autre ressource k_j si $i < j$ (avec $i, j \in \{1, \dots, N\}$).

6. C'est pourquoi, dans cette thèse, nous n'apportons pas une démonstration de ce résultat.

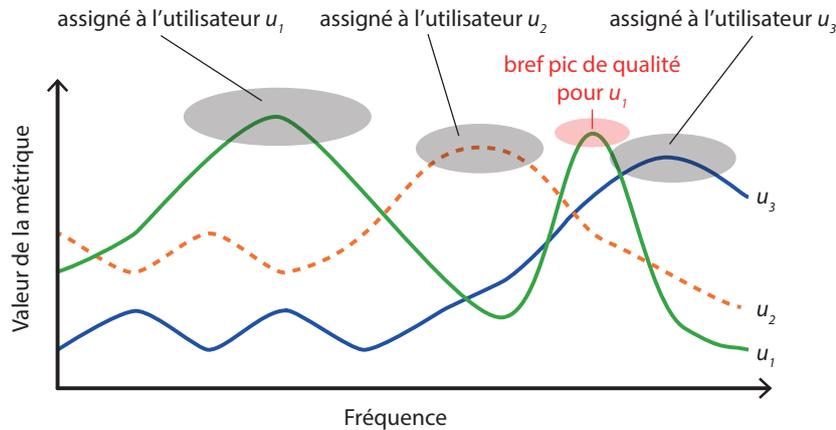


Fig. 2.2 – Exemple d'allocation avec Riding Peaks [63] dans un système composé de trois utilisateurs. Les valeurs des métriques sont exprimées en fonction de la fréquence. Afin de faciliter la visualisation, les valeurs des métriques sont représentées de façon continue.

alloués. Si ce n'est pas le cas, la deuxième meilleure métrique est considérée etc. Cette variante de l'algorithme précédent est plus efficace car les ressources sont allouées finement. L'allocation fine des ressources permet d'adapter facilement ce schéma d'ordonnement aux spécificités de LTE (Chapitre 5). Cependant, Riding Peaks peut être facilement mis en défaut lorsque la variation de la qualité des ressources est importante. Pour diminuer l'impact de ce problème, il est possible de réaliser les allocations par groupe de SUs afin de s'assurer de ne pas affecter les ressources sur une bande de fréquence présentant un "pic de qualité" trop bref. Cependant, les allocations sont alors moins fines. Un exemple d'allocation de l'algorithme Riding Peaks est illustré sur la Figure 2.2. Une légère variante de Riding Peaks, appelée *Iterative*, prenant en compte la contrainte du *Single MCS* (Contrainte 1) a été proposée dans [64]. Le calcul des métriques $W_{u,k}$ est modifié afin de considérer les allocations réalisées lors des étapes précédentes de l'exécution de l'algorithme. C'est cette version que nous utiliserons dans les simulations.

Multiple Carrier Scheduling Algorithm (MC-SA). L'algorithme MC-SA [65] permet d'offrir de la qualité de service aux utilisateurs. En effet, l'objectif de cette solution est d'assurer un débit moyen minimum ainsi que d'éviter un retard de paquets trop important. Lorsque le nombre d'utilisateurs actifs (i.e. le nombre d'utilisateurs ayant des données à transmettre) est supérieur au nombre de ressources alors les utilisateurs sont servis un par un dans un ordre déterminé par une métrique de qualité de service. L'allocation de ressources se déroule alors en deux phases :

1. allocation du SU qui maximise le débit de l'utilisateur sélectionné;

2. allocation du nombre de SUs (adjacents au SU déjà alloué) nécessaires pour satisfaire l'utilisateur en matière de débit minimum requis.

Cependant, si le nombre d'utilisateurs est inférieur au nombre de ressources du système, alors MC-SA distribue équitablement les SUs sans prendre en compte la qualité du canal de chacun des utilisateurs. Cela amène une gestion inefficace des ressources et de faibles performances. MC-SA est donc surtout adapté lorsqu'un grand nombre d'utilisateurs transmettent à de faibles débits.

Algorithme d'approximation. Ren et al. [66] ont montré qu'en plus d'être NP-Difficile, le problème est *MAX SNP-Difficile*⁷. En outre, l'article propose un algorithme basé sur la méthode du *Local Ratio* (LR) [68] qui permet de construire, relativement simplement, des algorithmes d'approximation en exploitant la structure combinatoire du problème. Malgré cela, les spécificités de LTE ne sont pas prises en compte ce qui rend l'algorithme difficilement implémentable dans un système réel.

Bilan. Même si le sens montant est moins étudié que le sens descendant, il existe tout de même de nombreuses solutions. Riding Peaks et IRME sont des solutions efficaces car elles permettent d'allouer finement les ressources. D'autre part, MC-SA permet d'assurer les débits des utilisateurs si la capacité totale du système le permet. Ces trois algorithmes présentent l'avantage de pouvoir être facilement adaptés aux spécificités de LTE. Nous les utiliserons donc pour comparer notre propre proposition (Chapitre 5).

2.4 Conclusion

Dans ce chapitre, nous avons étudié les principales méthodes permettant de partager des ressources entre un ensemble d'utilisateurs afin de répondre à un objectif global (maximisation de la capacité, équité entre les utilisateurs, etc.). Les ordonnanceurs majeurs conçus dans le cadre de réseaux filaires ont d'abord été présentés. Ces premiers algorithmes constituent une base pour l'allocation de ressources en proposant des solutions favorisant le partage équitable des ressources du système. Bien qu'ils ne soient pas totalement adaptés, le *Round Robin* et ses variantes restent toujours aujourd'hui utilisés au sein des équipements LTE [69]. Dans ce cas, l'intégralité de la largeur de bande est allouée à un même utilisateur par TTI. L'ordonnement est donc réalisé uniquement dans le domaine temporel.

Afin d'exploiter la diversité fréquentielle, le problème FDPS, i.e. l'allocation des ressources en fréquence, a été introduit. Pour répondre à cette nouvelle problématique, nous présentons l'algorithme classique (Algorithme 1) qui s'avère efficace dans un système OFDMA [70]. Quelques stratégies d'ordonnement sont également présentées. Celles-ci permettent d'adapter les allocations réalisées par l'algorithme classique à l'objectif de l'opérateur.

Cependant l'algorithme classique peine à fournir de bonnes performances sous les contraintes très restrictives de la norme LTE. C'est pourquoi, dans un dernier temps, nous avons présenté des algorithmes d'allocation de ressources conçus spécifiquement

7. Un problème est Max SNP-Difficile s'il n'existe pas d'algorithme, prenant en entrée une instance de ce problème et un paramètre $\epsilon > 0$, qui produit une solution en temps polynomial avec un facteur d'approximation égal à $1 - \epsilon$ [67].

pour les réseaux LTE. En particulier, dans le cas du sens descendant, une variante de l'algorithme classique [12] ainsi que l'algorithme USA [55] proposent des méthodes pour prendre en charge la contrainte du *Single MCS*.

Néanmoins, à notre connaissance, le calcul non-linéaire des débits n'a pas été considéré dans la littérature. L'impact de cette propriété sur les performances est pourtant important, surtout lorsqu'elle est associée à la contrainte du *Single MCS*. Dans cette thèse, nous souhaitons proposer des solutions pleinement adaptées aux spécificités de la norme.

Chapitre 3

Allocation de ressources dans le sens descendant

Dans ce chapitre nous nous intéressons à l'allocation de ressources dans le sens descendant. Nous prenons l'hypothèse des utilisateurs gourmands, i.e. la station a toujours des données à transmettre aux utilisateurs.

Après avoir formalisé le problème, nous montrons que ce dernier est NP-Difficile. Par conséquent, il n'est pas possible de le résoudre de façon optimale en moins d'une milliseconde. Afin d'apporter une solution efficace à ce problème, nous proposons un algorithme qui peut être adapté pour une large variété d'objectifs. Cet algorithme est composé de trois routines. La première calcule une solution approchée à un sous-problème défini comme une *relaxation* de notre problème principal. La deuxième routine modifie l'allocation réalisée pour l'algorithme précédent afin de prendre en compte la contrainte du *Single MCS*. La dernière routine permet de re-distribuer efficacement les ressources qui n'ont pas été allouées. La Figure 3.1 illustre l'objectif de chaque routine et leurs relations.

À l'aide de simulations basées sur la norme 3GPP LTE-B, nous comparons notre algorithme avec l'algorithme optimal ainsi que d'autres algorithmes issus de la littérature (que nous avons adaptés à LTE-B). L'évaluation des performances montre que notre solution permet d'obtenir une capacité supérieure à ce que l'on obtient avec les algorithmes existants, et proche de l'optimal. En association avec la stratégie *Proportional Fair* (PF), un gain de 16 % est observé en termes de capacité maximum par rapport au deuxième meilleur algorithme tout en obtenant une équité très convenable.

3.1 Formalisation du problème

Dans cette section, nous exprimons formellement le problème. Puis, nous introduisons un sous-problème, qui est une relaxation de notre problème principal. L'introduction du sous-problème nous permet :

- de montrer que notre problème principal est NP-Difficile ;
- d'utiliser ce sous-problème comme intermédiaire afin de faciliter la construction d'une solution pour le problème principal (première routine).

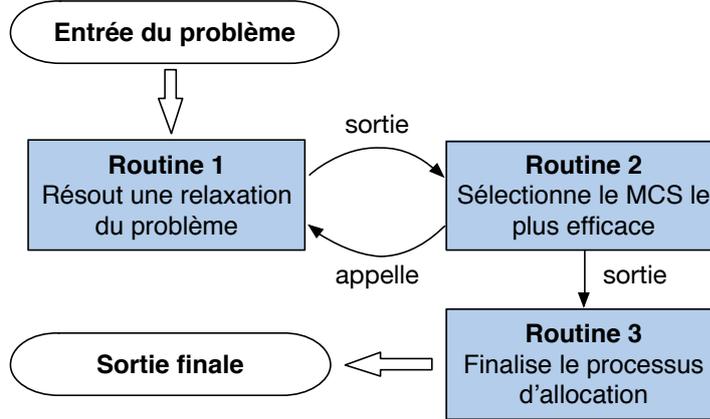


Fig. 3.1 – Les étapes de notre algorithme d'approximation.

Afin d'alléger les étapes et les notations, nous présentons notre travail sans considérer l'agrégation de porteuses et le mode MIMO (*Multiple Input Multiple Output*). Cependant, l'algorithme proposé peut facilement être adapté pour supporter ces fonctionnalités de la même façon que [13] et [55].

3.1.1 Le problème LTE DL FDPS

Fonctions d'utilité. Afin de rester général, nous souhaitons maximiser la somme de fonctions d'utilité définies de manière générique. Une fonction d'utilité, notée φ_u , est associée à chaque utilisateur $u \in \mathcal{U}$. Cette fonction dépend, au minimum, de l'ensemble des SUs alloués à l'utilisateur u et du MCS assigné à cet utilisateur :

$$\forall u, \varphi_u : 2^{\mathcal{N}} \times \mathcal{M} \rightarrow \mathbb{R}^+,$$

avec $2^{\mathcal{N}}$ l'ensemble des parties de \mathcal{N} , i.e. l'ensemble de tous les sous-ensembles de \mathcal{N} .

Nous considérons également que φ_u est une fonction croissante par rapport à ses deux principaux arguments. Plus formellement, $\forall u \in \mathcal{U}$:

- $\forall m$, soit $\varphi_{u,m}(\mathcal{N}_u) = \varphi_u(\mathcal{N}_u, m)$ alors $\forall S \subseteq \mathcal{N}$ et $\forall e \in \mathcal{N} \setminus S$, $\implies \varphi_{u,m}(S \cup \{e\}) \geq \varphi_{u,m}(S)$. En d'autres termes, $\varphi_{u,m}$ est une *fonction ensemble* monotone.
- $\forall \mathcal{N}_u \subseteq \mathcal{N}$, soit $\varphi_{u,\mathcal{N}_u}(m) = \varphi_u(\mathcal{N}_u, m)$ alors $\varphi_{u,\mathcal{N}_u}$ est croissante.
- $\forall m, \varphi_u(\emptyset, m) = 0$.

Le problème LTE DL FDPS (**P1**) est défini comme suit :

Entrées : La matrice de taille $U \times N$ définie par $\mathcal{I} = (m_{u,k})_{u,k}$ et l'expression des fonctions d'utilité $\{\varphi_u\}_u$.

Objectif :

$$\text{Maximiser} \quad \sum_{u \in \mathcal{U}} \varphi_u(\{k \in \mathcal{N} \mid b_{u,k} = 1\}, m_u), \quad (3.1.1)$$

sujet à :

$$\sum_u b_{u,k} \leq 1, \forall k \in \mathcal{N}, \quad (3.1.2)$$

$$m_u \leq b_{u,k} m_{u,k} + (1 - b_{u,k}) m_M, \forall u, \forall k, \quad (3.1.3)$$

$$b_{u,k} \in \{0, 1\}, \forall u, \forall k \text{ et } m_u \in \mathcal{M}, \forall u, \quad (3.1.4)$$

avec $b_{u,k}$ une variable binaire qui est égale à 1 si le SU k est alloué à l'utilisateur u . Par conséquent, $\sum_{k \in \mathcal{N}} b_{u,k}$ est égal au nombre de SUs alloués à l'utilisateur u . Notons que contrairement aux travaux précédents présentés dans le Chapitre 2, le problème n'est pas exprimé linéairement puisque le calcul non-linéaire du débit rend cette formulation inexploitable¹.

Le problème LTE DL FDPS a pour but de maximiser la somme des valeurs des fonctions d'utilité. La Contrainte (3.1.2) assure que chaque SU peut être alloué à, au plus, un utilisateur. En effet, il est parfois préférable de ne pas allouer un SU (voir Propriété 1 ci-dessous). Cette contrainte indique également qu'il n'est pas possible d'allouer plus de N SUs. La Contrainte (3.1.3) prend en considération la contrainte du *Single MCS* afin de déterminer le meilleur MCS à assigner à chaque utilisateur. Nous rappelons que m_u peut être exprimé de façon non-linéaire plus simplement par $\min(\{m_{u,k} \in \mathcal{I} \mid b_{u,k} = 1\})$.

Utilisateur 1	3	1	1	4	3
Utilisateur 2	4	2	4	2	1
	SU1	SU2	SU3	SU4	SU5

Fig. 3.2 – Exemple d'une allocation optimale (en vert) qui n'alloue pas toutes les ressources. La valeur de la fonction d'utilité d'un utilisateur est le minimum des entiers de chaque couple (utilisateur, SU) parmi les SUs alloués à cet utilisateur.

Une première propriété. La contrainte du *Single MCS* implique la propriété qui suit.

Propriété 1. *Une solution optimale peut ne pas allouer toutes les ressources du système.*

Cette propriété peut surprendre mais il s'agit d'une conséquence directe de la Contrainte 1 et des remarques faites ci-dessus. Il est facile de construire une instance dans laquelle une allocation optimale n'alloue pas toutes les ressources comme l'atteste la Figure 3.2. Dans cette figure, l'allocation d'un SU est représentée par un cadre vert. Le nombre situé à l'intérieur de chaque cadre représente l'indice du MCS $m_{u,k}$ (pour le cadre (u, k)). Notons que plus cet indice est grand, plus le MCS est efficace spectralement. La fonction d'utilité est définie comme le nombre de ressources allouées multiplié par l'indice du MCS utilisé : $\varphi_u(\mathcal{N}_u, m_u) = |\mathcal{N}_u| \times index(m_u)$. Par conséquent, le coût de

1. Plus précisément, il est tout de même possible d'exprimer le problème de façon linéaire en faisant intervenir une variable binaire par partie de \mathcal{N} . Il y aura alors un nombre exponentiel de variables (voir Proposition 3.2.2).

la solution optimale est égal à 14 et mène à un SU non-alloué (SU2), ce qui prouve la propriété.

En pratique, cette propriété indique qu'il n'est pas nécessaire de faire une allocation de tous les SUs afin d'aboutir à une allocation efficace (à défaut d'être optimale). C'est pourquoi les algorithmes et heuristiques que nous présenterons dans cette thèse ne procéderont pas nécessairement à une allocation exhaustive des ressources du système.

3.2 Sous-problème et résultats fondamentaux

Le sous-problème. Nous notons **P1** le problème principal (LTE DL FDPS). Afin de résoudre **P1**, nous chercherons d'abord à résoudre un sous-problème "plus simple" (noté **P2**). On définit ce sous-problème de la même manière que **P1** mais sans la contrainte du *single MCS*.

Par conséquent, l'entrée du problème **P2** reste la même que pour notre problème principal. En pratique, ce sous-problème correspondrait à un système LTE dans lequel chaque utilisateur pourrait utiliser un *transport block*² différent pour chaque MCS.

L'objectif de **P2** est de maximiser $\sum_u \Phi_u(\mathcal{N}_u)$ sujet à $\bigcap_u \mathcal{N}_u = \emptyset$ et $\mathcal{N}_u \subset \mathcal{N}, \forall u$; avec Φ_u la fonction d'utilité *libérée de la contrainte du single MCS* de l'utilisateur u . Bien sûr, Φ_u est étroitement liée à φ_u comme nous le montrons dans la preuve du Théorème 2.

Le problème du bien-être social. Nous décrivons ici le *problème de maximisation du bien-être social* (SWMP pour *Social Welfare Maximization Problem*), qui nous permettra de résoudre **P2** en utilisant des résultats connus.

L'entrée du problème est composée de :

- un ensemble X de q objets ;
- un ensemble P de p joueurs ;
- une *fonction d'évaluation* croissante pour chaque joueur $v_i : 2^X \rightarrow \mathbb{R}^+$.

L'objectif est de partitionner X en ensembles disjoints S_1, \dots, S_p de façon à maximiser le *bien-être social* $\sum_{i=1}^p v_i(S_i)$. Ce problème est NP-Difficile [71].

Résumé de nos résultats fondamentaux :

1. Le sous-problème **P2** est NP-Difficile puisqu'il est possible de réaliser une réduction polynomiale de SWMP vers **P2** en construisant une instance de **P2** avec un seul MCS disponible dans le système ($M = 1$). Il suit que notre problème principal (**P1**) est également NP-Difficile.
2. De façon similaire, on peut réaliser une réduction en temps linéaire de **P2** à SWMP.

Ces deux points permettent d'établir que **P2** et SWMP sont équivalents en termes de réductions polynomiales. Par conséquent, afin de résoudre **P2**, il est possible d'utiliser les algorithmes efficaces de la littérature qui ont été originellement construits pour résoudre le problème du bien-être social. Nous utiliserons cette méthode dans la section suivante.

2. Un *transport block* est une trame qui encapsule l'intégralité des données qui seront émises à destination d'un utilisateur sur un TTI (voir Annexe A).

Le reste de cette section présente les preuves des résultats avancés.

Théorème 1. *Le sous-problème P2 est NP-Difficile.*

Preuve. Soient P, X , et $v_i, \forall i \in P$, une instance du SWMP. Nous construisons l'instance pour le sous-problème P2 comme suit.

- Nous considérons un seul MCS ($M = 1$), et par conséquent $m_{u,k} = m_1, \forall (u, k)$; avec m_1 l'unique MCS disponible dans le système.
- L'ensemble des joueurs est mis en correspondance avec l'ensemble des utilisateurs ($\mathcal{U} = P$). De même, l'ensemble des objets est l'ensemble des SUs ($\mathcal{N} = X$).
- $\forall u \in \mathcal{U}, \mathcal{N}_u = S_u$ et $\forall S \subseteq \mathcal{N}, \varphi_u(S, m_1) = v_u(S)$;

De cette façon, l'objectif est donc bien d'allouer les ressources aux utilisateurs de façon à maximiser $\sum_u \Phi_u(\mathcal{N}_u)$. Plus formellement, soit $C \in \mathbb{R}^+$, nous affirmons qu'une partition S_1, \dots, S_p de X est telle que $\sum_{i \in P} v_i(S_i) = C$ si et seulement si l'allocation de ressources correspondante $\mathcal{N}_{u_1}, \dots, \mathcal{N}_{u_U}$ est telle que $\sum_u \Phi_u(\mathcal{N}_u) = C$.

En effet, comme un seul MCS est disponible, nous avons $\Phi_u(\mathcal{N}_u) = \varphi_u(\mathcal{N}_u, m_1), \forall u$. Ainsi, par construction $\sum_{i \in P} v_i(S_i) = \sum_{u \in \mathcal{U}} \varphi_u(\mathcal{N}_u, m_1) = \sum_u \Phi_u(\mathcal{N}_u)$.

En conclusion, le problème P2 est au moins aussi difficile que le problème du bien-être social. \square

Corollaire 1. *Le problème LTE DL FDPS (P1) est NP-Difficile.*

Preuve. La preuve est très similaire à la précédente. Soient P, X , et $v_i, i \in P$ une instance de SWMP. On peut construire une instance de P1 avec un seul MCS ($M = 1$) de sorte que la contrainte du single MCS ne rentre pas en considération. L'instance résultante de P1 est ainsi équivalente à l'instance du SWMP. Par conséquent, notre problème principal est au moins aussi difficile que le problème du bien-être social, i.e. une solution optimale ne peut pas être obtenue via un algorithme de complexité polynomiale (à moins que $P=NP$). \square

Théorème 2. *Il existe une réduction en temps linéaire de P2 à SWMP.*

Preuve. Tout d'abord, soient \mathcal{N}, \mathcal{U} et \mathcal{I} une instance de P2. Nous commençons par observer que l'ensemble des SUs alloués à l'utilisateur u peut être décomposé de la façon suivante : $\mathcal{N}_u = N_{u,m_1} \cup N_{u,m_2} \cup \dots \cup N_{u,m_M}$, avec $N_{u,m}$ l'ensemble des SUs alloués à l'utilisateur u avec le MCS m . Ainsi, chaque fonction Φ_u peut être exprimée par : $\Phi_u(\mathcal{N}_u) = \varphi_u(N_{u,m_1}, m_1) + \varphi_u(N_{u,m_2}, m_2) + \dots + \varphi_u(N_{u,m_M}, m_M)$. L'objectif est donc de maximiser :

$$\begin{aligned} \sum_{u \in \mathcal{U}} \sum_{m \in \mathcal{M}} \varphi_u(N_{u,m}, m) &= \sum_{(u,m)} \varphi_u(N_{u,m}, m) \\ &= \sum_{(u,m)} \varphi_{(u,m)}(N_{u,m}). \end{aligned}$$

Nous considérons que chaque joueur est une paire $(u, m) \in \mathcal{U} \times \mathcal{M}$. De plus, pour chaque joueur i (avec $i = (u, m) \in \mathcal{U} \times \mathcal{M}$), on définit une fonction $\pi_i : 2^{\mathcal{N}} \rightarrow 2^{\mathcal{N}}$,

$$\pi_i(S) = \pi_{(u,m)}(S) = \{k \in S \mid m \geq m_{u,k}\}. \quad (3.2.1)$$

Cette fonction permet d'écarter les SUs qui ne supportent pas le MCS m pour l'utilisateur u .

Enfin, soient $P = \mathcal{U} \times \mathcal{M}$, $X = \mathcal{N}$, $S_i = N_i (\forall i \in P)$ et $v_i = \varphi_i \circ \pi_i, \forall i$. Cette réduction est linéaire avec U , N et M . Ainsi, l'objectif de cette instance SWMP est en effet de trouver une partition $X = S_1 \cup S_2 \cup \dots \cup S_p$ telle que $\sum_{i \in \mathcal{P}} v_i(S_i)$ soit maximisée.

Plus formellement, soit $C \in \mathbb{R}^+$, nous affirmons qu'une allocation de ressources $\mathcal{N}_{u_1}, \dots, \mathcal{N}_{u_U}$ est telle que $\sum_u \Phi_{u \in \mathcal{U}}(\mathcal{N}_u) = \sum_{(u,m)} \varphi_{(u,m)}(N_{u,m}) = C$ si et seulement si la partition correspondante de X (S_1, \dots, S_p) est telle que $\sum_{i \in \mathcal{P}} v_i(S_i) = C$.

En conséquence, chaque instance de **P2** peut être réduite en une instance du SWMP. Cette propriété est un résultat important car nous pouvons maintenant utiliser les résultats de l'état-de-l'art sur le SWMP afin de résoudre notre sous-problème **P2**³.

Taille de l'espace de recherche. Comme **P1** fait partie des problèmes les plus difficiles, notre premier espoir est que l'espace de recherche soit suffisamment petit de façon à ce que l'on puisse considérer toutes les possibilités à l'aide d'un algorithme de force brute afin de retenir la meilleure allocation.

Proposition 1. La taille de l'espace de recherche pour le problème LTE DL FDPS (**P1**) est égale à :

$$\sum_{i=0}^{\min(U,N)} \binom{U}{i} \frac{N!}{(N-i)!} (i+1)^{N-i} M^i. \quad (3.2.2)$$

Preuve. Tout d'abord, considérons que seuls i utilisateurs ($0 \leq i \leq \min(U, N)$) reçoivent des SUs. Il y a $\binom{U}{i}$ ensembles d'utilisateurs de cardinal i possibles. On assigne ensuite un seul SU à chaque utilisateur. Une telle allocation peut être modélisée par une liste ordonnée sans répétition, ou encore un arrangement, de i éléments choisis dans \mathcal{N} . Le nombre d'arrangements possibles est égal à $\frac{N!}{(N-i)!}$.

Les SUs restants sont ensuite partagés entre les i utilisateurs sélectionnés ainsi qu'un utilisateur "virtuel" qui représente les SUs non-alloués. Le nombre de partages différents est donc égal à $(i+1)^{N-i}$. Pour finir, il y a M^i façons d'affecter les MCS aux i utilisateurs.

Il reste à considérer toutes les valeurs de i afin d'obtenir le nombre total de solutions possibles :

$$\sum_{i=0}^{\min(U,N)} \binom{U}{i} \frac{N!}{(N-i)!} (i+1)^{N-i} M^i.$$

□

Le nombre de possibilités est donc bien trop important pour que l'espace de recherche puisse être exploré par un algorithme de recherche exhaustive en une milliseconde, même pour de petites valeurs de U , N et M .

3. En réalité, même notre problème principal **P1** peut être vu comme le problème du bien-être social (d'une manière très générique). Cependant, dans ce cas, les fonctions d'utilité ne peuvent pas être classifiées facilement comme nous le montrons dans la Sous-section 3.3.1.

Un mot sur les programmes linéaires en nombres entiers. Même si notre problème est NP-Difficile, on pourrait penser à utiliser un logiciel dédié à la résolution des programmes linéaires en nombres entiers (ou *Integer Linear Program*) [72, 73]. Ces logiciels utilisent des résultats mathématiques et algorithmiques qui permettent de parcourir intelligemment l'espace de recherche des solutions afin de trouver une solution optimale. Malheureusement, nos fonctions d'utilité sont supposés génériques et donc potentiellement non-linéaires. D'autant plus que même pour un objectif simple tel que la maximisation de la capacité, la "fonction débit" étant non-linéaire, alors le problème (tel qu'il est exprimé) l'est également.

Pour toutes ces raisons, dans la suite, nous proposons un algorithme d'approximation de complexité polynomiale.

3.3 Algorithme d'approximation

Dans cette section, nous présentons un algorithme d'approximation pour le problème LTE DL FDPS (**P1**). Après avoir résolu le problème **P2** en utilisant les résultats de l'état-de-l'art sur le problème du bien-être social, nous proposons une solution pour **P1**. Cette solution est ensuite améliorée et complétée afin de proposer un algorithme d'allocation de ressources performant.

3.3.1 Première étape : vers la résolution de P2

Pour résoudre **P2**, nous nous appuyons sur le **problème du bien-être social** (SWMP) car ce dernier est très proche de **P2** comme démontré dans la section précédente. Comme le SWMP est NP-Difficile, nous cherchons à approximer la solution optimale. Nous rappelons qu'une α -approximation ($0 \leq \alpha \leq 1$) pour le SWMP est un algorithme qui retourne une partition $S_1 \cup \dots \cup S_p \subseteq X$ telle que pour toute autre partition $S'_1 \cup \dots \cup S'_p \subseteq X$ alors $\alpha \sum_{i \in P} v_i(S'_i) \leq \sum_{i \in P} v_i(S_i)$.

Malheureusement, il n'y a pas de résultat général applicable à n'importe quel type de fonction d'évaluation. C'est pourquoi nous restreignons ces fonctions à l'une des classes présentées ci-dessous. Ces classes couvrent tout de même une large variété d'objectifs. Par souci de concision, nous n'écrivons pas les algorithmes utilisés pour résoudre une instance de SWMP car ils sont disponibles dans les articles référencés.

Soit $f : 2^S \rightarrow \mathbb{R}^+$ une fonction ensemble qui peut être interprétée comme une fonction de valuation dans le problème du bien-être social ou comme une fonction d'utilité ($\varphi_{(u,m)}$) dans **P2** (voir preuve du Théorème 2). Nous présentons différentes classes de fonctions.

Additive : pour tout $S_1 \subseteq S$ et $S_2 \subseteq S$ alors $f(S_1 \cup S_2) = f(S_1) + f(S_2)$. Sous cette propriété, pour chaque sous-ensemble $S' \subseteq S$, il est toujours possible d'exprimer $f(S')$ comme une somme de "poids", i.e. $f(S') = \sum_{k \in S'} w_f(k)$, avec $w_f(k)$ le poids associé à l'élément $k \in S'$. C'est le cas considéré dans tous les travaux précédents traitant de l'allocation de ressources [13, 55, 56, 49]. Le problème est bien plus facile dans ce scénario puisqu'un algorithme optimal et polynomial existe pour le problème du bien-être social. D'ailleurs, cet algorithme n'est nul autre qu'une version adaptée

de l'algorithme classique (Algorithme 1). De surcroît, une 1/2-approximation a été construite pour le problème LTE DL FDPS (**P1**) sous l'hypothèse des fonctions d'utilité additives [13].

Sous-modulaire : pour tout $S_1, S_2 \subseteq S$ tels que $S_2 \subseteq S_1$ et pour tout $e \in S \setminus S_1$ alors $f(S_1 \cup \{e\}) - f(S_1) \leq f(S_2 \cup \{e\}) - f(S_2)$. Si les fonctions d'utilité sont sous-modulaires alors le problème **P2** peut être traité en utilisant un algorithme aléatoire avec un facteur d'approximation égal à $1 - 1/e$ [74]. Le cas sous-modulaire peut intervenir si les fonctions d'utilité sont concaves, i.e. plus le nombre de ressources allouées à un "joueur" (dans $\mathcal{U} \times \mathcal{M}$) alors plus l'accroissement du gain devient faible. En pratique, on peut imaginer que la satisfaction d'un utilisateur augmente lentement après avoir reçu un certain montant de ressources, i.e. la quantité $f(S' \cup \{e\})$ s'accroît plus lentement avec l'augmentation du nombre d'éléments dans S' .

Sous-additive : pour tout $S_1, S_2 \subseteq S$ alors $f(S_1 \cup S_2) \leq f(S_1) + f(S_2)$. Cette classe est plus générale que la classe des fonctions sous-modulaires puisque toute fonction sous-modulaire non-négative est également sous-additive. Dans ce cas, à notre connaissance, le meilleur facteur d'approximation atteint est 1/2 [75].

Super-additive : pour tout $S_1, S_2 \subseteq S$ alors $f(S_1 \cup S_2) \geq f(S_1) + f(S_2)$. Le problème **P1** avec des fonctions d'utilité super-additives couvre d'importants objectifs tels que les stratégies MaxSNR et PF. En effet, pour un MCS m donné, la "fonction débit" $R(\cdot, m)$ est super-additive. Pour cette classe de fonction, le meilleur facteur d'approximation atteint est de $\sqrt{\log(q)}/q$ avec q le nombre total d'objets de l'instance SWMP ($q = N$ après réduction de **P2** vers SWMP) [76, 77].

entrée : une instance de **P1**, que nous interprétons comme une instance de **P2**.

- 1 **Réduire** l'instance du problème **P2** en une instance SWMP (voir preuve du Théorème 2).
- 2 **Déterminer** la classe des fonctions d'utilité. Puis, **exécuter** l'algorithme de l'état-de-l'art correspondant sur l'instance SWMP.
- 3 Le résultat, $\{N_{u,m}\}$, est également valide pour l'instance originelle de **P2**.
- 4 **retourner** $\{N_{u,m}\}_{u \in \mathcal{U}, m \in \mathcal{M}}$

Algorithme 2 : SubProblem_Algo - Allocation de ressources pour **P2**.

Description de la première étape. Afin de résoudre **P2**, toutes les fonctions d'utilité doivent appartenir à la même classe de fonctions. Cette contrainte n'est pas vraiment une limitation puisqu'il est difficile d'interpréter la somme de fonctions complètement différentes. De plus, il est requis de faire la réduction présentée dans la preuve du Théorème 2. Par conséquent, il est nécessaire d'assurer que si φ_i ($i \in \mathcal{U} \times \mathcal{M}$) appartient à l'une des classes ci-dessus alors $\varphi_i \circ \pi_i$ appartient également à cette classe, avec π_i définie dans Eq. 3.2.1. Comme π_i supprime simplement les SUs qui ne respectent pas le MCS attribué à chaque joueur, cette propriété est respectée avec les classes ci-dessus.

L'Algorithme 2 décrit la première routine de notre solution. Nous appelons cette routine `SubProblem_Algo`. Comme déjà indiqué, notre première routine interprète l'instance de **P1** comme une instance de **P2**, et retourne donc une solution approximée pour **P2**. Plus précisément, cette routine réduit l'instance de **P2** en une instance de SWP équivalente. Une solution sous-optimale de l'instance de SWP est ensuite calculée en utilisant un algorithme d'approximation de la littérature qui dépend de la classe des fonctions d'utilité. De part le Théorème 2, la solution calculée est également valide pour l'instance de **P2**. Par conséquent, le facteur d'approximation de `SubProblem_Algo` est le même que celui de l'algorithme utilisé pour résoudre l'instance de SWMP. De plus, comme la réduction est en temps linéaire, la complexité de `SubProblem_Algo` est la même que la complexité de l'algorithme de l'état-de-l'art utilisé.

3.4 Deuxième étape : prise en compte du single MCS

Nos résultats pour cette seconde routine dépendent de l'algorithme utilisé pour trouver une solution pour **P2**. Pour chaque utilisateur u et pour chaque MCS m , `SubProblem_Algo` retourne $N_{u,m}$ l'ensemble de SUs assigné à l'utilisateur u avec le MCS m . Étant donné une solution déterminée par l'algorithme précédent, nous souhaitons maintenant prendre en compte la contrainte du *single MCS*.

Le pseudo-code de notre solution est présentée dans l'Algorithme 3. Nous appelons cet algorithme `Approx_Algo`. Tout d'abord, on considère que tous les ensembles $N_{u,m_1}, N_{u,m_2}, \dots, N_{u,m_M}$ sont *pré-alloués* à l'utilisateur à u .

A chaque étape de l'algorithme, nous choisissons un utilisateur u . La contrainte du *single MCS* est prise en compte en choisissant le MCS m_u qui maximise la fonction d'utilité en considérant successivement les différents sous-ensembles de ressources pré-alloués à l'utilisateur u :

$$m_u = \operatorname{argmax}_m \varphi_u \left(\bigcup_{m' \geq m} N_{u,m'}, m \right).$$

L'ensemble des SUs réellement alloués à l'utilisateur u est noté \mathcal{N}_u . Cet ensemble est donc déterminé comme l'ensemble des ressources qui lui ont été pré-allouées et qui supportent le MCS m_u :

$$\mathcal{N}_u = \bigcup_{m \geq m_u} N_{u,m}.$$

Après avoir alloué l'ensemble \mathcal{N}_u et le MCS m_u à l'utilisateur u , cet utilisateur est retiré de la liste des utilisateurs à ordonnancer. La première routine, `SubProblem_Algo`, est ensuite à nouveau exécutée sur l'ensemble des ressources restantes et sur les utilisateurs qui n'ont pas encore été ordonnancés. De cette façon, les SUs pré-alloués à u et qui n'ont pas encore été alloués (i.e. $\bigcup_{m < m_u} N_{u,m}$) peuvent être pré-alloués à un autre utilisateur.

L'algorithme retourne $\{\mathcal{N}_u, m_u\}_{u \in \mathcal{U}}$, i.e. l'ensemble des SUs alloués à chaque utilisateur ainsi que le *single MCS* associé.

Le théorème suivant donne une garantie (en facteur constant) sur la qualité de l'allocation par rapport à la solution optimale.

```

1  $\forall u, \mathcal{N}_u \leftarrow \emptyset$ 
2  $\mathcal{U}^r \leftarrow \mathcal{U}$ 
3  $\{(N_{u,m_1}, \dots, N_{u,m_M})\}_{u \in \mathcal{U}} \leftarrow \text{SubProblem\_Algo}$  sur l'ensemble des ressources  $\mathcal{N}$ 
   et l'ensemble des utilisateurs  $\mathcal{U}$ .
4 l //  $\bigcup_m N_{u,m}$  est l'ensemble des SUs alloués à  $u$ .
   pour chaque utilisateur  $u$  dans un ordre arbitraire faire
   | // Sélectionner le "meilleur" MCS :
5   |  $m_u \leftarrow \text{argmax}_m \varphi_u \left( \bigcup_{m' \geq m} N_{u,m'}, m \right)$ 
   | // Allocation des SUs qui supportent le MCS  $m_u$  :
6   |  $\mathcal{N}_u \leftarrow \bigcup_{m \geq m_u} N_{u,m}$ 
7   |  $\mathcal{U}^r \leftarrow \mathcal{U}^r \setminus \{u\}$ 
8   |  $\{(N_{u,m_1}, \dots, N_{u,m_M})\}_{u \in \mathcal{U}^r} \leftarrow \text{SubProblem\_Algo}$  sur l'ensemble des SUs
   | restants  $\mathcal{N} \setminus \bigcup_u \mathcal{N}_u$  et des utilisateurs non-encore ordonnancés  $\mathcal{U}^r$ .
9 fin
10 retourner  $\{\mathcal{N}_u, m_u\}_{u \in \mathcal{U}}$ 

```

Algorithme 3 : Approx_Algo - Première allocation pour **P1**.

Théorème 3. Soit α le facteur d'approximation du sous-problème **P2**. L'Algorithme 3 est une $\frac{\alpha}{M}$ -approximation pour le problème LTE DL FDPS, avec M le nombre total de MCSs.

Preuve. Considérons une instance de **P1**, qui est également valable pour le problème **P2**. Tout d'abord, notons que le coût de notre algorithme pour cette instance est égal à $RA = \sum_u \varphi(\mathcal{N}_u, m_u)$. Soit RA_1^* le coût de la solution optimale pour **P1** et RA_2^* le coût de la solution optimale pour **P2**. Comme **P1** correspond à un système soumis à une contrainte supplémentaire, il est clair que $RA_1^* \leq RA_2^*$. De plus, puisque $\sum_u \sum_m \varphi_u(N_{u,m}, m)$ est le coût de **SubProblem_Algo** utilisé pour calculer une solution pour **P2**, alors :

$$\alpha RA_1^* \leq \sum_u \sum_m \varphi_u(N_{u,m}, m) \leq RA_2^*.$$

Soit u un utilisateur. Par définition de m_u nous avons :

$$\forall m, \forall u, \varphi_u \left(\bigcup_{m' \geq m} N_{u,m'}, m \right) \leq \varphi_u \left(\bigcup_{m' \geq m_u} N_{u,m'}, m_u \right). \quad (3.4.1)$$

De plus, pour un m fixé, comme φ est une fonction croissante par rapport à son premier paramètre, alors :

$$\forall u, \sum_m \varphi_u(N_{u,m}, m) \leq \sum_m \varphi_u \left(\bigcup_{m' \geq m} N_{u,m'}, m \right).$$

Ainsi, en sommant (3.4.1) sur toutes les valeurs possibles de m , on déduit que :

$$\forall u, \sum_m \varphi_u(N_{u,m}, m) \leq M \times \varphi_u \left(\bigcup_{m' \geq m_u} N_{u,m'}, m_u \right).$$

Par conséquent,

$$\begin{aligned} \alpha RA_1^* &\leq M \sum_u \varphi_u \left(\bigcup_{m' \geq m_u} N_{u,m'} \right) \\ &= M \sum_u \varphi_u(\mathcal{N}_u, m_u) \\ &\leq M \times RA, \end{aligned}$$

ce qui termine la preuve. \square

3.5 Troisième étape : extension

Après la sélection du MCS, quelques SUs pré-alloués peuvent être perdus si ils ne supportent pas le “meilleur” MCS. Par conséquent, l’Algorithme 3 peut mener à un gaspillage de ressources.

Pour remédier à ce problème, nous proposons une troisième routine qui est une extension de `Approx_Algo`. Le pseudo-code de cette extension est présentée dans l’Algorithme 4.

Dans cette dernière étape, chaque utilisateur est ordonnancé au moins une fois. Ainsi, l’algorithme maintient un ensemble \mathcal{U}^r d’utilisateurs qui n’ont pas encore été sélectionnés par cette routine. L’allocation $\{\mathcal{N}_u, m_u\}_{u \in \mathcal{N}_u}$ réalisée par l’algorithme précédent est tout d’abord récupérée. Nous notons \mathcal{N}^r l’ensemble des SUs non-alloués après l’exécution des deux premières routines. L’ensemble \mathcal{U}^r est initialisé avec tous les utilisateurs dont la valeur de leur fonction d’utilité peut être améliorée, i.e. :

$$\mathcal{U}^r \triangleq \{u \in \mathcal{U} \mid \exists X \subset \mathcal{N}^r, \exists m \in \mathcal{M}, \varphi_u(X \cup \mathcal{N}_u, m) > \varphi_u(\mathcal{N}_u, m_u)\}. \quad (3.5.1)$$

Soit $u \in \mathcal{U}^r$ un utilisateur et m un MCS. Pour chaque couple (u, m) , nous construisons $N_{u,m}$ ⁴ qui est l’ensemble des ressources de \mathcal{N}^r tel que $\forall k \in N_{u,m}$ nous avons $m_{u,k} \geq m$. Notons que dans ce cas, les ensembles $\{N_{u,m}\}$ ne sont pas nécessairement disjoints.

A chaque étape de l’Algorithme 4 et pour chaque utilisateur u , nous lui associons une métrique g_u appelée *gain par ressource* et calculée de la façon suivante :

$$g_u = \max_{m \leq m_u} \frac{\varphi_u(\mathcal{N}_u \cup N_{u,m}, m) - \varphi_u(\mathcal{N}_u, m_u)}{|N_{u,m}|}, \text{ avec } |N_{u,m}| \text{ le cardinal de } N_{u,m}.$$

Soit m'_u le MCS qui maximise l’expression ci-dessus. L’utilisateur pour lequel la valeur de g_u est la plus grande est appelé l’utilisateur *élu*. On affecte alors à ce dernier le MCS m'_u et on lui alloue l’ensemble N_{u,m'_u} en plus de \mathcal{N}_u . Les ensembles \mathcal{N}^r et $N_{u,m}(\forall u, m)$ sont ensuite reconstruits et la procédure se répète. Cette troisième routine se termine lorsque plus aucune allocation n’est bénéfique, i.e. $\forall u, g_u \leq 0$. Notons que cette étape améliore la deuxième routine sans modifier le facteur d’approximation de l’algorithme complet.

4. Nous avons fait évoluer ici la définition du symbole $N_{u,m}$ donnée à la section précédente, les deux définitions étant proches, ceci dans le souci de ne pas augmenter le nombre de symboles.

```

1  $\{\mathcal{N}_u, m_u\}_{u \in \mathcal{U}} \leftarrow \text{Approx\_Algo}()$ ;
2  $\mathcal{N}^r \leftarrow \mathcal{N} \setminus \bigcup_u \mathcal{N}_u$ 
3  $\mathcal{U}^r \leftarrow$  calculer l'ensemble des utilisateurs restants en appliquant Eq. (3.5.1)
4 tant que  $\mathcal{U}^r$  n'est pas vide faire
5   pour chaque utilisateur  $u \in \mathcal{U}$  faire
6     // Calculer l'ensemble des SUs qui peuvent être utilisés avec les MCS  $m$ .
7      $\forall m \leq m_u, N_{u,m} \leftarrow \{k \in \mathcal{N}^r \mid m_{u,k} \leq m\}$ 
8      $\forall m \leq m_u, g_u(m) \leftarrow \frac{\varphi(N_{u,m} \cup \mathcal{N}_u, m) - \varphi(\mathcal{N}_u, m_u)}{|N_{u,m}|}$ 
9      $g_u \leftarrow \max_m g_u(m)$ 
10     $m'_u \leftarrow \operatorname{argmax}_m g_u(m)$ 
11  fin
12   $v \leftarrow \operatorname{argmax}_u g_u$ 
13   $m_v \leftarrow m'_v$ 
14   $\mathcal{N}_u \leftarrow \mathcal{N}_v \cup N_{v,m_v}$ 
15   $\mathcal{N}^r \leftarrow \mathcal{N}^r \setminus \mathcal{N}_u$ 
16   $\mathcal{U}^r \leftarrow$  appliquer Eq. (3.5.1)
17 fin
18 retourner  $\{\mathcal{N}_u, m_u\}_{u \in \mathcal{U}}$ 

```

Algorithme 4 : Extension - Fin de l'allocation.

3.5.1 Un mot sur la complexité

Comme la réduction peut être effectuée en $\mathcal{O}(1)$, la complexité de la première routine dépend uniquement de l'algorithme utilisé pour résoudre l'instance de SWP. Afin de rester général nous notons $\mathcal{O}(\delta(U, M, K))$ la complexité de la première routine, avec δ une fonction polynomiale. La seconde routine cherche le “meilleur” MCS pour chaque utilisateur. De plus, la première routine est de nouveau appelée après chaque affectation de MCS. Cela mène à une complexité temporelle de $\mathcal{O}(U(\delta(U, M, N) + M))$. Dans la troisième routine, le nombre maximum de calculs de la métrique *gain par ressource*, g_u , est égal à $\sum_{i=1}^U i$. En effet, à chaque étape du dernier algorithme, un utilisateur est retiré de la liste des utilisateurs à ordonnancer. Pour chaque utilisateur, le gain par ressource est calculé en considérant les différents MCS possibles. De plus, les ensembles $N_{u,m}$ doivent être calculés également. La complexité de cette dernière routine (sans compter l'exécution des algorithmes précédents) est donc de $\mathcal{O}(U^2MN)$. Il en résulte que la complexité de l'algorithme complet est en $\mathcal{O}(U\delta(U, M, N) + U^2MN)$, ce qui est adapté à LTE.

Par ailleurs, lors de nos simulations réalisées sur une machine standard, nous avons pu vérifier que notre algorithme était suffisamment rapide pour résoudre une instance réaliste en une milliseconde.

3.6 Évaluation des performances

L'objectif de cette section est de comparer, par simulation, les performances de notre algorithme d'approximation (avec ou sans l'extension), à d'autres procédures

Paramètre	Valeur
Modèle du canal	Affaiblissement de propagation avec atténuation de Rayleigh
Distance de référence	1 d_{ref}
SNR vers CQI	EESM [78]
Mode de duplexage	Time Division Duplex (TDD)
Configuration du TDD	Configuration 5 [11]
Nombre d'utilisateurs	10
Distance initiale des utilisateurs	1 d_{ref}
Modèle de mobilité	Random Way Point
Zone de mobilité	Cercle de rayon 2 d_{ref}
Modèle de trafic	Infinite backlogs buffers (utilisateurs gloutons)

Tableau 3.1 – Paramètres de simulations.

d'allocation de ressources.

Expressions des fonctions d'utilité. Même si notre solution peut être utilisée pour répondre à une large variété d'objectifs, nous considérons uniquement des fonctions d'utilité qui suivent (1) la stratégie du MaxRate et (2) la stratégie du PF (*Proportional Fairness*). Dans ces deux stratégies, nos fonctions d'utilité peuvent s'exprimer comme $\varphi_u(\mathcal{N}_u, m_u) = w_u R(\mathcal{N}_u, m_u)$, avec $w_u = 1$ pour la politique du MaxRate et $0 < w_u < 1$ pour la politique du PF. Puisque pour tout MCS m , la fonction $R(., m)$ peut être considérée comme une fonction super-additive, l'expression des fonctions d'utilité qui correspondent à ces deux politiques sont également super-additives.

Nous avons implémenté notre propre module LTE sur le logiciel *Riverbed Modeler* [9] (précédemment connu sous le nom d'*Opnet Modeler*). Les principaux paramètres de simulations sont disponibles dans le Tableau 3.1.

3.6.1 Modélisation de la couche physique

Afin de pouvoir valider nos algorithmes dans des scénarios proches de la réalité, nous détaillons la modélisation de la couche physique. Plus précisément, cette section a pour but de répondre à la question suivante : “*Comment modéliser correctement une instance du problème ? I.e. comment déterminer les valeurs de la matrice $\mathcal{I} = (m_{u,k})$?*”

Cette modélisation ne sera pas répétée intégralement lors de la présentation des simulations dans les chapitres suivants. Pour davantage d'informations sur la partie RAN de LTE, voir l'Annexe A.

Généralités

Nous considérons un modèle de canal tel que chaque sous-porteuse soit sujet à un affaiblissement en espace libre et une atténuation multi-trajet de Rayleigh [79]. Dans nos simulations, chaque SU est composé de 3 RBs (Resource Blocks). Le SNR peut être estimé uniquement sur les sous-porteuses sur lesquelles des signaux de référence *cell-specific* sont présents. Pour rappel, un RB correspond à 12 sous-porteuses (Fig. A.2). Parmi ces 12 sous-porteuses, seulement 4 contiennent des signaux de référence *cell-specific* et donc seulement celles-ci peuvent être utilisées pour estimer le SNR. Nous

appelons *sous-porteuse de référence* une sous-porteuse sur laquelle un utilisateur peut estimer la qualité du canal.

Une fois que l'utilisateur a estimé le SNR de chacune des sous-porteuses de référence, cette qualité de canal peut être rapportée à la station de base en utilisant un unique entier (le *Channel Quality Indicator*) codé sur 4 bits par groupe de 6 RBs. **Par conséquent, un CQI représente la qualité de deux SUs.** La valeur d'un CQI représente le MCS qui peut être utilisé par la station de base pour transmettre sur les SUs considérés sans dépasser le taux d'erreur binaire cible. En d'autres termes, pour tout SU k d'indice pair, un seul CQI est utilisé pour signaler à la fois $m_{u,k}$ et $m_{u,k+1}$.

Dans nos simulations, la valeur du CQI est calculée en se basant sur les N_{rs} signaux de référence présents dans un groupe de 6 RBs. Avec nos hypothèses, comme il y a 4 signaux de référence par sous-porteuse et 12 sous-porteuses par RB, nous avons donc $N_{rs} = 24$.

Obtention des $m_{u,k}$

Afin de modéliser le canal, nous définissons une variable aléatoire $Y_{u,n}$ ($u \in \mathcal{U}$ et $n \in \{1, \dots, N_{rs}\}$) uniformément distribuée sur $[0; 1]$. Ainsi, la variable aléatoire $\gamma_{u,n}$ définie par :

$$\gamma_{u,n} = M_{u,n} \sqrt{\frac{-4 \ln(Y_{u,n})}{\pi}}$$

suit une distribution de Rayleigh avec pour espérance $M_{u,n}$. Dans les simulations, nous fixons $M_{u,n} = 1, \forall u, \forall n$. Sur la $n^{\text{ième}}$ sous-porteuse de référence, l'atténuation perçue par l'utilisateur u à une distance d_u de la station de base est modélisée par :

$$a_{u,n} = a_{ref} \times \left(\frac{d_{ref}}{d_u}\right)^{3.5} \times \gamma_{u,n}.$$

Nous introduisons ensuite une distance de référence d_{ref} pour laquelle le SNR doit être égal à 31 dB :

$$SNR_{ref} = 10 \log_{10} \left(\frac{P_{max} T_s}{N_0} \times a_{ref} \right) = 31 \text{ dB},$$

avec P_{max} la puissance maximale de transmission de la station de base, N_0 la densité spectrale de bruit et a_{ref} le facteur d'atténuation de référence (i.e. qui correspond à une distance d_{ref} entre un utilisateur et la station de base). La valeur de 31 dB mène à des tailles de cellules réalistes et fournit une bonne diversité lors de la sélection des MCS ($m_{u,k}$). En effet, nous avons observé que tous les MCS sont régulièrement sélectionnés par les utilisateurs, ce que nous pensons être proche de la réalité. En effet, si cette valeur était trop haute, le MCS le plus efficace serait toujours choisi par tous les utilisateurs (et inversement si la valeur du SNR de référence était trop petite).

En notant T_s la durée d'un symbole OFDM, le SNR instantané perçu par l'utilisateur u sur la sous-porteuse n ($SNR_{u,n}$) peut être calculé de la façon suivante :

$$\begin{aligned} SNR_{u,n} &= 10 \log_{10} \left(\frac{P_{max} T_s}{N_0} \times a_{u,n} \right) \\ &= SNR_{ref} + 10 \log_{10} \left(\left(\frac{d_{ref}}{d_u} \right)^{3.5} \times \gamma_{u,n} \right). \end{aligned}$$

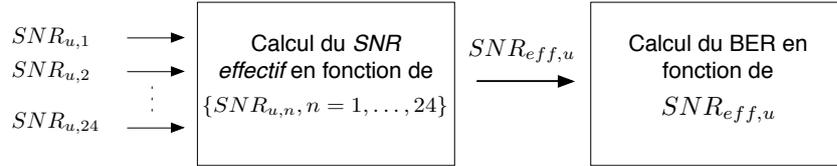


Fig. 3.3 – Principe d’EESM pour 2 SUs (6 RBs, 24 sous-porteuses de référence).

Indices des MCS	Modulation	Taux du code	β_m	SNR seuil (dB)
1	QPSK	0,3	1,48	2,90
2	QPSK	0,58	1,62	5,15
3	16QAM	0,36	3,10	7,12
4	16QAM	0,48	4,32	12,10
5	64QAM	0,45	7,71	15,85
6	64QAM	0,65	19,6	19,25

Tableau 3.2 – Valeurs des β_m et seuils de SNR pour chaque MCS.

Bien sûr, dans la réalité, le SNR de chaque sous-porteuse de référence est directement déduit des signaux de référence *cell-specific*.

Au niveau d’un utilisateur, nous avons donc les valeurs des SNRs de chaque sous-porteuse de référence. Il nous faut maintenant en déduire les CQIs qui seront envoyés à la station de base.

Il a été montré dans [78] que la moyenne des SNRs sur toutes les sous-porteuses de référence ne donne pas une bonne prédiction du BER. Dans nos simulations, nous avons utilisé une méthode appelée EESM (*Exponential Effective S(I)NR Mapping*) [78, 80, 81, 82, 83]. Etant donné un MCS et un ensemble de valeurs de SNR, cette technique permet de déduire une seule et unique valeur, le *SNR effectif* ($SNR_{eff,u}$ pour l’utilisateur u). Le SNR effectif peut être utilisé pour obtenir une estimation fiable du SNR sur un ensemble d’estimations. Après avoir calculé le SNR effectif, il est aisé de trouver le MCS le plus adapté pour le groupe de sous-porteuses associées à ces estimations.

Comme un seul CQI peut être envoyé pour deux SUs adjacents, nous considérons donc une paire de ces SUs. Le SNR effectif associé à ces deux SUs est calculé comme suit :

$$SNR_{eff,u}(\beta_m) = -\beta_m \ln \left(\frac{1}{N_{rs}} \sum_{n=1}^{N_{rs}} \exp \left(-\frac{SNR_{u,n}}{\beta_m} \right) \right)$$

avec β_m un paramètre qui dépend du MCS utilisé. Dans nos simulations, nous considérons 6 MCS différents. Nous avons considéré les paramètres EESM issus de [82] et résumés dans le Tableau 3.2. Ce tableau donne également le seuil à partir duquel chaque MCS peut être utilisé pour un BER cible inférieur à 10^{-3} .

Nous procédons ensuite à la sélection du MCS comme illustré sur la Figure 3.4. Au début du processus de sélection, le MCS le plus efficace spectralement (MCS_6) est choisi. L’utilisateur calcule ensuite le SNR effectif correspondant à ce MCS. Si cette

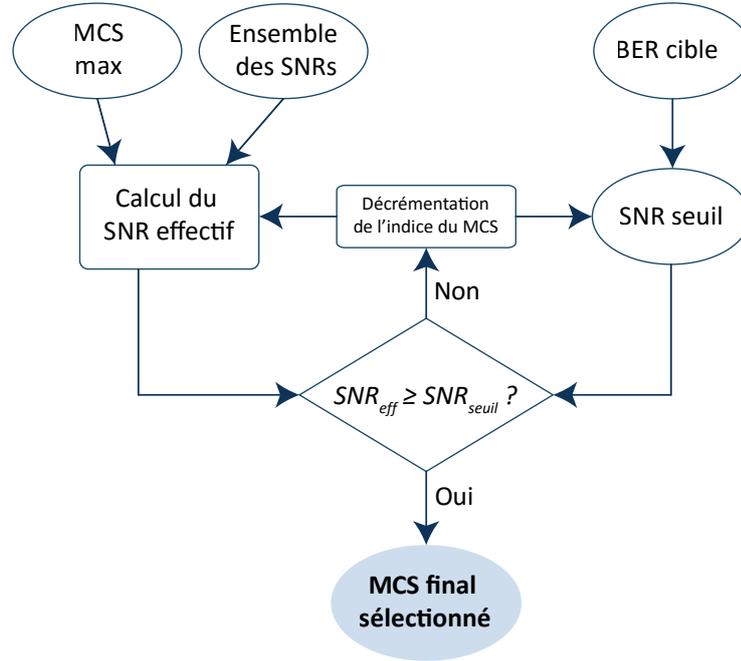


Fig. 3.4 – Processus de sélection du MCS pour une paire de SUs.

valeur est supérieure au seuil associé (19,25 dB pour MCS_6), ce MCS est sélectionné pour transmettre des données sur les deux SUs considérés. Si le SNR effectif est inférieur au seuil, l’opération est répétée avec le deuxième MCS le plus efficace, etc.

Le MCS retenu pour les SUs k et $k + 1$ correspond au $m_{u,k}$ et $m_{u,k+1}$ de l’instance du problème. L’indice de ce MCS correspond au CQI envoyé à la station de base pour ces deux SUs.

Adaptation des ordonnanceurs

Notre algorithme complet (*approx + ext*) est comparé avec :

- *approx*, le même algorithme d’approximation sans son extension (afin d’évaluer le gain de l’extension) ;
- l’*algorithme classique* [12], qui n’optimise pas l’allocation de ressources en fonction de la contrainte du single MCS et du calcul non-linéaire des débits, même si nous avons fait en sorte que ces spécificités soient respectées ;
- USA (*Unified Scheduling Algorithm*) détaillé dans [55], qui considère la contrainte du single MCS mais pas le calcul non-linéaire des débits ;
- l’*algorithme optimal* qui calcule une solution optimale.

Nous rappelons que le débit instantané de l’utilisateur u est noté $R(\mathcal{N}_u, m_u)$ avec \mathcal{N}_u le nombre de SUs alloués à l’utilisateur u et m_u le MCS qui lui a été assigné. L’algorithme classique et USA sont basés sur des métriques “par-SU” puisque les fonctions d’utilité sont supposées additives. Nous avons adapté ces algorithmes afin de calculer la métrique $\lambda_{u,k}$ de l’utilisateur u sur le SU k comme $\lambda_{u,k} = w_u \times R(\{k\}, m_{u,k})$, avec w_u qui dépend de la stratégie utilisée.

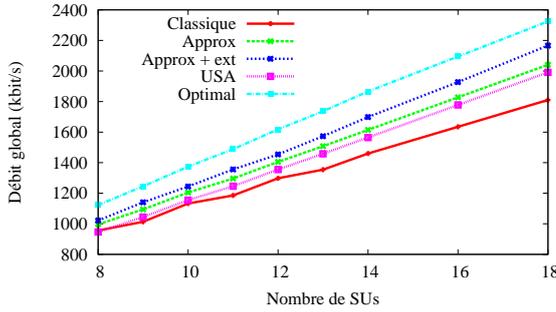


Fig. 3.5 – Capacité avec MaxRate.

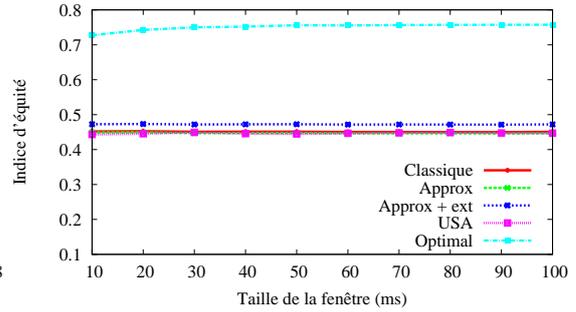


Fig. 3.6 – Équité avec MaxRate.

De plus, comme notre problème (tel que formulé) est non-linéaire, nous n'avons pas pu recourir à un logiciel dédié à la résolution des programmes linéaires en nombres entiers (Integer Linear Program solver) afin de trouver la solution optimale. Par conséquent, l'allocation optimale demande beaucoup de temps de calcul (complexité en $\mathcal{O}(U^N)$). De ce fait, le nombre d'utilisateurs est volontairement limité.

Afin d'évaluer l'équité entre les utilisateurs nous utilisons l'indice de Jain :

$$F(\Delta t) = \frac{(\sum_{u \in \mathcal{U}} \bar{r}_u(\Delta t))^2}{U \cdot \sum_{u \in \mathcal{U}} (\bar{r}_u(\Delta t))^2}, \quad (3.6.1)$$

avec $\bar{r}_u(\Delta t)$ le débit de l'utilisateur u obtenu sur une fenêtre de durée Δt . Cet indice varie de $1/U$ (faible équité) à 1 (équité parfaite).

Scenario 1 : stratégie du MaxRate

Dans ce premier scenario, nous souhaitons maximiser la capacité du système, i.e. nous exprimons les fonctions d'utilité de cette façon : $\varphi(n_u, m_u) = R(\mathcal{N}_u, m_u)$ pour tous les utilisateurs u et pour un *time slot* donné.

La Figure 3.5 montre la charge moyenne du système en fonction du nombre de SUs (i.e. la largeur de bande). Comme prévu, les courbes croissent linéairement avec le nombre de SUs. Notre solution réalise les meilleures performances parmi les algorithmes sous-optimaux. De plus, notre algorithme avec son extension permet d'obtenir des résultats encore supérieurs : la pente de la courbe correspondante est proche de la pente de la courbe optimale. De plus, l'écart entre notre algorithme avec extension et sans extension semble augmenter avec le nombre de SUs. En effet, plus le nombre de ressources augmente plus la probabilité d'avoir des ressources non-allouées (avec l'Algorithme 3) augmente également. Par conséquent, les bénéfices de notre extension deviennent de plus en plus importants. La Figure 3.6 compare l'indice de Jain par rapport à la taille de la fenêtre Δt . Comme l'équité n'est pas l'objectif ciblé par la stratégie du MaxRate, nous pouvons remarquer que les indices d'équité sont constants et plutôt faibles pour tous les algorithmes (sauf pour la solution optimale).

Scénario 2 : stratégie du PF

Nous cherchons maintenant à obtenir une allocation de ressources plus équitable sans compromettre totalement le débit des utilisateurs. Pour cela, nous utilisons une version modifiée du PF. Dans les systèmes traditionnels (non-FDPS et sans la contrainte du single MCS), l'objectif des ordonnanceurs PF est de maximiser $\sum_{u \in \mathcal{U}} \log R_u$, avec R_u le débit sur le long terme de l'utilisateur u . Dans [36], il est démontré que maximiser cet objectif est équivalent à maximiser $\sum_u r_u(t)/R_u(t)$, avec $r_u(t)$ le débit de l'utilisateur u lors du $t^{\text{ième}}$ TTI et $R_u(t)$ le *débit moyen par TTI* de u .

$$R_u(t+1) = \begin{cases} (1-\beta)R_u(t) + \beta r_u(t) & \text{si } u \text{ est ordonnancé} \\ & \text{sur le time slot } t, \\ (1-\beta)R_u(t) & \text{sinon,} \end{cases}$$

avec $0 \leq \beta \leq 1$.

Notre dernière contribution dans ce chapitre est d'étendre la stratégie PF en prenant en compte la contrainte du single MCS et les spécificités du FDPS. Pour cela, nous définissons $\mathcal{N}_u(t)$ l'ensemble des ressources allouées à l'utilisateur u lors du TTI t et $n_u(t) = |\mathcal{N}_u(t)|$ le cardinal de l'ensemble $\mathcal{N}_u(t)$. Nous commençons par redéfinir $R_u(t)$ comme le *débit moyen de l'utilisateur* u :

$$R_u(t+1) = \begin{cases} (1-\beta)R_u(t) + \beta R(\mathcal{N}_u(t), m_u(t))/n_u(t) & \text{si } \mathcal{N}_u(t) \neq \emptyset, \\ (1-\beta)R_u(t) & \text{sinon,} \end{cases}$$

avec β un facteur de configuration généralement fixé à 10^{-4} [36, 54]. Enfin, nous exprimons la stratégie du SM-PF (*Single MCS Proportional Fair*) en définissant la fonction d'utilité suivante :

$$\varphi_u(\mathcal{N}_u, m_u) = \frac{R(\mathcal{N}_u(t), m_u(t))}{n_u(t) \times R_u(t)}.$$

Cette fonction d'utilité mène à un bon compromis entre équité et capacité en favorisant l'allocation des ressources aux utilisateurs dont le débit résultant sera supérieur à leur débit moyen.

La Figure 3.7 montre la capacité moyenne en fonction du nombre de ressources tandis que la Figure 3.8 montre l'indice d'équité en fonction de la taille de fenêtre Δt . Le débit moyen obtenu avec les autres algorithmes sous-optimaux est clairement plus faible que celui obtenu avec l'algorithme que nous proposons. Bien sûr, l'algorithme optimal obtient le meilleur débit et la meilleure équité. L'indice de Jain de notre solution, avec extension ou sans son extension, est tout même très bon (environ 0,94). Les raisons pour lesquelles notre algorithme obtient la plus haute capacité tout en maintenant une excellente équité (comparé aux autres algorithmes sous-optimaux) sont que notre solution et la version modifiée du PF traitent (1) la contrainte du single MCS et (2) la non-linéarité des débits.

La Figure 3.9 illustre l'efficacité spectrale moyenne en fonction du nombre de SUs. Une nouvelle fois, notre algorithme en trois étapes alloue les SUs d'une manière plus efficace comparé aux autres algorithmes sous-optimaux. Dans cette figure, les deux courbes sont décroissantes ce qui peut être surprenant puisque les fonctions d'utilité sont super-additives. Cependant, plus le nombre de ressources est important plus la

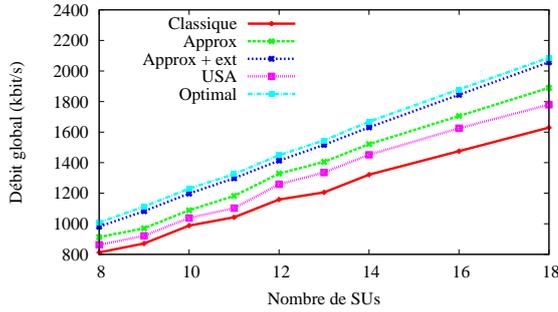


Fig. 3.7 – Capacité avec le PF.

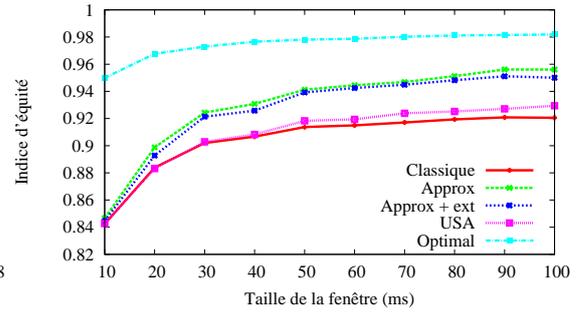


Fig. 3.8 – Indice d'équité avec le PF.

probabilité d'allouer des SUs de faible qualité est grande. Par conséquent, le *single MCS* utilisé par chaque utilisateur est globalement moins efficace comparé au cas où seulement quelques SUs sont disponibles. Cependant, notons que notre algorithme limite cet effet et reste proche de l'efficacité spectrale optimale (avec la stratégie du PF).

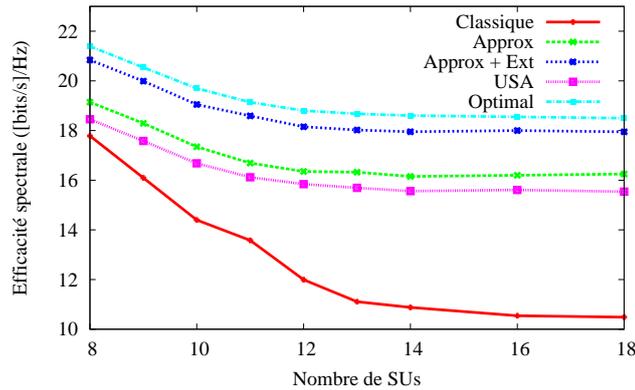


Fig. 3.9 – Efficacité spectrale avec le PF.

Passage à l'échelle. Enfin, nous considérons une instance de 50 utilisateurs. Figure 3.10 et Figure 3.11 comparent respectivement la capacité moyenne et l'équité des utilisateurs entre notre algorithme d'approximation et USA. De plus, nous avons étudié les performances jusqu'à 26 SUs ce qui est proche du nombre maximal de ressources disponibles dans un réseau LTE. L'algorithme optimal n'est plus utilisé comme référence car le nombre d'utilisateurs est maintenant trop important et empêche de calculer une solution en temps raisonnable⁵. On remarque que notre algorithme est une nouvelle fois supérieur à USA et croît toujours linéairement en débit. Cela confirme que notre proposition passe à l'échelle.

5. L'algorithme classique, étant inférieur à USA, n'a pas été comparé également.

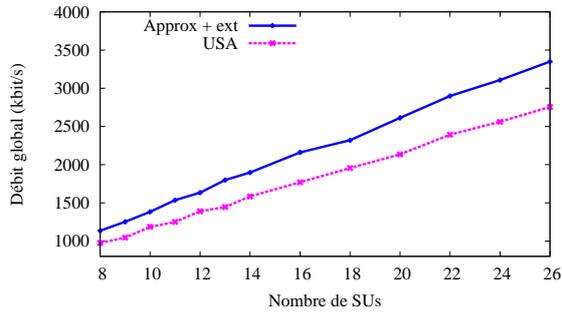


Fig. 3.10 – Capacité avec le PF (50 utilisateurs).

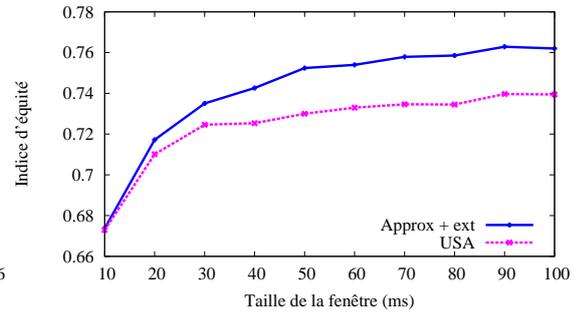


Fig. 3.11 – Indice d'équité avec le PF (50 utilisateurs et 18 SUs).

3.7 Conclusion

Dans ce chapitre, nous avons étudié un problème majeur des réseaux cellulaires : l'allocation de ressources sous les contraintes de la norme 3GPP LTE. Nous avons montré que le problème est NP-Difficile et par conséquent ne peut pas être résolu en temps raisonnable. Puis, nous avons proposé un algorithme d'approximation générique qui permet de calculer une solution au problème. Nos simulations montrent que notre solution permet d'obtenir une très bonne équité entre les utilisateurs tout en maintenant une capacité proche de la solution optimale. Par ce biais, nous avons montré que l'intégration au coeur de l'algorithme d'allocation des spécificités de LTE-B est essentiel pour l'obtention de résultats performants.

Discussion. Une limitation de notre système est que nous avons considéré un modèle de trafic irréaliste (utilisateurs gourmands). Cette hypothèse ne nous permet pas de poursuivre un objectif qui intègre le délai des paquets. Afin de traiter des trafics multimédias et fournir de la Qualité de Service aux utilisateurs nous allons maintenant relâcher cette hypothèse dans le chapitre suivant.

Chapitre 4

Allocation de ressources avec Trafics Multimédias

Dans ce chapitre, nous considérons que les utilisateurs reçoivent des données issues de trafics multimédias. Chacun des flux de trafics est associé à un *buffer* ou *buffer virtuel* d'émission implémenté au niveau de la station de base. Les paquets situés dans ces buffers sont en attente d'être transmis vers l'utilisateur destinataire.

Les buffers sont donc alimentés par des flux de données multimédias. Par conséquent, un buffer peut être occasionnellement vide de telle façon qu'il n'est pas utile d'allouer des ressources à ce flux. De ce fait, en considérant des trafics multimédias, le problème est encore plus complexe. En outre, nous supposons que chaque flux peut avoir différents besoins en termes de qualité de service. En effet, l'état d'un buffer n'est pas facilement prédictible puisqu'il dépend de nombreux facteurs :

- des propriétés statistiques des trafics à destination de l'utilisateur ;
- de la qualité du canal sur les différentes fréquences ;
- des allocations réalisées lors des TTI précédents.

Nous ne prenons aucune hypothèse restrictive sur chacun de ces trois points.

Contributions du chapitre. Les trafics multimédias réalistes présentent un débit variable qui doit être pris en compte sur une échelle de temps très courte afin d'assurer un niveau satisfaisant de QoS aux utilisateurs. Pour évaluer et quantifier la QoS offerte à un utilisateur, nous analysons tout d'abord une métrique de qualité de service, le *Packet Delay Outage Ratio*, qui peut être utilisé comme un outil de mesure fiable de la satisfaction d'un utilisateur à travers le temps.

Nous proposons ensuite AGSS une nouvelle heuristique d'allocation de ressources (*Adaptive and Generic Scheduling Scheme*). Cet algorithme adapte certaines idées énoncées dans le chapitre précédent afin d'offrir une allocation efficace. De ce fait, AGSS est également compatible avec les spécificités de la norme 3GPP LTE.

Cependant, AGSS répond à des critères différents car il permet :

- de prendre en compte les flux multimédias, et l'état des buffers ;
- de prendre en compte plusieurs flux destinés à un même utilisateur ;
- d'adresser des objectifs sensibles aux délais (*QoS aware scheduling*) ;
- d'élaborer une allocation très rapidement (complexité similaire à celle de l'algorithme classique) en se basant sur des listes triées.

Symbole	Signification
F	Nombre de flux de données dans le système
\mathcal{F}	Ensemble des flux du système
\mathcal{F}_u	Ensemble des flux appartenant à l'utilisateur $u \in \mathcal{U}$
R_f	Débit du flux f au cours de l'exécution d'AGSS
r_f	Gain (en termes de débit) obtenu par le flux f au cours de l'exécution d'AGSS
BO_f	PDOR cyclique du flux f
$PDOR_f$	Débit instantané nécessaire pour vider le buffer du flux f
$PDOR_f^{target}$	PDOR cible au delà duquel le flux f n'est plus satisfait
$PDOR_f^{cycle}$	PDOR cyclique du flux f

Tableau 4.1 – Notations additionnelles.

Enfin, nous proposons OPA (*Opportunistic PDOR Aware strategy*) notre propre stratégie d'allocation de ressources. Celle-ci est conçue pour réagir rapidement aux fluctuations de l'état du système. La construction de cette stratégie est basée sur le PDOR afin que l'allocation s'adapte aux besoins réels des applications utilisateurs.

Les résultats de simulations montrent qu'AGSS offre des performances bien supérieures à celles obtenues avec l'algorithme classique (pour une stratégie donnée). De plus, OPA obtient le plus haut taux d'utilisateurs satisfaits quelle que soit la charge du système.

Notations relatives aux flux. Plusieurs flux peuvent être destinés à un même utilisateur. Afin d'intégrer ceux-ci à notre modèle, nous définissons l'ensemble des flux actifs du système par $\mathcal{F} = \{f_1, \dots, f_F\}$; avec F le nombre total de flux. De plus, nous notons \mathcal{F}_u l'ensemble des flux à destination de l'utilisateur u . À noter que nous avons les relations suivantes :

$$\forall u, \mathcal{F}_u \subset \mathcal{F} \quad \text{et} \quad \mathcal{F} = \bigcup_u \mathcal{F}_u.$$

Les notations additionnelles du chapitre sont résumées dans le Tableau 4.1. Comme les applications recherchent un bon niveau de qualité de service sur le long terme, toutes les variables dépendent du TTI $t \in \mathbb{N}$ considéré. Lorsque nous souhaitons faire référence à une variable v durant un TTI t particulier, nous notons la variable $v(t)$. Cependant, lorsque qu'une variable fait référence au TTI courant, et si aucune confusion n'est possible, nous ignorons cette précision par souci de simplification.

4.1 Vers la quantification de la Qualité de Service

Tout d'abord, nous définissons un *profil de QoS* spécifique à un flux. Un profil est défini par trois critères :

- Le BER (*Bit Error Rate* ou Taux d'Erreur Binaire) maximum supporté par le flux. Dans ce chapitre, tout comme le précédent, nous supposons que transmettre avec un MCS adapté¹ est suffisant pour respecter cette contrainte.

1. Pour chaque utilisateur u , nous considérons qu'un MCS m est adapté si $m \geq m_{u,k}$ pour tout k alloué à u .

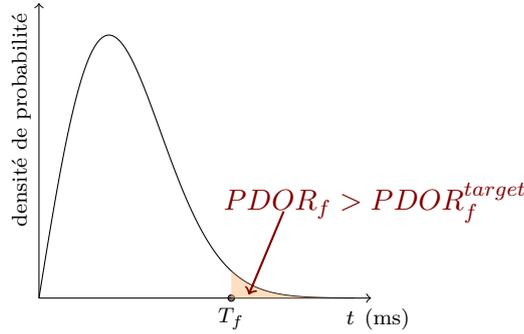


Fig. 4.1 – Exemple de distribution du délai des paquets.

- Le délai maximum toléré par les paquets du flux. Nous notons T_f cette borne (pour le flux f).
- Le PDOR cible, qui est défini ci-dessous.

Définition 3. Soit $\mathcal{P}_f(t)$ l'ensemble des paquets générés par la source du flux f durant l'intervalle $\llbracket 0, t-1 \rrbracket$. Pour chaque paquet $p \in \mathcal{P}_f(t)$, nous notons $d_p^f(t)$ le temps attendu dans le buffer d'émission par le $p^{\text{ième}}$ paquet du flux f au début du t -ième TTI. Pour finir, nous notons $D_f(t) = \{p \mid d_p^f(t) > T_f\}$, l'ensemble des paquets hors-délai du flux f . Le PDOR de ce flux est alors défini par :

$$PDOR_f(t) = \frac{|D_f(t)|}{|\mathcal{P}_f(t)|}.$$

Ainsi, le PDOR correspond au nombre de paquets hors-délais divisé par le nombre total de paquets générés. Le seuil à partir duquel le flux (et donc l'utilisateur) est considéré comme insatisfait est appelé le *PDOR cible*. Le PDOR cible du flux f est noté $PDOR_f^{target}$. Il est donc considéré qu'un flux f est satisfait lorsque la condition suivante est vérifiée :

$$PDOR_f(t) \leq PDOR_f^{target}, \forall t. \quad (4.1.1)$$

La Figure 4.1 donne un exemple de la densité de probabilité du délai des paquets pour un flux. Le PDOR du flux f correspond à l'aire orangée sur cette figure. Afin de répondre aux besoins des flux, cette aire doit être inférieure au PDOR cible.

Comparée à une contrainte forte sur les délais (i.e. une contrainte stipulant que tous les paquets doivent être transmis avant l'échéance), notre contrainte sur le PDOR est plus souple. En effet, il est souvent acceptable que quelques paquets soient transmis avec du retard si la plupart des paquets sont délivrés à temps. Par exemple, pour une application de type vidéo-conférence, la perte de quelques paquets n'affecte pas l'expérience de l'utilisateur [84].

De plus, comme notre contrainte est plus souple, l'allocation de ressources peut être réalisée de manière plus flexible. Notons que l'Equation 4.1.1 englobe également la contrainte forte sur les délais si on assigne au PDOR cible une valeur de zéro. Le seuil T_f peut également être paramétré pour s'adapter au degré de sensibilité des trafics par rapport au délai maximum supporté.

Enfin, le PDOR permet de limiter d'autres métriques de qualité de service telles que la gigue et le temps d'attente moyen des paquets dans le buffer. Nous donnons un exemple afin de donner une justification intuitive à cette propriété.

Exemple. Soit f un flux utilisateur. Nous supposons dans cet exemple que le délai des paquets est distribué selon une distribution de Rayleigh (comme illustré par la Figure 4.1) avec pour espérance μ et paramètre σ . Si nous supposons également que le taux de paquets hors-délai est inférieur à $PDOR_f^{target}$, alors l'inégalité suivante est vérifiée :

$$\int_{T_f}^{+\infty} \frac{t}{\sigma^2} \exp\left(\frac{-t^2}{2\sigma^2}\right) dt \leq PDOR_f^{target} \quad (4.1.2)$$

$$\Leftrightarrow \exp\left(\frac{-T_f^2}{2\sigma^2}\right) \leq PDOR_f^{target} \quad (4.1.3)$$

$$\Leftrightarrow \sigma \leq \sqrt{\frac{T_f^2}{-2 \ln(PDOR_f^{target})}}. \quad (4.1.4)$$

Comme l'écart type σ' d'une distribution de Rayleigh est égal à $\sigma\sqrt{\left(\frac{4-\pi}{2}\right)}$, alors :

$$\sigma' \leq T_f \sqrt{\frac{4-\pi}{-4 \ln(PDOR_f^{target})}}. \quad (4.1.5)$$

Ainsi, l'écart type (i.e. la gigue) est borné par une fonction du PDOR cible. Selon (4.1.5), plus le PDOR cible est faible, plus la gigue sera faible. Par conséquent, si la gigue est une métrique majeure pour un type de flux donné, il est possible de limiter sa valeur en paramétrant le PDOR cible avec une valeur adaptée.

Le raisonnement ci-dessus peut être étendu à d'autres distributions. Si la distribution n'est pas connue (comme nous le supposons dans ce chapitre), il est possible de tester empiriquement différentes valeurs du PDOR cible.

4.1.1 Objectif

L'objectif TFDPS. Dans ce chapitre, nous souhaitons traiter des objectifs permettant de fournir de la QoS à travers le temps. Avec un objectif tel que celui présenté dans la section précédente (Objectif 3.1.1), nous pouvons remarquer que l'allocation de ressources effectuée au TTI t aura un impact sur l'état du système aux TTIs $t+1$, $t+2$, etc... Ainsi, même une succession d'allocations optimales peut mener à de faibles performances globales. De plus, avec des trafics multimédias, cela n'a pas vraiment de sens de considérer le problème sur un seul TTI, puisque les débits de ces trafics varient avec le temps. Nous souhaitons donc intégrer le temps à la formulation du problème.

Pour cela, nous commençons par définir une *période de temps*.

Définition 4. Soit $n \in \mathbb{N}$. Une période de temps est une séquence de TTIs : $n+1, n+2, \dots, n+T$ avec $T \in \mathbb{N}$ la taille de la période.

Nous modifions également la définition d'une fonction d'utilité afin d'intégrer les flux au problème. Ainsi, nous notons $\varphi_f(R_f(t) | \mathcal{N}_u(t), m_u(t))$ la valeur de la fonction d'utilité pour le flux f au TTI t , avec :

- $\mathcal{N}_u(t)$ l'ensemble des ressources allouées à l'utilisateur u (u étant le propriétaire du flux),
- $m_u(t)$ le *single MCS* de l'utilisateur u ,
- $R_f(t)$ le débit accordé au flux f .

En effet, la contrainte du *single MCS* s'applique pour toutes les ressources allouées à un utilisateur (pas uniquement sur les ressources allouées à un flux). En d'autres termes, pour tous les flux de l'utilisateur u , le même MCS est utilisé sur un TTI. D'autre part, nous faisons apparaître explicitement le paramètre R_f dans la fonction d'utilité car une nouvelle contrainte s'applique aux débits des flux. Effectivement, la non-linéarité du calcul du débit rend le paramètre R_f dépendant des allocations réalisées pour tous les autres flux de l'utilisateur. C'est à l'ordonnanceur de partager le débit de l'utilisateur u , toujours noté $R(\mathcal{N}_u, m_u)$, entre tous les flux appartenant à u .

Nous appelons ce nouveau problème le LTE UL TFDPS (*Joint Time and Frequency Packet Scheduling*). Pour chaque utilisateur u et pour chaque TTI t d'une période de taille T , nous cherchons à trouver une allocation $(\mathcal{N}_u(t), m_u(t))$ et à calculer en conséquence les débits des flux $\{R_f(t)\}_f$ de façon à :

$$\text{Maximiser} \quad \sum_{t=n+1}^{n+T} \sum_{u \in \mathcal{U}} \sum_{f \in \mathcal{F}_u} \varphi_f(R_f(t) | \mathcal{N}_u(t), m_u(t)), \quad (4.1.6)$$

sujet à :

$$\sum_{f \in \mathcal{F}_u} R_f(t) \leq R(\mathcal{N}_u(t), m_u(t)), \forall u, \forall t \quad (4.1.7)$$

et les contraintes habituelles de LTE (3.1.1) appliquées à chaque TTI.

Objectif principal. Pour des raisons de concision, nous nous concentrons plus spécifiquement sur un objectif permettant de satisfaire les utilisateurs. Afin d'assurer la qualité de service, nous exprimons la fonction d'utilité de l'utilisateur u de la façon suivante (pour le TTI t) :

$$\varphi_f(R_f(t) | \mathcal{N}_u(t), m_u(t)) = \begin{cases} 1 & \text{si } PDOR_f(t) \leq PDOR_f^{target}. \\ 0 & \text{sinon.} \end{cases}$$

L'objectif est donc de maximiser le nombre de flux satisfaits (et par conséquent les utilisateurs) :

$$\text{maximiser Card} \left\{ (f, t) \mid PDOR_f(t) \leq PDOR_f^{target}, \forall t \right\}.$$

Bien sûr, cet objectif est uniquement cohérent si le débit total des flux est inférieur à la capacité du système. De plus, l'objectif dépend évidemment des débits offerts aux utilisateurs puisqu'un utilisateur satisfait est un utilisateur qui reçoit des données.

4.2 AGSS : Schéma d’ordonnement générique et adaptatif

Le problème présenté ci-dessus est extrêmement difficile à résoudre de manière optimale puisque de nombreuses informations ne sont pas connues. De plus, nous supposons que notre système est soumis à des modèles de trafics réalistes avec des besoins différents en termes de qualité de service. Ces éléments sont difficilement exploitables dans un modèle analytique autosuffisant.

Dans cette section, nous proposons une heuristique appelée AGSS (*Adaptive and Generic Scheduling Scheme*). Cet algorithme fournit une méthodologie générale pour allouer des ressources dans un contexte LTE. La combinaison d’AGSS et d’une stratégie d’ordonnement permet de définir complètement un ordonnanceur et d’adresser les objectifs présentés dans la section précédente.

Calcul du débit. Nous souhaitons également prendre en compte le fait que les buffers d’émission associés aux flux puissent être vides, ou très peu occupés. Soit BO_f (*Buffer Occupancy*), le débit nécessaire pour vider le buffer du flux f (en bit/s) sur un TTI. Notre solution s’assure que le débit accordé à chaque flux est borné par BO_f .

4.2.1 Préliminaires : un point sur les stratégies d’ordonnement

Les fonctions d’utilité sont nécessaires pour définir un objectif, et participent à la résolution des problèmes de type FDPS (Chapitre 3). Cependant, lorsque le temps est une donnée du problème, l’implémentation de ces fonctions dans un algorithme d’allocation de ressources peut être inadaptée. Par exemple, les fonctions d’utilité de l’Objectif 4.1.1 ne donnent pas une quantification suffisamment précise de l’état du système pour mesurer la qualité d’une allocation. En effet, si la valeur de la fonction d’utilité d’un utilisateur est égale à zéro, nous pouvons remarquer que cette seule information ne permet pas de savoir si un utilisateur est proche d’être satisfait ou est encore très éloigné du seuil de satisfaction.

Avantages de l’algorithme classique. Les stratégies “sensibles aux délais” du Chapitre 2 (Section 2.2.1) sont efficaces sur le long terme. En effet, historiquement, les stratégies d’ordonnement ont été proposées pour une méthode d’accès de type TDMA (*Time Division Multiple Access*), i.e. les ressources sont découpées uniquement dans le domaine temporel. L’algorithme classique est une solution naturelle pour la réalisation de ce type d’allocation en FDPS puisque les ressources sont allouées indépendamment les unes des autres (émulant ainsi un système composé de *time slots*). Sur chacune des ressources et pour chaque utilisateur, un poids est calculé. Chaque ressource est allouée à l’utilisateur qui obtient le poids le plus important sur cette dernière. C’est le calcul du poids qui définit une *stratégie d’ordonnement*.

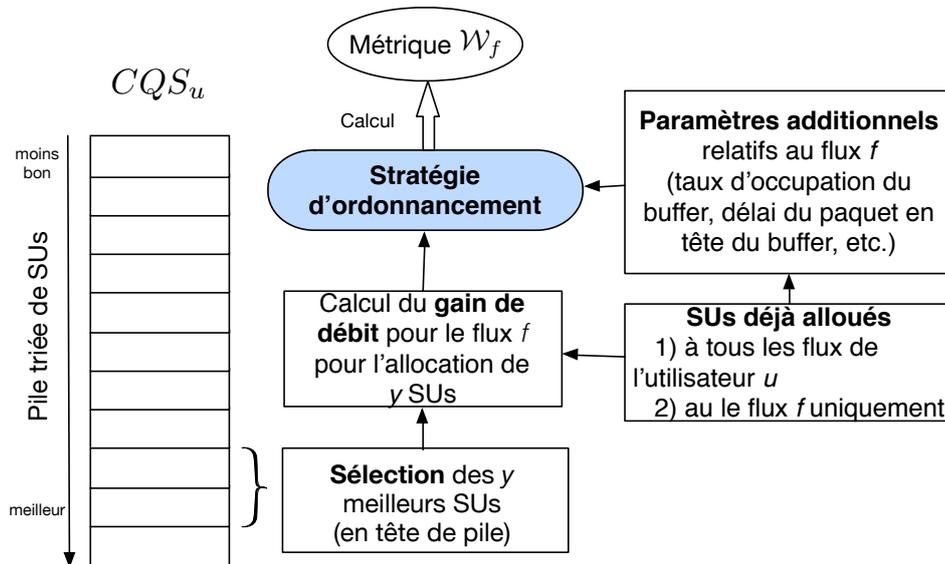
Stratégies d’ordonnement avec AGSS. De façon plus générale, nous avons remarqué que, pour rendre les stratégies d’ordonnement efficaces, l’allocation doit être réalisée de la façon la plus fine possible². En effet, lors de nos campagnes de simulations, nous avons remarqué qu’une allocation avec une granularité fine obtient de

2. Une allocation est fine, si à chaque itération de l’algorithme considéré, le nombre de ressources allouées est faible. Dans sa forme originale, l’algorithme classique (Algorithme 1) a une granularité

bien meilleures performances dans un contexte multimédia. Avec AGSS, notre objectif est de construire un algorithme qui alloue finement les ressources tout en étant adapté aux contraintes de LTE. Par ce biais, nous pouvons intégrer les stratégies d'ordonnancement, ce qui permet de conserver un héritage précieux de la théorie de l'allocation de ressources. Tout comme les fonctions d'utilité, une stratégie d'ordonnancement peut utiliser toutes les informations disponibles au niveau de la station de base. Nous traitons en Section 4.3 les détails des calculs de poids pour AGSS, car les stratégies nécessitent d'être adaptées à notre algorithme.

4.2.2 Présentation d'AGSS

Nous considérons que la matrice $\mathcal{I} = (m_{u,k})_{u,k}$ est mise-à-jour régulièrement³. À chaque mise-à-jour et pour chaque utilisateur, la station de base va trier les SUs en ordre de qualité décroissante. Le résultat de ce tri est placé dans une pile appelée *Channel Quality Stack* et notée CQS_u pour l'utilisateur u .



[h]

Fig. 4.2 – Prise en charge des flux avec AGSS. Dans cette figure, le flux f appartient à l'utilisateur u .

Principe d'AGSS. En substance, AGSS compare les poids calculés selon une stratégie d'ordonnancement sur chacun des SUs en tête des piles. Le résultat de cette comparaison (le meilleur SU parmi les meilleurs SUs) est alors alloué à l'utilisateur correspondant et supprimé de toutes les piles. La plupart du temps, ce processus est

d'allocation très fine puisqu'à chaque itération, une seule ressource est allouée. L'algorithme présenté dans le Chapitre 3 n'effectue pas des allocations fines puisqu'il évalue les allocations par *ensemble de ressources*. C'est pourquoi cet algorithme n'est pas adapté aux stratégies d'ordonnancement.

3. La fréquence de mise-à-jour de cette matrice dépend de la configuration TDD choisie, puisque les rapports sur l'état du canal sont transmis sur les sous-trames dédiées au sens montant (voir Annexe A).

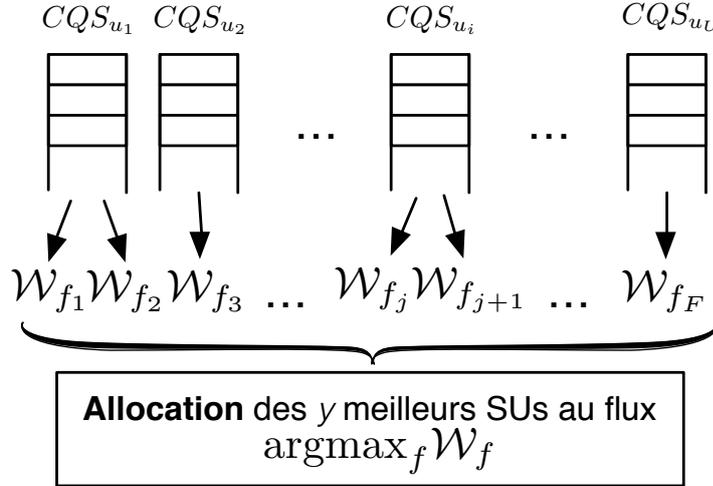


Fig. 4.3 – Processus d’élection. Un même utilisateur peut posséder plusieurs flux.

répété (1) jusqu’à ce qu’il n’y ait plus aucun SU à allouer ou (2) jusqu’à ce que les buffers soient vides. Idéalement, les SUs sont alloués un par un. Cependant, une conséquence de la Propriété 1 est qu’il n’est pas toujours bénéfique d’allouer un SU supplémentaire à un utilisateur, même si il s’agit du “meilleur” SU. En considérant cela, notre heuristique fournit également la possibilité d’allouer les SUs par groupe (au lieu d’un seul SU) afin de s’assurer que l’allocation soit bénéfique.

Initialisation. Soit \mathcal{N}^r l’ensemble des ressources qu’il reste à allouer aux utilisateurs. Au début de l’exécution, $\mathcal{N}^r = \mathcal{N}$. La phase d’initialisation (Algorithme 5) met à zéro toutes les variables utilisées par AGSS : \mathcal{N}_u , m_u et R_f ($\forall u, f$).

- 1 initialisation;
- 2 $\mathcal{N}^r \leftarrow \mathcal{N}$
- 3 **pour** pour chaque utilisateur $u \in \mathcal{U}$ **faire**
- 4 $\mathcal{N}_u \leftarrow \emptyset$
- 5 $m_u \leftarrow 0$
- 6 $R_f \leftarrow 0$ pour chaque $f \in \mathcal{F}_u$
- 7 **fin**

Algorithme 5 : AGSS : initialisation.

Allocation de ressources. Le pseudo-code de la phase d’allocation de ressources d’AGSS est présenté dans l’Algorithme 6. Nous notons $y \in \mathbb{N}$ le nombre de SUs à allouer lors de l’itération courante. La valeur de y est amenée à être modifiée durant l’exécution. Initialement, nous posons $y = 1$.

Étape 1 : processus d’élection d’un flux pour l’allocation de y SUs supplémentaires. Ce processus est illustré sur les figures 4.2 et 4.3. Pour chaque flux f dont le buffer d’émission est non-vidé, nous choisissons les y premiers SUs dans la pile CQS_u de

l'utilisateur u possédant le flux f . Soit k le dernier SU de la pile CQS_u parmi les y sélectionnés (i.e. $k = CQS_u[y]$). Puisque les piles sont triées, le MCS d'efficacité maximum qui pourra être utilisé par u si l'allocation est réalisée sera alors le MCS $m_{u,k}$. Pour tous les flux appartenant à cet utilisateur, le MCS maximum sera bien sûr le même. Nous notons m'_u le MCS qui sera sélectionné par l'utilisateur si les y SUs sont alloués à l'utilisateur u :

$$m'_u = m_{u,k} \text{ tel que } k = CQS_u[y].$$

Un poids \mathcal{W}_f^{STRAT} pour chaque flux f est ensuite calculé selon la stratégie d'ordonnement "*STRAT*". Le flux élu satisfait :

$$f_{max} = \arg \max_f \mathcal{W}_f^{STRAT}.$$

L'utilisateur élu est donc l'utilisateur u_{max} qui possède le flux f_{max} .

Comme déjà indiqué, nous supposons que $\mathcal{W}_f^{STRAT} \leq 0$ si l'allocation de y SUs supplémentaires au flux f n'augmente pas son débit. De cette façon, ces flux ne peuvent pas être élus (sauf si $\forall f, \mathcal{W}_f^{STRAT} \leq 0$). Par conséquent, deux cas peuvent se produire :

- $\mathcal{W}_{f_{max}}^{STRAT} \leq 0$, l'allocation de y SUs n'est bénéfique à aucun des flux. Si $y < |\mathcal{N}^r|$ (y est inférieur au nombre de SUs non-alloués), alors la valeur de y est incrémentée puis l'algorithme retourne au début de l'Étape 1. Dans le cas contraire, $y = |\mathcal{N}^r|$ (le nombre de SUs restants est égal à y), l'allocation est donc terminée. En effet, les y SUs restants ne peuvent pas être alloués sans réduire les performances du système.
- $\mathcal{W}_{f_{max}}^{STRAT} > 0$, alors le flux élu est f_{max} , et l'Étape 2 est ensuite exécutée.

Étape 2 : allocation des y SU(s) au flux f_{max} .

L'allocation des ressources à l'utilisateur u_{max} est réalisée. L'ensemble $\mathcal{N}_{u_{max}}$ est donc mis à jour. De plus, AGSS recalcule le débit instantané accordé au flux f_{max} . Pour finir, les SUs qui ont été alloués sont retirés des piles triées. Cela permet de ne pas ré-allouer un SU lors d'une itération ultérieure. Si $\mathcal{N}^r = \emptyset$, l'allocation est terminée. Sinon, l'heuristique retourne à l'Étape 1 avec $y = 1$.

En pratique, nous cherchons à ce que la valeur de y soit égale à 1 le plus souvent possible (un seul SU est alloué à chaque itération), ce qui permet d'assurer une granularité d'allocation fine. La valeur de y est incrémentée seulement lorsque l'allocation d'un SU n'est bénéfique à aucun utilisateur (n'augmente le débit d'aucun mobile). Dans ce cas, le processus d'allocation est répété en tentant d'allouer les SUs par groupe de deux. Si l'allocation n'est toujours pas bénéfique, nous allouons les SUs par groupe de trois, etc.

En effet, l'allocation d'un ou plusieurs SU(s) supplémentaire(s) peut uniquement amener à maintenir ou diminuer l'ordre du MCS assigné à l'utilisateur élu, et par conséquent, lui allouer davantage de ressources peut diminuer son débit. Dans tous les cas, les piles triées aident à rendre le processus d'allocation efficace en permettant l'allocation des meilleurs SUs pour l'utilisateur élu.

4.2.3 Un mot sur la complexité

Sans perte de généralité, nous supposons qu'à chaque itération de la boucle interne (ligne 4 à 26) un seul SU est toujours alloué, i.e. la condition $W_{max} > 0$ est toujours

vérifiée. Par conséquent, il peut être considéré qu'AGSS est composé d'une seule boucle principale (avec la variable y toujours égale à 1). Ainsi, il y a exactement $N = |\mathcal{N}|$ itérations pour une exécution complète.

De plus, la ligne 23 peut être exécutée en $\mathcal{O}(1)$ en utilisant des structures de données adéquates (listes chaînées) pour garder en mémoire les piles triées. En effet, l'ensemble \mathcal{A} fait toujours référence à un seul SU puisque y est égal à 1. Par conséquent, en inter-chaînant les SUs dans les piles triées, la boucle aux lignes 22-24 peut être effectuée en $\mathcal{O}(U)$. La complexité d'AGSS est donc simplement en $\mathcal{O}(N(F + U))$. Comme $F \geq U$, la complexité se réduit à $\mathcal{O}(N \times F)$. Cette complexité est équivalente à celle de l'algorithme classique.

Bien sûr, le tri des piles ajoute de la complexité à notre système. Cependant, ces tris ne sont pas effectués à chaque TTI et les valeurs de U et de N sont relativement peu importantes avec LTE.

4.3 Stratégie d'ordonnement

Les *stratégies d'ordonnement* sont implémentées via une définition appropriée des poids $\mathcal{W}_f^{STRAT}, \forall f$ qui sont utilisés dans l'Étape 1 d'AGSS.

Dans cette section, nous commençons par montrer brièvement comment adapter les stratégies de la littérature pour les rendre compatibles avec AGSS. Nous proposons ensuite notre propre stratégie d'ordonnement, que nous avons appelée OPA (*Opportunistic PDOR Aware policy*).

Le calcul des poids dépend de l'avancement de l'exécution d'AGSS. En particulier, ce calcul dépend de $r_f(y)$, le gain de débit obtenu en allouant y SU(s) au flux f . Pour tout flux f , $r_f(y)$ est calculé avant chaque processus d'élection de la manière suivante (en respectant les valeurs des variables à cette itération) :

$$r_f(y) = \min (R(\mathcal{N}_u \cup \mathcal{A}, m'_u) - R(\mathcal{N}_u, m_u), BO_f - R_f)$$

avec u l'utilisateur possédant le flux f et $\mathcal{A} = \{CQS_u[1], CQS_u[2], \dots, CQS_u[y]\}$ l'ensemble des y meilleurs SUs de l'utilisateur u ($CQS_u[1]$ est le SU en tête de pile). La valeur de r_f correspond au gain en termes de débit instantané auquel nous faisons référence dans la Figure 4.2.

Afin de pouvoir être utilisées avec AGSS, les stratégies existantes doivent être adaptées afin d'inclure les variables y et $r_f(y)$. Tout d'abord, nous présentons notre adaptation des stratégies MaxRate et PF pour AGSS. Puis, nous nous penchons sur les stratégies "sensibles aux délais" que sont EXP et M-LWDF. Enfin, nous proposons notre propre stratégie d'ordonnement. Afin de rester concis, nous renvoyons vers la Section 2.2.1 pour davantage d'informations sur les stratégies de l'état-de-l'art.

1) MaxSNR (ou MaxRate).

$$\mathcal{W}_f^{MaxSNR} = r_f(y).$$

Les y meilleurs SUs sont alloués à l'utilisateur qui peut recevoir la plus grande quantité de données sur ces ressources.

```

1 allocation de ressources ;
2 répéter
3    $y \leftarrow 0$ 
4   répéter
5      $y \leftarrow y + 1$ 
6     // Le nombre de SUs qui seront alloués est égal à  $y$ .
7     Étape 1 :
8     pour chaque flux  $f \in \mathcal{F} \setminus \{f \mid BO_f \leq R_f\}$  faire
9        $u \leftarrow$  utilisateur  $u$  tel que  $f \in \mathcal{F}_u$ 
10       $m'_u \leftarrow m_{u,k}$  tel que  $k$  est le  $y^{\text{ième}}$  SU dans  $CQS_u$ 
11       $W_f \leftarrow \mathcal{W}_f^{STRAT}$ 
12    fin
13    // Sélectionner le flux avec le poids le plus important
14     $W_{max} \leftarrow \max W_f, f \in \mathcal{F}$ 
15    Soit  $f_{max}$  le flux associé à  $W_{max}$  et  $u_{max}$  l'utilisateur qui possède ce flux.
16
17    // Si  $W_{max} \leq 0$ , aucun groupe de  $y$  SU(s) ne peut être alloué sans
18    // diminuer la capacité
19    si  $W_{max} > 0$  alors
20      Étape 2 :
21      // Allouer les  $y$  premiers SUs dans  $CQS_{u_{max}}$ 
22       $\mathcal{A} \leftarrow$  l'ensemble des  $y$  premiers SUs dans  $CQS_{u_{max}}$ 
23       $\mathcal{N}_{u_{max}} \leftarrow \mathcal{N}_{u_{max}} \cup \mathcal{A}$ 
24       $\mathcal{N}^r \leftarrow \mathcal{N}^r \setminus \mathcal{A}$ 
25      // (Re)calculer le débit du flux  $f_{max}$ 
26       $R_{f_{max}} \leftarrow R_{f_{max}} + R(\mathcal{N}_{u_{max}} \cup \mathcal{A}, m_{u_{max}}) - R(\mathcal{N}_{u_{max}}, m_{u_{max}})$ 
27       $m_{u_{max}} \leftarrow m'_{u_{max}}$ 
28      pour chaque utilisateur  $u \in \mathcal{U}$  faire
29        | Retirer tous les SUs compris dans  $\mathcal{A}$  de la pile  $CQS_u$ 
30      fin
31    fin
32  fin
33  jusqu'à  $W_{max} > 0$  ou  $y = |\mathcal{N}^r|$ ;
34 retourner  $\mathcal{N}^r = \emptyset$  ou  $y = |\mathcal{N}^r|$ ;
35 Retourner :
36 - Le débit accordé à chaque flux  $\{R_f\}_{f \in \mathcal{F}}$ 
37 - L'allocation de ressources pour chaque utilisateur  $\{\mathcal{N}_u, m_u\}_{u \in \mathcal{U}}$ 

```

Algorithme 6 : AGSS : allocation de ressources.

2) **Proportional Fair (PF).**

$$\mathcal{W}_f^{PF} = \frac{r_f(y)}{\bar{R}_f \times y},$$

avec \bar{R}_f le débit moyen par SU, obtenu grâce à un filtrage passe-bas [85]. Cette stratégie favorise l'allocation des y SUs à l'utilisateur dont le gain de débit "par ressource" est le plus important comparé à son débit moyen par ressource.

3) Modified Largest Weighted Delay First (MLWDF).

$$\mathcal{W}_j^{M-LWDF} = w_f(t) \times \mathcal{W}_f^{PF}.$$

Il s'agit d'une version modifiée du PF qui prend également en compte le délai du paquet en tête du buffer (w_f pour le flux f) dans la gestion des priorités.

4) Exponential (EXP).

Cette stratégie permet de réagir rapidement aux fluctuations des délais :

$$\mathcal{W}_f^{EXP} = \beta_f \exp \left(\frac{a_f w_f(t)}{1 + \sqrt{1/F \sum_{f \in \mathcal{F}} w_f(t)}} \right) \times r_f(y),$$

avec u l'utilisateur possédant le flux f . Les valeurs paramètres a_f et β_f sont données dans la Section 2.2.1.

5) Opportunistic PDOR Aware (OPA).

Afin de pleinement traiter l'Objectif 4.1.1, nous définissons OPA, une stratégie d'ordonnancement originale. Notre idée est de profiter à la fois des avantages de la stratégie exponentielle (pour la gestion du court terme) et du PDOR (pour la gestion du long terme). Par conséquent, OPA prend en considération le PDOR afin d'intégrer à la stratégie l'historique des paquets transmis hors-délai.

PDOR cyclique. Nous utilisons le *PDOR cyclique* au lieu du PDOR "classique" afin de réagir plus rapidement aux *bursts*⁴ des trafics et de s'assurer que les flux bénéficient d'une bonne qualité de service même sur de courtes périodes de temps. Le PDOR cyclique est calculé comme le PDOR mais sur une fenêtre glissante de durée T . Le PDOR cyclique est noté $PDOR_f^{cycle}, \forall f$.

Pour chaque flux u , nous définissons la stratégie OPA comme suit :

$$\mathcal{W}_f^{OPA} = \frac{r_f(y) \times \delta_{OPA}}{PDOR_f^{target} + 1 - PDOR_f^{cycle}}, \quad (4.3.1)$$

avec δ_{OPA} défini par :

$$\delta_{OPA} = 1 + 10^6 \left(b \times PDOR_f^{cycle} + (1 - b) \times \frac{F \times w_f}{\sum_{f \in \mathcal{F}} w_f} \right)^2,$$

w_f étant toujours le délai du paquet en tête du buffer du flux f .

Dans l'ensemble, la combinaison de w_f et $PDOR_f^{cycle}$ permet de diminuer le PDOR tout en évitant que des paquets attendent trop longtemps dans les buffers. Le facteur 10^6 dans l'expression de δ_{OPA} permet de faire en sorte que ce paramètre soit du même ordre de grandeur que $r_f(y)$. Le dénominateur de (4.3.1) compare le PDOR cyclique au PDOR cible. Si le PDOR d'un utilisateur est proche du PDOR cible, alors le trafic devient prioritaire. La priorité d'un flux ne cesse de s'accroître avec l'augmentation du PDOR. Notons que nous avons construit cette stratégie empiriquement en réalisant de nombreuses simulations pour mettre au point son expression.

4. Transfert de données important sur une courte période de temps.

Paramètre	Valeur
Temps simulé	15 minutes (630 kTTIs en sens descendant)
SNR de référence	31 dB correspond à une distance de référence d_{ref}
Modèle du canal	Affaiblissement de propagation avec atténuation de Rayleigh
Distance de référence	$1 d_{ref}$
Distance initiale des utilisateurs	Voir Tableau 4.3
Choix du MCS	Méthode EESM (Section 3.6)
Mode de duplexage	Time Division Duplex (TDD)
Configuration du TDD	Configuration 2 [11]
Largeur de bande	6,48 MHz (36 RBs / 12 SUs)
Distance initiale des utilisateurs	1, 2 ou 3 d_{ref} (Tableau
Modèle de mobilité	Random Way Point
Zone de mobilité	Cercle de rayon 4 d_{ref}
Taille de la fenêtre pour le calcul du PDOR cyclique (T)	15 secondes

Tableau 4.2 – Paramètres de simulations.

4.4 Évaluation des performances

L’objectif de cette section est d’évaluer les performances d’AGSS associé à la stratégie OPA. Pour cela, nous avons mené une nouvelle campagne de simulations se basant sur un modèle similaire à celui présenté dans la Section 3.6.

Les paramètres de simulations généraux sont donnés dans le Tableau 4.2. De plus, nous supposons l’existence d’un mécanisme de contrôle d’accès performant. Par conséquent, la somme des débits des flux n’excède jamais la capacité du système. Dans le cas contraire, quel que soit l’ordonnanceur, aucun flux ne pourrait être satisfait. D’autre part, notons qu’il n’est pas nécessaire d’évaluer le débit moyen de notre système car, en régime stationnaire, le débit de sortie de la station de base est égal à la somme des débits en entrée du système puisque chacun des paquets finit par être servi. Enfin, nous définissons la charge du système comme le débit moyen auquel est soumis celui-ci.

Le modèle de mobilité utilisé dans les simulations est le *Random Way Point* [86]. Chaque utilisateur sélectionne une vitesse aléatoire compris entre $1/60 d_{ref}.s^{-1}$ et $1/2 d_{ref}.s^{-1}$ (avec une distribution uniforme).

D’autre part, nous considérons des applications en temps réel avec des contraintes restrictives sur les délais des paquets. En effet, il est important que l’ordonnanceur puisse s’adapter à des trafics hétérogènes et réalistes. Les caractéristiques d’un trafic modifient complètement les performances du système en comparaison avec un modèle dans lequel les buffers sont considérés comme “toujours pleins”.

Nous supposons que chaque mobile possède deux flux, i.e. un utilisateur reçoit des données issues de deux flux bien différents :

- Un flux de données MPEG-4 [87]. Ce trafic de type vidéo-conférence est complexe mais très réaliste. Il peut provoquer des pics de débits très importants (Figure 4.4). Ces *bursts* influencent grandement les performances du système en remplissant soudainement les buffers sur de très courtes périodes de temps.
- Un flux de voix AMR [88]. Ce modèle est basé sur une chaîne de Markov à deux états (état ON et état OFF). La durée moyenne de séjour dans chacun des

GID	Distance initiale	Délai max T_f	Débit moyen	Vitesse ($d_{ref}.s^{-1}$)
1	$1d_{ref}$	80 ms	288,5 kbit/s	[1/60, 1/2]
2	$2d_{ref}$	80 ms	288,5 kbit/s	[1/60, 1/2]
3	$3d_{ref}$	80 ms	288,5 kbit/s	[1/60, 1/2]

Tableau 4.3 – Paramètres de mobilité.

deux états est fixée à 3 secondes. Dans l'état ON, la source transmet à un débit constant de 12,2 kbit/s. Dans l'état OFF, le débit est fixé à 4,75 kbit/s. Le temps moyen inter-arrivées des paquets est égal à 20 ms.

Ainsi, le débit moyen d'un utilisateur est environ égal à 288,5 kbit/s. Notons que tous les trafics sont désynchronisés les uns des autres de façon à éviter que tous les paquets arrivent exactement au même moment, ce qui ne serait pas réaliste.

Le délai maximum à partir duquel un paquet est considéré comme hors-délai est fixé à 80 ms pour les deux flux ($T_f = 80$ ms, $\forall f \in \mathcal{F}$). De plus, nous visons un PDOR inférieur ou égal à 5 % pour tous les flux ($PDOR_f^{target} = 0,05, \forall f \in \mathcal{F}$). Nous supposons que le mécanisme de contrôle d'accès refuse l'arrivée de nouveaux flux/utilisateurs si le trafic total généré rend l'objectif impossible à atteindre. Par conséquent, nous ne simulons pas de charge (somme des débits des flux) qui mène à un PDOR moyen excédant 6 % (pour le meilleur ordonnanceur et pour l'utilisateur le moins performant). Les paramètres sont résumés dans le Tableau 4.3.

Le plus petit débit global que nous expérimentons est de 3 Mbit/s. Cela correspond à 10 utilisateurs. Les utilisateurs sont répartis selon trois groupes appelés respectivement GID 1, 2 et 3 dans le Tableau 4.3. Initialement, le premier groupe est composé de 4 utilisateurs alors que les deuxième et troisième groupes sont composés de trois utilisateurs. Les distances de départ sont différentes pour chacun des groupes (voir Tableau 4.3). L'augmentation du débit global est réalisée en augmentant successivement le nombre d'utilisateurs dans le système (en alternant entre les trois groupes).

Tous les résultats de cette section sont donnés avec un intervalle de confiance de mi-largeur inférieure à 5 % de la valeur correspondante.

Calibration du paramètre b . Dans la stratégie OPA, nous avons fixé le paramètre b à 0,9. Cette valeur minimise le PDOR moyen comme cela peut être observé sur la Figure 4.5 qui montre le PDOR moyen en fonction de la valeur de b pour une charge globale de 4,5 Mbit/s (16 utilisateurs). Des résultats similaires ont été obtenus pour d'autres valeurs de charge du système.

4.4.1 Algorithme classique vs AGSS

Tout d'abord, nous comparons AGSS avec l'algorithme d'allocation de ressources classique (Section 2.2.1). L'algorithme classique est la solution la plus utilisée dans les travaux sur l'allocation de ressources dans le sens descendant de LTE [4, 89, 70, 90, 91, 46]. En effet, bien que simple, cet algorithme est optimal dans un système OFDMA classique (sans les contraintes spécifiques à LTE). L'avantage de cet algorithme est d'être suffisamment générique pour permettre l'implémentation des différentes stratégies d'ordonnancement. C'est pourquoi nous pouvons comparer AGSS à l'algorithme

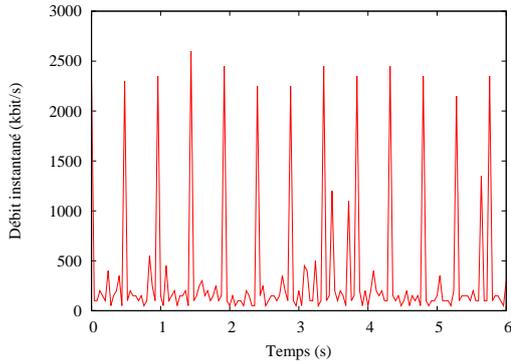


Fig. 4.4 – Débit instantané de notre modèle MPEG-4.

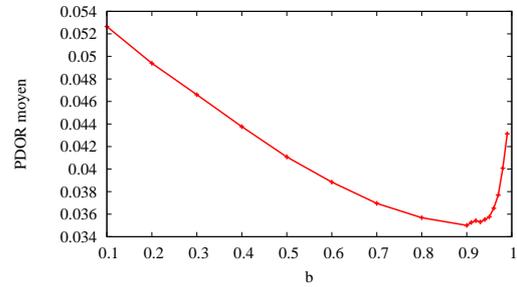


Fig. 4.5 – Calibration du paramètre b dans la stratégie OPA.

classique.

La Figure 4.6 compare les deux algorithmes associés à une même stratégie d’ordonnement. Comme prévu, AGSS est bien plus performant que le schéma classique en termes de PDOR moyen et ce quelle que soit la stratégie d’ordonnement. Pour les stratégies standards telles que le MaxSNR et le PF, AGSS est en moyenne trois fois supérieur à l’algorithme classique. Les résultats sont encore meilleurs pour les stratégies “sensibles aux délais” (M-LWDF, EXP) : le gain de performance est jusqu’à 4 fois supérieur pour les hauts débits.

Dans le reste de cette sous-section, nous avons utilisé OPA comme stratégie d’ordonnement afin de comparer les algorithmes plus en détail. Des résultats aux tendances similaires ont été obtenus avec les autres stratégies. La comparaison des stratégies d’ordonnement est traitée dans la prochaine section.

Dans la Figure 4.7, le ratio d’utilisateurs satisfaits est évalué par rapport à la charge du système (i.e. le débit total auquel le système est soumis). Pour toutes les charges, il apparaît que le nombre d’utilisateurs satisfaits est supérieur avec AGSS. Même pour une charge de 3 Mbit/s, le schéma classique n’obtient que 45 % d’utilisateurs satisfaits, ce qui n’est pas acceptable. Les résultats sont confirmés par la Figure 4.8 qui montre qu’AGSS surpasse l’algorithme classique en matière de délai moyen des paquets.

L’écart de performance entre les deux algorithmes s’explique par deux raisons :

- La considération des contraintes de LTE fait partie intégrante de la conception d’AGSS, tandis que l’algorithme classique ne tente pas d’optimiser l’allocation de ressources selon ces contraintes (même si nous avons fait en sorte que les contraintes soient respectées, voir Chapitre 2).
- Contrairement à l’algorithme classique, AGSS n’alloue pas séquentiellement les ressources. Plus précisément, AGSS tente d’allouer le meilleur groupe de SUs au meilleur utilisateur à chaque itération.

4.4.2 Comparaison des stratégies d’ordonnement

Nous comparons ensuite les différentes stratégies d’allocation de ressources lorsqu’elles sont utilisées avec AGSS.

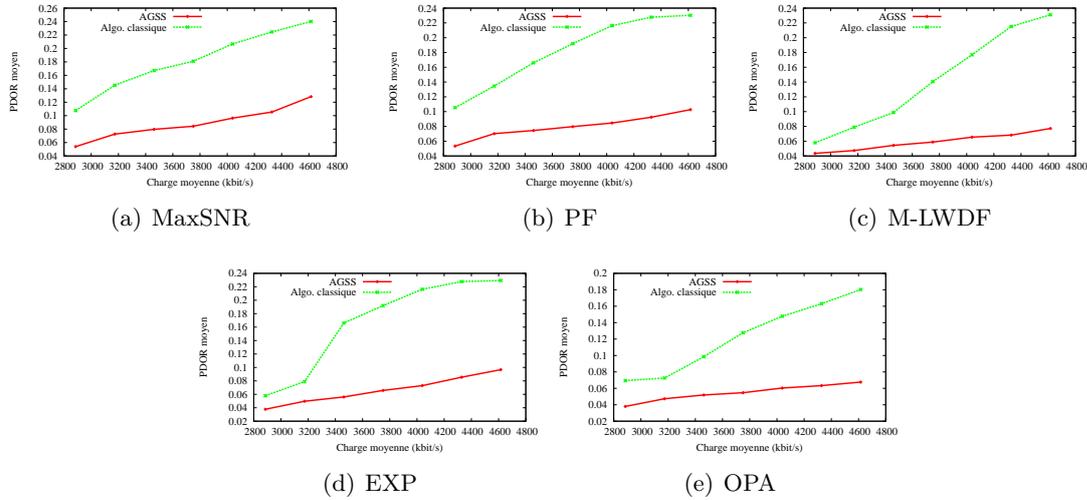


Fig. 4.6 – PDOR en fonction du débit moyen avec l'algorithme classique et AGSS.

Cas statique. Pour ce cas particulier, nous considérons des utilisateurs statiques afin de mieux justifier l'équité en matière de PDOR obtenu par chacune des stratégies.

Pour cela, nous avons séparé les utilisateurs en deux groupes : (1) les utilisateurs proches et (2) les utilisateurs éloignés. Les utilisateurs proches sont placés à une distance de $2d_{ref}$ de la station de base tandis que les utilisateurs éloignés sont placés à $3d_{ref}$. Par conséquent, la qualité du canal est relativement stable mais ce dernier est toujours soumis à une atténuation multi-trajet de Rayleigh. L'augmentation de la charge du réseau est toujours réalisée en ajoutant des utilisateurs dans le système (en alternant entre le groupe des utilisateurs proches et le groupe des utilisateurs éloignés).

Les courbes de la Figure 4.9 montrent l'évolution du PDOR pour tous les utilisateurs en fonction du débit global. Le PDOR moyen est différencié pour les utilisateurs proches et éloignés. Pour des raisons de visibilité, nous avons utilisé différentes échelles pour chaque figure. Pour le MaxSNR et le PF, le PDOR moyen des utilisateurs situés

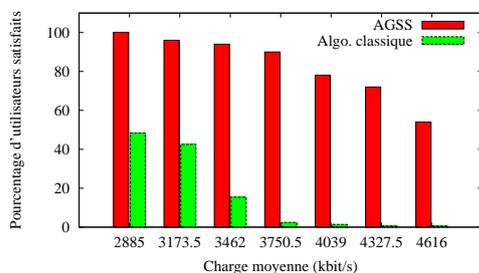


Fig. 4.7 – Pourcentage d'utilisateurs satisfaits en fonction de la charge du système (avec OPA).

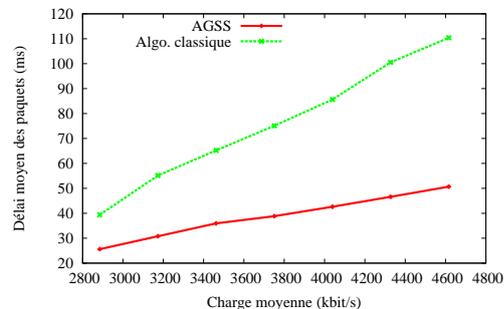


Fig. 4.8 – Délai moyen de paquets en fonction de la charge du système (avec OPA).

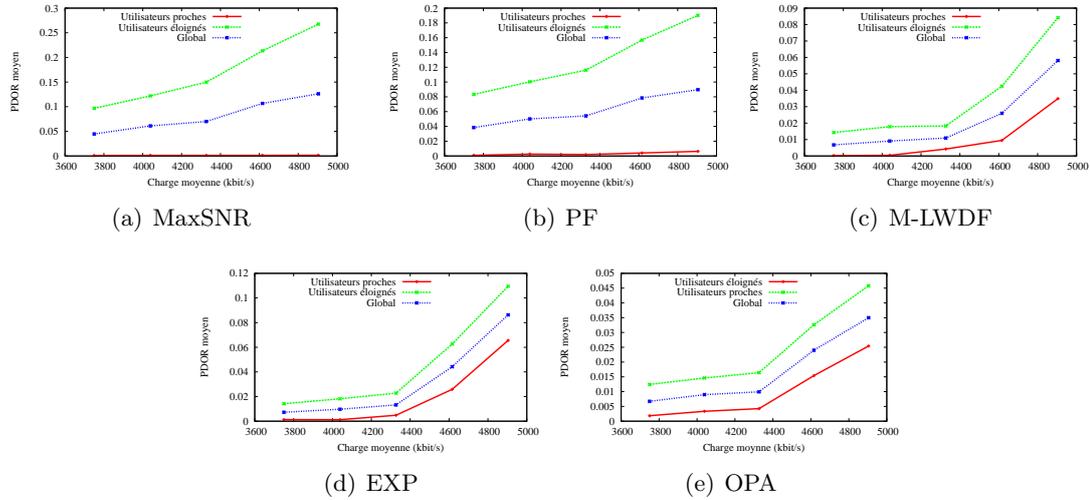


Fig. 4.9 – PDOR en fonction du débit moyen. Les utilisateurs proches et éloignés sont différenciés (cas statique).

à $2d_{ref}$ de la station de base est proche de zéro ; tandis que le PDOR moyen des utilisateurs situés à $2d_{ref}$ est très haut, ce qui n'est pas acceptable. En effet, la stratégie du MaxSNR alloue les ressources à l'utilisateur qui peut recevoir la plus grande quantité de données sur ces dernières. Par conséquent, les utilisateurs éloignés sont clairement désavantagés. De façon similaire, le PF alloue un SU à un utilisateur qui peut recevoir plus de données que d'ordinaire sur ce SU. Cependant, les utilisateurs proches sont toujours favorisés car la station de base peut transmettre avec un meilleur MCS sur les ressources allouées à ces derniers.

En ce qui concerne EXP, M-LWDF et OPA, ces stratégies fournissent un bon PDOR moyen, et offrent une qualité de service très similaire aux deux groupes. Pour EXP et M-LWDF, si le paquet en tête du buffer est en attente depuis un long moment, le flux devient prioritaire quelle que soit sa position. Pour OPA, si le PDOR d'un utilisateur devient trop important par rapport à son PDOR cible alors le flux obtient bien plus de chance d'être élu. De surcroît, OPA est construit pour maintenir une bonne diversité multi-utilisateur, i.e. cette stratégie permet de conserver un grand nombre de flux actifs (avec un buffer non-vidé). Cela est très important pour bénéficier pleinement de la diversité des flux lors de l'étape d'élection d'AGSS. Ces caractéristiques permettent d'obtenir de bonnes performances.

Cas mobile. Nous revenons maintenant au scénario impliquant des utilisateurs mobiles. La Figure 4.10 montre le PDOR moyen en fonction de la charge du système pour les différentes stratégies (OPA, MaxSNR, PF et EXP) en utilisant AGSS. Entre 2,8 et 3,5 Mbit/s, les stratégies OPA, M-LWDF et EXP sont quasiment équivalentes. En effet, M-LWDF et EXP prennent en compte le délai des paquets ce qui affecte le PDOR. Cependant, pour des débits plus importants, OPA devient significativement meilleur en matière de PDOR. Cela peut être expliqué par la nature instable des paramètres

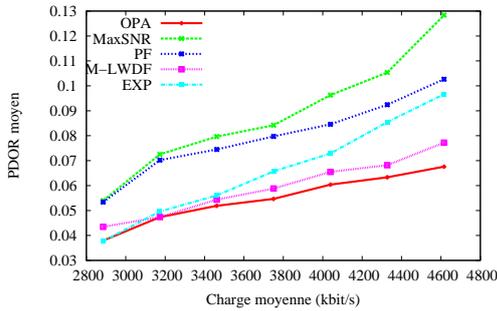


Fig. 4.10 – PDOR moyen pour les différentes stratégies.

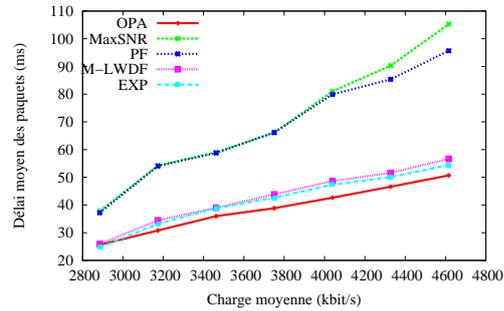


Fig. 4.11 – Délai moyen pour les différentes stratégies.

w_f (délai du paquet en tête de buffer) dans le calcul des poids de ces stratégies. A haut débit, les métriques stables mènent à un système stable, ce qui offre généralement de meilleures performances. Comme on pouvait s’y attendre, le MaxSNR est la stratégie la moins efficace en raison de l’inéquité totale entre les flux.

Les Figures 4.11 et 4.12 montrent respectivement le délai moyen des paquets (temps d’attente dans les buffers d’émissions) et la gigue moyenne en fonction de la charge du système. Les performances obtenues en matière de délai et de gigue avec M-LWDF, EXP et OPA sont similaires, même à hauts débits. En effet, les stratégies M-LWDF et EXP prennent directement en compte le délai des paquets, et par conséquent, obtiennent de bonnes performances. Cependant, la stratégie OPA devient légèrement supérieure à haut débit ce qui montre que le PDOR est également très bien adapté à la minimisation du temps d’attente des paquets.

La Figure 4.13 montre le pourcentage d’utilisateurs satisfaits (au sens de l’Équation 4.1.1) pour trois charges différentes. Pour 3173,5 kbit/s, toutes les stratégies “sensibles aux délais” obtiennent plus de 90 % d’utilisateurs satisfaits. Avec l’augmentation de la charge du système, le ratio d’utilisateurs satisfaits diminue pour le M-LWDF et EXP mais reste très correct pour la stratégie OPA, qui obtient clairement le meilleur ratio. Les stratégies PF et MaxSNR restent stables car les utilisateurs proches sont toujours satisfaits. À haut débit, le nombre d’utilisateurs satisfaits chute pour le M-LWDF et

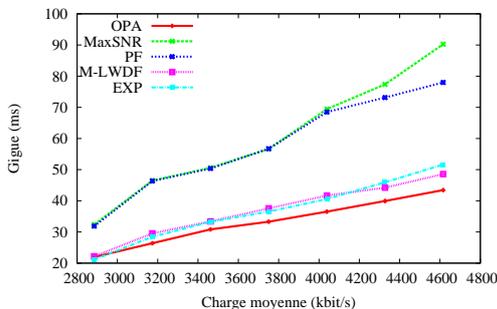


Fig. 4.12 – Gigue moyenne pour les différentes stratégies.

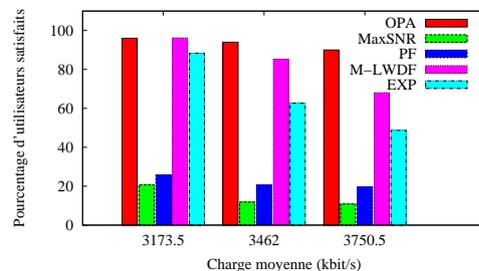


Fig. 4.13 – Pourcentage d’utilisateurs satisfaits en fonction du débit global du système.

EXP.

La stratégie OPA est la seule à obtenir de bonnes performances comparée aux autres stratégies. En effet, la prise en compte du PDOR dans le calcul des poids permet de réduire le PDOR des utilisateurs et donc d'augmenter le nombre d'utilisateurs satisfaits. En moyenne, OPA est environ 25 % plus efficace que la deuxième meilleure stratégie, M-LWDF.

Comme déjà indiqué, nous avons conduit des simulations pour des charges adaptées à la capacité maximale du système. Par conséquent, la Figure 4.13 montre également la charge maximale supportée pour chacune des stratégies. Par exemple, si nous visons 90 % d'utilisateurs satisfaits, la stratégie OPA peut supporter une charge de 3750 kbit/s. De même, M-LWDF et EXP supportent 3173,5 kbit/s, tandis que le MaxSNR et le PF supportent seulement quelques utilisateurs dans ce scénario.

4.5 Conclusion

Dans ce chapitre, nous avons proposé AGSS, une heuristique d'allocation de ressources générique adaptée aux spécificités de LTE. AGSS permet d'intégrer des *stratégies d'ordonnancement* via une définition appropriée de poids. Nous avons développé notre propre stratégie, appelée OPA, qui permet de fournir le niveau de Qualité de Service demandé par les applications multimédias. Afin de parvenir à cet objectif, nous avons utilisé le PDOR, une métrique de QoS inhabituelle. Nos simulations montrent qu'AGSS obtient de meilleures performances que l'algorithme classique (quelle que soit la stratégie utilisée). De plus, OPA surclasse également les autres stratégies d'ordonnancement. L'association d'AGSS et d'OPA fournit un ordonnanceur pour le sens descendant de LTE très efficace en termes de charge maximale supportée, délais des paquets et de PDORs moyens. De plus, OPA assure une meilleure équité pour des mobiles ayant des qualités de liens radios différentes.

Discussion. Dans ce chapitre ainsi que le précédent, nous avons étudié le problème de l'allocation de ressources sous différents niveaux de contraintes et d'objectifs. Cependant, ces solutions sont seulement valables pour le sens descendant de LTE. Le prochain chapitre sera dédié à l'étude du sens montant.

Chapitre 5

Allocation de ressources dans le sens montant

Dans ce chapitre, nous traitons le cas du sens montant de LTE. Les équipements utilisateurs emploient les ressources allouées par l'ordonnanceur pour transmettre des données en direction de la station de base. Dans ce sens de transmission, de nouvelles contraintes s'appliquent :

- une allocation est valide si les SUs alloués à chaque utilisateur sont adjacents dans le domaine fréquentiel ;
- le nombre maximum d'utilisateurs ordonnancés sur un TTI est borné¹.

Ces contraintes réduisent drastiquement la flexibilité offerte à un algorithme d'allocation de ressources. La diversité fréquentielle et la diversité des utilisateurs ne peuvent pas être pleinement exploitées. Par conséquent, les algorithmes présentés dans les chapitres précédents ne sont plus valides. Nous cherchons alors à trouver de nouvelles solutions adaptées au sens montant.

Contribution du chapitre. Dans ce chapitre, nous proposons APASS (*Adaptive and Potential Aware Scheduling Scheme*), une heuristique adaptée :

- aux trafics multimédias ;
- aux contraintes de la norme 3GPP LTE-A/LTE-B.

Tout comme l'algorithme présenté dans le Chapitre 3, APASS est composé de trois routines. Chacune de ces routines répond à un objectif précis. La Figure 5.1 résume les opérations d'APASS.

Routine 1. Pour cette première routine, nous relâchons la contrainte du nombre maximum d'utilisateurs. L'algorithme réalise une première allocation de ressources. De plus, nous montrons que cette première routine est un algorithme d'approximation si les utilisateurs sont gourmands.

1. En réalité, cette contrainte existe également dans le sens descendant mais la borne est bien plus large (50 utilisateurs dans le sens descendant, 8-10 utilisateurs dans le sens montant). Avec l'*allocation de type 0* (voir Annexe A) que nous avons utilisée, cette borne n'est même jamais atteinte. Dans le cas où l'*allocation de type 0* n'est pas utilisée, la diversité fréquentielle ne peut pas du tout être exploitée et le problème d'allocation de ressources ne se pose pas.

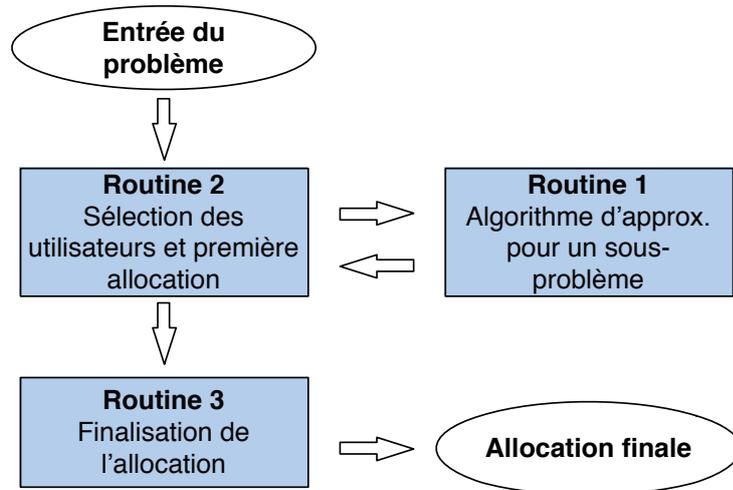


Fig. 5.1 – Résumé des opérations d'APASS.

Routine 2. Cette deuxième routine est chargée de traiter la contrainte du nombre maximum d'utilisateurs. Pour cela, cette étape utilise les allocations réalisées par la première routine afin de sélectionner les utilisateurs qui seront ordonnancés sur le prochain TTI.

Routine 3. La dernière routine d'APASS termine l'allocation de ressources. Cette dernière étape est basée sur le concept de *zone de potentiel maximal* qui fournit une abstraction des contraintes LTE en sélectionnant un ensemble de ressources qui sont *intéressantes* pour un utilisateur donné.

Nous validons notre approche par simulation. Les résultats montrent qu'APASS est en moyenne trois fois plus performant en matière de PDOR moyen et de temps d'attente des paquets par rapport aux solutions existantes. Le nombre d'utilisateurs satisfaits de notre solution est également supérieur.

Simplification. APASS est une procédure d'allocation de ressources avancée qui nécessite l'introduction de nombreuses notations. Afin de simplifier un maximum la présentation de notre solution, nous avons fait le choix de ne pas inclure les flux dans la formulation du problème. Cependant, ces derniers peuvent tout de même être intégrés au problème de la même façon que dans la Section 4.1.1. De plus, nous avons volontairement simplifié la formulation du problème, notamment sur la formalisation des *stratégies d'ordonnancement*. Une formulation plus ambitieuse peut être trouvée dans [92].

5.1 Formulation du problème

Nous notons toujours $\varphi_u(\mathcal{N}_u, m_u)$ la valeur de la fonction d'utilité de l'utilisateur u lorsqu'on lui alloue l'ensemble \mathcal{N}_u avec le MCS m_u .

5.1.1 Problème du LTE UL TFDPS.

Tout comme pour les chapitres précédents, les fonctions $\{\varphi_u\}$ étant non-linéaires, nous préférons ne pas exprimer le problème comme un programme linéaire en nombres entiers. Pour chaque TTI t de la période de taille T , le problème LTE UL TFDPS consiste à trouver un sous-ensemble $\mathcal{V}(t)$ inclus dans \mathcal{U} afin de construire des sous-ensembles disjoints $\mathcal{N}_{u_1}, \dots, \mathcal{N}_{u_{|\mathcal{V}|}}$ ($u_1, \dots, u_V \in \mathcal{V}(t)$) de façon à maximiser :

$$\sum_{t=n+1}^{n+T} \sum_{u \in \mathcal{V}(t)} \varphi_u(\mathcal{N}_u(t), m_u(t)) \quad (5.1.1)$$

soumis aux contraintes suivantes ($\forall t$) :

- le cardinal de l'ensemble $\mathcal{V}(t)$ doit être inférieur à une constante entière V (contraintes sur le nombre d'utilisateurs);
- les SUs alloués à chaque utilisateur sont adjacents dans le domaine fréquentiel;
- $m_u(t)$ est le seul MCS qui peut être utilisé par u ;
- $m_u(t)$ doit être choisi de façon à ce que le taux d'erreurs sur chaque ressource soit inférieur au taux d'erreur binaire maximum toléré.

Comme déjà indiqué, cette formulation peut être adaptée pour prendre en compte plusieurs flux utilisateurs.

Exemple. Comme pour le chapitre précédent, notre travail s'applique à une grande variété d'objectifs. Cependant, notre priorité est de fournir de la qualité de service aux utilisateurs. Nous reprenons donc l'objectif traité précédemment (satisfaction du plus grand nombre d'utilisateurs à travers le temps).

5.1.2 Expression des stratégies d'ordonnancement et sous-problèmes

Nous supposons dans cette sous-section qu'aucun utilisateur n'a reçu de ressource, comme cela est le cas au début de la procédure d'allocation de ressources.

Afin de traiter l'Objectif 5.1.1, nous raffinons le concept de stratégie d'ordonnancement. Dans la Section 4.3, nous avons défini des poids pour des groupes de y SU(s); avec $y \in \mathbb{N}^*$. Ainsi, pour un utilisateur u , un sous-ensemble de ressources \mathcal{N}_u et un MCS m , nous pouvons calculer le poids de la stratégie sur ce sous-ensemble via l'expression de \mathcal{W}_u^{STRAT} . Le poids dépend donc à la fois (1) de l'utilisateur, (2) du sous-ensemble de ressources considéré et (3) du MCS qui sera utilisé pour transmettre sur ces ressources. Si nous supposons qu'aucun utilisateur n'a reçu de ressource, nous pouvons noter explicitement simplement ces dépendances : nous désignons par $\mathcal{W}_u^{STRAT}(\mathcal{N}_u, m)$ le poids calculé par la stratégie *STRAT* avec les paramètres précédents. Notons que nous avons également utilisé ce type de notation dans la Section 2.2.1.

Afin de faciliter la construction de notre algorithme, nous définissons $\mathcal{N}_u(m)$ comme le plus grand ensemble de SUs contigus, inclus dans \mathcal{N}_u , et qui supportent le MCS m (pour l'utilisateur u). Cela signifie qu'il est possible de transmettre sur tout SU dans $\mathcal{N}_u(m)$ avec le MCS m (1) sans excéder le taux d'erreur maximum toléré pour l'utilisateur u et (2) sans enfreindre la contrainte sur les allocations contiguës.

Dans ce chapitre, le poids d'une stratégie est calculé ainsi :

$$c_{u, \mathcal{N}_u} \triangleq \max_{m \in \mathcal{M}} \mathcal{W}_u^{STRAT}(\mathcal{N}_u(m), m). \quad (5.1.2)$$

Si nous notons $j = \mathcal{N}_u$, la valeur du paramètre $c_{u,j}$ fait référence au poids de la stratégie d'ordonnancement pour l'ensemble j et l'utilisateur u . Nous avons modifié les notations du calcul de poids afin de rendre intuitif le programme linéaire qui sera résolu lors de la première routine.

Sous-problème 1. En utilisant l'expression des poids, nous définissons un objectif intermédiaire qui sera adressé par la première routine. Pour ce sous-problème, nous relâchons la contrainte de la limitation du nombre d'utilisateurs. L'objectif consiste à calculer les sous-ensembles disjoints j_{u_1}, \dots, j_{u_U} de façon à maximiser $\sum_{u \in \mathcal{U}} c_{u,j}$.

Sous problème 2. Ce sous-problème est traité par la deuxième routine. L'objectif est semblable au sous-problème précédent, mais considérant cette fois la contrainte sur la limitation du nombre maximum d'utilisateurs ordonnancés. Ainsi, nous cherchons à trouver l'ensemble des utilisateurs élus \mathcal{V} conjointement avec les sous-ensembles disjoints $j_{u_1}, \dots, j_{u_{|\mathcal{V}|}}$ (avec $u_i \in \mathcal{V}, \forall i$) de façon à maximiser $\sum_{u \in \mathcal{V}} c_{u,j}$.

Remarques. Nous rappelons que les stratégies d'ordonnancement n'ont pas pour vocation de remplacer les fonctions d'utilité. Dans notre cas, nous les utilisons pour fixer des objectifs intermédiaires. Dans la Section 4.3, nous avons insisté sur le fait que, pour qu'une stratégie d'allocation soit efficace, il faut que les allocations soient réalisées finement. Cependant, ce n'est pas le cas pour les deux premières routines puisque les SUs seront évalués par groupe afin d'apporter une solution au sous-problème considéré. L'objectif de nos deux premières routines est en effet :

- d'élire les V utilisateurs qui seront ordonnancés ;
- et de réaliser une première allocation.

Cette allocation est une solide base de travail pour la troisième routine mais ne constitue en aucun cas l'allocation finale. C'est la dernière routine qui est chargée de terminer l'allocation. Cette routine utilise les stratégies d'ordonnancement à leurs pleins potentiels en allouant les ressources le plus finement possible.

5.1.3 Difficulté du problème

Le problème est clairement NP-Difficile (voir Chapitre 2). Cependant, avec l'ajout de la contrainte sur le nombre d'utilisateurs ainsi que la contrainte sur la contiguïté des allocations, on peut se demander si l'espace de recherche n'est pas devenu suffisamment petit pour exécuter un algorithme de force brute. En particulier, nous souhaitons savoir si le deuxième sous-problème peut être résolu par un tel procédé.

La proposition suivante montre que, malheureusement, la taille de l'espace de recherche est toujours bien trop importante.

Proposition 2. *La taille de l'espace de recherche du deuxième sous-problème est égale à :*

$$\sum_{i=0}^{\min(U,V,N)} \frac{U!}{(U-i)!} \binom{N+i}{2i}.$$

Preuve. Tout d'abord, considérons que seulement i ($0 \leq i \leq \min(U, V, N)$ ²) utilisateurs reçoivent au moins une ressource. Nous notons $L = u'_1, \dots, u'_i$ une liste ordonnée d'utilisateurs telle que (1) $u'_k \in \mathcal{U}$ avec $1 \leq k \leq i$ et (2) aucun élément dans L n'est présent plus d'une fois. Une telle liste ordonnée est appelée arrangement. Le nombre d'arrangements possibles est égal à $\frac{U!}{(U-i)!}$.

Supposons que tous les SUs de a_k à b_k sont alloués à u'_k , $1 \leq k \leq i$, tel que :

$$1 \leq a_1 \leq b_1 < a_2 \leq b_2 < a_3 \leq b_3 < \dots < a_i \leq b_i \leq N.$$

Soit $x_k = b_k - a_k$, et soit

$$y_k = \begin{cases} a_1 - 1 & \text{if } k = 1, \\ a_k - b_{k-1} - 1 & \text{if } 2 \leq k \leq i, \\ N - b_N & \text{if } k = i + 1. \end{cases}$$

Ainsi, $x_k + 1$ ($1 \leq k \leq i$) correspond au nombre de SUs alloués à u'_k . De même, y_k ($1 \leq k \leq i + 1$) est le nombre de SUs non-alloués dans le $k^{\text{ème}}$ slot disponible.

Comme il n'y a que N SUs, les variables x_k et y_k doivent respecter la contrainte suivante :

$$\sum_{k=1}^i x_k + \sum_{k=1}^{i+1} y_k = N - i, x_k \geq 0, y_k \geq 0. \quad (5.1.3)$$

Le nombre de solutions entières pour cette équation est égal à $\binom{N+i}{N-i} = \binom{N+i}{2i}$.

Il reste à considérer toutes les valeurs de i afin d'obtenir le nombre total d'allocations possibles :

$$\sum_{i=0}^{\min(U, V, N)} \frac{U!}{(U-i)!} \binom{N+i}{2i}. \quad (5.1.4)$$

□

De même que pour la Proposition 3.2.2, l'espace de recherche est bien trop grand pour être exploré de façon exhaustive.

5.2 APASS : algorithme d'allocation de ressources dans le sens montant

Dans cette section, nous présentons notre solution pour le problème d'allocation de ressources dans le sens montant. Cette solution est composée de trois routines. Nous présentons successivement chacune de ces étapes.

5.2.1 Allocation de ressources pour une relaxation du problème

Cette étape traite le premier sous-problème de la Section 5.1.2.

2. En effet, si $U > N$ et $V > N$ alors le nombre maximum d'utilisateurs ordonnancés est égal à N . De même, si $V < U$ et $V < N$ alors le nombre maximum d'utilisateurs ordonnancés est égal à V . Enfin, si $V > U$ et $U < N$, alors le nombre maximum d'utilisateurs ordonnancés est égal à U .

L'Algorithme 7 décrit le pseudo-code de notre première routine. Afin de simplifier les notations, un SU est référencé par son indice. Dans cet algorithme, un ensemble S_i est une partition de \mathcal{N} constitué de sous-ensembles de 2^i SUs contigus à l'exception du "dernier" sous-ensemble qui est composé des SUs restants, comme indiqué à la ligne 3 de l'Algorithme 7.

Par conséquent, la première partition est définie comme $S_0 = \{\{1\}, \{2\}, \dots, \{N\}\}$ et la dernière partition est exprimée par $S_p = \{\{1, \dots, N\}\}$, avec $p = \lceil \log_2(N) \rceil$.

Soit i , $0 \leq i \leq p$. Chaque élément de S_i est un sous-ensemble de SUs contigus. On construit un problème d'affectation dans lequel chaque élément dans S_i doit être assigné à, au plus, un utilisateur de \mathcal{U} . Ce problème est un programme linéaire en nombres entiers qui est exprimé à la ligne 6 de l'Algorithme 7. Dans ce problème, $b_{u,j}^i$ est une variable de décision qui est égale à 1 si et seulement si le sous-ensemble j est assigné à l'utilisateur u .

Ainsi, nous avons défini $p + 1$ problèmes d'affectation, un pour chacune des partitions. Un problème d'affectation est un programme linéaire en nombres entiers particulier qui peut être résolu de façon optimale par un algorithme de complexité polynomiale. Pour le problème i , la complexité temporelle d'un problème d'affectation est en $\mathcal{O}(\max(U, |S_i|)^3)$ [93]³.

L'Algorithme 7 retourne une description de toutes les allocations, i.e. $(b_{u,j}^i)_{u \in \mathcal{U}, j \in S_i}$ conjointement avec chacune des valeurs agrégées $W_i = \max \sum_u \sum_j c_{u,j}^i b_{u,j}^i$. Nous utilisons ces valeurs retournées dans la prochaine section.

L'allocation qui permet d'obtenir la meilleure valeur agrégée, notée $W_{i_{max}}$, est une bonne solution pour le problème considéré si la contrainte sur la limitation du nombre d'utilisateurs est relâchée. Cependant, pour notre problème principal, il est important de noter que cet algorithme n'alloue aucune ressource lors de cette étape.

Théorème 4. *L'Algorithme 7 est une $\frac{1}{\lceil \log_2(N) \rceil + 1}$ -approximation pour le sous-problème 1.*

Preuve. *Nous rappelons que la contrainte sur la limitation du nombre d'utilisateurs est relâchée.*

Pour une instance du problème, soit $RA^ = \sum_u c_{u, \mathcal{N}_u^*}$ le coût (ou valeur agrégée) de la solution optimale, avec $\{\mathcal{N}_u^*\}_u$ l'allocation optimale.*

Tout d'abord, nous définissons $S = S_0 \cup S_1 \cup \dots \cup S_p$ comme l'ensemble de tous les sous-ensembles qui peuvent être assignés lors de l'exécution de l'algorithme.

Soit $g : \mathcal{U} \rightarrow S$ une fonction exprimée comme suit :

$$g(u) = j \text{ si et seulement si } \begin{cases} 1) & j \in S; \\ 2) & \mathcal{N}_u^* \subseteq j; \\ 3) & \text{le cardinal de } j \text{ est minimum.} \end{cases}$$

3. Le problème d'affectation "d'origine" est défini avec le même nombre d'utilisateurs et d'"objets" à assigner. Autrement dit, pour chaque problème i , U devrait être égal à $|S_i|$ afin que l'instance du problème d'affectation soit valide. Dans notre cas, cela n'est généralement pas vrai. Cependant, il est possible de construire une instance équivalente et valide. Tout d'abord, supposons que $U < |S_i|$. Dans ce cas, nous ajoutons $|S_i| - U$ utilisateurs virtuels au système. Pour chacun de ces utilisateurs virtuels, nous fixons $c_{u,j}^i = 0, \forall j \in S_i$. Inversement, if $U > |S_i|$, nous ajoutons $U - |S_i|$ sous-ensembles virtuels à S_i et nous fixons $c_{u,j}^i = 0, \forall u \in \mathcal{U}$ pour chacun des sous-ensembles virtuels j .

input : L'ensemble des utilisateurs \mathcal{U} .
 L'ensemble des ressources \mathcal{N} .
 L'expression des fonctions d'utilité.
 La qualité du canal pour chaque utilisateur $\mathcal{I} = (m_{u,k})_{u,k}$.
output : Le résultat des $p + 1$ problèmes d'affectation.

```

1  $p \leftarrow \lceil \log_2(N) \rceil$ 
2 pour chaque  $i \in \{0, \dots, p\}$  faire
3    $S_i \leftarrow \{\{1, \dots, 2^i\}, \{2^i + 1, \dots, 2^{i+1}\}, \dots, \{\dots, N\}\}$ 
4   Caculer les poids,  $\{c_{u,j}^i\}_{u \in \mathcal{U}, j \in S_i}$ , via l'Equation 5.1.2.
5   Résoudre le programme linéaire suivant :
6   
$$\begin{array}{l} W_i = \max \sum_u \sum_j c_{u,j}^i b_{u,j}^i \\ \text{subject à :} \\ \sum_u b_{u,j}^i \leq 1, \forall j \in S_i \\ \sum_j b_{u,j}^i \leq 1, \forall u \in \mathcal{U} \\ b_{u,j}^i \in \{0, 1\}, \forall u \in \mathcal{U}, \forall j \in S_i \end{array}$$

7 fin
8  $i_{max} \leftarrow \operatorname{argmax}_i W_i, i \in \{0, \dots, p\}$ 
   /* L'allocation définie par  $(b_{u,j}^{i_{max}})_{u \in \mathcal{U}, j \in S_{i_{max}}}$  est une bonne solution
   si la contrainte sur le nombre d'utilisateurs est relâchée. */
9 Return  $\{i_{max}, (b_{u,j}^i)_{i,u,j}, (W_i)_i\}$ .

```

Algorithme 7 : Algorithme d'approximation pour une relaxation du problème LTE UL FDPS (Sous-problème 1).

En d'autres termes, $g(u)$ fait référence au plus petit ensemble de ressources de S qui contient \mathcal{N}_u^* (i.e. l'ensemble des ressources allouées à u dans la solution optimale).

Pour chaque utilisateur u , nous avons $c_{u,g(u)} \geq c_{u,\mathcal{N}_u^*}$ puisque $c_{u,j}$ croît avec le cardinal de j . Ainsi,

$$\sum_u c_{u,g(u)} \geq \sum_u c_{u,\mathcal{N}_u^*}. \quad (5.2.1)$$

D'autre part, nous montrons que g est injective, i.e. pour deux utilisateurs u_1 et u_2 tels que $g(u_1) = g(u_2)$ alors $u_1 = u_2$. De façon équivalente nous montrons que pour deux utilisateurs u_1 et u_2 tels que $g(u_1) = g(u_2)$ alors $\mathcal{N}_{u_1}^* = \mathcal{N}_{u_2}^*$. En effet, dans une solution valide, comme un sous-ensemble de ressources ne peut pas être alloué à deux utilisateurs différents, il en résulte que $u_1 = u_2$.

Ainsi, considérons deux utilisateurs u_1 et u_2 tels que $g(u_1) = g(u_2) = j$. Si $j \in S_0$, alors j est un ensemble composé d'un seul SU et l'allocation optimale alloue cette ressource à la fois à u_1 et u_2 (i.e. $\mathcal{N}_{u_1}^* = \mathcal{N}_{u_2}^* = j$). Par conséquent, $u_1 = u_2$. Sinon si $|j| > 0$, alors il existe un i tel que $j \in S_i$. Par définition de g , nous savons que $\mathcal{N}_{u_1}^* \subseteq j$ et $\mathcal{N}_{u_2}^* \subseteq j$. Supposons que $j_1, j_2 \in S_{i-1}$ les deux sous-ensembles qui satisfont $j_1 \cup j_2 = j$. Par construction, ces deux sous-ensembles existent. En effet, chaque sous-

ensemble dans S_i est composé de deux sous-ensembles dans S_{i-1} ⁴. Comme j est le sous-ensemble de cardinal minimum qui contient $\mathcal{N}_{u_1}^*$ et $\mathcal{N}_{u_2}^*$, alors ces deux derniers sous-ensembles (issus de la solution optimale) doivent contenir à la fois j_1 et j_2 . À cause de la contrainte de contiguïté des ressources, cela est uniquement possible si $\mathcal{N}_{u_1}^* = \mathcal{N}_{u_2}^*$. Par conséquent, $u_1 = u_2$.

Notons que, même si g est injective, $\{g(u)\}_{u \in \mathcal{U}}$ n'est généralement pas une allocation valide. En effet, l'intersection de deux sous-ensembles différents $g(u_1)$ et $g(u_2)$ peut être non-vide. C'est pourquoi, pour chaque problème i , nous définissons $g_i : \mathcal{U} \rightarrow S_i \cup \emptyset$ une fonction définie par :

$$g_i(u) = \begin{cases} g(u) & \text{si } g(u) \in S_i; \\ \emptyset & \text{sinon.} \end{cases}$$

De cette façon, pour chaque problème d'affectation, $\{g_i(u)\}_{u \in \mathcal{U}}$ est une allocation valide. En effet, comme g est injective, deux utilisateurs ne peuvent pas être associés au même sous-ensemble de S_i . De plus, comme S_i est une partition de \mathcal{N} alors tous les sous-ensembles de S_i sont disjoints.

Pour finir, comme W_i est le coût optimal du problème d'affectation i , nous en déduisons que :

$$\sum_{u \in \mathcal{U}} c_{u, g_i(u)} \leq W_i, \forall i. \quad (5.2.2)$$

Soit $RA = W_{i_{max}} = \sum_u c_{u, \mathcal{N}_u}$ le coût de l'allocation réalisée par l'Algorithme 7. Ainsi :

$$\begin{aligned} RA^* &= \sum_{u \in \mathcal{U}} c_{u, \mathcal{N}_u^*} \\ &\leq \sum_{u \in \mathcal{U}} c_{u, g(u)} \leq \sum_{i=0}^p \sum_{u \in \mathcal{U}} c_{u, g_i(u)} && \text{par (5.2.1)} \\ &\leq \sum_{i=0}^p W_i && \text{par (5.2.2)} \\ &\leq (p+1)W_{i_{max}} = (p+1)RA, \end{aligned}$$

ce qui termine la preuve. □

Nous utilisons l'Algorithme 7 comme première étape d'APASS. Nous souhaitons maintenant prendre en compte la contrainte sur la limitation du nombre d'utilisateurs.

4. Il existe cependant une exception. En effet, soit j le "dernier" sous-ensemble de S_i , i.e. le sous-ensemble composé des SUs restants. Ainsi, j est le seul sous-ensemble de S_i qui peut être composé d'un ou deux sous-ensembles de S_{i-1} . Si j est l'union de deux sous-ensembles de S_{i-1} , alors nous avons bien trouvé deux sous-ensembles j_1 et j_2 tels que $j_1 \cup j_2 = j$. Dans le cas contraire, j est à la fois dans S_i et S_{i-1} . Il est alors facile de montrer que, soit $j \in S_0$ (ce qui correspond à notre cas initial), soit il existe un $i' < i$ tel que nous avons deux sous-ensembles j_1 et j_2 de $S_{i'}$ qui satisfont $j_1 \cup j_2 = j$.

5.2.2 Sélection des utilisateurs

Dans la suite de ce chapitre, nous considérons à nouveau des trafics multimédia venant remplir les buffers des utilisateurs.

L'Algorithme 8 détaille la seconde routine d'APASS. Ce deuxième algorithme sélectionne les V utilisateurs qui seront ordonnancés lors du prochain *TTI*. Un utilisateur sélectionné est appelé un utilisateur élu. L'ensemble des utilisateurs élus est noté \mathcal{V} . De plus, l'algorithme réalise la première allocation de ressources en prenant en compte la sortie de l'Algorithme 7. Notons que seules les lignes 1 et 7 ainsi que les suivantes sont exécutées si $U < V$.

Cet algorithme est basé sur une boucle principale. A chaque tour de boucle, l'Algorithme 7 est appelé afin de trouver une solution à chacun des problèmes d'affectation. Notre deuxième algorithme supprime de la liste des utilisateurs à ordonnancer l'utilisateur qui obtient la *performance moyenne* la plus faible, i.e. l'utilisateur u^- qui minimise la quantité

$$\sum_{i=0}^p \sum_{j \in S_i} b_{u,j}^i \frac{c_{u,j}}{2^i}.$$

En d'autres termes, l'utilisateur qui obtient, en moyenne, les moins bonnes performances (relativement à son poids) sur les $p+1$ problèmes d'affectation n'est pas sélectionné pour être ordonnancé lors du prochain *TTI*. Cette procédure est répétée jusqu'à ce que le nombre d'utilisateurs soit égal à V .

Enfin, l'algorithme retourne la meilleure allocation réalisée par l'Algorithme 7 sur les V utilisateurs élus. Cette allocation constitue une bonne base pour la dernière étape.

input : Entrée identique à l'Algorithme 7.

output : L'ensemble des utilisateurs élus \mathcal{V} et la première allocation $\{\mathcal{N}_u\}_{u \in \mathcal{V}}$.

```

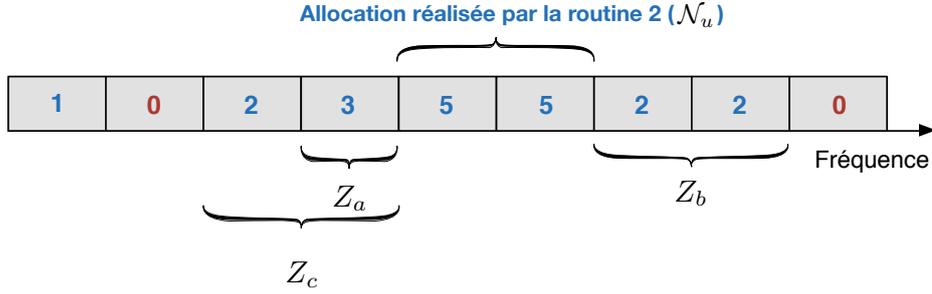
1  $\mathcal{V} \leftarrow \mathcal{U}$ 
2 tant que  $|\mathcal{V}| > V$  faire
3   Exécuter l'Algorithme 7 avec l'ensemble d'utilisateurs  $\mathcal{V}$  et recupérer
    $\{b_{u,j}^i\}, i = 0, \dots, p, j \in S_i, u \in \mathcal{V}$ .
4    $u^- \leftarrow$  l'utilisateur qui minimise  $\sum_{i=0}^p \sum_{j \in S_i} b_{u,j}^i \frac{c_{u,j}}{2^i}$ 
5    $\mathcal{V} \leftarrow \mathcal{V} \setminus \{u^-\}$ 
6 fin
7 Exécuter un dernière fois l'Algorithme 7 avec les utilisateurs élus  $\mathcal{V}$  et
recupérer la sortie :
```

$$i_{max}, (b_{u,j}^i)_{i,u,j}, (W_i)_i$$

$$\mathcal{N}_u \leftarrow j \text{ si } b_{u,j}^{i_{max}} = 1, \forall u \in \mathcal{V}$$

8 **Retourner** \mathcal{V} et $\{\mathcal{N}_u\}_{u \in \mathcal{V}}$.

Algorithme 8 : Sélection des utilisateurs et première allocation.



$Z_b \cup Z_c$: Zone de potentiel maximal (Z_u)

Z_a : Zone de potentiel max. avec cardinal 1 (Z_u^1)

$Z_a \cup Z_b$: Zone de potentiel max. avec cardinal 3 (Z_u^3)

Fig. 5.2 – Exemple de zones de potentiel maximal avec un seul utilisateur $u \in \mathcal{V}$ et avec les SUs pré-alloués \mathcal{N}_u .

5.3 Améliorations et finalisation de l'allocation

L'allocation de ressources réalisée par la deuxième routine (Algorithme 8) avec les V utilisateurs élus n'est pas complètement satisfaisante. En effet :

- Si le nombre de sous-ensembles dans $S_{i_{max}}$ est plus grand que le nombre d'utilisateurs, plusieurs ressources vont rester non-allouées après l'exécution de l'Algorithme 8 ; même si certains SUs auraient pu être bénéfiques à un des utilisateurs élus.
- L'Algorithme 7 utilise des partitions pré-déterminées de \mathcal{N} et cela structure l'allocation obtenue par l'Algorithme 8. Des améliorations peuvent donc être réalisées en adaptant l'allocation.

Pour ces raisons, nous avons construit une dernière routine qui adapte l'allocation réalisée par l'Algorithme 8.

5.3.1 Préliminaires

Dans cette sous-section, nous introduisons quelques définitions qui permettent de construire simplement la dernière étape d'APASS.

Ces définitions sont liées à une allocation inachevée réalisée par l'Algorithme 8. Cette allocation est décrite par $\{\mathcal{N}_u\}_{u \in \mathcal{V}}$, avec \mathcal{N}_u l'ensemble contigu de ressources actuellement allouées à l'utilisateur u .

Définition 5. (Zone). Pour un utilisateur $u \in \mathcal{U}$, un sous-ensemble $Z \subseteq \mathcal{N}$ est appelé **une zone** si $Z \cup \mathcal{N}_u$ est un ensemble de ressources contiguës.

Une zone est donc un sous-ensemble de SUs qui peuvent être alloués à un utilisateur spécifique sans enfreindre la contrainte de contiguïté. Notons qu'une zone est toujours définie par rapport à un utilisateur.

Définition 6. (Potentiel d'une zone).

Nous mesurons le potentiel d'une zone Z de l'utilisateur u grâce à la métrique $P(u, Z)$ qui s'exprime par :

$$c_{u, \mathcal{N}_u \cup Z} - \left(c_{u, \mathcal{N}_u} + \sum_{v \in \mathcal{V} \setminus \{u\}} (c_{v, \mathcal{N}_v} - c_{v, \mathcal{N}_v \cap Z}) \right).$$

Ainsi, pour un utilisateur u , le potentiel d'une zone fait référence au gain qui peut être obtenu en allouant cette zone à l'utilisateur u (en comparaison avec l'allocation précédente). Une zone Z à haut potentiel $P(u, Z)$ est un ensemble de SUs adjacents qui est bénéfique à cet utilisateur. En conséquence, APASS donne une plus haute priorité aux zones à fort potentiel. Notons que les zones à haut potentiel sont différentes pour chaque utilisateur puisque le potentiel d'une zone dépend de la qualité du canal de l'utilisateur.

Définition 7. (Zone de potentiel maximal).

La zone de potentiel maximal de l'utilisateur u , notée par Z_u , est une zone $Z \subseteq \mathcal{N}$ qui maximise $P(u, Z)$. Si plusieurs zones vérifient cette définition, alors Z_u fait référence indifféremment à l'une de ces zones.

Définition 8. (Zone de potentiel maximal avec contrainte de cardinalité).

Une zone de potentiel maximal avec cardinalité $y \in \mathbb{N}$, notée Z_u^y , est une zone $Z \subseteq \mathcal{N}$ de cardinal y exactement qui maximise $P(u, Z)$.

Cas particuliers :

- Si une telle zone n'existe pas, Z_u^y est l'ensemble vide ;
- Si plusieurs zones vérifient cette définition, Z_u^y fait référence indifféremment à une de ces zones .

La Figure 5.2 illustre le potentiel d'une zone. Le nombre indiqué dans chaque ressource correspond à l'indice du MCS d'ordre maximum qui peut être utilisé sur la ressource ($m_{u,k}$ pour l'utilisateur u et la ressource k). Ainsi, si tous les SUs dans Z_u sont alloués à u , le cardinal de Z_u (i.e. $|Z_u|$) est donc le nombre maximum de SUs qui peuvent être alloués à u sans enfreindre aucune contrainte. Dans cette figure, la zone Z_u^1 correspond au SU de plus haut potentiel et contigu à \mathcal{N}_u (les SUs déjà alloués à u). Dans ce cas, il n'y a qu'une seule possibilité pour Z_u^1 puisque le SU situé sur "la gauche" de \mathcal{N}_u supporte m_3 tandis que le SU situé "sur la droite" de \mathcal{N}_u supporte seulement m_2 . De même, Z_u^3 est la zone composée de trois SUs et de plus haut potentiel.

5.3.2 Dernière étape de l'allocation de ressources

Le principe de cette dernière routine est repris d'AGSS, l'algorithme présenté dans le chapitre 4, mais la méthode de sélection des utilisateurs est différente. Le concept de "zone de potentiel maximal" permet de définir la dernière étape d'APASS de façon simple (Algorithme 9). Tout d'abord, cette heuristique récupère l'ensemble des utilisateurs élus \mathcal{V} et la première allocation réalisée par l'Algorithme 8. Notre dernière routine est également basée sur une boucle principale. À chaque tour de boucle, un utilisateur est sélectionné. Nous notons u^* cet utilisateur.

Quand un utilisateur est sélectionné, l'allocation est graduellement étendue sur la droite et sur la gauche des ressources précédemment allouées. Notons qu'un utilisateur peut être sélectionné plusieurs fois durant l'exécution de l'algorithme, lui permettant d'augmenter rapidement le nombre de ressources qui lui sont allouées. Le processus de sélection dépend des poids calculés par les stratégies d'ordonnancement (voir Section 4.3).

input : Identique à l'Algorithme 7.
output : Une allocation valide $\{\mathcal{N}_u\}_{u \in \mathcal{V}}$ pour le TTI courant.
 /* Récupération des utilisateurs élus et leurs zones de potentiel maximal. */

```

1 Soit  $\{\mathcal{N}_u\}_{u \in \mathcal{V}}$  l'allocation réalisée par l'Algorithme 8.
2  $y \leftarrow 1$ 
3 Pour chaque utilisateur  $u$  :
4    $Z_u^y \leftarrow$  la zone de potentiel maximal de cardinal  $y$  de l'utilisateur  $u$  (Déf. 8)
5 do
6   /* Sélectionner un utilisateur */
7    $u^* \leftarrow \arg \max_{u \in \mathcal{V}} P(u, Z_u^y)$ 
8   si  $P(u^*, Z_{u^*}^y)$  est négatif alors
9     /* Augmenter la taille des zones */
10     $y \leftarrow y + 1$ 
11    Recalculer les zones  $Z_u^y$  selon Déf. 8.
12    /* Puis commencer un nouveau tour de boucle. */
13  sinon
14    /* Procéder à l'allocation de tous les SUs dans  $Z_{u^*}^y$  à  $u^*$ . */
15     $\mathcal{N}_{u^*} \leftarrow \mathcal{N}_{u^*} \cup Z_{u^*}^y$ 
16     $\forall u \neq u^*, \mathcal{N}_u \leftarrow \mathcal{N}_u \setminus Z_{u^*}^y$ 
17    pour chaque utilisateur  $u \in \mathcal{V}$  faire
18      | Recalculer les zones  $Z_u^y$  selon Déf. 8.
19    fin
20     $y \leftarrow 1$ 
21  fin
22 tant qu' il existe un utilisateur  $u$  tel que  $P(u, Z_u) > 0$ ;
23   /*  $Z_u$  fait référence à la zone de potentiel maximal de
24    l'utilisateur  $u$  (Déf. 7). */
25 Retourner  $\{\mathcal{N}_u\}_{u \in \mathcal{U}}$ .
```

Algorithme 9 : Finalisation de l'allocation de ressources.

Processus d'allocation. Tout comme pour AGSS, l'algorithme présenté dans le Chapitre 4, cette dernière routine a pour objectif d'allouer les ressources avec une granularité fine, i.e. l'ordonnanceur alloue un minimum de SU(s) à l'utilisateur élu à chaque tour de boucle. Cela permet de prendre des décisions plus précises et donc d'augmenter progressivement la valeur à maximiser. La granularité des allocations successives est définie par le paramètre y , qui représente le nombre de SUs à allouer à chaque itération.

Tout d'abord, pour chaque utilisateur, nous calculons la zone Z_u^y qui va être affectée à l'utilisateur u s'il est sélectionné. Le calcul de ces zones est une étape importante de notre heuristique. Chaque zone Z_u^y est calculée afin d'assurer que les SUs sont contigus et que l'utilisateur u puisse transmettre sur toutes les ressources de la zone. De plus, Z_u^y est un ensemble de cardinal y (ou zéro si cet ensemble n'existe pas). Ainsi, y SUs seront alloués si l'utilisateur considéré est sélectionné.

Paramètre de granularité. Le nombre y de SUs qui sont alloués à l'utilisateur sélectionné varie durant l'exécution de l'algorithme. Notre but est de garder ce paramètre de granularité aussi faible que possible afin d'allouer les ressources finement. Par conséquent, y est initialement fixé à 1. Cette valeur est modifiée à chaque fois que le processus d'élection échoue à sélectionner un utilisateur. En effet, le processus d'élection peut échouer si le potentiel de la "meilleure" zone $P(u^*, Z_{u^*}^y)$ est négatif ou nul.

Cela signifie que l'allocation des y SUs n'augmente le débit à aucun des utilisateurs, i.e. l'allocation des y SUs n'est bénéfique à personne. Nous résolvons ce problème en augmentant la valeur de y . En effet, si l'allocation des y SUs n'augmente le débit d'aucun utilisateur, il est possible que l'allocation de $y + 1$ SUs augmente tout de même le débit de certains.

Condition de terminaison. Si plus aucune zone n'est bénéfique aux utilisateurs ($P(u, Z_u) \leq 0, \forall u$), l'allocation est terminée.

5.3.3 Un mot sur la complexité

Tout d'abord, l'Algorithme 7 résout $\log_2(N)$ sous-problèmes d'affectations. Puisque la complexité temporelle de l'algorithme que nous avons utilisé afin de résoudre un problème d'affectation est d'au plus $\mathcal{O}(\max(U, N)^3)$ [93], alors la complexité de l'Algorithme 7 est en $\mathcal{O}(\max(U, N)^3 \log_2(N))$.

L'Algorithme 8 appelle à $U - V$ reprises l'Algorithme 7 et cherche l'utilisateur "le moins efficace" à chaque itération de la boucle. Nous en déduisons que la complexité temporelle de l'Algorithme 8 est en $\mathcal{O}(U \max(U, N)^3 \log_2(N))$.

Sans perte de généralité, nous considérons qu'à chaque itération de la boucle principale de l'Algorithme 9 un seul SU est alloué, i.e. la condition " $P(u^*, Z_{u^*}^y)$ est négatif" n'est jamais satisfaite. Par conséquent, le paramètre de granularité y est toujours égal à 1⁵. Ainsi, il y a au plus N itérations de la boucle principale pour une exécution complète. Le calcul d'une zone Z_u^1 (ligne 14) consiste seulement à évaluer le premier SU à la droite et à la gauche des SUs déjà alloués de l'utilisateur u (2 opérations), ce calcul est donc en $\mathcal{O}(1)$. La complexité de la boucle interne (ligne 13-15) est donc en $\mathcal{O}(U)$. Par conséquent, l'Algorithme 9 est rapide et sa complexité est simplement $\mathcal{O}(NV)$ (sans considérer l'exécution de l'Algorithme 8 au début de la routine).

Pour finir, la complexité globale d'APASS est $\mathcal{O}(U \max(U, N)^3 \log_2(N) + NV)$ ce qui correspond à $\mathcal{O}(U \max(U, N)^3 \log_2(N))$ puisque $NV < U \max(U, N)^3 \log_2(N)$.

5. Il n'y a pas de perte de généralité car si la condition " $P(u^*, Z_{u^*}^y)$ est négatif" est satisfaite, le paramètre y est incrémenté. À la prochaine itération de la boucle principale, deux SUs seront alloués.

Paramètre	Valeur
Temps simulé	15 minutes
Modèle du canal	Affaiblissement de propagation avec atténuation de Rayleigh
SNR de référence	31 dB correspond à une distance de référence d_{ref}
Mode de duplexage	Time Division Duplex (TDD) [11]
Rayon de la cellule	$3d_{ref}$
Largeur de bande	12,9 MHz (72 SUs)
Nombre de MCSs	6 MCSs (voir Tableau 3.2)
Choix du MCS	Équation 5.4.1 et Tableau 3.2
Nombre d'utilisateurs	66 à 107 utilisateurs (12,1 à 19,7 Mbit/s)
Nombre maximum d'utilisateurs ordonnancés par TTI	$V = 8$ utilisateurs [11]
Modèle de mobilité	Random Way Point [94]
Vitesse des utilisateurs	$[1/60, 10/60] d_{ref}.s^{-1}$
Période (pour OPA)	$T = 20$ s
Taux d'erreur binaire cible	10^{-3}
Paramètres de trafics	Voir Tableau 5.2

Tableau 5.1 – Paramètres de simulation.

5.4 Évaluation des performances

Tout comme pour le sens descendant, nous avons implémenté la partie RAN du sens montant de LTE-A (Release 13) sous le logiciel Riverbed Modeler [9]. Cette fois, nous supposons une largeur de bande de 12,9 MHz, ce qui correspond à 72 RBs (ou 72 SUs).

Modélisation de la couche physique. Nous considérons le même modèle que dans le Chapitre 3 (Section 3.6.1) avec P_{max} la puissance maximale des utilisateurs. Notons que, dans ce cas, la station de base peut estimer les SNRs des utilisateurs directement. Cependant, dans le sens montant, la méthode EESM (*Exponential Effective S(I)NR Mapping*) ne peut être utilisée pour approximer le SNR effectif [95, 96]. Par ailleurs, le SNR effectif peut être calculé sur chaque SU. Le SNR effectif de l'utilisateur u pour le SU k est alors calculé comme la moyenne des valeurs $\{SNR_{u,n}\}$, avec n l'indice d'une sous-porteuse du SU k :

$$SNR_{eff,u} = \frac{1}{N_{sc}} \sum_{n=1}^{N_{sc}} SNR_{u,n}, \quad (5.4.1)$$

avec N_{sc} ($= 4$) le nombre de sous-porteuses de référence sur chaque SU. La méthode de sélection du MCS est elle identique à celle présentée en Section 3.6.1.

Trafics. Chaque utilisateur est associé à trois flux de données. Nous reprenons les applications de vidéo-conférence et d'appel vocal présentées dans le chapitre précédent. De plus, nous ajoutons à chaque utilisateur un trafic de radiodiffusion de haute qualité. Nous supposons que cette application utilise une compression WMA 320 kbit/s. Ainsi, l'application génère des paquets de 320 bits toutes les millisecondes. Le délai maximum toléré par cette application est fixé à 500 ms (application temps réel mais non-interactive). Le Tableau 5.2 résume les paramètres des trafics.

Application	T_f	Débit moyen
Vidéo-conférence	80 ms	220 kbit/s
Appel vocal	60 ms	8 kbit/s
Radiodiffusion	500 ms	320 kbit/s

Tableau 5.2 – Paramètres des applications.

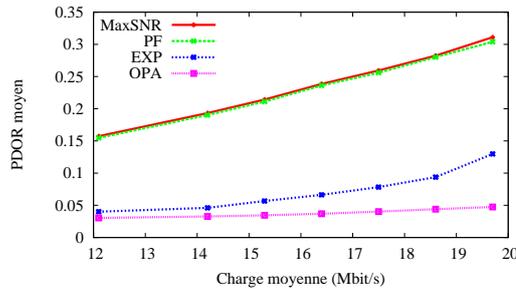


Fig. 5.3 – PDOR moyen en fonction de la charge du système pour les différentes stratégies.

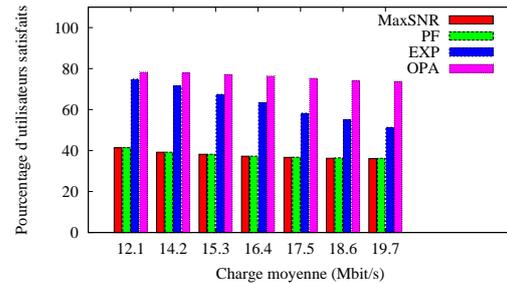


Fig. 5.4 – Pourcentage d'utilisateurs satisfaits en fonction de la charge du système.

5.4.1 Comparaison des stratégies d'ordonnement

Nous montrons une nouvelle fois les bénéfices d'OPA, notre stratégie d'allocation de ressources, utilisée conjointement avec APASS. Notre stratégie est comparée avec le MaxSNR, le PF et EXP⁶. Comme nous avons déjà étudié en détail la comparaison des stratégies d'ordonnement dans le Chapitre 4, nous passons plus rapidement ici sur les résultats.

La Figure 5.3 montre une comparaison, en matière de PDOR moyen, des différentes stratégies. Contre toute attente, le PDOR moyen obtenu avec le MaxSNR est le même que celui obtenu avec le PF. Cela montre que le MaxSNR et le PF sont quasiment équivalents avec APASS. Ce résultat peut paraître surprenant car il est bien connu que le PF permet d'obtenir de meilleures performances que le MaxSNR sur le long terme [70], comme nous l'avons d'ailleurs observé dans le chapitre précédent. En fait, avec APASS, le choix de la première allocation (première routine) définit essentiellement les performances sur l'ensemble du TTI. Puisque ces deux stratégies favorisent les utilisateurs avec une bonne qualité de canal, leurs performances sont similaires. Pour chaque stratégie, la Figure 5.4 montre le ratio d'utilisateurs satisfaits en fonction de la charge du système. Notre stratégie surpasse le PF et le MaxSNR en termes de PDOR et d'utilisateurs satisfaits. De plus, OPA obtient également un gain de performances significatif comparé à la stratégie EXP. Avec l'accroissement de la charge, ce gain semble augmenter. Les bons résultats d'OPA s'expliquent par la prise en compte du PDOR (et du PDOR cible) directement dans l'expression du calcul des poids. En effet, le PDOR est particulièrement adapté aux trafics multimédias et à notre objectif (EXP

6. Ces stratégies sont présentées en Section 2.2.1 et 4.3. Nous avons fait le choix de ne pas intégrer M-LWDF dans la présentation des résultats, car ce dernier obtenait toujours des performances inférieures par rapport à la stratégie EXP.

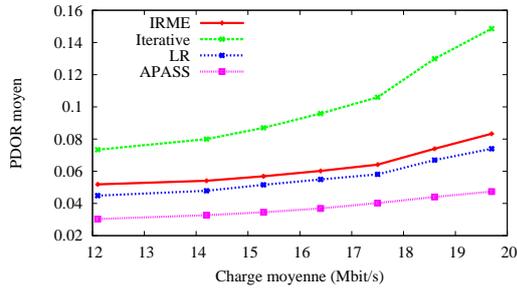


Fig. 5.5 – PDOR moyen en fonction de la charge du système (pour les différents algorithmes).

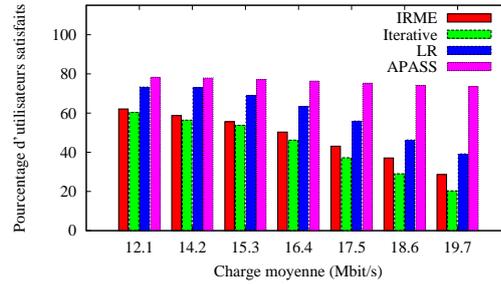


Fig. 5.6 – Pourcentage d'utilisateurs satisfaits en fonction de la charge du système.

considère seulement le délai du paquet en tête du buffer).

Ces résultats montrent, une nouvelle fois, que le choix de la stratégie a un impact significatif sur les performances de notre algorithme.

5.4.2 Comparaison des algorithmes

Nous comparons ensuite APASS avec d'autres algorithmes populaires présentés dans la Section 2.3.2 : IRME[62], Iterative [64] et Local Ratio [66]. Comme OPA est la stratégie la plus efficace pour notre objectif, nous avons utilisé uniquement cette stratégie lors des simulations.

Le PDOR moyen des différents algorithmes est évalué à travers la Figure 5.5. De même, la Figure 5.6 montre le ratio d'utilisateurs satisfaits en fonction de la charge du système. IRME et Local Ratio obtiennent des performances similaires en matière de PDOR, tandis qu'Iterative est moins efficace. En effet, Iterative alloue les ressources une par une en déterminant successivement la paire "(utilisateur, SU)" qui maximise un *gain* [64]. Comme attendu, la simplicité de cet algorithme ne lui permet pas de supporter la charge. En fournissant un système de classement entre les paires "(utilisateur, SU)", IRME offre de meilleures performances par rapport à Iterative. Pour finir la technique du Local Ratio montre une nouvelle fois qu'il s'agit d'une méthodologie efficace pour la conception d'algorithmes.

Contrairement à Iterative et IRME, Local Ratio et APASS ont la possibilité d'allouer plus d'un SU au sein d'une même itération de boucle, ce qui est adapté pour traiter la contrainte du *Single MCS* (tant que la granularité des allocations reste faible). Quel que soit la charge du système, le PDOR moyen obtenu avec APASS est le plus faible. En moyenne, APASS est deux fois plus performant que Local Ratio et obtient un gain de 100% en matière de PDOR avec la même fonction d'utilité (Figure 5.5). Pour des charges importantes, le ratio d'utilisateurs satisfaits est également multiplié par deux (Figure 5.6) comparé à Local Ratio.

Ces résultats valident l'efficacité d'APASS et montrent l'importance d'intégrer la prise en compte des contraintes LTE au sein de l'algorithme.

Pour finir, nous comparons les algorithmes avec des métriques communes, et nous montrons leurs relations avec le PDOR. La Figure 5.7 présente le délai moyen des paquets en fonction de la charge du système. De même, la Figure 5.8 compare les

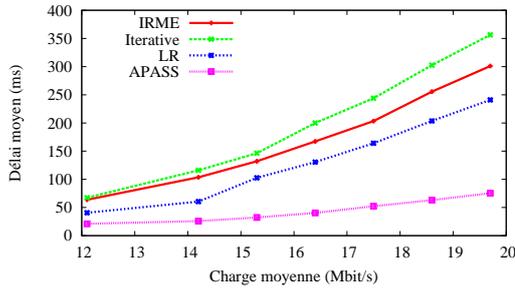


Fig. 5.7 – Délai moyen en fonction de la charge du système.

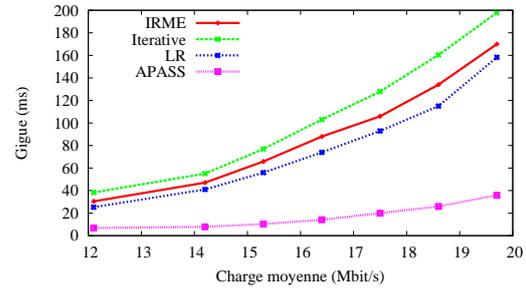


Fig. 5.8 – Gigue en fonction de la charge du système.

algorithmes en termes de gigue (écart type moyen du délai des paquets). Dans ces deux figures, les courbes ont des tendances similaires à celles de la Figure 5.5. De ce fait, notre solution aide efficacement à la réduction du délai des paquets et de la gigue. Cela s'explique par le fait que le PDOR a une influence directe sur ces deux métriques (voir Section 4.1).

5.5 Conclusion

Dans ce chapitre, nous avons construit un algorithme d'allocation de ressources pour le sens montant de LTE. Notre solution est composée de trois routines qui fonctionnent ensemble pour fournir une solution efficace. En particulier, le concept de *zone de potentiel maximal* est introduit pour abstraire les contraintes LTE et permet de finaliser l'allocation de ressources avec flexibilité. De plus, APASS est construit pour être adapté à de nombreux objectifs via le choix d'une *stratégie d'ordonnement* adaptée. L'évaluation de performance montre qu'APASS obtient un gain de 100 % en matière d'utilisateurs satisfaits comparé à Local Ratio (Figure 5.6), une solution reconnue efficace de l'état de l'art.

Nous avons également montré que, dans un contexte multimédia, OPA surclasse les stratégies d'ordonnement existantes (de même que pour le sens descendant).

Chapitre 6

Conclusion

6.1 Synthèse

Dans le cadre de cette thèse, nous avons étudié le problème de l'allocation de ressources dans les réseaux LTE. L'originalité de nos travaux réside dans la prise en compte des contraintes de la norme. Nous avons travaillé autour de ces dernières pour construire des solutions efficaces et adaptées. Par conséquent, nos algorithmes sont susceptibles d'être implémentés dans un système réel.

Les contraintes que nous avons mises en avant modifient la façon d'allouer les ressources. En particulier, la contrainte du *Single MCS* et le calcul non-linéaire des débits complexifient le système en diminuant les libertés offertes à l'ordonnanceur. Il en résulte que les problèmes sont NP-Difficiles et ne peuvent donc pas être résolus de façon optimale avec un algorithme polynomial. C'est pourquoi, nous avons élaboré des solutions sous-optimales mais rapidement calculables.

Dans le sens descendant, nous avons tout d'abord proposé un algorithme d'approximation efficace lorsque les buffers sont "toujours pleins". Puis, nous avons souhaité atteindre un objectif plus ambitieux : satisfaire des applications multimédias sur le long terme. C'est pourquoi nous avons conçu AGSS, qui offre un haut niveau de différenciation de services s'il est utilisé avec une *stratégie ordonnancement* adaptée. L'objectif d'AGSS est de conserver l'avantage de l'algorithme classique tout en allouant les ressources plus efficacement.

Dans le sens montant, de nouvelles contraintes s'appliquent. En effet, seules les ressources contiguës dans le domaine fréquentiel peuvent être allouées à un utilisateur. De plus, le nombre maximum d'utilisateurs ordonnancés sur un *time slot* est borné. Les libertés d'actions en sont alors d'autant plus réduites. Nous avons donc proposé APASS, une solution dédiée au sens montant de LTE. Cet algorithme est composé de plusieurs étapes qui répondent à des objectifs spécifiques pour aboutir à une allocation prenant en compte l'ensemble des spécificités de la norme.

Nos simulations ont montré que nos algorithmes sont efficaces et apportent souvent de meilleures performances en matière de pourcentage d'utilisateurs satisfaits que les solutions de l'état de l'art. Ainsi, cette thèse a démontré à plusieurs reprises que la considération minutieuse des spécificités de LTE au sein de l'algorithme d'allocation permet d'aboutir à des allocations de bien meilleure qualité.

6.2 Perspectives.

L'allocation de ressources est un vaste domaine de recherche et de nombreuses pistes s'ouvrent suite à nos travaux.

Problème de minimisation. Nous avons majoritairement traité des problèmes de maximisation même si d'autres objectifs peuvent être soulevés en sélectionnant une stratégie d'allocation adaptée. Une piste intéressante serait d'étudier plus formellement des problèmes de minimisation et d'adapter nos algorithmes en conséquence.

Caractéristiques des trafics. Dans cette thèse, nous avons :

- soit traité un problème sous l'hypothèse des utilisateurs gourmands (buffers toujours pleins) ;
- soit nous n'avons pris aucune hypothèse sur la nature et les caractéristiques des trafics, ceci afin que nos solutions soient valides quelle que soit l'application générant le flux de données.

Une approche "à mi-chemin" peut être envisagée. En effet, faire des suppositions sur les caractéristiques statistiques/probabilistes des trafics permettrait de prédire le comportement des flux. Avec ce type d'hypothèse, il est ainsi possible d'obtenir des résultats fondamentaux sur les performances des allocations. Quelques travaux ont déjà étudié ce type de problème [97] ; cependant, à notre connaissance, aucun n'a été proposé en considérant les contraintes très restrictives de LTE.

Mécanisme d'accès au réseau. Nous avons supposé l'existence d'un mécanisme d'accès au réseau performant afin que nos solutions puissent s'appliquer. Cependant, la conception d'un mécanisme d'accès performant n'est pas nécessairement aisée si l'on souhaite éviter un maximum les faux-positifs [98]. Ce sujet de recherche soulève également de nombreuses problématiques en matière de sécurité [99].

Impact de la mobilité. L'allocation de ressources est fortement dépendante de la qualité du canal expérimentée par chaque utilisateur. D'autre part, la mobilité des utilisateurs influence grandement la qualité du canal perçue. Dans l'ensemble de cette thèse, nos simulations ont été réalisées avec des utilisateurs qui se déplacent aléatoirement au sein de la cellule. Il serait donc intéressant d'étudier plus précisément l'impact de la mobilité sur l'allocation de ressources. La détection d'un schéma de mobilité (voyage en train, en voiture, à pieds, etc.) pourrait même être utilisée pour modifier les allocations. Il serait également possible d'utiliser les informations de mobilité d'un utilisateur afin d'estimer la qualité de canal qu'il percevra dans un futur proche.

Interférences entre cellules. Une autre hypothèse forte de notre système est que nous n'avons considéré qu'une seule cellule. En réalité, plusieurs cellules mitoyennes peuvent entrer en interférences à leurs extrémités. Diminuer ces interférences peut-être réalisé en allouant moins de ressources aux utilisateurs qui se situeraient à l'intersection de deux cellules. Une meilleure solution consisterait à allouer aux utilisateurs de bordures des ressources qui correspondent à des bandes de fréquences qui ne rentrent pas en conflit avec les fréquences utilisées par la cellule adjacente considérée [100].

Allocation de ressources pour la 5G. La méthode d'accès la plus probable pour la 5G est appelée NOMA (*Non-Orthogonal Multiple Access*) [101]. Les utilisateurs sont superposés dans le domaine de la puissance et le mode de décodage est basé sur l'annulation d'interférences successives. Avec NOMA, l'allocation de ressources est donc complètement différente. Ce nouveau sujet de recherche est très prometteur et sera sûrement très étudié dans les années à venir. La première version du standard 3GPP pour la technologie 5G est prévue pour l'année 2020.

Annexe A

Description du RAN de LTE

Cette première annexe décrit la partie RAN (*Radio Access Network*) de LTE.

Il est en effet essentiel de décrire la structure d'une trame LTE ainsi que l'architecture des protocoles puisque ces éléments interviennent directement dans :

- le modèle étudié ;
- le modèle et les paramètres de simulation.

Plus précisément, avant de proposer un algorithme d'allocation de ressources, plusieurs questions concernant la faisabilité se posent :

- *Qu'est-ce qu'une ressource ?*
- *Comment une ressource peut-elle être allouée ? Quelle est la granularité de l'allocation ?*
- *Comment sont distingués les sens de transmission ? Quel est l'impact sur l'allocation de ressources ?*
- *Comment l'ordonnanceur peut-il connaître la qualité (SNR) de chaque ressource ? Comment ces informations sont-elles échangées ?*
- *Est-il possible que plusieurs flux d'un même utilisateur utilisent les ressources d'un même slot ?*
- *Combien d'utilisateurs peuvent être ordonnancés dans un même slot (TTI) ?*

L'objectif de cette annexe est de répondre à ces questions.

A.1 L'architecture des protocoles RAN

Le réseau d'accès radio (RAN pour *Radio Access Network*) est composé de plusieurs couches protocolaires afin d'assurer et de fiabiliser l'accès aux ressources radios. On peut considérer que l'ordonnanceur est situé au niveau MAC même si cet élément a accès à de nombreuses informations des autres couches protocolaires. Les protocoles sont brièvement décrits ci-dessous et sur la figure [A.1](#) :

- PDCP (*Packet Data Convergence Protocol*), qui se charge de la compression des en-têtes IP ainsi que du chiffrement (si besoin). Pour cela, une en-tête PDCP est ajoutée.
- RLC (*Radio Link Control*), dont le service principal est de segmenter/assembler des PDCP-PDUs (segments PDCP). Par conséquent, les données des flux d'un

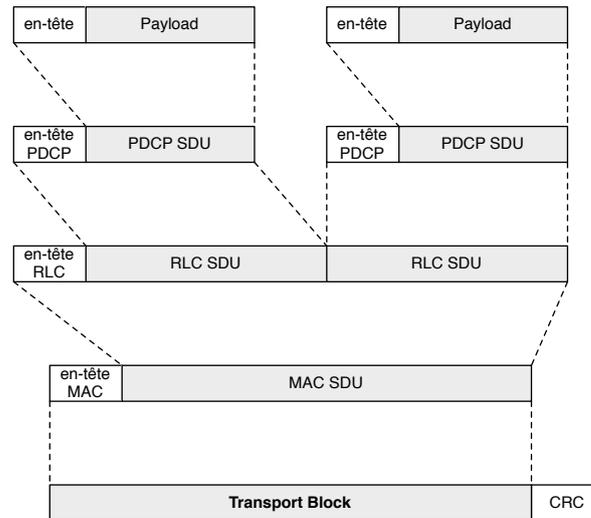


Fig. A.1 – Exemple d’encapsulation successive de deux flux de données à travers l’architecture des protocoles RAN de LTE.

même utilisateur sont multiplexées dans un RLC-PDU. C’est ce qui explique pourquoi plusieurs flux peuvent être transmis sur un même TTI. Les tailles des RLC-PDUs (segment RLC) sont dynamiques afin d’éviter le *padding* ou un montant trop important d’*overhead*.

- MAC (*Medium Access Control*) qui multiplexe les RLC-PDUs. Cela permet donc de multiplexer plusieurs flux de données. La couche MAC assure également les retransmissions (H-ARQ, *Hybrid Automatic ReQuest*). L’en-tête MAC associée à sa charge utile (*payload*) constitue un *transport block*. De plus, l’ordonnancement est une fonctionnalité majeure de la couche MAC.
- PHY (couche PHYsique) qui fournit les fonctionnalités de codage et de modulation du signal.

Dans la suite, nous négligeons la taille des en-têtes protocolaires. Cependant, toutes nos solutions restent valides puisqu’il suffit d’ajouter à chaque paquet un montant fixe de bits afin de simuler les en-têtes.

A.2 Structure de la trame LTE

La Figure A.2 représente une trame physique LTE. Le temps de transmission d’une trame T_{trame} est égal à 10 ms [102]. Une trame est ensuite divisée en dix sous-trames de durée 1 ms. Enfin, une sous-trame est elle-même composée de deux *slots* de 0,5 ms. Dans le domaine fréquentiel, un slot est divisé en Ressource Blocks (RBs). Ces RBs correspondent à 12 sous-porteuses de largeur de bande égale à 15 kHz. Ainsi, la largeur de bande d’un RB est de 180 kHz. Le nombre de RBs dans le domaine fréquentiel varie de 6 à 100 en fonction de la largeur de bande totale du système (entre 1 MHz et 20 MHz). Évidemment, plus le nombre de RBs est important, plus la capacité du système sera grande. La norme définit 6 ou 7 symboles OFDM par RB en fonction de la taille

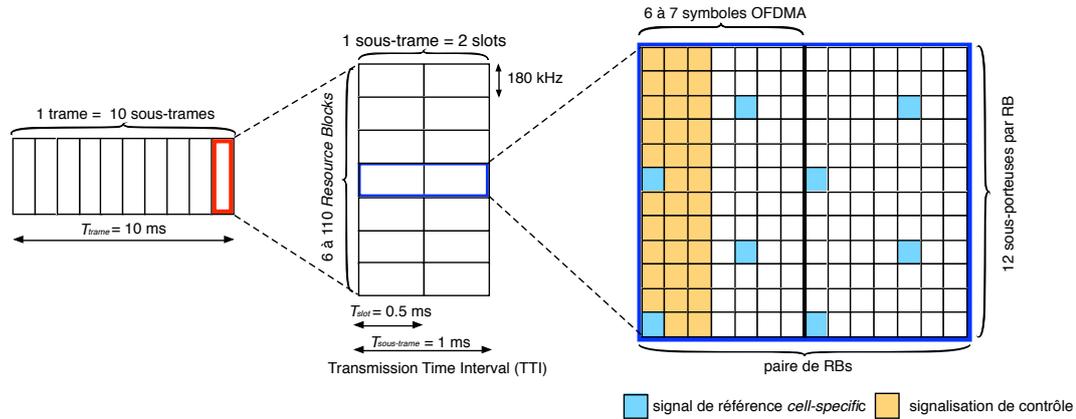


Fig. A.2 – Structure d'une trame LTE.

du *préfixe cyclique* situé à la fin de chacun des symboles OFDM. Nous désignons par T_s la durée d'un symbole OFDM. Les préfixes cycliques sont utilisés pour réduire l'Interférence Entre Symboles (IES). Un *resource element* désigne une sous-porteuse pendant la durée de transmission d'un symbole OFDM. Si l'on considère le préfixe cyclique *normal* (le plus court), il y a 7×12 , i.e. 84 *resource elements*.

L'allocation de ressources est un processus périodique de période appelée TTI (*Transmission Time Interval*). Un TTI correspond à la durée d'une sous-trame. La plus petite unité de données allouable par l'ordonnanceur est une **paire de RBs** adjacents dans le domaine temporel. La partie droite de la Figure A.2 illustre une paire de RBs.

Dans la suite, par abus de langage, nous utiliserons le terme RB pour faire référence à une paire de RBs¹.

En outre, une sous-trame est composée de deux régions :

- La *région de contrôle* située au début de chaque sous-trame. Sa taille varie dynamiquement en fonction de la quantité d'informations de contrôle, allant de 1 à 3 symboles OFDM. Ces informations de contrôle (appelées *L1/L2 control signaling*) indiquent à chaque mobile (1) la taille de la région de contrôle (et donc la taille de la région de données), (2) les RBs alloués à l'utilisateur, (3) les acquittements des données si nécessaires (H-ARQ), et (4) des informations concernant la couche physique (pour la démodulation).
- La *région de données* principalement utilisée pour la transmission des données utilisateurs mais qui peut contenir des informations de signalisation supplémentaires tels que des signaux de référence (*cell-specific reference signals*).

A.3 Mode de duplexage

Deux modes de duplexage sont définis dans LTE [11] :

- FDD (Frequency Division Duplex). Dans ce mode, les bandes dédiées au sens

1. Cet abus de langage est également présent dans la norme.

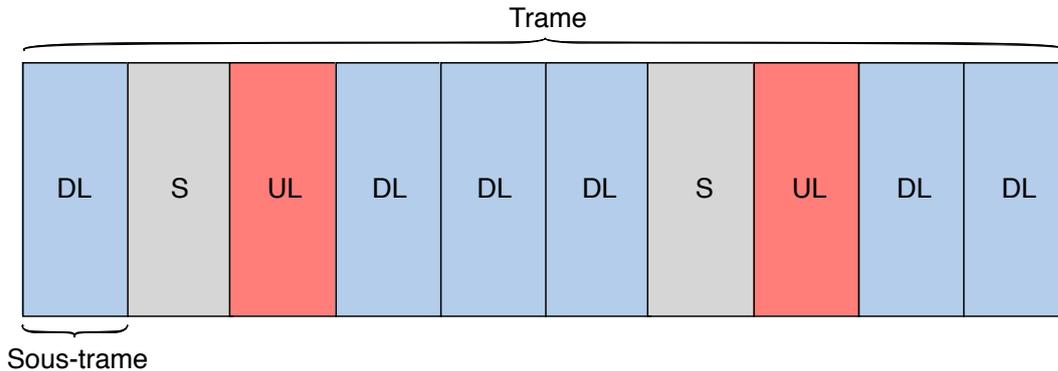


Fig. A.3 – Exemple d’une trame dans le mode TDD. Les trames dédiées au sens montant sont désignées par *UL*, les trames dédiées au sens descendant sont désignées par *DL*, et les trames spéciales sont désignées par *S*.

descendant et au sens montant sont séparées par une *bande de garde*. Un mobile peut donc à la fois transmettre et recevoir des données.

- TDD (Time Division Duplex). Les sens montant et descendant sont séparés dans le domaine fréquentiel. Chaque sous-trame est soit dédiée au sens montant soit au sens descendant.

Un problème majeur du mode FDD est l’ajout de complexité au niveau des équipements mobiles. En effet, chaque équipement utilisateur doit être capable de transmettre et de recevoir en même temps sur des fréquences différentes. Cela a un impact non négligeable sur le prix de ces équipements. De plus, le mode FDD ne peut pas être utilisé si l’opérateur n’a pas la largeur de bande nécessaire pour mettre en place une *bande de garde* de taille suffisante. Par conséquent, le mode FDD manque de flexibilité dans un contexte mobile. Dans cette thèse, nous nous appuyons sur le mode TDD pour expliquer nos contributions. Cependant, comme nous dissociions le sens descendant et le sens montant, nos solutions peuvent facilement être adaptées au mode FDD.

Comme expliqué dans la section précédente, une trame est une séquence de 10 sous-trames. Avec le mode TDD, chaque sous-trame est, soit dédiée au sens montant, soit au sens descendant. Cependant, des sous-trames spéciales permettent de laisser suffisamment de temps aux mobiles pour passer du mode transmission au mode réception (ou inversement). La Figure A.3 illustre un exemple de trame TDD. Les trames spéciales ne sont pas utilisées pour la transmission de données utilisateurs. La norme définit différentes configurations pour la succession des sous-trames montantes/descendantes. Ces configurations sont résumées dans le Tableau A.1. Le choix de la configuration est relativement statique dans une cellule [11]. Il n’est donc pas possible de modifier dynamiquement la configuration.

A.4 Détermination de la taille d’un *transport block*

Dans un contexte multimédia, un mobile peut envoyer plusieurs flux en parallèle. Dans ce cas, les données de chaque flux sont encapsulées dans un RLC-PDU qui est lui-

#	sf0	sf1	sf2	sf3	sf4	sf5	sf6	sf7	sf8	sf9
0	DL	S	UL	UL	UL	DL	S	UL	UL	UL
1	DL	S	UL	UL	DL	DL	S	UL	UL	DL
2	DL	S	UL	DL	DL	DL	S	UL	DL	DL
3	DL	S	UL	UL	UL	DL	DL	DL	DL	DL
4	DL	S	UL	UL	DL	DL	DL	DL	DL	DL
5	DL	S	UL	DL						
6	DL	S	UL	UL	UL	DL	S	UL	UL	DL

Tableau A.1 – Configurations des sous-trames pour le mode TDD. Par exemple, la première configuration est constituée d’une sous trame DL, d’une sous-trame spéciale, de trois sous-trames UL, d’une sous trame DL, d’une sous-trame spéciale et de trois sous-trames UL. Ce schéma est ensuite répété.

même multiplexé dans la couche MAC. Un *transport block* (ou MAC-PDU) se compose d’une en-tête MAC et un ou plusieurs RLC-PDUs. Si une seule antenne est utilisée, un seul transport block par utilisateur peut être transmis durant un TTI. En MIMO (mode multi-antennes), deux *transport blocks* peuvent être transmis durant un TTI.

Pour un mobile, la taille d’un *transport block* dépend de deux facteurs principaux :

- la qualité du lien radio du mobile ;
- le nombre de RBs alloués à ce mobile par l’ordonnanceur.

Notons qu’il n’est possible d’utiliser qu’un seul MCS par *transport block* pour la transmission de données (quel que soit le nombre de flux du mobile). C’est ce qui explique la contrainte du *Single MCS*.

A.5 Estimation de la qualité du canal

Dans toutes les versions de LTE / LTE-A / LTE-B, les signaux de références *cell specific* fournissent des informations essentielles pour l’estimation de la qualité du canal. Ces signaux de référence sont envoyés sur des *resource elements* avec une puissance pré-déterminée permettant au récepteur d’estimer la qualité du canal sur les ressources proches. Ces signaux sont espacés dans les domaines temporels et fréquentiels de manière régulière (comme illustré dans la Figure A.2).

À titre d’information, d’autres types de signaux de référence sont présents et peuvent être utilisés pour estimer la qualité du canal. Cependant, les signaux de référence *cell specific* permettent d’obtenir la meilleure granularité pour la récupération d’informations concernant la qualité des ressources.

Dans le cas du sens montant, la station de base peut estimer la qualité de chaque signal de référence et utiliser ces informations pour l’allocation de ressources. Pour le sens descendant, les mobiles estiment la qualité du canal grâce aux signaux de référence envoyés par la station de base puis lui transmettent les résultats de ces estimations via des rapports d’information appelés *channel-state reports*. Il est important de savoir comment les *channel-state reports* sont envoyés et quelles sont les informations contenues dans ces rapports car ils définissent la matrice $\mathcal{I} = (m_{u,k})_{u,k}$ représentant la qualité du canal de chaque utilisateur.

Pour réduire la quantité de signalisation, la bande passante totale est divisée en sous-bandes. Chaque sous-bande est constituée d'un ensemble de 4 à 8 RBs consécutifs en fonction de la largeur de bande totale du système. Les *channel-state reports* envoyés par le mobile à la station de base contiennent un CQI (*Channel Quality Indicator*) spécifique pour chaque sous-bande. Un CQI est un entier codé sur 4 bits (correspondant à 15 valeurs de CQI différentes), qui est interprété par la station de base comme le meilleur MCS qui peut être utilisé sur les ressources appartenant à la sous-bande considérée. Ainsi pour un utilisateur u et pour tout $k \in \mathcal{N}$ appartenant à une sous-bande particulière, les $m_{u,k}$ sont identiques. Ceci permet d'éviter qu'un CQI ne soit transmis pour chaque RB, et donc de réduire le volume de signalisation.

Grâce à ces rapports, l'ordonnanceur connaît la qualité du canal avec une granularité de 4-8 RBs (en fonction de la largeur de bande totale). En outre, dans le mode TDD, comme les *channel-state reports* sont transmis uniquement sur les sous-trames UL (dédiées au sens montant), le nombre de rapports envoyés par un mobile sur une trame entière dépend de la configuration choisie (Tableau A.1). Par exemple, en utilisant la deuxième configuration, les CQIs sont mis-à-jour deux fois par trame puisqu'il y a seulement deux sous-trames UL non-adjacentes.

A.5.1 Allocation dans le sens descendant

Dans le sens descendant, une fois que l'ordonnanceur a pris une décision, un certain nombre de RBs sont alloués à chaque utilisateur. De plus, un MCS (*Modulation-and-Coding Scheme*) est également affecté à chacun d'eux en prenant en compte les CQIs et potentiellement toutes les autres informations à la disposition de l'ordonnanceur. Il faut alors annoncer aux utilisateurs quelles ressources leurs ont été allouées afin qu'ils puissent décoder les données correctement. Cette annonce de l'allocation de ressources est implémentée en utilisant des *rapports d'allocations*.

La façon la plus simple d'annoncer les ressources allouées à un utilisateur est d'utiliser un *bitmap* de longueur égal au nombre de RBs dans une sous-trame. Cependant, un tel *bitmap* générera un surplus de signalisation non-souhaitable. Afin de réduire cette quantité de signalisation, LTE introduit trois modes d'allocation pour le sens descendant [14]. Nous mettons l'accent uniquement sur le mode d'allocation appelé *allocation de ressources de type 0* puisqu'il apporte le plus haut niveau de diversité fréquentielle. Ce mode regroupe les RBs en *Resource Block Groups* (RBGs). Les RBs d'un RBG doivent être contigus dans le domaine fréquentiel. Chacun des RBGs est composé de 2 à 8 RBs en fonction de la largeur de bande totale. Un rapport d'allocation contient un *bitmap* de la longueur du nombre de RBGs. Par conséquent, dans le sens descendant, la plus petite unité allouable par l'ordonnanceur est un RBG. C'est la raison pour laquelle la granularité des allocations est différente pour le sens descendant par rapport au sens montant.

A.5.2 Allocation dans le sens montant

L'accès au sens montant utilise une forme modifiée de l'OFDMA appelée SC-FDMA (*Single Carrier Frequency Division Multiple Access*). Cette méthode d'accès obtient des performances similaires à l'OFDMA en termes de capacité [103]. De plus, le facteur

Indices des MCS	Nombre de (paires de) RBs alloués				
	1	2	3	4	5
1	16	32	56	88	120
2	24	56	88	144	176
3	32	72	144	176	208
4	56	120	208	256	328

Tableau A.2 – Extrait du tableau permettant de déterminer la taille (en bits) des données incluses dans un *transport block*.

de crête² de SC-FDMA est bien plus faible que celui de l’OFDMA. Par conséquent, la puissance énergétique requise dans le sens montant est plus faible que dans le sens descendant. Ceci est parfaitement adapté à la durée de vie limitée des batteries au sein des équipements mobiles.

En contrepartie, SC-FDMA impose une contrainte majeure sur l’allocation de ressources : toutes les ressources allouées à un utilisateur doivent être contiguës dans le domaine fréquentiel.

Cette contrainte facilite cependant l’annonce des ressources allouées aux utilisateurs. En effet, il suffit d’indiquer le premier et le dernier RB à allouer à un utilisateur pour attribuer toutes les ressources situées entre ces deux RBs. La signalisation est donc beaucoup moins importante qu’un *bitmap*. Cela implique que dans le sens montant, la plus petite unité allouable par l’ordonnanceur est le RB (au lieu du RBG pour le sens descendant).

A.5.3 Taille des Transport Blocks

Afin d’éviter que les tailles des *transport blocks* soient trop variables³, la norme utilise un ensemble pré-défini de tailles pour les transport blocks. Les tailles sont référencées dans un tableau indexé par deux paramètres (1) le nombre de RBs alloués à l’utilisateur concerné et (2) le MCS affecté à cet utilisateur. Le Tableau A.2 donne un extrait de ce tableau. Une version complète peut être trouvée dans [14]. C’est pourquoi la “fonction débit” n’est pas linéaire. À notre connaissance, nos travaux sont les seuls à prendre en compte cette table dans le processus d’allocation de ressources.

2. *peak to average ratio* en anglais.

3. En effet, les tailles doivent être multiples de 8 bits et il est essentiel de réduire le nombre de tailles possibles afin de faciliter les retransmissions.

Annexe B

Publications

B.1 Articles publiés

A. Ragaleux, S. Baey et C. Gueguen, “*Adaptive and Generic Scheduling Scheme for LTE/LTE-A Mobile Networks*”, Wireless Networks, 2015.

A. Ragaleux et S. Baey, “*Allocation de Ressources dans les Réseaux Cellulaires pré-5G*”, AlgoTel 2016.

A. Ragaleux, S. Baey et A. Fladenmuller, “*An Efficient and Generic Downlink Resource Allocation Procedure for pre-5G Networks*”, Wireless Communications and Mobile Computing, 2016.

B.2 Articles en cours de soumission

A. Ragaleux, S. Baey et M. Karaca, “*Standard-compliant LTE-A Uplink Scheduling with QoS Awareness*”, IEEE Transactions on Vehicular Technology (**Round 3**).

B.3 Rapports techniques

A. Ragaleux et S. Baey, “*Generic Downlink Resource Allocation Procedure for pre-5G Networks*”, Technical Report, http://www-npa.lip6.fr/~ragaleux/TR_01.pdf.

A. Ragaleux, S. Baey et M. Karaca, “*Generic Scheduling Scheme for LTE-A Uplink with Multimedia Traffic*”, Technical Report, http://www-npa.lip6.fr/~ragaleux/TR_02.pdf .

Bibliographie

- [1] O. Signal, “LTE coverage worldmap,” (2015). URL : <http://opensignal.com/>. 1.1
- [2] N. Solutions., “LTE deployment map,” (2015). URL : <http://ltemaps.org/home/>. 1.1
- [3] A. Pokhariyal, T. Kolding, et P. Mogensen, “Performance of downlink frequency domain packet scheduling for the UTRAN long term evolution,” dans “IEEE PIMRC,” p. 1–5. 1.1, 2.2.1
- [4] P. Viswanath, D. Tse, et R. Laroia, “Opportunistic beamforming using dumb antennas,” *IEEE Transactions on Information Theory* **48**, 1277–1294 (2002). 1.1, 1, 6, 4.4.1
- [5] X. Liu, E. K. P. Chong, et N. B. Shroff, “A framework for opportunistic scheduling in wireless networks,” *Computer Networks* **41**, 451–474 (2003). 1.1
- [6] C.-L. Chen, J.-W. Lee, C.-Y. Wu, et Y.-H. Kuo, “Fairness and QoS guarantees of WiMAX OFDMA scheduling with fuzzy controls,” *EURASIP J. Wireless Comm. and Networking* **2009**, 1–14 (2009). URL : <http://dx.doi.org/10.1155/2009/512507>. 1.1
- [7] G. Song et Y. Li, “Cross-layer optimization for OFDM wireless networks-part i : theoretical framework,” *IEEE Transactions on Wireless Communications* **4**, 614–624 (2005). 1.1
- [8] G. Song et Y. Li, “Cross-layer optimization for OFDM wireless networks-part ii : algorithm development,” *IEEE Transactions on Wireless Communications*, **4**, 625–634 (2005). 1.1
- [9] RiverBed, “Riverbed modeler with wireless suite,” (2015). URL : <http://www.riverbed.com/>. 1.2, 3.6, 5.4
- [10] P.-H. Huang, Y. Gai, B. Krishnamachari, et A. Sridharan, “Subcarrier allocation in multiuser OFDM systems : Complexity and approximability,” dans “Wireless Communications and Networking Conference (WCNC),” (2010), p. 1–6. 3
- [11] E. Dahlman, S. Parkvall, et J. Skold, *4G : LTE/LTE-Advanced for Mobile Broadband* (Elsevier, 2011). URL : <http://books.google.fr/books?id=DLbsq9GD0zMC>. 1.3.1, ??, ??, 5.1, A.3, A.3
- [12] R. Kwan, C. Leung, et J. Zhang, “Multiuser scheduling on the downlink of an LTE cellular system,” *Rec. Lett. Commun.* **2008**, 3 :1–3 :4 (2008). URL : <http://dx.doi.org/10.1155/2008/323048>. 1.3.2, 2.3.1, 2.3.1, 2.4, 3.6.1

- [13] H.-S. Liao, P.-Y. Chen, et W.-T. Chen, “An efficient downlink radio resource allocation with carrier aggregation in LTE-advanced networks,” *Mobile Computing, IEEE Transactions on* p. 1–11 (2014). [1.3.2](#), [2.1](#), [2.3.1](#), [3.1](#), [3.3.1](#)
- [14] 3GPP, “Physical layer procedures (Release 13),” TS 36.213, 3GPP (2016). [1.3.3](#), [A.5.1](#), [A.5.3](#)
- [15] J. Blazewicz, W. Domschke, et E. Pesch, “The job shop scheduling problem : Conventional and new solution techniques,” *European Journal of Operational Research* **93**, 1–33 (1996). URL : <http://EconPapers.repec.org/RePEc:eee:ejores:v:93:y:1996:i:1:p:1-33>. [2](#)
- [16] R. K. Ahuja, T. L. Magnanti, et J. B. Orlin, *Network Flows : Theory, Algorithms, and Applications* (Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993). [2](#)
- [17] D. E. Knuth, *The Art of Computer Programming, Volume 3 : (2nd Ed.) Sorting and Searching* (Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 1998). [2](#)
- [18] J. Nagle, “On packet switches with infinite storage,” *IEEE Transactions on Communications* **35**, 435–438 (1987). [2](#), [2.2](#)
- [19] A. G. Qureshi et A. S. Ween, “Dynamic resource allocation for GSM-GPRS services over a LEO satellite system,” dans “ICICS 1997,” , vol. 1 (1997), vol. 1, p. 20–24 vol.1. [2](#)
- [20] J.-L. C. Wu, W.-Y. Chen, et H.-H. Liu, *Information Networking : Wired Communications and Management : International Conference, ICOIN 2002 Cheju Island, Korea, January 30 – February 1, 2002 Revised Papers, Part I* (Springer Berlin Heidelberg, Berlin, Heidelberg, 2002), chap. Radio Resource Allocation in GSM/GPRS Networks, p. 457–468. URL : http://dx.doi.org/10.1007/3-540-45803-4_41. [2](#)
- [21] C. M. Papadimitriou, *Computational complexity* (Addison-Wesley, Reading, Massachusetts, 1994). [2.1](#)
- [22] A. Kuurne et A. P. Miettinen, “Weighted round robin scheduling strategies in (E)GPRS radio interface,” dans “IEEE 60th Vehicular Technology Conference, 2004. VTC2004-Fall.”, , vol. 5 (2004), vol. 5, p. 3155–3159 Vol. 5. [2.2](#)
- [23] A. Sen, L. Mohammed, R. Samprathi, et S. Bandyopadhyay, “Fair queuing with round robin : a new packet scheduling algorithm for routers,” dans “ISCC 2002,” (2002), p. 1001–1006. [2.2](#)
- [24] B. Dekeris, L. Narbutaite, et T. Adomkus, “A new adaptive fair queueing (AFQ) scheduler for support SLA,” dans “2007 29th International Conference on Information Technology Interfaces,” (2007), p. 597–602. [2.2](#)
- [25] A. Francini et F. M. Chiussi, “A weighted fair queueing scheduler with decoupled bandwidth and delay guarantees for the support of voice traffic,” dans “Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE,” , vol. 3 (2001), vol. 3, p. 1821–1827 vol.3. [2.2](#)
- [26] H. Tayvar et H. Alnuweiri, “The complexity of computing virtual-time in weighted fair queuing schedulers,” dans “Communications, 2004 IEEE International Conference on,” , vol. 4 (2004), vol. 4, p. 1996–2002 Vol.4. [2.2](#)

- [27] J. C. R. Bennett et H. Zhang, “WF2Q : worst-case fair weighted fair queueing,” dans “INFOCOM ’96. Fifteenth Annual Joint Conference of the IEEE Computer Societies. Networking the Next Generation. Proceedings IEEE,” , vol. 1 (1996), vol. 1, p. 120–128 vol.1. [2.2](#)
- [28] D. Pan et Y. Yang, “Max-min fair bandwidth allocation algorithms for packet switches,” dans “2007 IEEE International Parallel and Distributed Processing Symposium,” (2007), p. 1–10. [2.2](#)
- [29] B. Song, Y. H. Lin, et R. L. Cruz, “Weighted max-min fair beamforming, power control, and scheduling for a MISO downlink,” *IEEE Transactions on Wireless Communications* **7**, 464–469 (2008). [2.2](#)
- [30] M. Hosaagrahara et H. Sethu, “Max-min fair scheduling in input-queued switches,” *IEEE Transactions on Parallel and Distributed Systems* **19**, 462–475 (2008). [2.2](#)
- [31] Z. Zhaohua, “Performance of the frequency domain packet scheduling for LTE downlink,” dans “2nd International Conference on Future Computer and Communication (ICFCC) 2010,” , vol. 1 (2010), vol. 1, p. V1–635–V1–638. [2.2.1](#)
- [32] C. Gueguen et S. Baey, “A fair MaxSNR scheduling scheme for multiuser OFDM wireless systems,” dans “2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications,” (2009), p. 2935–2939. [2.2.1](#)
- [33] R. Knopp et P. A. Humblet, “Information capacity and power control in single-cell multiuser communications,” dans “Communications, 1995. ICC ’95 Seattle, ‘Gateway to Globalization’, 1995 IEEE International Conference on,” , vol. 1 (1995), vol. 1, p. 331–335 vol.1. [6](#)
- [34] C. Y. Wong, R. S. Cheng, K. B. Lataief, et R. D. Murch, “Multiuser OFDM with adaptive subcarrier, bit, and power allocation,” *IEEE Journal on Selected Areas in Communications* **17**, 1747–1758 (1999). [6](#)
- [35] V. K. N. Lau, “A proportional fair space time scheduling for wireless communications,” *IEEE Transactions on Communications* **53**, 747–747 (2005). [6](#)
- [36] M. Andrews, *Wireless Communications* (Springer New York, New York, NY, 2007), chap. A Survey of Scheduling Theory in Wireless Data Networks, p. 1–17. URL : http://dx.doi.org/10.1007/978-0-387-48945-2_1. [6](#), [3.6.1](#)
- [37] H. J. Zhu et R. H. M. Hafez, “Novel scheduling algorithms for multimedia service in OFDM broadband wireless systems,” dans “2006 IEEE International Conference on Communications,” , vol. 2 (2006), vol. 2, p. 772–777. [6](#)
- [38] T. E. Kolding, “QoS-aware proportional fair packet scheduling with required activity detection,” dans “IEEE Vehicular Technology Conference,” (2006), p. 1–5. [6](#)
- [39] C. Yang, W. Wang, Y. Qian, et X. Zhang, “A weighted proportional fair scheduling to maximize best-effort service utility in multicell network,” dans “2008 IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications,” (2008), p. 1–5. [6](#)
- [40] Zhouhua, Yangdacheng, Qiweishi, et MaMin, “On performance of multiuser diversity in SISO and MIMO wireless communication,” dans “14th IEEE Proceedings on PIMRC 2003.”, , vol. 3 (2003), vol. 3, p. 2872–2875 vol.3. [6](#)

- [41] G. Song et Y. Li, “Asymptotic throughput analysis for channel-aware scheduling,” *IEEE Transactions on Communications* **54**, 1827–1834 (2006). [6](#)
- [42] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, P. Whiting, et R. Vijayakumar, “Providing quality of service over a shared wireless link,” *Communications Magazine, IEEE* **39**, 150–154 (2001). [6](#)
- [43] S. Shakkottai et A. L. Stolyar, “Scheduling for multiple flows sharing a time-varying channel : The exponential rule,” *American Mathematical Society Translations, Series 2*, 2002 (2000). [6](#)
- [44] S. Shakkottai et A. Stolyar, “Scheduling Algorithms for a mixture of real-time and non-real-time data in HDR,” dans “Proceeding of ITC-17,” (2001), p. 793–804. URL : <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.22.2619>. [6](#)
- [45] C. Gueguen et S. Baey, “Comparison Study of Resource Allocation Strategies for OFDM Multimedia Networks,” *Hindawi Journal of Electrical and Computer Engineering* **2012**, 3 (2012). Special Issue on Resource Allocation in Communications and Computing. [6](#)
- [46] F. Capozzi, G. Piro, L. A. Grieco, G. Boggia, et P. Camarda, “Downlink packet scheduling in LTE cellular networks : Key design issues and a survey,” *IEEE Commun. Surveys and Tutorials* **15**, 678–700 (2013). Doi :10.1109/SURV.2012.060912.00100. [6](#), [2.3.1](#), [4.4.1](#)
- [47] F. Shams, G. Bacci, et M. Luise, “A survey on resource allocation techniques in OFDM(A) networks,” *Computer Networks* **65**, 129 – 150 (2014). [6](#)
- [48] H. Ahmed, K. Jagannathan, et S. Bhashyam, “Queue-aware optimal resource allocation for the LTE downlink,” p. 4122–4128 (2013). [2.3.1](#)
- [49] M. Carpin, A. Zanella, J. Rasool, K. Mahmood, O. Grondalen, et O. Osterbo, “A performance comparison of lte downlink scheduling algorithms in time and frequency domains,” dans “Communications (ICC), 2015 IEEE International Conference on,” (2015), p. 3173–3179. [2.3.1](#), [3.3.1](#)
- [50] X. Xiao, X. Tao, et J. Lu, “Energy-efficient resource allocation in LTE-based MIMO-OFDMA systems with user rate constraints,” *Vehicular Technology, IEEE Transactions on* **64**, 185–197 (2015). [2.3.1](#)
- [51] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods (Optimization and Neural Computation Series)* (Athena Scientific, 1996). [2.3.1](#)
- [52] A. E. Lozano et N. Jindal, “Transmit diversity v. spatial multiplexing in modern MIMO systems,” *CoRR abs/0811.3887* (2008). URL : <http://arxiv.org/abs/0811.3887>. [2.3.1](#)
- [53] S.-B. Lee, S. Choudhury, A. Khoshnevis, S. Xu, et S. Lu, “Downlink MIMO with frequency-domain packet scheduling for 3gpp lte,” dans “IEEE INFOCOM 2009,,” (2009), p. 1269–1277. [2.3.1](#)
- [54] S.-B. Lee, I. Pefkianakis, S. Choudhury, S. Xu, et S. Lu, “Exploiting spatial, frequency, and multiuser diversity in 3gpp lte cellular networks,” *Mobile Computing, IEEE Transactions on* **11**, 1652–1665 (2012). [2.3.1](#), [3.6.1](#)

- [55] H. Zhang, N. Prasad, et S. Rangarajan, “Mimo downlink scheduling in LTE systems,” dans “INFOCOM, 2012 Proceedings IEEE,” (2012), p. 2936–2940. [2.3.1](#), [2.1](#), [2.3.1](#), [2.4](#), [3.1](#), [3.3.1](#), [3.6.1](#)
- [56] S. Niafar, Z. Huang, et D. Tsang, “An optimal standard-compliant MIMO scheduler for LTE downlink,” *IEEE Transactions on Wireless Communications*, **13**, 2412–2421 (2014). [2.3.1](#), [3.3.1](#)
- [57] H. Karloff, *Linear Programming* (Boston, MA, 1991), chap. The Simplex Algorithm, p. 23–47. URL : http://dx.doi.org/10.1007/978-0-8176-4844-2_2. [2.3.1](#)
- [58] F. Z. Kaddour, E. Vivier, L. Mroueh, M. Pischella, et P. Martins, “Green opportunistic and efficient resource block allocation algorithm for LTE uplink networks,” *IEEE Transactions on Vehicular Technology* **64**, 4537–4550 (2015). [2.3.2](#)
- [59] N. Prasad, H. Zhang, H. Zhu, et S. Rangarajan, “Multiuser scheduling in the 3GPP LTE cellular uplink,” *IEEE Transactions on Mobile Computing* **13**, 130–145 (2014). [2.3.2](#)
- [60] L. Ruiz de Temino, G. Berardinelli, S. Frattasi, et P. Mogensen, “Channel-aware scheduling algorithms for sc-fdma in LTE uplink,” dans “Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on,” (2008), p. 1–6. [2.3.2](#)
- [61] H. Safa et K. Tohme, “LTE uplink scheduling algorithms : Performance and challenges,” dans “Telecommunications (ICT), 2012 19th International Conference on,” (2012), p. 1–6. [2.3.2](#)
- [62] F. Liu, X. She, L. Chen, et H. Otsuka, “Improved recursive maximum expansion scheduling algorithms for uplink single carrier fdma system,” dans “Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st,” (2010), p. 1–5. [2.3.2](#), [5.4.2](#)
- [63] S.-B. Lee, I. Pefkianakis, A. Meyerson, S. Xu, et S. Lu, “Proportional fair frequency-domain packet scheduling for 3GPP LTE uplink,” dans “INFOCOM 2009, IEEE,” (2009), p. 2611–2615. [2.3.2](#), [2.2](#)
- [64] M. Kalil, A. Shami, et A. Al-Dweik, “QoS-aware power-efficient scheduler for LTE uplink,” *IEEE Transactions on Mobile Computing* **14**, 1672–1685 (2015). [2.3.2](#), [5.4.2](#)
- [65] O. Delgado et B. Jaumard, “Scheduling and resource allocation for multiclass services in LTE uplink systems,” dans “Wireless and Mobile Computing, Networking and Communications (WiMob), 2010 IEEE 6th International Conference on,” (2010), p. 355–360. [2.3.2](#)
- [66] F. Ren, Y. Xu, H. Yang, J. Zhang, et C. Lin, “Frequency domain packet scheduling with stability analysis for 3GPP LTE uplink,” *Mobile Computing, IEEE Transactions on* **12**, 2412–2426 (2013). [2.3.2](#), [5.4.2](#)
- [67] S. Arora, “The approximability of np-hard problems,” dans “Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing,” (ACM, New York, NY, USA, 1998), STOC ’98, p. 337–348. URL : <http://doi.acm.org/10.1145/276698.276784>. [7](#)

- [68] R. Bar-Yehuda, K. Bendel, A. Freund, et D. Rawitz, “Local ratio : A unified framework for approximation algorithms. in memoriam : Shimon even 1935-2004,” *ACM Comput. Surv.* **36**, 422–463 (2004). URL : <http://doi.acm.org/10.1145/1041680.1041683>. 2.3.2
- [69] 3rd Generation Partnership Project, “Further advancements for e-utra physical layer aspects,” Technical Specification Group Radio Access Network (2009). 2.4
- [70] C. Gueguen et S. Baey, “A fair opportunistic access scheme for multiuser OFDM wireless networks,” *EURASIP J. Wireless Comm. and Networking* **2009** (2009). 2.4, 4.4.1, 5.4.1
- [71] U. Feige et R. Izsak, “Welfare maximization and the supermodular degree.” *Electronic Colloquium on Computational Complexity (ECCC)* p. 95. 3.2
- [72] A. Schrijver, *Theory of Linear and Integer Programming* (John Wiley & Sons, Inc., New York, NY, USA, 1986). 3.2
- [73] “IBM ILOG CPLEX Optimizer,”
urlhttp : [://www-01.ibm.com/software/integration/optimization/cplex-optimizer/](http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/) (2010). 3.2
- [74] G. Calinescu, C. Chekuri, M. Pál, et J. Vondrák, “Maximizing a submodular set function subject to a matroid constraint,” dans “Proceedings of the 12th International Conference on Integer Programming and Combinatorial Optimization,” (Springer-Verlag, Berlin, Heidelberg, 2007), IPCO ’07, p. 182–196. 3.3.1
- [75] U. Feige, “On maximizing welfare when utility functions are subadditive,” *SIAM Journal on Computing* **39**, 122–142 (2009). 3.3.1
- [76] V. Mirrokni, M. Schapira, et J. Vondrak, “Tight information-theoretic lower bounds for welfare maximization in combinatorial auctions,” dans “Proceedings of the 9th ACM Conference on Electronic Commerce,” . 3.3.1
- [77] R. Holzman, N. Kfir-Dahav, D. Monderer, et M. Tennenholtz, “Bundling equilibrium in combinatorial auctions,” *Games and Economic Behavior* **47**, 104 – 123 (2004). 3.3.1
- [78] Y. Blankenship, P. Sartori, B. Classon, V. Desai, et K. Baum, “Link error prediction methods for multicarrier systems,” dans “IEEE Vehicular Technology Conference,” , vol. 6 (2004), vol. 6, p. 4175–4179. ??, 3.6.1
- [79] J. D. Parsons, *The Mobile Radio Propagation Channel, 2nd Edition* (Wiley, 2000). URL : <http://www.worldcat.org/isbn/047198857X>. 3.6.1
- [80] T. Tao et A. Czylik, “Combined fast link adaptation algorithm in LTE systems,” 7th International Conference on Communications and Networking in China **0**, 415–420 (2011). 3.6.1
- [81] J. C. Ikuno, M. Wrulich, et M. Rupp, “System level simulation of lte networks,” dans “IEEE Vehicular Technology Conference (Spring),” (2010), p. 1–5. 3.6.1
- [82] J. Fan, Q. Yin, G. Li, B. Peng, et X. Zhu, “MCS selection for throughput improvement in downlink LTE systems,” dans “Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN),” (Maui, Hawaii, 2011), p. 1–5. 3.6.1, 3.6.1

- [83] F.-T. Chen et G.-L. Tao, “A novel MCS selection criterion for supporting AMC in LTE system,” dans “International Conference on Computer Application and System Modeling (ICCASM),” , vol. 6 (2010), vol. 6, p. 598–603. [3.6.1](#)
- [84] 3GPP, “Policy and charging control architecture (Release 10),” TS 23.203, 3rd Generation Partnership Project (3GPP) (2012). [4.1](#)
- [85] J. Niu, D. Lee, X. Ren, G. Y. Li, et T. Su, “Scheduling exploiting frequency and multi-user diversity in LTE downlink systems.” dans “International Symposium on Personal, Indoor and Mobile Radio Communications,” (IEEE, 2012), p. 1412–1417. [4.3](#)
- [86] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, et J. Jetcheva, “A performance comparison of multi-hop wireless ad hoc network routing protocols,” dans “Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking,” (ACM, New York, NY, USA, 1998), MobiCom ’98, p. 85–97. URL : <http://doi.acm.org/10.1145/288235.288256>. [4.4](#)
- [87] S. Baey, “Modeling MPEG4 video traffic based on a customization of the DB-MAP,” dans “Proc. Int. Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS’04),” (San Jose, CA, USA, 2004), p. 705–714. [4.4](#)
- [88] P. T. Brady, “A model for generating on-off speech patterns in two-way conversation,” Bell System Technical Journal **48** (1969). [4.4](#)
- [89] H. J. Kushner et P. A. Whiting, “Convergence of proportional-fair sharing algorithms under general conditions,” IEEE Trans. Wireless Commun p. 1250–1259 (2004). [4.4.1](#)
- [90] J. Huang, V. Subramanian, R. Agrawal, et R. Berry, “Downlink scheduling and resource allocation for OFDM systems,” Wireless Communications, IEEE Transactions on **8**, 288–296 (2009). [4.4.1](#)
- [91] W. Miao, G. Min, Y. Jiang, X. Jin, et H. Wang, “QoS-aware resource allocation for LTE-A systems with carrier aggregation,” dans “Wireless Communications and Networking Conference (WCNC), 2014 IEEE,” (2014), p. 1403–1408. [4.4.1](#)
- [92] A. Ragaleux, S. Baey, et M. Karaca, “Generic scheduling scheme for lte-a uplink with multimedia traffic,” Vehicular Technology (**Round 2**), Submission to IEEE Transactions on (2016). [5](#)
- [93] D. Bertsekas, “A new algorithm for the assignment problem,” Mathematical Programming **21**, 152–171 (1981). URL : <http://dx.doi.org/10.1007/BF01584237>. [5.2.1](#), [5.3.3](#)
- [94] D. B. Johnson et D. A. Maltz, “Dynamic source routing in ad hoc wireless networks,” dans “Mobile Computing,” (Kluwer Academic Publishers, 1996), p. 153–181. [5.1](#)
- [95] C. Isheden et G. Fettweis, “Energy-efficient multi-carrier link adaptation with sum rate-dependent circuit power,” dans “Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE,” (2010), p. 1–6. [5.4](#)
- [96] D. Dechene et A. Shami, “Energy-aware resource allocation strategies for lte uplink with synchronous harq constraints,” Mobile Computing, IEEE Transactions on **13**, 422–433 (2014). [5.4](#)

- [97] J. Huang, Z. Li, M. Chiang, et A. K. Katsaggelos, “Joint source adaptation and resource allocation for multi-user wireless video streaming.” *IEEE Trans. Circuits Syst. Video Techn.* **18**, 582–595 (2008). [6.2](#)
- [98] C. Yang et S. Jordan, “A novel coordinated connection access control and resource allocation framework for 4g wireless networks,” *IEEE/ACM Transactions on Networking* **23**, 1328–1341 (2015). [6.2](#)
- [99] Q. h. Bai et Y. Zheng, “Study on the access control model,” dans “Cross Strait Quad-Regional Radio Science and Wireless Technology Conference (CSQRWC), 2011,” , vol. 1 (2011), vol. 1, p. 830–834. [6.2](#)
- [100] M. A. AboulHassan, M. Yassin, S. Lahoud, M. Ibrahim, D. Mezher, B. Cousin, et E. A. Sourour, “Classification and comparative analysis of inter-cell interference coordination techniques in LTE networks,” dans “2015 7th International Conference on New Technologies, Mobility and Security (NTMS),” (2015), p. 1–6. [6.2](#)
- [101] S. Timotheou et I. Krikidis, “Fairness for non-orthogonal multiple access in 5G systems,” *IEEE Signal Processing Letters* **22**, 1647–1651 (2015). [6.2](#)
- [102] 3GPP, “Physical channels and modulation (release 13),” TS 36.211, 3rd Generation Partnership Project (3GPP) (2016). [A.2](#)
- [103] H. Myung, J. Lim, et D. Goodman, “Single carrier FDMA for uplink wireless transmission,” *Vehicular Technology Magazine, IEEE* **1**, 30–38 (2006). [A.5.2](#)

Sujet : Mécanismes D'Accès Multiple dans les Réseaux Sans Fil Large Bande

Résumé : Dans cette thèse, nous étudions le problème de l'allocation de ressources dans le cadre des réseaux 4G LTE. La méthode d'accès OFDMA qui est utilisée partage les ressources radios à la fois dans le domaine fréquentiel et temporel. En raison des déficiences du canal, les utilisateurs ne bénéficient pas toujours des mêmes débits d'émission/réception sur chacune des ressources. Dans ce cadre, notre problème consiste à distribuer ces ressources radios aux mobiles afin de leur permettre de transmettre/recevoir des données. L'algorithme utilisé pour allouer les ressources a une importance fondamentale sur les performances du système.

La norme LTE ajoute des contraintes supplémentaires à ce problème et rend l'exploitation de la diversité fréquentielle et de la diversité multi-utilisateurs plus difficile. En effet, sous ces contraintes, nous montrons que le problème de l'allocation de ressources fait alors partie de la classe des problèmes "difficiles". Par conséquent, les algorithmes classiques de la littérature sont souvent inadaptés à un réseau LTE réel.

Nous proposons des algorithmes d'allocation de ressources à la fois pour le sens montant et descendant de LTE. Les contraintes de la norme sont rigoureusement prises en compte afin de construire des solutions efficaces. De plus, les algorithmes proposés sont génériques et peuvent donc s'adapter à une grande variété d'objectifs. En particulier, nous nous attachons à prendre en charge les trafics multimédias dont les débits et les besoins en qualité de service sont très hétérogènes (taux d'erreurs binaires, retard, gigue, etc.). En effet, l'augmentation progressive des débits et la forte popularité des équipements mobiles intelligents amènent à une utilisation toujours plus massive des applications multimédias.

Tous nos algorithmes sont validés par simulation. Par ce biais, nous montrons que la prise en compte des contraintes de LTE est essentielle à l'obtention de performances élevées.

Mots clés : LTE, OFDMA, allocation de ressources

Subject : Multiple Access Mechanisms in Broadband Wireless Networks

Résumé : In this thesis, we study the resource allocation problem within the framework of 4G LTE networks. The OFDMA access method divides the radio resources both in the frequency and time domains. Due to channel impairments, users do not always have the same transmit/receive rates on each resource. In this context, our problem is to share the radio resources between users and enable them to transmit/receive data. The algorithm used to allocate resources is of fundamental importance on system performance.

The LTE standard adds constraints to this problem and makes harder the exploitation of the frequency and the multi-user diversity. Indeed, under these constraints, we show that the resource allocation problem becomes part of the “most difficult” problems. Therefore, the conventional algorithms are often not adapted to a real LTE network.

We provide resource allocation algorithms for both the uplink and downlink of LTE. The constraints of the standard are rigorously taken into account in order to build effective solutions. In addition, the proposed algorithms are generic and can adapt to a wide variety of objectives. In particular, we focus on the support of multimedia traffic with heterogeneous quality of service requirements (bit error rate, delay, jitter, etc.). Indeed, the gradual increase of the offered throughput and the strong popularity of smart mobile devices lead to a massive use of multimedia applications.

Our algorithms are validated through extensive simulation. By this means, we show that the inclusion of LTE constraints is essential to achieving high performance.

Keywords : LTE, OFDMA, resource allocation