



HAL
open science

Allocation de ressources et partage de charge adaptés aux architectures de réseaux d'accès basées sur le cloud

Hadjer Touati

► **To cite this version:**

Hadjer Touati. Allocation de ressources et partage de charge adaptés aux architectures de réseaux d'accès basées sur le cloud. Réseaux et télécommunications [cs.NI]. Institut Polytechnique de Paris, 2020. Français. NNT : 2020IPPAS009 . tel-02905421

HAL Id: tel-02905421

<https://theses.hal.science/tel-02905421>

Submitted on 23 Jul 2020

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.



INSTITUT
POLYTECHNIQUE
DE PARIS

NNT : 2020IPPAS009

Thèse de doctorat



Allocation de Ressources et Partage de Charge adaptés aux architectures de réseaux d'accès basées sur le cloud

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à Télécom SudParis

École doctorale n°ED 626 Ecole doctorale IP Paris (ED IPP)
Spécialité de doctorat : Mathématique et informatique

Thèse présentée et soutenue à Evry, le 07/07/2020, par

HADJER TOUATI

Composition du Jury :

Marcelo Dias Amorim Directeur de recherche, CNRS	Président
Pascal Lorenz Professeur, Université de Haute Alsace (MIPS)	Rapporteur
Adlen Ksentini Professeur, Eurecom (SC)	Rapporteur
Lila Boukhatem Maître de conférences, LRI (ROCS)	Examinatrice
Nazim Agoulmine Professeur, Université d'Evry (IBISC)	Examineur
Hind CASTEL-TALEB Professeur, Télécom SudParis (METHODES)	Directrice de thèse
Badii JOUABER HDR, Télécom SudParis (R3S)	Co-encadrant de thèse
Sara AKBERZADEH Consultante experte radio, Davidson Consulting	Invitée

Table des matières

Liste des figures	7
Liste des tableaux	10
Abréviations	11
1 Introduction	14
1.1 Contexte et motivations	16
1.2 Objectifs et contributions de la thèse	18
1.3 Plan des travaux de thèse	19
1.4 Liste des publications	22
2 La 5G, la révolution des réseaux mobiles	24
2.1 Introduction	24
2.2 L'évolution des réseaux mobiles (4G/ 5G)	24
2.2.1 Exigences techniques de la 5G	25
2.2.2 Vision de l'accès radio 5G	26
2.2.3 Interface radio du LTE/ LTE-A	26
2.2.3.1 Modes de duplexage	27
2.2.3.2 Structure de la trame LTE-FDD	27
2.2.4 Armes de LTE-A	29
2.3 Présentation de l'architecture Cloud Radio Access Network (C-RAN)	30
2.3.1 Composants de l'architecture C-RAN	31
2.3.1.1 Remote Radio Head (RRH)	32
2.3.1.2 Baseband Unit (BBU)	33
2.3.1.2.a Couche physique	33
2.3.1.2.b Couche MAC	37
2.3.1.2.c Couche Radio Resource Control (RRC)	37
2.3.1.3 Le fronthaul	38
2.3.2 Architecture H-CRAN	40

2.3.2.1	Technique de remédiation aux interférences eICIC	41
2.3.2.2	Technique de remédiation aux interférences CoMP	42
2.3.2.2.a	Coordinated Scheduling/Coordinated Beamforming (CS/CB)	42
2.3.2.2.b	Joint Transmission (JT)	43
2.3.2.2.c	Dynamic Point Selection (DPS)	44
2.4	Conclusion	46
3 Modèles de files d'attente et chaînes de Markov pour le dimensionnement des		
C-RAN/H-CRAN		47
3.1	Introduction	47
3.2	Chaînes de Markov	48
3.3	Modèles de files d'attente	50
3.4	Principaux modèles de files d'attente utilisés	50
3.4.1	La file $M/G/S/S$	52
3.4.2	La file $M^{[X]}/G/S/S$	52
3.4.3	La file $G/G/1$	54
3.4.4	La file $G^{[X]}/G/1$	55
3.5	Conclusion	55
4 Modélisation de eICIC/ABS et allocation optimale de ressources dans les C-		
RAN/ H-CRAN		57
4.1	Problématique et contexte	57
4.2	Formulation du problème	61
4.3	Chaînes de Markov pour la modélisation de eICIC/ ABS	63
4.4	Modèles de file d'attente pour la modélisation de eICIC/ ABS	67
4.4.1	Modèle de file d'attente : $M/G/S/S$ pour la modélisation de eICIC/ ABS	68
4.4.2	Modèle de file d'attente : $M^{[X]}/G/S/S$ pour la modélisation de eICIC/ ABS	71
4.5	Analyse des performances	73
4.5.1	Algorithme de génération et de calcul des conditions radio des UE pour eICIC/	
	ABS	73
4.5.2	Etude comparative entre les performances de la $M/G/S/S$ et celles des chaînes	
	de Markov simulées	78
4.5.3	Etude comparative entre les performances de la $M^{[X]}/G/S/S$ et celles des	
	chaînes de Markov simulées	85
4.5.4	Etude comparative entre les trois approches proposées : chaînes de Markov,	
	$M/G/S/S$ et $M^{[X]}/G/S/S$	91
4.5.4.1	Comparaison des débits par classe en fonction du taux d'arrivée	91
4.5.4.2	Comparaison des temps d'exécution des trois approches proposées :	
	chaînes de Markov, $M/G/S/S$ et $M^{[X]}/G/S/S$	94

4.5.5	Influence de la micro et du nombre d'ABS	95
4.6	Optimisation de eICIC/ ABS basée sur les modèles de files d'attente	97
4.6.1	Description de l'algorithme d'optimisation	99
4.6.2	Application numérique du modèle d'optimisation de eICIC/ABS	101
4.6.3	Influence d'épsilon sur le modèle d'optimisation de eICIC/ ABS	104
4.7	Conclusion	106
5	Modélisation de JT-CoMP et allocation optimale de ressources dans les C-RAN	108
5.1	Problématique et contexte	108
5.2	Formulation du problème	110
5.3	Chaîne de Markov pour la modélisation de JT-CoMP	112
5.4	Modèles de file d'attente pour la modélisation de JT-CoMP	117
5.4.1	Modèle de file d'attente : $M/G/S/S$ pour la modélisation de JT-CoMP	118
5.4.2	Modèle de file d'attente : $M^{[X]}/G/S/S$ pour la modélisation de JT-CoMP	121
5.5	Analyse des performance	123
5.5.1	Algorithme de génération et de calcul des conditions radio des UE pour JT-CoMP	123
5.5.2	Etude comparative entre les trois approches proposées : chaînes de Markov, $M/G/S/S$ et $M^{[X]}/G/S/S$	126
5.6	Optimisation de JT-CoMP basée sur les modèles de files d'attente	131
5.6.1	Application numérique du modèle d'optimisation de JT-CoMP	134
5.6.2	Influence d'épsilon sur le modèle d'optimisation de JT-CoMP	136
5.6.3	Influence de la distribution des clients sur le modèle d'optimisation de JT-CoMP	138
5.7	Conclusion	140
6	Etude comparative entre eICIC/ABS et JT-CoMP	142
6.1	Problématique et contexte	142
6.2	Description des systèmes à comparer	144
6.3	Analyse des performances	145
6.4	Conclusion	150
7	Dimensionnement du fronthaul dans les C-RAN/ H-CRAN	152
7.1	Problématique et contexte	152
7.2	Etat de l'art et revue des splits fonctionnels	154
7.3	Proposition d'un nouveau split physique en UL	156
7.3.1	Analyse de la contrainte de latence sur le fronthaul	158
7.3.2	Split Π_U : nouveau split physique en UL sur le fronthaul	160
7.4	Modélisation et analyse des performances	163
7.5	Résultats numériques	165

7.5.1	Analyse des temps de séjour	165
7.5.2	Calcul du nombre maximal de RRH synchrones à merger	165
7.6	Conclusion	167
8	Conclusion générale et perspectives	171
8.1	Contributions sur eICIC/ABS	171
8.1.1	Modélisation de la technique de remédiation aux interférences eICIC/ABS	171
8.1.2	Dimensionnement et optimisation du débit global pour eICIC/ABS	172
8.2	Contributions sur JT-CoMP	172
8.2.1	Modélisation de la technique de remédiation aux interférences JT-CoMP	172
8.2.2	Dimensionnement et optimisation du débit global pour JT-CoMP	172
8.3	Contribution comparant les performances de eICIC/ABS et de JT-CoMP	172
8.4	Contribution sur le dimensionnement du fronthaul	173
8.5	Travaux futurs	173
8.5.1	Perspectives à court terme	173
8.5.1.1	Gestion des interférences dans des clusters avec eICIC/ ABS	173
8.5.1.2	Gestion des interférences dans des clusters avec JT-CoMP	174
8.5.1.3	Considérer la mobilité des utilisateurs dans nos études sur eICIC/ ABS et JT-CoMP	175
8.5.1.4	Considérer d'autres bandes passantes dans nos études sur eICIC/ ABS et JT-CoMP	175
8.5.1.5	Effectuer une analyse approfondie numérique et théorique de nos modèles de file d'attente	176
8.5.1.6	Dimensionnement du fronthaul pour des RRH asynchrones	176
8.5.2	Perspectives à long terme	177
8.5.2.1	Stratégie de scheduling des utilisateurs	178
8.5.2.2	Modélisation d'autres techniques de remédiation aux interférences en UL	178
8.5.2.3	Intégration de nos outils de dimensionnement dans un réseau 5G basé sur le slicing	179
8.5.2.4	Développer des méthodes d'intelligence artificielle	179
	Annexe	180
	Bibliographie	207

Table des figures

1.1	Ce qui est fait aujourd'hui en une minute sur Internet [Eri14a]	15
1.2	Architecture C-RAN	17
1.3	Plan de la thèse	20
2.1	Vision de l'accès radio 5G [ZAM ⁺ 18]	26
2.2	Allocation de ressources en OFDM et OFDMA/SC-FDMA	27
2.3	Trame et slot de l'OFDMA et du SC-FDMA dans LTE-FDD	28
2.4	Grille de ressources temps-fréquence en voie montante et descendante	29
2.5	Composition de l'architecture C-RAN	32
2.6	Constellation des modulations possibles en LTE	34
2.7	Technique de remédiation aux interférences eICIC/ ABS	41
2.8	Coordinated Scheduling (CS) CoMP	42
2.9	Coordinated Beamforming (CB) CoMP	43
2.10	CS/CB CoMP	44
2.11	JT CoMP	44
2.12	DPS CoMP	45
4.1	Résumé de notre démarche pour la modélisation de eICIC/ ABS	61
4.2	H-CRAN cluster : 7 macro cells et 1 micro cell	61
4.3	Exemple d'arrivée d'un client, avec une taille de batch égal à 2 RBG, dans une trame	63
4.4	Arrivée des clients et partage de ressources entre la macro et la micro	69
4.5	Exemple de service d'un client dans la chaîne de Markov et dans la $M/G/S/S$	69
4.6	Exemple de service d'un client dans la chaîne de Markov et dans la $M^{[X]}/G/S/S$	72
4.7	Explication détaillée sur notre démarche pour la modélisation de eICIC/ ABS	77
4.8	Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de débit (macro sans micro, $size(u)=1024$ bits)	80
4.9	Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de débit (macro sans micro, $size(u)=2048$ bits)	81
4.10	Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de débit (macro sans micro, $size(u)=3200$ bits)	81

4.11 Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de débit global (macro avec micro $n_{ABS} = 3$, $size(u)=1024$ bits)	83
4.12 Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de débit global (macro avec micro $n_{ABS} = 1$, $size(u)=2048$ bits)	84
4.13 Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de débit global (macro avec micro $n_{ABS} = 1$, $size(u)=3200$ bits)	84
4.14 Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de de débit $X_{M,1}$ (macro sans micro, $size(u) = 1024$ bits)	86
4.15 Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de de débit $X_{M,2}$ (macro sans micro, $size(u) = 1024$ bits)	86
4.16 Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de débit de sortie dans la classe 2 (macro sans micro, $size(u) = 2048$ bits)	87
4.17 Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de débit de sortie dans la classe 4 (macro sans micro, $size(u) = 2048$ bits)	88
4.18 Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de débit de sortie dans la classe 1 de la macro (macro avec micro, $n_{ABS} = 3$, $size(u) = 1024$ bits)	89
4.19 Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de débit de sortie dans la classe 2 de la macro (macro avec micro, $n_{ABS} = 3$, $size(u) = 1024$ bits)	90
4.20 Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 1024$ bits)	91
4.21 Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 1024$ bits)	92
4.22 Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 2048$ bits)	93
4.23 Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 2048$ bits)	93
4.24 Influence du déploiement de la micro	95
4.25 Débit de sortie global en fonction du n_{ABS} ($size(u) = 1024$ bits)	96
4.26 Comparaison entre X (le débit global optimal) et X^* (le débit global sans déploiement de la micro) où $\epsilon = 0.02$	102
4.27 Influence d'epsilon sur le nombre d'ABS optimal	105
5.1 Résumé de notre démarche pour la modélisation de JT-CoMP	110
5.2 C-RAN cluster : 10 macro cells	110
5.3 Arrivée des clients et partage de ressources entre les utilisateurs CoMP et non-CoMP	119
5.4 Explication détaillée sur notre démarche pour la modélisation de JT-CoMP	126
5.5 Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 1 de la macro M1 ($size(u) = 1024$ bits)	128

5.6	Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 2 de la macro M1 ($size(u) = 1024$ bits)	128
5.7	Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 1 de la macro M2 ($size(u) = 1024$ bits)	129
5.8	Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 2 de la macro M2 ($size(u) = 1024$ bits)	129
5.9	Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 1 de la région de coopération ($size(u) = 1024$ bits)	130
5.10	Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 2 de la région de coopération ($size(u) = 1024$ bits)	130
5.11	Influence d'epsilon sur le nombre de serveurs optimal dans la région de coopération	137
6.1	C-RAN cluster : 10 macro cells	144
6.2	H-RAN cluster : 10 macro cells et 2 micro cells	145
7.1	Cas d'étude de la topologie du FH du C-RAN	153
7.2	Décomposition fonctionnelle au niveau de la couche RAN [E+17]	154
7.3	Splits eCPRI de la couche physique [E+17]	155
7.4	Temps HARQ en Uplink	158
7.5	Proposed Π_U uplink physical layer split	161

Liste des tableaux

2.1	Largeur de la bande du système en nombre de PRB	28
2.2	Correspondance $SINR/ CQI$ et $CQI/ Bits_{MCS}/ CR$	35
2.3	Taille des RBG en fréquence en fonction de la bande passante du système	36
3.1	Récapitulatif sur les modèles de files d'attente étudiés et les performances calculées	50
3.2	Symboles des modèles de file d'attente utilisés	51
4.1	Paramètres des chaînes de Markov pour la modélisation de eICIC/ABS	64
4.2	Paramètres pour le calcul du nombre de RBG requis par chaque UE	75
4.3	Paramètres de simulation	78
4.4	Temps d'exécution des trois approches (macro avec micro, $size(u)= 1024$ bits)	95
4.5	n_{ABS} optimal, $\lambda_m^{clients}$ optimal, débit global optimal en kbits/trame et gain ($X-X^*$) en kbits/trame selon le taux d'arrivée des clients obtenus par l'algorithme Global Search et l'algorithme 2	102
4.6	Influence d'epsilon sur le gain apporté suite au déploiement optimale de la micro	105
5.1	Paramètres des chaînes de Markov pour la modélisation de JT-CoMP	114
5.2	Paramètres extraits de la simulation des chaînes de Markov	127
5.3	S_c optimal, $\lambda_c^{clients}$ optimal et débit global optimal en fonction du taux d'arrivée des clients ($\lambda_{cmax}^{clients} = 10\% \times \lambda^{clients}$)	135
5.4	Influence d'epsilon sur le gain apporté suite à l'application de JT-CoMP	138
5.5	S_c optimal, $\lambda_c^{clients}$ optimal et débit global optimal en fonction du taux d'arrivée des clients ($\lambda_{cmax}^{clients} = 40\% \times \lambda^{clients}$)	139
6.1	Paramètres extraits des simulations de la topologie 1 (voir FIGURE 6.1)	146
6.2	Paramètres extraits des simulations de la topologie 2 (voir FIGURE 6.2)	147
6.3	Comparaison en termes de débit optimal entre la topologie 1 et la topologie 2	149
7.1	Paramètres utilisés	157
7.2	Profils de trafic en UL pour les splits physiques, $N_{ant} = 2$ MIMO, $N_{bit} = 15$ bits/sample	162

7.3	Nombre maximal de RRH mergées sous différentes bandes passantes LTE et différentes capacités de switch 40G, 100G, 200G et 1T dans un C-RAN tel que les RRH sont synchrones	166
7.4	Valeurs des temps de séjour pour le transport de plusieurs secteurs (RRH) sous différentes bandes passantes et différentes capacités de switches 40G, 100G, 200G et 1T dans une architecture C-RAN avec le split I_U où les RRH sont synchrones.	168
7.5	Valeurs des temps de séjour pour le transport de plusieurs secteurs (RRH) sous différentes bandes passantes et différentes capacités de switches 40G, 100G, 200G et 1T dans une architecture C-RAN sous le split II_U par slot où les RRH sont synchrones.	169
7.6	Valeurs des temps de séjour pour le transport de plusieurs secteurs (RRH) sous différentes bandes passantes et différentes capacités de switches 40G, 100G, 200G et 1T dans une architecture C-RAN sous le split II_U par TTI où les RRH sont synchrones.	170
8.1	Comparaison entre les temps de séjour obtenus par simulation et par la G/G/1 pour une BP= 100 MHz sous différentes capacités de switch 100 G, 200 G et 400 G lorsque le split II_U par slot est appliqué avec des RRH asynchrones.	177
2	Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit (macro sans micro, $size(u)=1024$ bits)	180
3	Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit (macro sans micro, $size(u)=2048$ bits)	180
4	Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit (macro sans micro, $size(u)=3200$ bits)	181
5	Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit global (macro avec micro $n_{ABS} = 3$, $size(u)=1024$ bits)	181
6	Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit global (macro avec micro $n_{ABS} = 1$, $size(u)=2048$ bits)	181
7	Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit global (macro avec micro $n_{ABS} = 1$, $size(u)=3200$ bits)	182
8	Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe (macro sans micro, $size(u) = 1024$ bits)	182
9	Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe (macro sans micro, $size(u) = 2048$ bits)	183
10	Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe dans la macro (micro déployée $n_{ABS} = 3$, $size(u) = 1024$ bits)	184

11	Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe dans la micro (micro déployée $n_{ABS} = 3$, $size(u) = 1024$ bits)	185
12	Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe dans la macro (micro déployée $n_{ABS} = 1$, $size(u) = 2048$ bits)	186
13	Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe dans la micro (micro déployée $n_{ABS} = 1$, $size(u) = 2048$ bits)	187
14	Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 1024$ bits)	188
15	Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 2048$ bits)	189
16	Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de probabilité de blocage par classe (macro M1, $size(u) = 1024$ bits)	190
17	Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de débit par classe (macro M1, $size(u) = 1024$ bits)	191
18	Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de probabilité de blocage par classe (macro M2, $size(u) = 1024$ bits)	192
19	Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de débit par classe (macro M2, $size(u) = 1024$ bits)	193
20	Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de probabilité de blocage par classe (région de coopération, $size(u) = 1024$ bits)	194
21	Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de débit par classe (région de coopération, $size(u) = 1024$ bits)	195
22	Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage par classe (macro sans micro), $\lambda^{clients} = 50$ clients/trame	196
23	Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage par classe (macro sans micro), $\lambda = 200$ clients/trame	197

Abréviations

3GPP : 3rd Generation Partnership Project
ABS : Almost Blank Subframe
ADC : Analog-to-Digital Converter
ADMM : Alternating Direction Method of Multipliers
AMC : Adaptive Modulation and Coding
BBU : Base Band Unit
BLER : Block Error Rate
BPSK : Binary Phase-Shift Keying
CA : Carrier Aggregation
CAPEX : Capital Expenditure
CoMP : Coordinated Multi-Point
CP : Cyclic Prefix
CPRI : Common Public Radio Interface
CQI : Channel Quality Indicator
C-RAN : Cloud Radio Access Network
CRC : Cyclic Redundancy Check
CRE : Cell Range Expansion
CS/ CB : Coordinated Scheduling/ Coordinated Beamforming
DAC : Digital-to-Analog Converter
DL : Downlink
DPS : Dynamic Point Selection
DRX : Discontinuous Reception
eICIC : enhanced Inter-Cell Interference Coordination
eNB : evolved NodeB
EPC : Evolved Packet Core
FDD : Frequency Division Duplex
FFO : Fractional Frequency Offset
FH : Fronthaul
FIFO : First In First Out

GSMA : Global System for Mobile Communications Association
HARQ :Hybrid Automatic Repeat Request
H-CRAN :Heterogeneous Cloud Radio Access Network
HetNets : Heterogeneous Networks
I/Q :Inphase/Quadrature
IETF : Internet Engineering Task Force
Iot : Internet of Things
IP : Internet Protocol
IPG :Inter-Packet Gap
ITU-R : International Telecommunication Union Radiocommunication Sector
JT-CoMP :Joint Transmission CoMP
LTE : Long Term Evolution
LTE-A : LTE-Advanced
MAC : Media Access Control
MCN :Mobile Cloud Network
MCS :Modulation and Coding Scheme
MIMO :Multi-Input Multi-Output
MINLP :Mixed Integer Non-Linear Programming
MNO :Mobile Network Operators
MTU :Maximum Transfer Unit
NGFI :Next Generation Fronthaul Interface
NGMN :Next Generation Mobile Network
NLP : Non-Linear Programming
NR : New Radio
NTP :Network Time Protocol
OBSAI :Open Base Station Architecture Initiative
OFDM : Orthogonal Frequency Division Multiplexing
OFDMA : Orthogonal Frequency Division Multiple Access
OPEX : Operational Expenditure
ORI :Open Radio Interface
OSI :Open System Interconnection
OTN :Optical Transport Network
PAPR :Peak-to-Average Power Ratio
PDCP :Packet Data Compression Protocol
PDU :Packet Data Unit
PF : Proportional Fairness
PPP : Poisson Point Process
PRB : Physical Resource Block

PTP :Precision Time Protocol
QAM :Quadrature Amplitude Modulation
QoE :Quality of Experience
QoS : Quality of Service
QPSK :Quadrature Phase-Shift Keying
RAB :Radio Access Bearer
RBG :Resource Block Group
RCC :Radio Cloud Center
RE : Resource Element
REC : Radio Equipment Control
RF : Radio Frequency
RH :Remote Head
RLC :Radio Link Control
RRC :Radio Resource Control
RRH : Remote Radio Head
RRM : Radio Resource Management
RU :Remote Unit
SC-FDMA : Single Carrier-Frequency Division Multiple Access
SDN :Software Defined Network
SINR : Signal to Interference plus Noise Ratio
SRC :Sample Rate Conversion
TDD :Time Division Duplex
TTI :Transmission Time Interval
UE : User Equipment
UL : Uplink
WDM :Wavelength Division Multiplexing
WNC :Wireless Cloud Network

Chapitre 1

Introduction

Ces dernières décennies ont connu un bouleversement spectaculaire avec l'évolution des réseaux mobiles, faisant émerger à chaque fois une nouvelle génération mobile plus performante que celle qui la précède. Suite à cette évolution, accroître significativement la capacité du réseau et les débits offerts par chaque génération devient un véritable défi pour les entreprises exerçantes dans le domaine de la téléphonie mobile. En effet, leurs principaux objectifs sont la remédiation à l'augmentation du trafic qui croît à un rythme très rapide, et la volonté de maîtriser les coûts de déploiement, d'exploitation, de maintenance et de consommation énergétique.

Tout le long de cette évolution, les réseaux mobiles ont offert des services dont l'essor est remarquable. Effectivement, tout a commencé avec les deux premières générations qui étaient dédiées aux communications vocales, ont suivi ensuite la troisième et la quatrième génération qui ont permis à Internet et ses services de gagner nos téléphones mobiles tout en offrant des services multimédia enrichis. La naissance de la technologie LTE (Long Term Evolution) et de la quatrième génération, couplée à la démocratisation des smartphones et tablettes, a mené à une augmentation très forte des volumes de données échangés en mobilité. L'utilisation d'un mobile et de ses applications est désormais fermement ancrée dans le quotidien de nos concitoyens : les appareils portables connectés sont de plus en plus performants : ils remplacent bien souvent le téléphone fixe, l'appareil photo, voire l'ordinateur et même le téléviseur. Aujourd'hui, 5 millions de vidéos sont vues sur YouTube et 67 000 images sont chargées sur Instagram chaque minute [Eri14a] (voir FIGURE 1.1).

L'augmentation du nombre d'applications, leur diversification ainsi que l'amélioration de la qualité des réseaux mobiles ont conduit à l'augmentation de la demande, à l'apparition de nouveaux usages (objets connectés, drones, etc. . .) et de nouveaux utilisateurs. Actuellement, des efforts considérables de l'industrie et de la recherche visent à concevoir et définir une nouvelle génération «5G» prévue pour 2020 qui s'annonce évolutive et révolutionnaire permettant un débit Internet de plusieurs gigabits et intégrant des services riches et innovants tout en introduisant de nouveaux concepts et de nouvelles technologies. En effet, beaucoup d'entreprises réfléchissent actuellement à leur transformation digitale, en étudiant la manière dont la numérisation de l'économie pourrait



FIGURE 1.1 – Ce qui est fait aujourd’hui en une minute sur Internet [Eri14a]

modifier leurs produits, leurs marchés, leurs logistiques et bien sûr, leurs manières de produire. Suivant cette réflexion, le mouvement Industrie 4.0 prend de plus en plus d’importance. Au cours des dernières années, de nombreuses technologies ont émergé et pourraient changer radicalement les méthodes de fabrication. Les technologies les plus prometteuses sont les suivantes :

- L’Internet des Objets (IoT) qui mènera à de nouveaux produits & services ;
- L’impression 3D et l’efficacité énergétique qui apportent de nouveaux modes de fabrication ;
- La réalité augmentée et les robots collaboratifs, ou cobots, pour améliorer la productivité des opérateurs ;
- Les unités de production reconfigurables, les systèmes cyber-physiques (CPS), l’identification radiofréquence (RFID), le réseau industriel pour accélérer et sécuriser le flux de fabrication ;
- Le Big Data industriel pour améliorer l’efficacité des usines ;
- La simulation d’une usine virtuelle, La capitalisation des données de production pour améliorer le flux et l’ingénierie de conception.

En permettant des communications sans fil haute performance, la 5G possède un potentiel considérable pour des cas d’utilisation tels que l’automatisation de ligne d’assemblage, la prise de décisions par les véhicules à guidage automatique et la collecte de données par des capteurs pour informer les solutions de machine-learning et les applications de réalité augmentée et de réalité virtuelle. Bien que ces cas d’utilisation aient tous des besoins différents en termes de vitesse, de temps de latence ou de bande passante, la 5G est en mesure de les gérer dans le cadre d’un réseau unique. Selon leurs besoins spécifiques, certains cas d’utilisation peuvent être plus complexes et exigeants que d’autres.

Les systèmes de commande de mouvement, par exemple, chargés de contrôler les mouvements bien définis des machines, ont des exigences très strictes en termes de temps de latence, de fiabilité et de déterminisme, alors que la transmission constante de données au sein de l'énorme réseau IoT requiert une haute capacité et que la transmission de flux vidéo haute définition vers et à partir des périphériques de réalité augmentée est fortement tributaire de débits élevés.

1.1 Contexte et motivations

Augmenter la capacité d'un réseau, afin de contenir la croissance exponentielle du trafic de données, représente un défi continu pour les opérateurs de téléphonie mobile. Pour ce faire, les architectures C-RAN ont été définies pour être la nouvelle architecture d'accès radio des réseaux mobiles 5G. Elles sont proposées en tant que solutions potentielles rentables avec pour objectif de fournir un meilleur traitement des signaux radio, un partage des ressources de calcul/radio optimisé et une efficacité énergétique au niveau des stations de base.

L'architecture C-RAN, illustrée dans la FIGURE 1.2, redéfinit l'accès radio, grâce à son principe de séparation de l'interface LTE en deux unités, l'unité radio appelée Remote Radio Head (RRH) qui constitue la cellule et qui embarque entre autres des fonctions d'amplification et de conversion du signal (analogique-numérique (Analog-to-Digital Converter (ADC)), numérique-analogique (Digital-to-Analog Converter (DAC))) et l'unité de traitement appelée BaseBand Unit (BBU) qui représente l'élément « intelligent » du réseau d'accès regroupant les fonctions des couches protocolaires (L1/L2/L3) de l'interface radio. Les différentes BBU sont centralisées et virtualisées au sein d'un « Cloud », nommé BBU pool, se trouvant à une certaine distance des RRH. Les deux unités du C-RAN sont alors interconnectées à travers un réseau Fronthaul (FH).

L'architecture centralisée du C-RAN présente de nombreux avantages, tels que :

- Une meilleure efficacité spectrale et une adaptation dynamique de l'allocation de ressources en fonction du trafic utilisateurs ;
- Une densification du réseau incluant plus d'antennes et donc une meilleure réponse aux exigences de la 5G pour supporter le massive MIMO (Multi-Input Multi-Output) et l'introduction de plus de bandes de fréquences (bande 1.4 GHz, bande 3.5 Ghz et bande millimétrique) ;
- Une réduction considérable des coûts vu qu'un seul élément de processing du signal serait requis pour un plus grand nombre de stations, ce qui implique la réduction de la consommation électrique. De plus, les liens Backhaul reliant chaque station au cœur du réseau vont être remplacés par un seul lien reliant le BBU pool au réseau cœur ;
- Une vue d'ensemble permettant de gérer des applications très diverses avec différentes exigences en termes de latence et de débit ainsi que de faciliter l'implémentation des techniques de remédiation aux interférences et des algorithmes d'allocation conjointe des ressources.

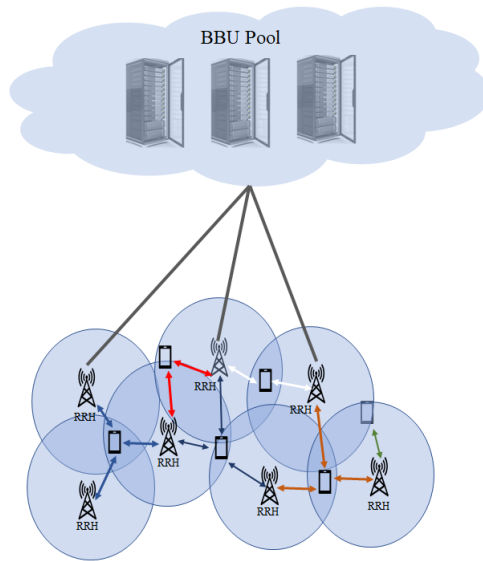


FIGURE 1.2 – Architecture C-RAN

En contrepartie, la connexion RRH-BBU requiert l'utilisation d'une large bande passante, la garantie d'une très faible latence dans le FH et la synchronisation des horloges entre les deux unités du C-RAN. Fournir de telles conditions implique nécessairement le déploiement de liaisons de fibre optiques entre les RRH et les BBU, la mise en place de nouveaux équipements dans l'infrastructure réseau et l'implémentation de certaines techniques de multiplexage.

Afin d'alléger ces contraintes, plusieurs modèles de centralisation (que nous détaillerons davantage dans le chapitre 7) qui diffèrent principalement dans le nombre de couches protocolaires centralisées dans le BBU pool sont proposés. La centralisation partielle est ainsi conçue pour atténuer l'utilisation de la bande passante, réduire les contraintes de latence/synchronisation ainsi que minimiser les CAPEX/OPEX. Un certain nombre d'opérateurs pensent par ailleurs à adopter ce modèle de division afin de réutiliser leurs infrastructures existantes et standardiser le transport du trafic de données à travers des technologies moins coûteuses telle que Ethernet, dans des réseaux Fronthaul/Backhaul unifiés.

De ce fait, l'utilisation d'Ethernet, entre le BBU pool et les RRH, peut présenter de nombreux avantages, notamment l'utilisation d'équipements moins coûteux, l'utilisation partagée d'infrastructures avec des réseaux d'accès fixes, pour obtenir un multiplexage statistique et des performances optimisées. Cependant, il existe un certain nombre de défis : des exigences de débit binaire ultra élevé provenant du transport de flux radio à bande passante accrue pour plusieurs antennes dans les futurs réseaux de téléphonie mobile, et une latence et une gigue faibles pour répondre aux exigences de délai et aux exigences de traitement commun. Par conséquent, la politique de planification des paquets au niveau du switch Ethernet a un impact sur la latence et l'équité pour toutes les RRH connectées.

1.2 Objectifs et contributions de la thèse

Dans un monde où tout est connecté, la concentration de la population dans les milieux urbains provoque la saturation des cellules à cause du nombre élevé d'appareils simultanément connectés. Pour relever le défi de la saturation des cellules existantes à densifier, les micro cells sont déployées pour répartir les utilisateurs sur de petites cellules à très hauts débits. Ce déploiement peut causer un niveau élevé d'interférences entre cellules voisines et une augmentation des coûts de déploiement et d'exploitation du réseau (unités de processing, batteries, cooling, électricité) et du nombre des transceivers (switch, ...). L'interférence entre stations de base est considérée comme le principal facteur limitant les performances des réseaux cellulaires, notamment en bord de cellule. Nous nous intéressons dans cette thèse aux techniques de remédiation aux interférences eICIC/ABS (enhanced Inter-Cell Interference Coordination/ Almost Blank Subframe) et JT-CoMP (Joint Transmission Coordinated Multi-Point). Le choix de ces deux techniques de remédiation aux interférences est expliqué par le fait que plusieurs études montrent que généralement JT-CoMP fournit les meilleurs gains de performances par rapport aux autres techniques CoMP (remédiation aux interférences inter-cellulaires) et que eICIC/ABS est la technique de remédiation aux interférences intra-cellulaires la plus utilisée. En outre, nous nous intéressons au Fronthaul dont l'infrastructure réseau doit prendre en considération les contraintes strictes exigées (large bande passante, faible latence et synchronisation).

Par conséquent, trouver un compromis entre des performances optimales et un coût de déploiement réduit, représente le principal défi dans la conception d'une architecture C-RAN. A travers ce travail de thèse, nous tenterons de relever ce défi, en proposant des outils de dimensionnement sur la partie radio (RRH) et des règles de dimensionnement du Fronthaul qui permettent d'assurer une architecture C-RAN à capacité optimale et à faible coût de déploiement, qui pourrait exploiter les équipements ainsi que les liaisons Ethernet, existant dans l'infrastructure d'un réseau cellulaire traditionnel. Ainsi, les objectifs de cette thèse sont les suivants :

- En premier lieu, nous avons proposé des solutions de partage de ressources entre cellules voisines et d'allocation dynamique de ressources tout en remédiant aux interférences et en assurant un débit global optimal. Pour ce faire, nous avons :

- proposé des modèles de chaîne de Markov qui reflètent le comportement de la technique eICIC /ABS et JT-CoMP et qui permettent d'analyser le débit selon la charge du trafic, les conditions radio et la distribution des ressources disponibles entre les cellules coordonnées
- proposé des modèles d'approximation avec des formes closes pour eICIC/ABS et JT-CoMP
- effectué des simulations sur Matlab en reproduisant des conditions radio réalistes afin de comparer et de valider les modèles proposés
- proposé des outils de dimensionnement en temps réels basés sur les formes closes pour la technique eICIC / ABS et la technique JT-CoMP dans les architectures C-RAN

Cette modélisation est proposée pour permettre l’optimisation dynamique des décisions d’allocation de ressources et de gestion d’interférences entre cellules afin de répondre aux exigences des différents utilisateurs en termes de débit et de latence. La gestion des interférences est basée sur des mécanismes de coopérations (JT-CoMP et eICIC/ABS) entre différentes cellules hétérogènes (macro et micro) nécessitant un échange intensif de messages de contrôle entre stations de bases souvent distantes et nécessitant des latences ultra faibles pour que ces algorithmes soient efficaces.

Ensuite, nous avons considéré la problématique de latence sur le Fronthaul afin de pouvoir mettre en place ces mécanismes de gestion d’interférences. Les spécifications 5G ont défini eCPRI (evolved Common Public Radio Interface) qui présente des mécanismes de partage des fonctionnalités de la BBU entre les différentes entités de calcul localisées au niveau des RRH et celles au niveau du BBU pool. Plusieurs niveaux de partage appelé splits fonctionnels ont été introduit par eCPRI pour répondre à différentes exigences des services.

eCPRI assure le transport sur fibre optique des données utilisateurs et des messages de contrôle entre BBU pool et RRH utilisant le protocole Ethernet (IEEE 802.3) permettant l’interconnexion d’équipements 2G, 3G, 4G et bientôt 5G provenant de plusieurs constructeurs. Malgré son adoption par le standard 3GPP et sa primordialité pour permettre la mise en place de la densification des nœuds et le massive MIMO, la mise en place de eCPRI dans les réseaux actuels rencontre des problèmes de dimensionnement du Fronthaul ainsi que des problèmes de partage de ressources entre cellule voisines.

- En deuxième lieu, nous avons donc effectué une étude sur le dimensionnement du Fronthaul et la capacité du C-RAN en termes d’agrégation de RRHs pour différents niveaux de splits fonctionnels permettant la prise en charge des mécanismes de gestion d’interférences par le BBU pool tout en respectant les contraintes de latence des utilisateurs. Cette partie vise à :

- comprendre les différents niveaux de splits en uplink (UL) et en downlink (DL) selon les standards CPRI et eCPRI
- proposer un nouveau split physique en UL
- comparer les splits physiques en termes de latence et de débit sur le Fronthaul dans le cas de RRH synchrones sous différents scénarios (slow fading et fast fading)
- dimensionner le Fronthaul pour le split proposé et calculer le nombre maximal de RRH à merger respectant la contrainte de latence

1.3 Plan des travaux de thèse

Pour une meilleure compréhension de l’ensemble de nos contributions, la FIGURE [1.3](#) présente l’organisation du manuscrit et le lien entre les différents chapitres.

Le chapitre 2 est dédié à l’évolution des réseaux mobiles (4G/ 5G) et l’architecture C-RAN en détaillant les fonctions de chacun des éléments le composant. Par ailleurs, nous avons présenté

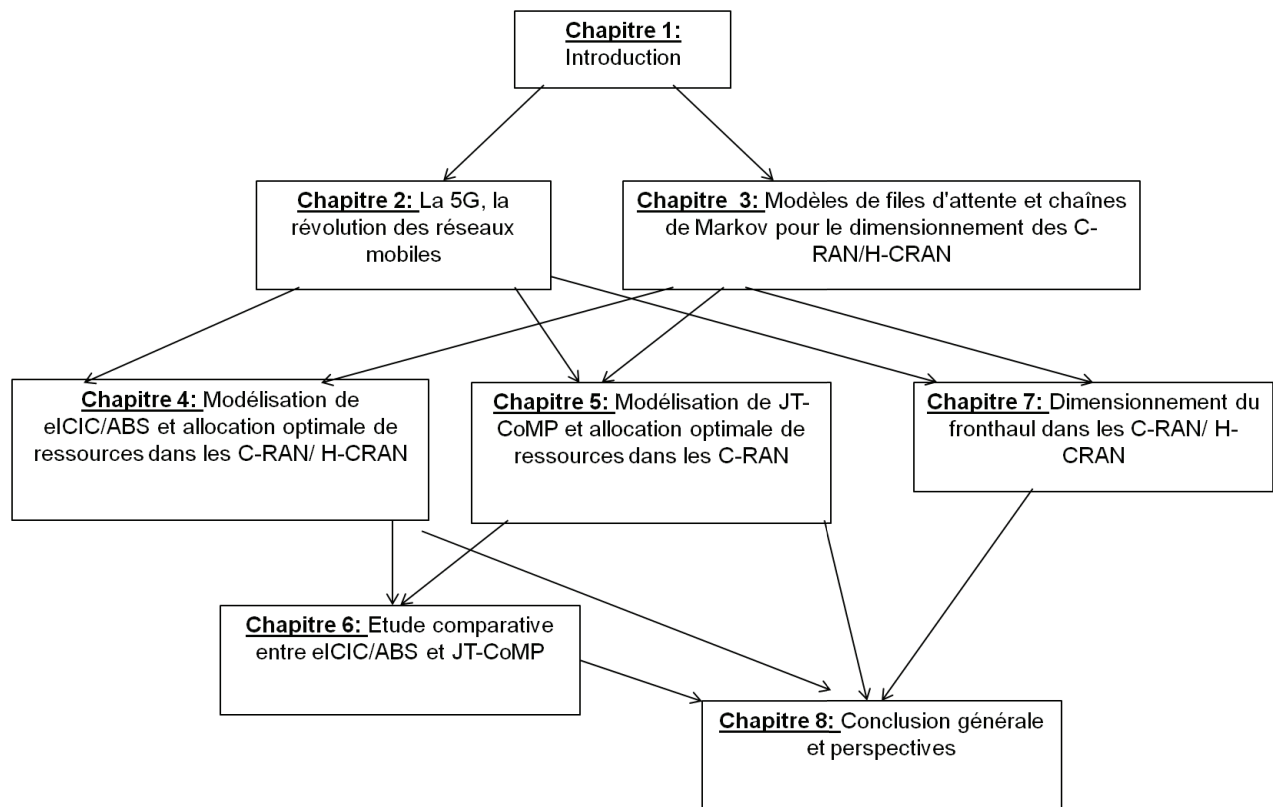


FIGURE 1.3 – Plan de la thèse

l'architecture H-CRAN qui permet d'allier les avantages des C-RAN et des HetNets ainsi que les différentes techniques de remédiation aux interférences : eICIC et CoMP.

Le chapitre 3 décrit les modèles mathématiques pour le dimensionnement des C-RAN/H-CRAN que nous avons utilisés et nécessaires à la compréhension des contributions proposées dans cette thèse. L'objectif de ces modèles mathématiques est d'une part de modéliser les techniques de remédiation aux interférences (voire eICIC/ ABS et JT-CoMP) et d'autre part d'utiliser les techniques d'optimisation sur les formes closes pour proposer des outils de dimensionnement qui, selon la charge du trafic, décident rapidement de la distribution des ressources entre les cellules coopératives afin d'assurer un débit global optimal.

Le chapitre 4, le chapitre 5, le chapitre 6 et le chapitre 7 présentent nos contributions proprement dites.

Le chapitre 4 est dédié à la modélisation du mécanisme eICIC/ABS pour allouer de façon optimale les ressources dans les C-RAN/ H-CRAN. Nos contributions dans ce chapitre sont les suivantes :

- Proposer des chaînes de Markov pour la modélisation de eICIC/ ABS ;
- Proposer des modèles de files d'attente avec formes closes pour évaluer les performances de eICIC/ ABS ;
- Comparer les résultats numériques des modèles de files d'attente avec les résultats des chaînes de Markov où des conditions radio et des distributions d'utilisateurs réalistes ont été considérées ;
- Proposer des outils de dimensionnement qui offrent aux opérateurs les décisions sur l'attribution des ressources entre la macro et la micro cellule pour atteindre la capacité optimale du système de façon rapide et adaptative.

Dans le chapitre 5, nous nous intéressons à modéliser mathématiquement JT-CoMP dans le but de déterminer précisément et en temps réel le nombre optimal d'utilisateurs en coopération avec CoMP ainsi que le nombre optimal de ressources à dédier pour la coopération tout en assurant un débit global optimal.

Pour ce faire, nos contributions dans ce chapitre sont les suivantes :

- Proposer des chaînes de Markov pour la modélisation de JT-CoMP ;
- Proposer des modèles de files d'attente avec formes closes pour évaluer les performances de JT-CoMP ;
- Comparer les résultats numériques des modèles de files d'attente avec les résultats des chaînes de Markov où des conditions radio et des distributions d'utilisateurs réalistes ont été considérées ;
- Proposer des outils de dimensionnement qui offrent aux opérateurs les décisions sur l'attribution des ressources entre les utilisateurs CoMP et non-CoMP ainsi que le nombre d'utilisateurs à servir en coopération pour atteindre le débit global optimal de façon rapide et adaptative.

Après avoir proposé les outils de dimensionnement pour la technique eICIC/ ABS et pour la technique JT-CoMP, dans le chapitre 6, nous proposons une étude comparative entre eICIC/ABS et JT-CoMP qui consiste à déterminer quelle est la technique de remédiation aux interférences, eICIC/ABS ou JT-CoMP, qui permettrait d’assurer un débit global optimal. Autrement dit, pour une charge de trafic donnée serait-il intéressant d’effectuer eICIC/ABS ou JT-CoMP afin d’atteindre le débit global optimal ? Lors de cette étude, nous déterminons la technique optimale et la configuration optimale sur l’allocation de ressources et des utilisateurs entre les cellules en coopération pour atteindre un débit global optimal.

Le chapitre 7 vise à proposer un dimensionnement du FH avec un temps de latence faible, une bande passante efficace et un déploiement économique sur le FH. Dans ce chapitre, nous limitons notre étude à la liaison montante (UL) du transport FH basé sur Ethernet dans une architecture C-RAN, car le traitement nécessaire à cet effet impose des délais plus stricts que ceux de la liaison descendante (DL). Nos principales contributions sont :

- Proposer un split alternatif en UL, notée II_U , dans un transport FH basé sur Ethernet ;
- Evaluer les besoins en capacité sur le FH selon le split proposé ;
- Proposer des formules mathématiques lorsque les RRH sont synchronisées afin de déterminer le nombre maximal de RRH à merger qui répond aux contraintes de latence et à la capacité du switch ;
- Analyser le split proposé II_U pour les situations de canal à atténuation rapide (fast fading) et lente (slow fading) et comparer les résultats de performance de ces variantes avec ceux du split physique eCPRI normalisé I_U sous une grande variété de scénarios (différentes bandes passantes et différentes capacités du switch).

Enfin, une conclusion générale résumera le travail accompli durant cette thèse. Des perspectives seront proposées pour apporter des améliorations à ce travail.

1.4 Liste des publications

Conférences internationales :

- H.TOUATI, H. CASTEL-TALEB, B. JOUABER, S. AKBERZADEH, A. KHLASS, “Simulation, modeling and analysis of the eICIC/ABS in H-CRAN”, The, 7th IFIP/IEEE International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks, September 26-28, 2018, Toulouse, Blagnac, France (publié)
- H. TOUATI, H. CASTEL-TALEB, B. JOUABER and S. AKBERZADEH, “Model-Based optimization for JT-CoMP in C-RAN”, The Third International Workshop on Flexible and Agile Networks : 5G and Beyond (IEEE FlexNets 2019) in conjunction with IEEE INFOCOM, Paris, France, 29 avril – 2 mai 2019 (publié)

- H. TOUATI, H. CASTEL-TALEB, B. JOUABER and S. AKBERZADEH, “Queue-based model approximation for inter-cell coordination with service differentiation”, IEEE Symposium on Computers and Communications (IEEE ISCC 2019), Barcelone, Espagne, 30 juin-03 juillet, 2019 (publié)
- H. TOUATI, H. CASTEL-TALEB, B. JOUABER and S. AKBERZADEH, “Split analysis and fronthaul dimensioning in 5G C-RAN to guarantee ultra low latency”, IEEE Consumer Communications & Networking Conference (IEEE CCNC 2020), Las Vegas, USA, 10-13 janvier 2020 (publié)
- H. TOUATI, H. CASTEL-TALEB, B. JOUABER and S. AKBERZADEH, “A new lower cost UL split option for ultra-low latency 5G fronthaul”, IEEE International conference on Wireless Communications & Mobile Computing (IEEE IWCMC 2020), Limassol, Cyprus, 15-19 juin 2020 (accepté)

Conférences nationales :

- H. TOUATI, H. CASTEL-TALEB, B. JOUABER, S. AKBERZADEH, A. KHLASS, “Modeling and performance evaluation of the eICIC/ABS in H-CRAN”, Cores 2018, Roscoff, France, 28-29 Mai, 2018 (publié)

Chapitre 2

La 5G, la révolution des réseaux mobiles

2.1 Introduction

L'industrie des télécoms prépare les technologies qui devraient avoir vocation à prendre la relève de la 4G, qui est en cours de déploiement actuellement. De nombreux travaux sont donc en cours pour préparer ces nouvelles technologies, dites « 5G ». L'idée qui commence à se forger derrière la notion de 5G est que celle-ci ne correspondrait pas à une simple augmentation des débits, comme cela a été le cas pour les précédentes générations. Les communications mobiles grand public, le téléchargement de vidéos et l'utilisation d'applications mobiles représentent l'essentiel de l'utilisation actuelle des ressources radio dans les réseaux 4G ; avec la 5G, l'objectif serait que le spectre des usages et la diversité des utilisateurs soient grandement élargis. La 5G vise des secteurs très variés, qui n'auraient pas nécessairement d'autre élément commun que cette technologie, mais qui sont des piliers importants d'une société : l'énergie, la santé, les médias, l'industrie ou le transport.

Ainsi, l'architecture C-RAN a été proposée [CLRH⁺14], avec pour objectif de fournir un meilleur traitement des signaux radio, un partage des ressources de calcul/radio optimisé et une efficacité énergétique au niveau des stations de base. De ce fait, le grand gain de traitement distribué peut être combiné avec la technologie émergente C-RAN. Cette combinaison fournira l'amélioration de l'efficacité spectrale. En outre, la gestion des ressources de façon centralisée et globale engendrera l'amélioration de la consommation d'énergie du réseau et la réduction des coûts.

Ce chapitre est composé de deux parties distinctes. La première partie consiste à décrire l'évolution des réseaux mobiles (4G et 5G) et la seconde à décrire les réseaux d'accès C-RAN.

2.2 L'évolution des réseaux mobiles (4G/ 5G)

Le LTE-A (ou LTE avancé, LTE evolution ou advanced en anglais) est une évolution du LTE normalisée dans le cadre de la Release 10 du 3GPP. Le LTE, tout comme les autres systèmes normalisés par le 3GPP, évolue au cours du temps, notamment en fonction des progrès techniques

et des demandes du marché. Chaque palier d'évolution est appelé une Release. La première Release du LTE est la Release 8, dont les spécifications fonctionnelles ont été finalisées en décembre 2008. Quant aux spécifications fonctionnelles de la Release 10, elles ont été finalisées en mars 2011. LTE-A apporte des améliorations au LTE, notamment afin d'offrir une capacité de cellule supérieure et des débits utilisateur plus élevés. LTE-A est naturellement compatible avec le LTE, ce qui signifie qu'un utilisateur en LTE-A peut utiliser un réseau LTE Release 8 et qu'un UE (User Equipment) LTE Release 8 peut être utilisé dans un réseau LTE-A. Dans ces deux cas, les UE ne pourront naturellement pas bénéficier des améliorations apportées par le LTE-A. Le nom LTE-A provient du processus de réglementation appelé IMT-Advanced du secteur Radio communications de l'Union Internationale des Télécommunications (UIT-R), dans lequel s'est inscrit son développement.

Dans cette première partie, nous commençons par énumérer les exigences techniques de la 5G et présentons la vision de son accès radio qui comprend à la fois 5G NR (New Radio) et LTE-A. Puis, vu que nous nous focalisons sur LTE-A comme accès radio 5G, nous décrivons donc les principes de l'interface radio du système LTE-A et fournissons les connaissances nécessaires à la compréhension des chapitres suivants.

2.2.1 Exigences techniques de la 5G

D'après l'édition 2020 du rapport de l'économie mobile de la Global System for Mobile Communications Association (GSMA), le nombre de personnes ayant souscrit à une offre de téléphonie mobile, devrait atteindre les 9.7 milliards à la fin de cette année [GSM20]. De plus, les équipements mobiles actuels génèrent et échangent de plus en plus de données, à travers les réseaux sociaux, les applications de partage/streaming vidéo, etc. Le volume de ces données devrait par ailleurs dépasser les 38 Exaoctets (Eo) par mois [Ind17]. Cette tendance devrait continuer lors des prochaines années (160 Eo prévus pour 2025 [Ind17]) en raison de l'orientation du contenu internet vers le format vidéo et en raison de l'apparition de nouvelles solutions IT génératrices et consommatrices d'une grande quantité de données (Internet of Things (IoT), Cloud gaming, etc.). D'autre part, certaines technologies «critiques» requièrent la présence d'une infrastructure réseau fiable et performante, pouvant fournir une très faible latence de transport, un haut débit de transmission et une disponibilité accrue (99,999%). Citons comme exemple la télémédecine et particulièrement la chirurgie assistée, dont la mise en pratique exige une forte disponibilité et une faible latence de la connexion, en raison de son fonctionnement en temps réel.

Bien que l'architecture LTE-A, actuellement déployée, permet de répondre aux besoins suscités, grâce notamment à un délai de transmission réduit à 50 ms ou encore à des débits théoriques descendants/montants (downlink/uplink) de 1 Gbit/s et 500 Mbits/s respectivement, sur des bandes passantes de 100 MHz [GRM+10], les réseaux cellulaires devront rapidement évoluer pour s'adapter à la croissance du trafic, aux exigences émises, en terme de qualité de service (Quality of Service (QoS)), par les fournisseurs de contenus/services ainsi qu'aux besoins des utilisateurs, en termes

d'accès internet et de qualité d'expérience (Quality of Experience (QoE)). La nouvelle génération (5G) est justement définie pour atteindre ces différents objectifs, avec les contraintes d'atténuation de la consommation d'énergie et de réduction des dépenses d'investissement/exploitation (CAPEX et OPEX) [ABC⁺14b].

2.2.2 Vision de l'accès radio 5G

La vision du réseau d'accès pour la 5G est illustrée dans la FIGURE 2.1. Le réseau d'accès 5G comprend 5G NR et LTE-A ou LTE evolution. LTE évolue continuellement pour répondre aux exigences de la 5G. LTE-A fonctionnera au-dessous de 6 GHz et NR fonctionnera de 1 GHz à 100 GHz. 5G NR est optimisé pour des performances supérieures ; il n'est pas rétrocompatible avec LTE-A. Cependant, une intégration étroite des évolutions NR et LTE-A sera nécessaire pour agréger efficacement le trafic de ces deux évolutions.

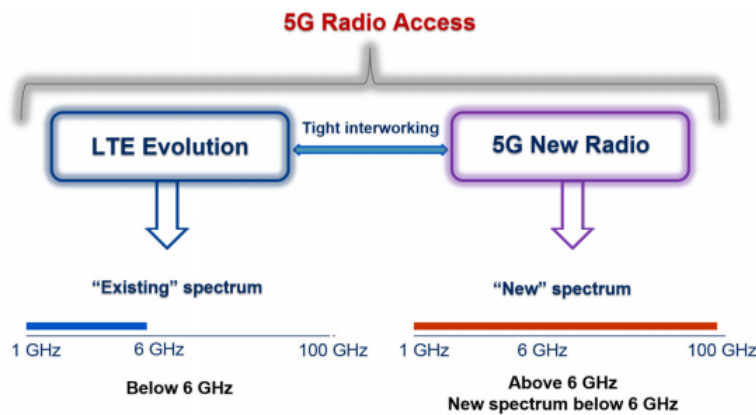


FIGURE 2.1 – Vision de l'accès radio 5G [ZAM⁺18]

Dans cette thèse, nous considérons que notre réseau d'accès pour la 5G se base sur LTE-A. Ceci nous amène donc à fournir une vue d'ensemble de l'interface radio du système LTE-A et à énumérer ses principales fonctionnalités qui constituent une base pour la compréhension des chapitres 4 et 5.

2.2.3 Interface radio du LTE/ LTE-A

L'interface radio assure le rôle clé de transférer par la voie des airs les données issues de la couche IP associées au service demandé par l'utilisateur. Ce transfert doit respecter des exigences de qualité de service (latence, débit) malgré un medium extrêmement variable, tout en optimisant l'accès à une ressource spectrale limitée. En outre, la disponibilité du spectre, variable selon les régions, impose de pouvoir s'adapter à différents types de bandes disponibles.

Cette section fournit une vue d'ensemble de l'interface radio du système LTE (identique à celle de LTE-A) qui constitue une base pour la compréhension des chapitres 4 et 5.

2.2.3.1 Modes de duplexage

Le duplexage définit la manière dont sont séparées les transmissions sur le DL (de la station de base vers l'UE) et sur le UL (de l'UE vers la station de base). Il existe deux principaux modes de duplexage, tous deux gérés par l'interface radio du LTE [ZM07] :

- **le duplexage en fréquence ou Frequency Division Duplex (FDD)** : la voie montante et la voie descendante opèrent sur deux fréquences porteuses séparées par une bande de garde.
- **le duplexage en temps ou Time Division Duplex (TDD)** : les voies montante et descendante utilisent la même fréquence porteuse, le partage s'effectue dans le domaine temporel.

Dans cette thèse, nous considérons le mode de duplexage FDD.

2.2.3.2 Structure de la trame LTE-FDD

Afin d'offrir des débits élevés, le LTE emploie la technologie OFDMA (Orthogonal Frequency Division Multiple Access) sur la voie descendante, et le SC-FDMA (Single Carrier - Frequency Division Multiple Access) sur la voie montante. Ces deux technologies sont des alternatives séduisantes à l'OFDM (Orthogonal Frequency-Division Multiplexing). Il est utile de rappeler que l'allocation de ressources aux utilisateurs s'effectue dans le domaine temporel uniquement (voir FIGURE 2.2) et que la transmission s'effectue en parallèle sur plusieurs porteuses à bande étroite appelées sous-porteuses. Ces sous-porteuses sont orthogonales, de sorte qu'un symbole de modulation (voir FIGURE 2.4) transmis sur une sous-porteuse ne subit pas d'interférence de la part des sous-porteuses adjacentes. Contrairement à l'OFDM, l'OFDMA et le SC-FDMA allouent les ressources dans le domaine temporel et fréquentiel (voir FIGURE 2.2). Ceci grâce à la fonction de multiplexage temporel qu'ils intègrent (avec une base de temps "TTI" (Transmission Time Interval) de 1 ms dans les réseaux mobiles LTE) qui permet de partager la bande de fréquence radio entre un plus grand nombre d'utilisateurs et avec plus de souplesse, cela permet notamment une adaptation rapide aux variations de débits qui caractérisent les flux d'accès à internet.

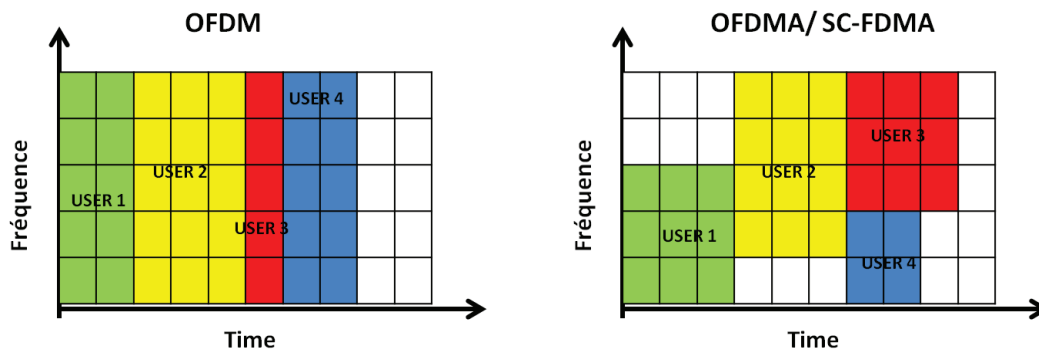


FIGURE 2.2 – Allocation de ressources en OFDM et OFDMA/SC-FDMA

Plus précisément, l'utilisation de l'OFDMA procure une grande flexibilité du nombre d'utilisateurs desservis simultanément et des débits de données hétérogènes pour différents utilisateurs. D'autre part, le principal avantage du SC-FDMA tient à un facteur de crête et à un PAPR (peak-to-average power ratio en anglais) plus faibles que ceux du codage OFDMA ; cela améliore l'efficacité énergétique du terminal, en diminuant la puissance crête d'émission et donc le poids et le coût du terminal (smartphone ou tablette tactile) et en contribuant à l'augmentation de l'autonomie de la batterie. Rappelons que la longueur de la trame LTE-FDD (identique à celle du LTE-A) est de 10 ms, divisée en 10 sous-trames d'une durée de 1ms. Chaque sous-trame consiste en 2 slots de longueur de 0.5 ms (voir FIGURE 2.3).

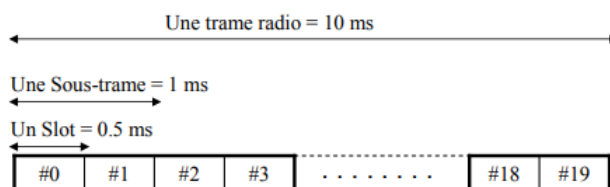


FIGURE 2.3 – Trame et slot de l'OFDMA et du SC-FDMA dans LTE-FDD

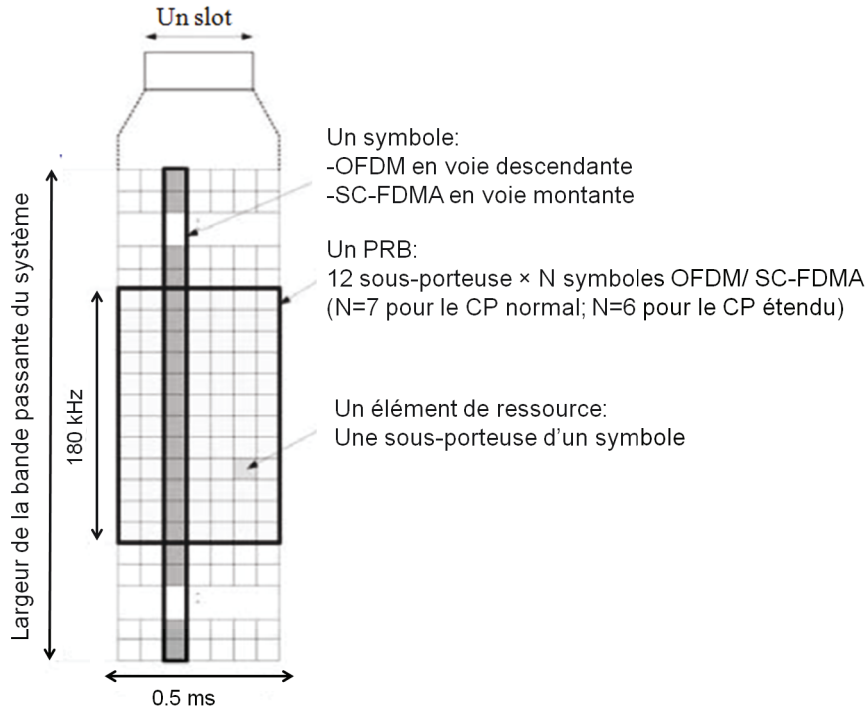
Ainsi, étant donné que l'allocation de ressources en LTE s'effectue à la fois dans les dimensions temporelle et fréquentielle, les ressources radio en voie descendante et voie montante se présentent sous la forme d'une grille temps-fréquence représentée dans la FIGURE 2.4.

Comme illustré dans la FIGURE 2.4, un PRB (Physical Resource Block), la plus petite unité de ressource fréquentielle pouvant être allouée à un UE, dure 0.5 ms, soit un slot. Un PRB est constitué de plusieurs symboles OFDM (ou SC-FDMA). Rappelons qu'un symbole OFDM représente le signal dans le domaine temporel correspondant à un bloc de symboles de modulation émis sur les différentes sous-porteuses de la bande du système. La durée d'un symbole dépend de la valeur du préfixe cyclique (CP (Cyclic Prefix) normal pour un mode d'opération point à point typique et un CP étendu pour de très grandes cellules). La largeur de bande d'un PRB est de 12 sous-porteuses, soit 180 KHz. Une sous-porteuse d'un symbole OFDM/SC-FDMA est appelée un élément de ressource. Un élément de ressource porte un symbole de modulation en OFDM.

Le nombre de PRB dans la dimension fréquentielle dépend de la largeur de la bande passante, comme indiqué dans le tableau suivant.

Largeur de bande (MHz)	1.4	3	5	10	15	20
Largeur de bande en termes de PRB	6	15	25	50	75	100

Tableau 2.1 – Largeur de la bande du système en nombre de PRB



71

FIGURE 2.4 – Grille de ressources temps-fréquence en voie montante et descendante

2.2.4 Armes de LTE-A

Comme expliqué précédemment, LTE-A vise simultanément une meilleure efficacité spectrale, des débits binaires plus élevés et une performance améliorée à la périphérie de la cellule tout en répondant à toutes les exigences du système 4G défini par l'UIT. Pour répondre à ces exigences, plusieurs fonctionnalités ont été développées dans LTE-A, parmi lesquelles nous énumérons quatre qui nous semblent essentielles :

- **La Carrier Aggregation (CA)** : C'est l'agrégation ou le groupement de plusieurs bandes passantes qui peuvent être de la même bande de fréquence ou de plusieurs fréquences. En LTE Release 8, la porteuse a une limite de 20 MHz, avec LTE-A l'objectif est d'aller jusqu'à 5 porteuses (100 MHz) ainsi le débit est multiplié par un facteur de 5. L'utilisateur pourra donc bénéficier, dans ce cas, de plusieurs ressources afin de satisfaire ses demandes (VoIP, streaming, gaming online, web browsing, ...).
- **Haute Modulation** : Ça revient à utiliser une modulation haute qui peut atteindre jusqu'à 256 QAM dans le sens descendant (c'est-à-dire envoyer 8 bits par symbole) et aussi 64 QAM dans le sens ascendant (c'est-à-dire 6 bits par symbole). Ainsi, l'utilisateur ayant de très bonnes conditions radio (près de l'eNB (evolved Node B) en absence d'aléas d'atténuation du signal) pourra utiliser de telles modulations ce qui permettrait d'améliorer le débit de l'utilisateur et par conséquent la capacité de la cellule.

- **Les répéteurs (Relay Node) et les réseaux hétérogènes** : Les répéteurs sont des stations de base avec une puissance faible utilisés afin d'améliorer la couverture et la capacité des zones en périphérie de la cellule. Ces nœuds agissent comme des répéteurs pour améliorer la qualité du signal et rediffuser le signal. Ils sont utilisés dans le contexte d'un réseau hétérogène (HetNet).
- **MIMO** : MIMO est une fonction qui utilise plusieurs antennes d'émission et de réception pour atteindre un débit binaire plus élevé. Cette fonctionnalité existait déjà dans LTE. Cependant, le LTE ne peut gérer que la configuration MIMO 4x4 même si 2x2 est la plus couramment utilisée. Pour LTE-A, il est possible de prendre en charge des configurations 8x8 dans la liaison descendante et 4x4 dans la liaison montante. L'augmentation de l'ordre MIMO est une clé pour améliorer l'efficacité spectrale et le débit. Il a été théoriquement prouvé que 8 flux spatiaux peuvent atteindre des vitesses 8 fois plus rapides qu'un système à entrée et sortie uniques (SISO).

Dans la seconde partie de ce chapitre, nous allons introduire l'architecture C-RAN. Nous commencerons par décrire ses différents composants et détailler les différentes solutions pouvant être intégrées. Enfin, nous nous intéresserons à l'architecture H-CRAN obtenue grâce à l'incorporation des HetNets dans les C-RAN où nous nous focaliserons sur les techniques de remédiation aux interférences.

2.3 Présentation de l'architecture Cloud Radio Access Network (C-RAN)

Introduit par IBM sous le nom de Wireless Network Cloud (WNC) [LSZ⁺10], le C-RAN a été par la suite décrit dans [CLRH⁺14] comme une architecture « green », conçue pour pallier à certaines limites présentes dans l'architecture RAN (LTE-A) actuelle. Quelques années plus tard, l'architecture C-RAN a été introduite et a été motivée dans de nombreux projets, tels que « Interworking and JOINT Design of an Open Access and Backhaul Network Architecture for Small Cells based on Cloud Networks (iJOIN) » [COR12], lancée en 2012 par la Commission européenne, et « Mobile Cloud Networking (MCN) » [Net13], lancé en 2013 également par la Commission européenne. En outre, de nombreux MNO (Mobile Network Operators) d'Asie-Pacifique ont été tentés par les avantages de l'architecture C-RAN et ont commencé à planifier son déploiement. Par exemple, Korean SK Telecom et NTT-DoCoMo ont annoncé des essais préliminaires du C-RAN en 2019. Il sera également adopté aux Jeux Olympiques en 2020 [SUFM16]. Le C-RAN a également suscité l'intérêt de nombreux opérateurs de réseau mobile et fournisseurs locaux, tels que AT&T, Vodafone, Verizon Communications, Orange, Intel, ZTE, Huawei et Nokia Bell Labs, qui investissent encore dans ses avantages et définissent son écosystème.

Dans ce but, le C-RAN devrait relever les défis suivants :

- Réduire les dépenses d'investissement (CAPEX), liées à l'acquisition de sites d'installation, à l'achat d'équipements composant une cellule, à l'installation du réseau d'accès, ainsi que celles d'exploitation (OPEX), liées aux opérations de maintenance, d'entretien des équipements et de la mise à jour des systèmes logiciels ;
- Réduire la forte consommation d'énergie dans les réseaux d'accès, causée par le déploiement d'un grand nombre de stations de base, nécessaire pour offrir une meilleure couverture. Pour cela, le C-RAN doit intégrer des solutions logicielles et matérielles, permettant d'optimiser la consommation d'énergie tout en améliorant les performances réseau. De plus, de nouvelles stratégies de déploiement d'antennes doivent être pensées pour améliorer la couverture du réseau ;
- Augmenter la capacité du réseau d'accès afin de prendre en charge le nombre croissant d'utilisateurs ainsi que les données générées par leurs terminaux mobiles ;
- Améliorer la gestion de la mobilité des utilisateurs et l'impact de leur consommation de données sur les stations de base, se trouvant dans leur zone géographique. Pour ce faire, il est nécessaire de prendre en considération le taux d'utilisation de chaque station de base, en fonction de la période de temps, pour déterminer les zones à forte densité d'utilisateurs. Cette solution aurait également pour objectif d'optimiser la consommation d'énergie en mettant en veille active les stations localisées dans des zones à très faible densité humaine ;
- Ouvrir le réseau d'accès radio aux fournisseurs de services pour le déploiement de solutions logicielles qui permettraient, d'une part, d'offrir aux utilisateurs de nouveaux services, contenus et applications avec de meilleures QoE et de performances possibles, et d'autre part, de pouvoir traiter et acheminer le trafic de données au plus près des terminaux, sans forcément passer par le réseau cœur de l'opérateur (EPC), ce qui permettrait de réduire le coût de transport des données vers l'EPC.

2.3.1 Composants de l'architecture C-RAN

L'eNodeB représente l'élément principal de la partie d'accès radio des réseaux LTE actuels. Il se charge de la réception/transmission des signaux radio à partir de/vers les terminaux utilisateurs (UE), de la gestion des ressources radio (Radio Resource Management (RRM)), de la compression des entêtes IP, de la sécurité, etc. Distribués aux extrémités du réseau, les eNodeB servent de stations de base aux UE ainsi que de relais vers l'EPC et internet. Toutefois, dans l'architecture C-RAN (FIGURE 2.5), les eNodeB sont remplacés par de simples antennes radio qui permettent de réceptionner et/ou de transmettre les signaux radio. Nommées Remote Radio Head/Unit (RRH/RRU), ces antennes sont déployées au plus près de l'utilisateur et de ses terminaux. Les fonctions de traitement de l'interface radio, appelées BaseBand Unit (BBU), sont par ailleurs séparées des antennes pour être rassemblées et centralisées au sein d'un nouvel élément appelé le BBU pool ou Cloud BBU (C-BBU). A la manière d'un Cloud, le BBU pool permet de traiter tous les signaux reçus à partir

des différentes RRH reliées à lui à travers un lien en fibre optique appelé fronthaul. Le BBU pool va alors exécuter les fonctions BBU en offrant de meilleures performances et en consommant moins d'énergie que les eNodeB qu'il remplace dans le réseau.

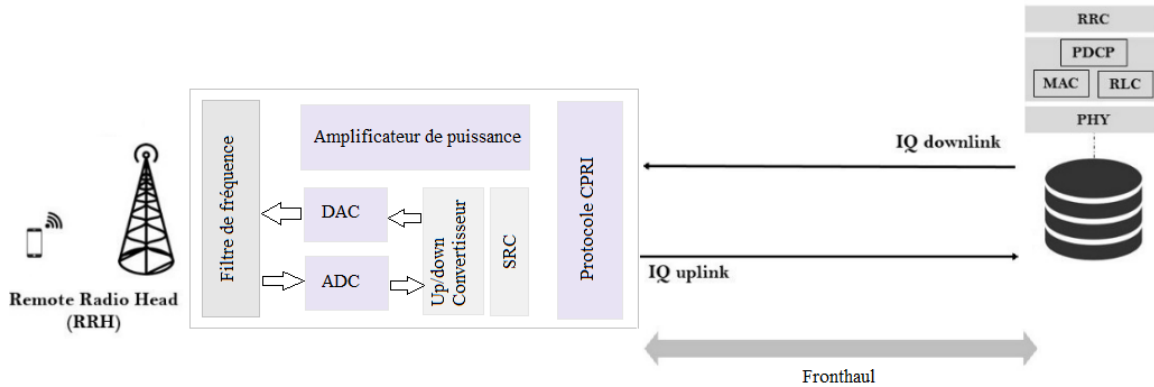


FIGURE 2.5 – Composition de l'architecture C-RAN

2.3.1.1 Remote Radio Head (RRH)

La RRH est composée des éléments suivants :

- Un filtre permettant d'enlever certaines bandes ou fréquences du signal transitant par l'antenne ;
- Un amplificateur qui permet d'accentuer la puissance du signal ;
- Un convertisseur de signal analogique-numérique (ADC) et numérique-analogique (DAC) ;
- Un convertisseur up/down et un module de conversion d'échantillons du signal (Sample Rate Conversion (SRC)) ;
- Un module d'encapsulation des données In-phase/Quadrature (I/Q) dans des paquets, utilisant le protocole d'interface, Common Public Radio Interface (CPRI) [CPR15] [Eri14b] ou d'autres standards tels que, Open Base Station Architecture Initiative (OBSAI) [I+16] [I+05] et Open Radio Interface (ORI) [Rya16] [ETS].

Les RRH permettent ainsi d'amplifier, filtrer et récupérer/transmettre les signaux radio à partir/vers des UE ou du BBU pool. Elles n'exécutent aucun calcul en local. Les RRH sont généralement déployées aux extrémités du réseau mobile et sont localisées à une distance, pouvant atteindre au maximum 40 Km de la BBU [CCY+14]. Toutefois, pour réduire la latence, la Next Generation Mobile Networks Alliance (NGMN) recommande une distance maximale de 10 Km [All15b].

2.3.1.2 Baseband Unit (BBU)

La Baseband Unit (BBU), appelée également Radio Cloud Center (RCC) dans certaines architectures, est composée d'un ensemble de couches et de fonctions, servant à traiter les signaux radio, transitant par le réseau d'accès. Alors qu'elle est associée à l'unité radio (RU/RH) dans un eNodeB dans l'architecture RAN de la 4G, la BBU est séparée de cette même RU (appelée alors RRH/RRU) dans une architecture C-RAN pour être, par la suite, regroupée avec d'autres unités de traitement au sein d'un même élément, appelé BBU pool. Ce dernier est généralement virtualisé et centralisé dans un serveur (ou un mini Datacenter) se trouvant entre les RRH et le réseau cœur (EPC). Le BBU pool fournit ainsi les ressources physiques nécessaires pour le traitement des signaux.

La centralisation des unités BBU en un seul pool offre un grand nombre d'avantages, dont nous énumérons :

- Le partage des ressources virtuelles et matérielles entre les différentes BBU ;
- L'augmentation de la capacité de calcul pour le traitement des signaux radio ;
- L'efficacité énergétique grâce à la possibilité de désactiver le fonctionnement de certaines unités BBU selon leur charge de traitement,
- La gestion améliorée de la mobilité des UE ainsi que des opérations de handover ;
- La simplification des opérations de configuration des cellules dont principalement la gestion des interférences et le partage de ressources.

Néanmoins, afin de fournir ces performances, il est nécessaire d'intégrer des outils de dimensionnement à temps réel pour l'ordonnement des ressources partagées au sein du BBU pool et de déterminer le bon placement de ce dernier par rapport aux RRH déployées dans le réseau.

Une unité BBU intègre les trois couches protocolaires composant l'eNodeB : la couche physique (PHY/L1), la couche (Media Access Control (MAC)/L2) ainsi que la couche contrôle (Radio Resource Control (RRC)/L3). Les fonctions de chacune de ces couches sont présentées ci-dessous.

2.3.1.2.a Couche physique

La couche physique représente la couche basse (L1) de la BBU. Elle se charge de la transmission/réception des données I/Q à travers le fronthaul grâce à l'exécution des trois fonctions fondamentales de la chaîne d'émission/réception : Le codage du canal, la modulation et l'adaptation de lien. Sans aucune prétention de notre part de fournir les détails (se référer à [BHW12]), nous expliquons très brièvement le principe de chacune de ces fonctionnalités, ceci va permettre au lecteur de bien comprendre la FIGURE 7.3 qui illustre les différents niveaux de splits de la couche physique.

- Codage du canal

Le codage du canal a pour but de protéger l'information émise en introduisant de la redondance dans le signal à transmettre, c'est-à-dire en liant les bits transmis entre eux par une règle connue par le récepteur, ce dernier pourra ainsi reconstruire des parties reçues corrompues. Notons d'ores

et déjà qu'une fonction de retransmission des blocs de transport erronés ne pouvant être corrigés est également disponible : l'HARQ (Hybrid Automatic Repeat Request), qui sera décrite dans la suite (voir chapitre 7).

Nous notons quatre fonctionnalités principales du codage canal : détection d'erreurs, codage correcteur d'erreurs, adaptation de débit et embrouillage. On pourra se rapporter aux ouvrages [Ber07] et [BHW12] pour une introduction complète au codage de canal.

- Modulation

Les bits codés et embrouillés sont ensuite convertis en symboles de modulation. L'opération de modulation imprime l'information sur le signal physique (une onde électromagnétique), en faisant varier l'une de ses caractéristiques physiques, typiquement sa phase et son amplitude. En fonction de la modulation utilisée, un symbole de modulation peut être associé à un ou plusieurs bit(s). Les combinaisons possibles d'une modulation sont souvent représentées par une constellation de point dans le plan des voies en phase (voie I) et en quadrature (voie Q) [JG07] déphasées de 90°. Pour une constellation, l'éloignement du point par rapport à l'origine indique l'amplitude, son angle indique le décalage de phase (voir FIGURE 2.6). Le LTE emploie les modulations BPSK (Binary Phase-Shift Keying), pour des informations de contrôle uniquement, QPSK (Quadrature Phase-Shift Keying), 16 QAM (16 Quadrature Amplitude Modulation) et 64 QAM (64 Quadrature Amplitude Modulation) dont chaque symbole porte respectivement 1, 2, 4 et 6 bits [JG07] et dont les points de constellations sont représentés par la FIGURE 2.6. Notons qu'une modulation, où un signal modulé transmet x bits par symbole, requiert 2^x combinaisons binaires différentes soit 2^x points de constellations [JG07].

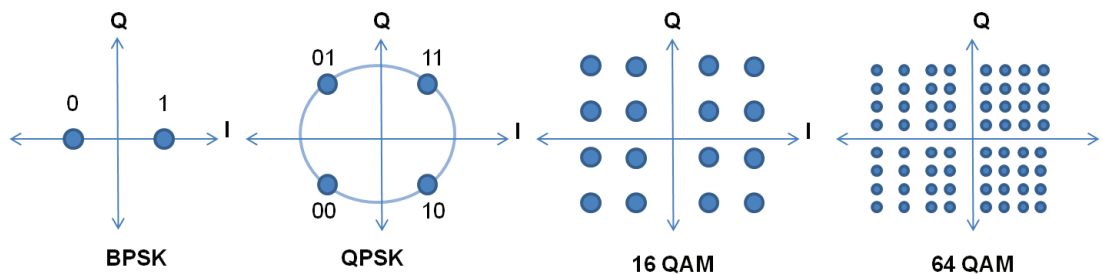


FIGURE 2.6 – Constellation des modulations possibles en LTE

- Adaptation de lien

L'adaptation de lien (Adaptive Modulation and Coding (AMC) en anglais) consiste à adapter le débit instantané de la transmission à la qualité du canal, ce qui revient à décider du MCS (Modulation and Coding Scheme) à allouer à l'UE pour chaque transmission. Le MCS est déterminé en fonction des conditions radio de l'UE, sujettes aux interférences, à l'atténuation et au bruit. En pratique, le MCS optimal est sélectionné sur la base du SINR (Signal to Interference plus Noise Ratio). Le SINR (en db) de l'UE u est obtenu grâce à l'équation suivante :

$$SINR(u) = 10 \log_{10} \left(\frac{PRx(u)}{\sum_i PRx_i(u) + N_o} \right) \quad (2.1)$$

où $PR_x(u)$ (resp. $PR_{x_i}(u)$) sont les puissances de réception (en watt) de eNB qui sert l'UE (resp. des eNB i interférents avec l'eNB qui le sert), et N_0 est la puissance du bruit thermique exprimée en watt sur toute la bande passante considérée. D'après [Acc17], le bruit thermique par Hz est égal à -174 dBm.

Notons que la puissance de réception donnée dans l'équation (2.1) dépend du pathloss $PL(u)$ entre l'UE u et l'eNB exprimé en db (décibels). Le calcul du pathloss diffère selon l'environnement et le type de cellules considérés. Dans le cadre cette thèse nous considérons un environnement urbain où des macro et des micro cellules peuvent être déployées. Dans nos études, nous considérons un environnement urbain normalisé par le 3GPP [Acc17] où le pathloss $PL(u)$ de la macro cellule est donné par :

$$PL(u) = 128.1 + 37.6 \times \log_{10}(R) \quad (2.2)$$

et celui de la micro cellule par :

$$PL(u) = 38 + 30 \times \log_{10}(R) \quad (2.3)$$

où R est la distance entre l'UE et l'eNB, R est exprimé en kilomètres dans le cas d'une macro cellule et en mètre dans le cas d'une micro [Acc17].

Une fois que le SINR pour un UE est déterminé, le CQI (Channel Quality Indicator) est sélectionné par l'UE selon la table de correspondances (SINR/ CQI) normalisée par le 3GPP [Acc11] (voir Tableau 2.2). Le CQI est un nombre compris entre 0 (pire) et 15 (meilleur) indiquant le MCS

SINR [dB]	CQI	CQI	Bits _{MCS}	CR
< -8	0	0	0	0
-8 ≤ SINR ≤ -6	1	1	2	0.076
-6 ≤ SINR ≤ -4	2	2	2	0.12
-4 ≤ SINR ≤ -2	3	3	2	0.19
-2 ≤ SINR ≤ 0	4	4	2	0.3
0 ≤ SINR ≤ 2	5	5	2	0.44
2 ≤ SINR ≤ 4	6	6	2	0.59
4 ≤ SINR ≤ 6	7	7	4	0.37
6 ≤ SINR ≤ 8	8	8	4	0.48
8 ≤ SINR ≤ 10	9	9	4	0.6
10 ≤ SINR ≤ 12	10	10	6	0.45
12 ≤ SINR ≤ 14	11	11	6	0.55
14 ≤ SINR ≤ 16	12	12	6	0.65
16 ≤ SINR ≤ 18	13	13	6	0.75
18 ≤ SINR ≤ 20	14	14	6	0.85
SINR ≥ 20	15	15	6	0.93

Tableau 2.2 – Correspondance $SINR/ CQI$ et $CQI/ Bits_{MCS}/ CR$

($Bits_{MCS}$ et CR) le plus efficace qui donnerait un taux d'erreur (Block Error Rate BLER) de 10% au plus. Le BBU pool requiert de chaque utilisateur les mesures du canal de liaison descendante, il

correspond à un retour indiquant le CQI de chaque bloc de ressources alloué à l'utilisateur. Après avoir reçu les mesures des utilisateurs, le BBU pool mappe les valeurs CQI de chaque utilisateur sur une valeur de schéma de modulation et de taux de codage.

Après avoir calculé le MCS, le scheduler attribue les ressources (RE) à chaque UE. Nous supposons que l'allocation de ressource type 0 est appliqué qui alloue des groupes (éventuellement disjoints) de PRB consécutifs en fréquence. Un groupe de PRB consécutifs en fréquence est appelé groupe de blocs de ressources (RBG, pour Resource Block Group). La taille des RBG en nombre de PRB en fréquence dépend de la bande passante du système, comme décrit dans le Tableau 2.3 [Acc11].

Bande passante (MHz)	Taille des RBG (PRB)
1.4	1
3	2
5	2
10	3
15	4
20	4

Tableau 2.3 – Taille des RBG en fréquence en fonction de la bande passante du système

Ainsi, le nombre de RBG disponible dans une trame noté S est calculé comme suit :

$$S = \frac{N \times 2 \times 10}{N_{PRB}} \quad (2.4)$$

où N représente la largeur de la bande du système en nombre de PRB (voir Tableau 2.1) que nous multiplions par 2×10 pour avoir la largeur de la bande du système en nombre de PRB sur une trame (2 correspond au nombre de slots par sous-trame et 10 au nombre de sous-trames par trame) et N_{PRB} représente le nombre de PRB dans un RBG (voir le Tableau 2.3).

Par ailleurs, le nombre de RBG attribués par trame pour un UE u , avec une taille ($size(u)$) en bits, et un nombre de bits de modulation ($Bits_{MCS}$), est obtenu comme suit :

(a) déterminer le nombre d'OFDM symboles (RE) requis :

$$N_{Symb}(u) = \frac{size(u)}{Bits_{MCS}(u)} \quad (2.5)$$

(b) déterminer le nombre de sous-canaux correspondants (c'est à dire le nombre de PRB) :

$$N_{Subch}(u) = \frac{N_{Symb}(u)}{7 \times 12} \quad (2.6)$$

(c) déterminer le nombre de RBGs correspondant :

$$N_{RBG}(u) = \frac{N_{Subch}(u)}{N_{PRB}} \quad (2.7)$$

2.3.1.2.b Couche MAC

La Couche MAC, appelée également couche 2 (L2), est composée de trois sous couches qui interviennent pour la transmission des paquets de données et des paquets de contrôles. La description de chacune de ces sous couches est donnée ci-dessous :

- Packet Data Compression Protocol (PDCP) : permet de compresser les entêtes des paquets pour améliorer l'efficacité spectrale. Elle permet également de chiffrer et de protéger l'intégrité des données de signalisation RRC. Enfin, cette sous couche se charge de la détection et de la suppression des doublons (paquets de données) qui apparaissent généralement lorsqu'un handover se produit entre deux cellules.
- Radio Link Control (RLC) : assure le contrôle des liaisons de données grâce à des fonctions de détection d'erreurs, de retransmission de paquets (en cas d'erreur), d'ordonnancement de ces derniers et d'optimisation des transmissions grâce à l'utilisation de fenêtres d'émission/réception.
- Medium Access Control (MAC) : permet d'accéder et d'adapter la transmission des données au canal de transport correspondant. Pour cela, la sous couche MAC utilise des fonctions d'allocation dynamique des ressources radio, de maintien de synchronisation (en lien montant) et de priorisation des flux de données. La sous couche MAC utilise également le mécanisme Hybrid Automatic Repeat Request (HARQ) pour une transmission fiable des paquets. Ce mécanisme sera décrit dans la section [7.3.1](#).

2.3.1.2.c Couche Radio Resource Control (RRC)

Cette troisième couche de la pile protocolaire assure la configuration et le contrôle des couches sous-jacentes. Ainsi, elle est responsable des fonctions suivantes :

- diffusion et décodage des informations système,
- gestion des envois/réceptions de radio messagerie (paging),
- gestion des connexions entre les couches RRC de l'UE et de l'unité de traitement,
- établissement, configuration et maintenance des connexions point à point des Radio Access Bearer (RAB),
- gestion des fonctions et des clefs de sécurité,

- gestion et contrôle de la mobilité des UE en mode connecté et en mode veille,
- gestion de la QoS et des mesures de l'UE.

2.3.1.3 Le fronthaul

L'ensemble des RRH sont reliées au BBU pool à travers un réseau FH, lequel permet de transporter les données I/Q généralement encapsulées par le protocole CPRI à travers un lien point-à-point. Alors que dans une architecture LTE-A, l'échange des signaux entre l'antenne et la BBU se fait au niveau d'un eNB (en interne), cet échange peut être étendu sur plusieurs kilomètres dans une architecture C-RAN. Par conséquent, l'infrastructure réseau du fronthaul doit prendre en considération les contraintes strictes, citées ci-dessous :

- une large bande passante, avec un débit supérieur à 10 Gbit/s, en raison de la charge supplémentaire apportée par le transport des données I/Q à travers le fronthaul [7],
- une faible latence de transmission nécessaire pour le traitement des signaux par la couche PHY du réseau d'accès (se trouvant dans le BBU pool),
- une synchronisation des horloges de l'équipement radio (Radio Equipment (RE)), se trouvant dans la RRH et celle du contrôle de l'équipement radio (Radio Equipment Control (REC)) qui se trouve dans le BBU pool.
- une faible fraction de fréquence du signal Fractional Frequency Offset (FFO), ainsi qu'une faible gigue (variation de latence au niveau du FH).

Afin de répondre à ces contraintes techniques, un certain nombre de solutions réseau ont été proposées pour intégrer le fronthaul et transporter le trafic radio entre les RRH et le BBU pool. Les avantages et les inconvénients de chaque solution sont décrits ci-dessous.

- Fibre optique dédiée (fibre noire) : La fibre optique dédiée peut fournir un débit puissant (jusqu'à 100 Gbit/s) sur de longues distances (jusqu'à 50 km) et elle représente la solution la plus efficace, en terme de performance. De ce fait, elle permet de relier les antennes radio au BBU pool et de satisfaire toutes les contraintes liées au transport des données I/Q à travers le fronthaul. En outre, elle peut être déployée de manière rapide et ne nécessite pas d'équipements particuliers pour son fonctionnement. Cependant, la fibre optique dédiée reste coûteuse à étendre et à maintenir, en raison de la forte utilisation de ressources de fibre et pourrait par conséquent ne pas atteindre un des principaux objectifs, définis pour le C-RAN qui est celui de la réduction des CAPEX/OPEX sur l'infrastructure réseau.
- Multiplexage en longueur d'onde (Wavelength Division Multiplexing (WDM)) : WDM est une technique qui permet de transporter plusieurs longueurs d'ondes optiques (40-80) sur une seule et même fibre [CCY+14], en utilisant différents canaux (chaque canal correspond à une longueur d'onde) grâce à l'intégration de modules de transpondeur au niveau des deux parties du C-RAN. L'avantage de WDM est son coût d'exploitation réduit comparé à celui de la

fibre optique dédiée et son optimisation des ressources optiques. En outre, cette technique garantit des débits élevés et une faible latence, paramètres indispensables pour le transport du trafic de données à travers le FH. WDM est considéré actuellement comme la solution la plus appropriée pour l'architecture C-RAN en raison de ces performances, sa scalabilité, ainsi que son coût amoindri. Par conséquent, son déploiement dans le fronthaul à travers un réseau de transport optique (Optical Transport Network (OTN)) est envisagé par un grand nombre d'opérateurs [Che16].

- Ondes millimétriques : Pouvant être utilisée sur de larges bandes de fréquence (70-80 GHz [49]), la transmission par ondes millimétriques représente une alternative intéressante pour le transport des données I/Q à travers le FH. Cette solution peut être mise en place dans des stades, campus universitaires ou encore dans des aéroports, où il est généralement difficile de déployer de la fibre optique. Les ondes millimétriques offrent une faible latence et permettent d'atteindre des débits de 10-40 Gbit/s sur de courtes distances (inférieures à 1.5 km [50]). Néanmoins, ses performances restent en deçà de celle définies pour le fronthaul C-RAN. Les ondes millimétriques restent tout de même une solution envisageable pour une utilisation dans le backhaul [HE11], notamment dans certaines régions où il peut être compliqué d'établir des connexions filaires.
- Ethernet : Ethernet représente un standard réseau mature, largement déployé dans les infrastructures existantes. Alors qu'il constitue le médium de transport principal dans le backhaul, l'adoption de ce protocole dans le FH est sérieusement envisagée par un grand nombre d'opérateurs et d'équipementiers qui voient en Ethernet un moyen peu coûteux et efficace pour relier les RRH au BBU pool. En outre, Ethernet présente de nombreux avantages pour le transport des données, tels que : 1) utilisation de fonctions d'opération, d'administration et de maintenance pour la détection et la localisation d'erreurs, le diagnostic des données ainsi que l'apport d'informations liées aux performances du réseau ; 2) faible coût d'exploitation des équipements Ethernet actuellement déployés dans les réseaux d'accès des opérateurs, ainsi que la possibilité de les partager avec d'autres infrastructures pour le transport des données ; 3) possibilité de contrôler, gérer et orchestrer le trafic de données transitant dans le réseau grâce à l'utilisation de technologies de type SDN (Software Defined Network et 4) transmission des données à très haut débit, pouvant atteindre les 1 Tbit/s entre deux extrémités du réseau [LdlCdD⁺14a]. Bien qu'Ethernet soit une alternative intéressante et moins coûteuse que WDM, cette solution possède néanmoins des limitations, pouvant restreindre son utilisation dans le FH. La première limitation est l'absence de synchronisation des horloges qui est nécessaire lors de la transmission des données I/Q entre la RRH et le BBU pool qui peut conduire à l'apparition d'erreurs de transmission et/ou à la corruption du signal source. La seconde limitation concerne les performances de latence et de gigue apportées par Ethernet qui restent en deçà du seuil requis pour le transport de données dans le FH. Ces performances peuvent par ailleurs se dégrader si des nœuds ou switches intermédiaires se trouvent entre les

RRH et la BBU.

Ethernet fait l'objet de nos études dans cette thèse. Afin d'adapter Ethernet à la transmission radio, un certain nombre de travaux ont été menés pour répondre aux contraintes imposées par le transport des données I/Q à travers le protocole CPRI [CPRI15]. Nous citons, comme exemple, le standard Synchronous Ethernet (SyncE), protocole défini par l'UIT. Le principe de SyncE est de fournir une synchronisation stricte des horloges avec une fraction de fréquence faible dans la couche physique du modèle Open Systems Interconnection (OSI), nécessaire pour le transport des fréquences radio à travers un réseau Ethernet [FGJ+08]. D'autres standards ont par ailleurs été définis prenant en considération cette synchronisation des transmissions à travers Ethernet, tels que, IEEE 1588 Precision Time Protocol (PTP) (synchronisation dans la couche liaison) [EL02] et IETF Network Time Protocol (NTP) (synchronisation dans la couche réseau) [M+85]. Une nouvelle spécification du protocole CPRI, appelée eCPRI [E+17] (eCPRI sera présenté dans le chapitre 7), est étudiée pour intégrer la 5ème génération de réseaux mobiles. Proposée par de grands noms de l'industrie des télécommunications, tels que, Huawei, Ericsson, Nokia et NEC et connus également sous le nom du groupe de travail NGFI, eCPRI aura pour objectif de réduire l'utilisation de la bande passante, d'exploiter les infrastructures existantes à travers Ethernet, d'adopter Ethernet comme la principale solution de transport dans le fronthaul, etc. Cette réduction de la bande passante est assurée grâce à l'ajout de fonctionnalités aux RRH (avec des options de partage de niveau du BBU pool). Toutefois, dans la littérature, il est admis que le fait d'ajouter plus de fonctionnalités aux RRH augmenterait certainement l'efficacité de la bande passante, mais réduirait les avantages du traitement radio centralisé et coopératif. Par conséquent, des splits des fonctionnalités de la couche physique sont envisagés. Ils permettent d'assurer un compromis entre ce que l'on gagne en termes d'assouplissement des exigences de FH et ce que l'on perd en termes d'avantages de centralisation C-RAN. Nous détaillerons les différents niveaux de splits eCPRI dans le chapitre 7.

2.3.2 Architecture H-CRAN

Afin d'allier les avantages des C-RAN et des réseaux hétérogènes (HetNets), l'architecture H-CRAN est proposée pour les réseaux cellulaires 5G [ABC+14a] [PLZW15]. Rappelons qu'un HetNet peut utiliser différentes technologies d'accès sans fil et différentes tailles de cellules pour améliorer les performances des cellules en périphérie et l'efficacité spectrale. En pratique, l'utilisation de petites cellules (small cells) peut augmenter le niveau d'interférences [DMMS13]. Le C-RAN permet donc d'atteindre les objectifs des HetNets puisque les décisions peuvent être centralisées et basées sur la connaissance globale dans le BBU pool [CLHD+14]. Par ailleurs, afin de remédier aux interférences entre les macro et small cellules, les mécanismes eICIC (enhanced Inter-Cell Interference Coordination) et CoMP (Coordinated Multi-Point) ont été proposés par le 3GPP pour LTE-A [DMMS13] [GJD+11] [JL12].

2.3.2.1 Technique de remédiation aux interférences eICIC

eICIC, technique de remédiation aux interférences intra-cellulaires, est défini par 3GPP Release 10. C'est une version améliorée de ICIC, précédemment défini dans 3GPP Release 8, évoluée pour supporter les environnements HetNets. Rappelons que pour remédier aux interférences inter-cellulaires, ICIC permet aux UE en périphérie (edge UEs) dans des cellules voisines d'utiliser différentes fréquences (PRB ou sous-porteuses). Par ailleurs, eICIC leur permet d'utiliser différents intervalle de temps (sous-trames) pour le même objectif. Ainsi, avec eICIC, une macro cell et une small cell qui partagent la bande passante peuvent utiliser les mêmes ressources dans des intervalles de temps différents (sous-trames différentes).

eICIC présente deux caractéristiques principales : la technologie ABS (Almost Blank Subframe) définie dans la Release 10 et la technologie CRE (Cell Range Expansion) définie dans la Release 11. L'ABS peut empêcher les UE edge dans les small cell d'interférer avec la cellule macro voisines grâce à l'utilisation des mêmes ressources radio mais dans différentes sous-trames. Quant à CRE, il étend la couverture d'une small cell afin qu'un plus grand nombre d'UE proches de la périphérie de la cellule puissent accéder à la small cell. Dans cette thèse, nous nous intéressons uniquement à eICIC/ABS (se référer à [TCE17] pour plus de détails sur eICIC/ CRE). La FIGURE 2.7 illustre le principe de eICIC/ABS.

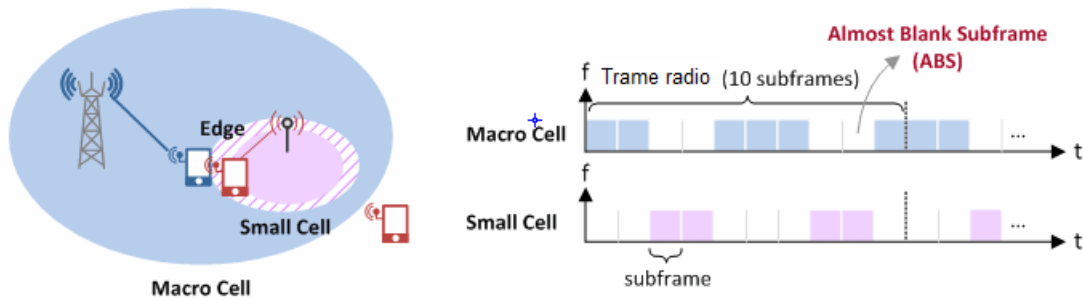


FIGURE 2.7 – Technique de remédiation aux interférences eICIC/ ABS

Avec eICIC/ABS, une part des ressources radio (sous-trame : subframes appelés ABS) est déléguée à la small cell (voir subframes en rose dans la FIGURE 2.7) et l'autre part est déléguée à la macro cell (voir subframes en bleu dans la FIGURE 2.7). Pendant les ABS, la macro cell reste silencieuse ainsi la small cell peut servir ses UE avec des interférences réduites.

Toutefois, pour que eICIC/ABS soit efficacement déployé, cette évolution nécessite des outils et approches pour analyser (et prévoir), en temps réel, l'attribution des sous-trames selon la charge du trafic ainsi que les exigences du trafic sur le FH et le BBU pool. S'ils ne sont pas correctement dimensionnés, le FH et le BBU pool peuvent constituer un goulot d'étranglement empêchant les RRH d'utiliser pleinement les ressources radio disponibles.

2.3.2.2 Technique de remédiation aux interférences CoMP

CoMP a été normalisée par 3GPP pour la première fois dans la Release 11 puis dans la Release 12 pour gérer à la fois les interférences entre small/macro cellules et les interférences inter-cellulaires (entre macro cellules voisines). Les interférences inter-cellulaires restent un problème majeur dans les réseaux cellulaires, étant donné qu'elles affectent les performances des edge UE et réduisent la capacité globale du réseau. Avec CoMP, plusieurs cellules coordonnent leurs transmissions pour que les signaux transmis ne s'interfèrent pas. Pour ce faire, les cellules coopératives peuvent être organisées en cluster. Ainsi, la gestion d'interférences peut être facilitée grâce au C-RAN où le BBU pool centralisé peut avoir une connaissance globale sur toutes les cellules d'un même cluster. Lorsque la ou les cellule(s) envoie(ent) des données aux UEs, elle(s) peut(vent) utiliser l'un des types de CoMP suivants en fonction de l'étendue de la coordination entre les cellules et la charge du trafic. Bien que différents types de CoMP peuvent être utilisés ensemble, nous allons expliquer les types spécifiques, un par un pour faciliter leur compréhension.

2.3.2.2.a Coordinated Scheduling/Coordinated Beamforming (CS/CB)

Dans le but de minimiser les interférences au niveau des edge UEs, CS et CB CoMP sélectionnent une des cellules coopératives comme cellule de transmission et l'utilise pour servir le UE.

- Coordinated Scheduling (CS) : L'idée de base du CS CoMP est d'allouer différentes fréquences de ressources (PRB ou sub-carriers) aux edge UEs. Un exemple est illustré dans la FIGURE 2.8.

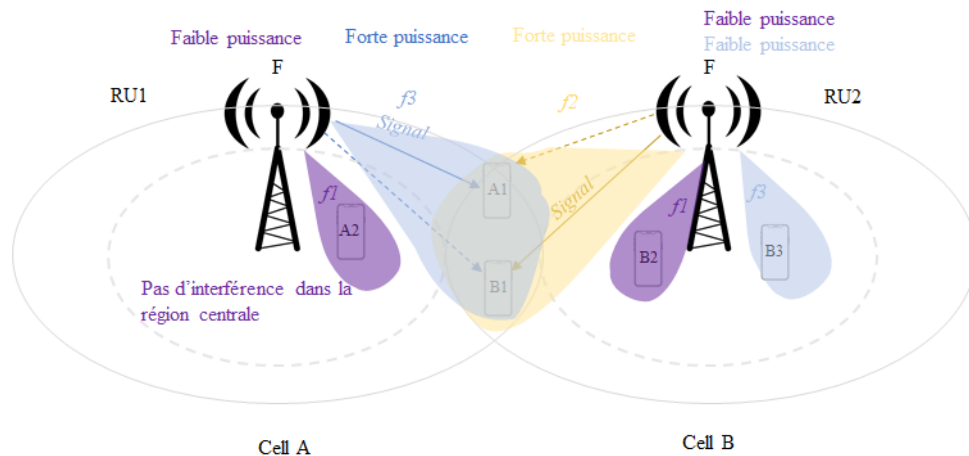


FIGURE 2.8 – Coordinated Scheduling (CS) CoMP

Dans la FIGURE 2.8, A1 et B1 deux UE en périphérie de cellule (edge UE) se voient attribuer une fréquence de ressource différente (f_3 et f_2 respectivement), évitant ainsi les interférences, et améliorant par conséquent leurs débits. Chaque UE va recevoir le signal de l'autre UE.

Toutefois, ces signaux ne vont pas interférer entre eux mais ils vont causer une dégradation de réception de leurs propres signaux.

- Coordinated Beamforming (CB) : CB CoMP alloue différentes ressources spatiales (beam patterns) aux UE en périphérie en utilisant la technologie d'antennes intelligentes. Sans CS, A1 et B1 peuvent utiliser la même fréquence (f_3 dans la FIGURE 2.9). CB CoMP permet à Cell A et Cell B de coopérer et d'allouer différentes ressources spatiales (beam pattern 1, beam pattern 2) à A1 et B1 en périphérie. Ces deux cellules préviennent les interférences en allouant le faisceau principal à leurs propres UE et un faisceau nul pour les UE voisins.

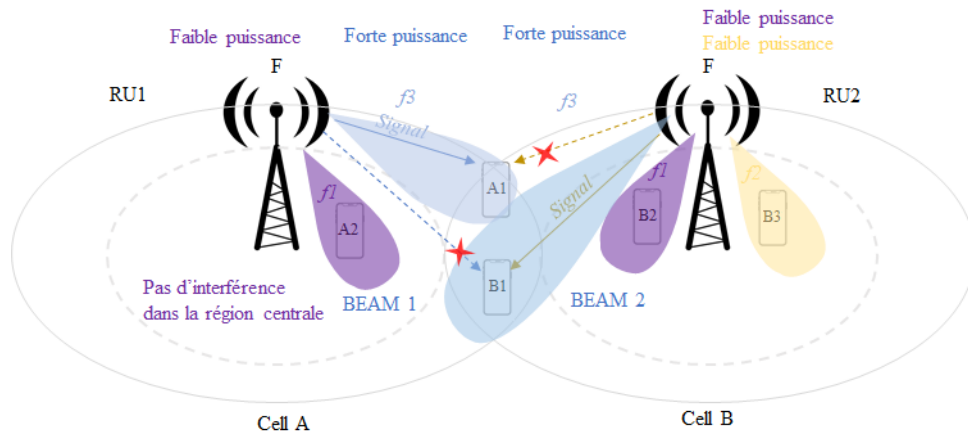


FIGURE 2.9 – Coordinated Beamforming (CB) CoMP

Généralement, CB est souvent utilisé avec CS. La FIGURE 2.10 illustre un cas où CS et CB sont conjointement utilisés. Cell A et Cell B coopèrent l'une avec l'autre pour allouer différentes ressources fréquentielles (f_3 , f_2) et différentes ressources spatiales (beam pattern 1, beam pattern 2) à l'UE A1 et l'UE B1, respectivement. Cette coopération est assez efficace : CS seul peut facilement éviter les interférences, et de plus CB peut assurer une meilleure qualité de réception. S'il est utilisé avec CB, CS peut obtenir de meilleurs débits pour les utilisateurs en edge, car CB aide A1 et B1 à éviter les signaux envoyés à l'autre, et à mieux recevoir ceux qui leurs sont destinés.

2.3.2.2.b Joint Transmission (JT)

Avec JT, plusieurs cellules peuvent transmettre la même information en utilisant la même ressource radio (fréquence et temps). Puisque les mêmes données sont envoyées, les performances en réception vont être améliorées. Dans la FIGURE 2.11, le edge UE A1 (dont la serving cell est la Cell A) reçoit les données de la Cell A, et également les mêmes données de la part de la Cell B, ce qui conduit à une meilleure qualité de réception au niveau du UE A1. Ainsi, le signal reçu d'une

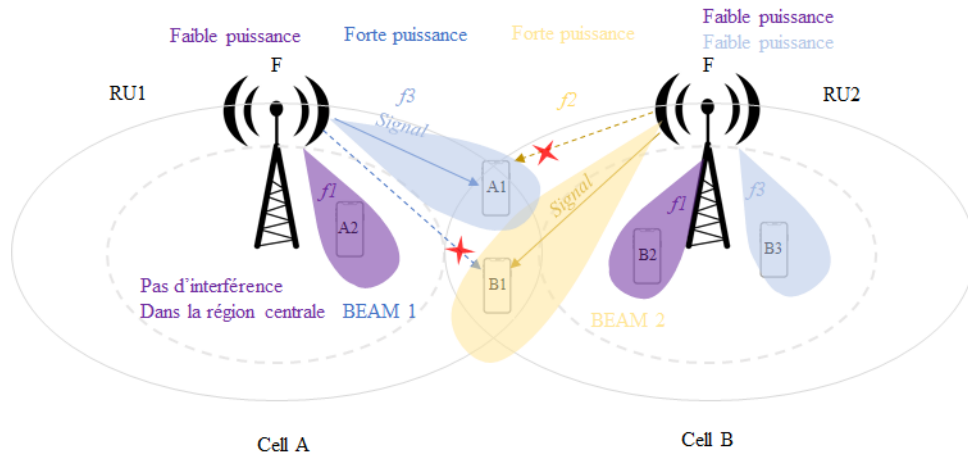


FIGURE 2.10 – CS/CB CoMP

autre cellule ne provoque pas d'interférences, mais renforce le signal destiné à l'UE A1. L'HARQ est réalisé par la serving cell

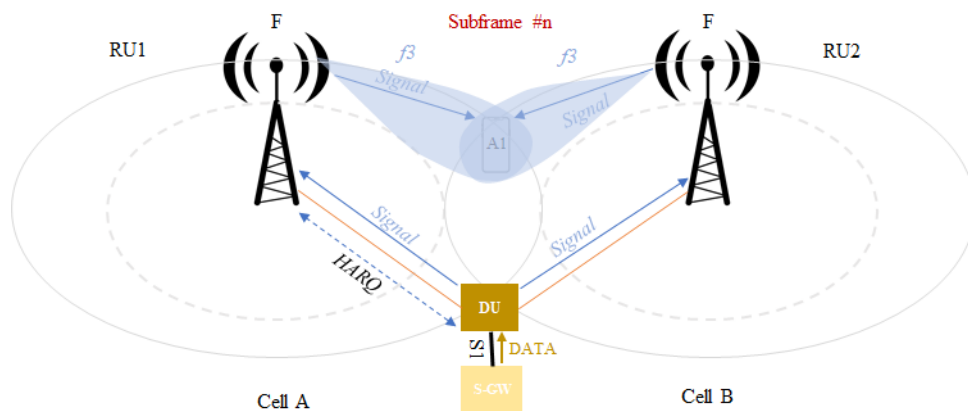


FIGURE 2.11 – JT CoMP

Grâce à l'amélioration des performances en réception, les débits des edge UE de la cellule peuvent être considérablement améliorés. Toutefois, pour que JT CoMP fonctionne efficacement, une synchronisation étroite entre les cellules est nécessaire.

2.3.2.2.c Dynamic Point Selection (DPS)

A la phase de préparation, DPS fonctionne de la même manière que JT en ce qui concerne l'envoi des mêmes données. Toutefois, lors de la phase de transmission des données, DPS fonctionne différemment. En premier lieu, la qualité des canaux des UE est contrôlée dans chaque sous-trame,

et les données sont envoyées par la cellule dont le pathloss est minimal. Les autres cellules qui ne sont pas sélectionnées restent silencieuses pour le edge UE. La qualité de réception au niveau des UE est améliorée plus efficacement étant donné que les données sont envoyées à partir de la cellule ayant la meilleure qualité du canal.

La FIGURE 2.12 montre comment DPS CoMP fonctionne. Contrairement à JT, une synchronisation précise n'est pas nécessaire, l'HARQ est réalisé par la cellule qui envoie les données.

Dans la littérature, plusieurs études montrent que généralement JT CoMP fournit les meilleurs

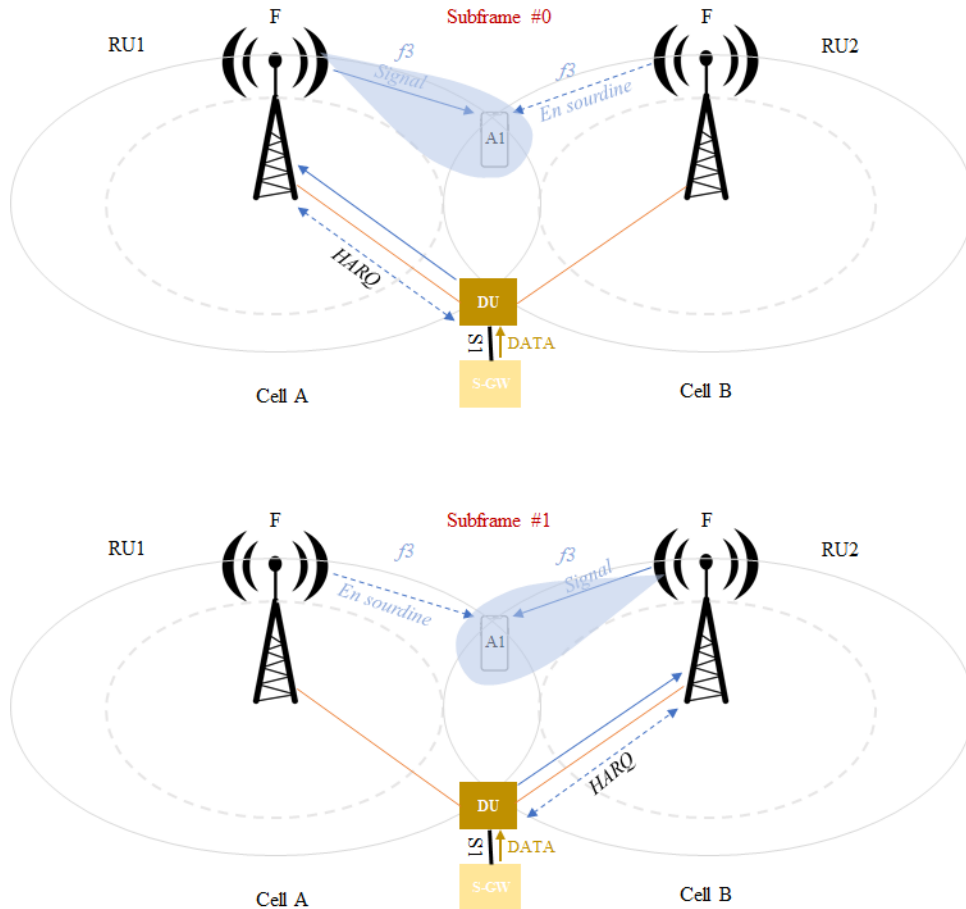


FIGURE 2.12 – DPS CoMP

gains de performances par rapport aux autres techniques CoMP [Abb16b] [LSC⁺12] [Rad13]. Le fait que JT CoMP coordonne les cellules voisines coopératives pour qu'elles transmettent le même contenu aux mêmes utilisateurs sur les mêmes ressources blocs améliore par la diversité et donc permet d'éviter les retransmissions. Toutefois, ceci réduit le débit total des deux cellules coopératives. C'est pour cela, nous nous focalisons sur JT CoMP et nous proposons des outils de dimensionnement qui offrent un déploiement efficace de cette technique.

2.4 Conclusion

Ce chapitre présente l'état de l'art concernant les problématiques étudiées dans le cadre de cette thèse. Cette analyse nous a permis de poser les cadres technologiques et architecturaux qui vont nous guider sur l'ensemble des travaux présentés dans les chapitres suivants.

Dans un premier temps, nous avons présenté les principales évolutions technologiques et architecturales des réseaux LTE et qui conduisent vers les réseaux 5G. Nous nous sommes essentiellement focalisés sur les aspects architecturaux. Nous avons présenté le principe de base du concept C-RAN qui consiste à séparer l'unité radio (RRH) de l'unité de traitement en bande de base (BBU). Dans les architectures traditionnelles, ces deux entités étaient jusqu'à présent regroupées en un seul élément (la station de base). Le C-RAN est aujourd'hui considéré comme une architecture candidate pour la nouvelle architecture d'accès radio des réseaux mobiles 5G. Mais plusieurs difficultés de dimensionnement et d'optimisation restent ouvertes. Nous avons décrit les fonctions de chacun des éléments composant l'architecture C-RAN, en détaillant les différentes solutions pouvant être intégrées dans les RRH, BBU et le réseau les reliant (le FH).

Dans ce chapitre, nous avons également énuméré les principaux défis à relever par le C-RAN, en perspective des besoins exprimés par les nouvelles générations de réseaux mobiles. Nous avons considéré pour cela les aspects performances, flexibilité, coût et efficacité énergétique.

Enfin, nous avons présenté l'architecture H-CRAN, permettant de combiner des réseaux d'accès de technologies différentes et/ou des cellules de tailles différentes et qui permet d'allier les avantages des C-RAN et des HetNets. Cette architecture permet de remédier aux interférences de façon plus efficace.

Ces différentes évolutions architecturales ont également été placées dans leurs contextes, à savoir, répondre aux exigences nécessaires aux techniques avancées de remédiation aux interférences, en particulier avec les techniques eICIC/ ABS et JT-CoMP que nous avons décrites dans ce chapitre. Ces deux techniques seront reprises dans la suite de ce manuscrit de thèse pour les modéliser et étudier leurs performances.

Dans cette optique, pour que eICIC/ABS et JT CoMP soient efficacement déployés, ces techniques nécessitent des outils et approches pour analyser (et prévoir), en temps réel, l'attribution des ressources selon la charge du trafic ainsi que le dimensionnement du FH qui peut constituer un goulot d'étranglement et empêcher les RRH d'utiliser pleinement les ressources radio disponibles.

Dans le chapitre suivant, nous présenterons les principaux outils de modélisations que nous avons adopté pour proposer des modèles et des approches de dimensionnement aussi bien pour l'allocation des ressources radio (avec eICIC/ABS et JT-CoMP) que sur le FH.

Chapitre 3

Modèles de files d'attente et chaînes de Markov pour le dimensionnement des C-RAN/H-CRAN

3.1 Introduction

Dans cette thèse, nous nous sommes intéressés à certains modèles mathématiques (chaînes de Markov et files d'attente multi-serveurs) pour le dimensionnement des C-RAN et des H-CRAN. L'analyse mathématique de ces modèles nous a permis de calculer les probabilités de blocage et de déduire les débits de sortie dans les C-RAN/H-CRAN. Ensuite, en nous basant sur les modèles de files d'attente, nous nous sommes intéressés à des techniques d'optimisation afin de maximiser la capacité globale du réseau en distribuant efficacement les ressources entre les cellules coopératives dans le cadre des techniques de remédiation aux interférences eICIC/ABS et JT-CoMP.

Ce chapitre consiste donc à faire un rappel sur les modèles mathématiques utilisés (modèles de files d'attente et chaînes de Markov) afin de proposer des solutions de dimensionnement des réseaux.

La théorie des files d'attente a été introduite en 1917 par Erlang dans le but de modéliser les réseaux téléphoniques. De nos jours, cette théorie intervient dans de nombreux réseaux actuels (réseaux informatiques, réseaux mobiles fixes ou mobiles), et systèmes informatiques. Dans certains cas, la dynamique d'un modèle de réseau de files d'attente (ouvert ou fermé) peut-être représentée par une chaîne de Markov, dont la distribution stationnaire est calculable soit analytiquement ou numériquement (plus ou moins rapidement). Cette théorie a pour objet l'étude de systèmes et réseaux, où des entités, appelées clients, cherchant à accéder à des ressources, généralement limitées, afin d'en obtenir un service. La demande concurrente d'une même ressource par plusieurs clients engendre des délais dans la réalisation des services et la formation de files de clients désireux d'accéder à une ressource indisponible. L'analyse théorique de tels systèmes permet de prévoir à l'avance les

performances, d'identifier les éléments critiques (goulôt d'étranglement) ou, encore, d'appréhender les effets d'une modification des conditions de fonctionnement. Les systèmes de files d'attente sont très étudiés et une abondante littérature couvre ce sujet (voir [Bay00], [Erl17], [Ken53], [Kle76]).

Dans ce chapitre, nous rappelons d'abord certaines notions sur les chaînes de Markov et les files d'attente. Ensuite, nous décrivons les modèles de file d'attente utiles dans le contexte de nos études. Enfin, nous donnons des formes closes et des bornes pour le calcul des paramètres de performance des files d'attente.

3.2 Chaînes de Markov

Les chaînes de Markov occupent une place très importante dans l'analyse des systèmes actuels. L'intérêt de ces modèles est la propriété d'absence de mémoire ce qui facilite l'analyse grâce à une théorie mathématique précise et efficace. Il existe deux types de chaînes de Markov : à temps discret et à temps continu. Nous les présentons ci-dessous.

Définition 3.2.1. [Med02] Un processus stochastique $X(n), n \geq 0$ à temps discret et à espace d'états E discret est une chaîne de Markov, si pour tout $n \geq 0$ et pour toutes séquences $i_0, i_1, \dots, i_{n+1} \in E$,

$$P(X_{n+1} = i_{n+1} | X_n = i_n, X_{n-1} = i_{n-1}, \dots, X_0 = i_0) = P(X_{n+1} = i_{n+1} | X_n = i_n), \quad (3.1)$$

où X_n représente l'état du processus à l'instant n .

Pour une chaîne de Markov, la prédiction du futur à partir du présent n'est pas rendue plus précise par des éléments d'information supplémentaires concernant le passé, car toute l'information utile pour la prédiction du futur est contenue dans l'état présent du processus. L'équation (3.1) représente la propriété de Markov. N'importe quel processus stochastique satisfaisant la propriété de Markov sera considéré comme une chaîne de Markov.

Nous appelons la probabilité conditionnelle suivante, notée par $P_{ij}(n)$,

$$P_{ij}(n) = P(X_{n+1} = j | X_n = i), i, j \in E, \quad (3.2)$$

la probabilité de transition de l'état i à l'état j à l'étape n .

Définition 3.2.2. [Med02] Une chaîne de Markov discrète est dite homogène, si $P_{ij}(n)$ ne dépend pas de n . En d'autres termes,

$$P(X_{n+1} = j | X_n = i) = P(X_1 = j | X_0 = i), \quad \forall n \in \mathbb{N} \quad (3.3)$$

Nous notons par P la matrice de probabilité de transition de la chaîne de Markov $X(n), n \geq 0$, dont chaque élément P_{ij} représente la probabilité de transition de l'état i à l'état j .

Définition 3.2.3. [Med02] Un processus stochastique $X(t), t \geq 0$ à temps continu et à espace d'état E discret est une chaîne de Markov, si pour tout $0 = t_0 < t_1 < t_2 < \dots < t_n < t_{n+1}$ avec $(t_0, t_1, t_2, \dots, t_n, t_{n+1}) \in \mathbb{R}^n$,

$$P(X_{t_{n+1}} = S_{n+1} | X_{t_n} = S_n, X_{t_{n-1}} = S_{n-1}, \dots, X_{t_0} = S_0) = P(X_{t_{n+1}} = S_{n+1} | X_{t_n} = S_n), \quad S \in \mathbb{R} \quad (3.4)$$

Définition 3.2.4. [Med02] Une chaîne de Markov à temps continu est homogène ssi,

$$P(X(S+t) = j | X(S) = i) = P(X(t) = j | X(0) = i) \quad (3.5)$$

Nous notons par Q le générateur infinitésimal de la chaîne de Markov $X(t), t \geq 0$.

Définition 3.2.5. [HB13] Une chaîne de Markov à temps discret est ergodique si elle est irréductible et les états sont récurrents non nuls et apériodiques.

Définition 3.2.6. [HB13] Une chaîne de Markov à temps continu est ergodique si elle est irréductible et les états sont récurrents non nuls.

Nous nous intéressons à des chaînes de Markov ergodiques et au calcul de la distribution stationnaire $p = (p_1, p_1, \dots, p_i, \dots)$, où p_i est la probabilité stationnaire à l'état i .

Théorème 3.2.1. *Pour la chaîne de Markov à temps discret $X(n), n \geq 0$ ergodique, il existe une distribution limite p , indépendante de la distribution initiale. Elle vérifie :*

$$\begin{cases} p = pP \\ \sum p_i = 1 \end{cases} \quad (3.6)$$

Théorème 3.2.2. *Pour la chaîne de Markov à temps continu $X(t), t \geq 0$ ergodique, il existe une distribution limite p , indépendante de la distribution initiale. Elle vérifie :*

$$\begin{cases} pQ = 0 \\ \sum p_i = 1 \end{cases} \quad (3.7)$$

Dans certains cas, le calcul de la distribution p (c'est à dire la résolution des équations (3.7)) mène à une formule «simple» dite «forme close» c'est le cas de certaines formules obtenues par des modèles de files d'attente classiques et que nous présentons ci-dessous. Dans d'autres cas, il s'agit de résoudre numériquement le système par des méthodes classiques (méthodes itératives, méthodes directes), ou simuler le système.

3.3 Modèles de files d'attente

Ce formalisme de modélisation est basé sur deux types d'entités : des clients et des ressources (serveurs). Un client qui nécessite un service, s'il trouve le serveur occupé alors il attend (c'est la file d'attente) jusqu'à ce qu'il reçoive son service. Ensuite, il quitte la file et se dirige vers une autre file ou quitte le système. Ce formalisme est largement utilisé pour modéliser des systèmes de partage de ressources : dans la vie quotidienne, l'informatique, les télécommunications, la gestion de production, etc. Un modèle de file d'attente est caractérisé par la notation de Kendall.

Dans le cadre de cette thèse, nous nous intéressons pas seulement aux modèles Markoviens. Ceci est dû au fait que la modélisation des systèmes étudiés nous écarte de l'hypothèse d'exponentialité de l'une ou des deux quantités stochastiques suivantes : loi des temps des inter-arrivées et de la durée de service, ou en considérant des arrivées groupées, ainsi nous pouvons aboutir à un modèle général.

Dans nos études, soit nous avons utilisé les chaînes de Markov pour représenter exactement le système ou soit nous avons simplifié certains paramètres du système pour se ramener à un modèle de file d'attente avec une forme close -il s'agit d'approximer le système exact afin de calculer rapidement des mesures de performances- ces résultats d'approximation ont été comparés avec des simulations.

3.4 Principaux modèles de files d'attente utilisés

Plusieurs modèles de files d'attente ont été développés pour permettre de concevoir des systèmes ou de représenter un système réel afin d'en analyser les performances. Dans cette section, nous allons d'abord décrire la terminologie utilisée puis nous présentons les modèles étudiés dans le chapitre 4, le chapitre 5 et le chapitre 7, enfin nous donnons les formules correspondantes utilisées (formes closes, bornes) pour calculer les performances qui nous intéressent. Le but n'est pas d'étudier de façon exhaustive tous les paramètres de performances de ces modèles, mais plutôt d'analyser un certain nombre d'entre eux utiles dans le contexte de chaque chapitre (voir Tableau 3.1).

	Modèles de files d'attentes	Paramètres de performance
Chapitre 4 et 5	$M/G/S/S$ $M^{[X]}/G/S/S$	- Probabilité d'occupation du serveur - Probabilité de blocage - Débit de sortie
Chapitre 7	$G/G/1$ $G^{[X]}/G/1$	- Temps d'attente/ séjour moyens - Temps d'attente/ séjour du <i>pième</i> percentile

Tableau 3.1 – Récapitulatif sur les modèles de files d'attente étudiés et les performances calculées

Afin de faciliter l'utilisation des modèles, le Tableau 3.2 présente les symboles et la terminologie utilisés. Nous rappelons dans la suite certaines définitions de ces paramètres.

λ	Taux d'arrivée moyen des clients
μ	Taux de service moyen
$\rho = \frac{\lambda}{\mu}$	Taux d'utilisation moyen du système
$t_s = \frac{1}{\mu}$	Temps de service moyen
W_n	Temps d'attente du client n dans la file
$E[W]$	Temps d'attente moyen dans la file
$W^{(p)}$	Temps d'attente du <i>pième</i> percentile
R_n	Temps de séjour du client n dans le système (temps d'attente en file, plus le temps de service)
$E[R]$	Temps de séjour moyen dans le système
$R^{(p)}$	Temps de séjour du <i>pième</i> percentile
p_0	Probabilité qu'il y ait zéro client dans le système
p_n	Probabilité qu'il y ait n clients dans le système
p^b	Probabilité que tous les serveurs soient occupés
X_j	Débit moyen de sortie pour les clients de classe j
X	Débit de sortie du système

Tableau 3.2 – Symboles des modèles de file d'attente utilisés

Définition 3.4.1. L'intervalle de temps entre le moment où le client arrive jusqu'à ce qu'il démarre son service, est connu comme le temps d'attente.

Définition 3.4.2. On appelle temps de séjour d'un client, l'intervalle de temps aléatoire qui sépare son entrée et sa sortie du système. Le temps de séjour s'obtient en additionnant le temps d'attente avec le temps de service de ce client.

Dans nos études, nous nous intéressons à des temps moyens (calculés entre tous les clients).

Définition 3.4.3. Le temps d'attente du *pième* percentile correspond au temps d'attente $W^{(p)}$ tel que :

$$P(W_n \leq W^{(p)}) = p, \quad \forall n = 1, 2, \dots, N \quad (3.8)$$

où N représente le nombre total de clients dans le système.

Définition 3.4.4. Le débit de sortie représente l'espérance mathématique du nombre de clients servis par unité de temps.

Définition 3.4.5. La probabilité de blocage reflète la probabilité que tous les serveurs soient occupés.

Ces paramètres de performances peuvent être calculés exactement ou bornés par des formules, nous les donnons par la suite pour les files d'attente étudiées dans cette thèse.

3.4.1 La file M/G/S/S

- . M (Markovien) : Processus d'arrivée Poissonien, de paramètre λ
- . G (générale) : temps de service de loi générale, avec une moyenne $\frac{1}{\mu}$
- . S : nombre de serveurs disponibles
- . S : capacité du système, incluant l'espace d'attente (=0) et le nombre de serveurs (=S)

C'est donc une file sans attente (tous les clients sont en service). Le service est assuré par S serveurs identiques, chaque serveur pouvant servir un client à la fois. Les clients arrivent selon un processus de Poisson de taux $\lambda > 0$. Les durées de service suivent une loi générale (aucune hypothèse n'est faite sur la distribution et les formules obtenues sont applicables quelque soit la distribution de probabilité) de moyenne finie $\frac{1}{\mu}$. Un client qui arrive entre en service s'il y a au moins un serveur disponible, sinon il sera rejeté et ne revient pas plus tard.

Il a été démontré que la M/M/S/S a une distribution de probabilité stationnaire donnée par une forme close [Kle76] permettant de calculer les probabilités d'occupation des serveurs tel que :

$$p_n = \frac{\frac{\rho^n}{n!}}{\sum_{i=0}^n \frac{\rho^i}{i!}} \quad (3.9)$$

où $\rho = \frac{\lambda}{\mu}$.

Ainsi, la probabilité de blocage p^b , correspondant à la probabilité que tous les serveurs (les S serveurs) sont occupés, est donnée par l'équation suivante :

$$p^b = p_S = \frac{\frac{\rho^S}{S!}}{\sum_{i=0}^S \frac{\rho^i}{i!}} \quad (3.10)$$

Cette égalité correspond à la formule d'Erlang B, et il a été démontré que cette formule est insensible à la distribution des temps de service des clients, et donc est valable pour une file M/G/S/S [Fak90]. A partir de la probabilité de blocage, nous pouvons facilement calculer le débit du système comme suit :

$$X = \lambda \times (1 - p^b) \quad (3.11)$$

3.4.2 La file M^[X]/G/S/S

Nous nous intéressons à un modèle de file d'attente avec des arrivées en groupes (ou batch). Un batch contient un ensemble de clients.

- . M^[X] : Processus d'arrivée Poissonien par batch, de paramètre λ
- . G (générale) : temps de service de loi générale, avec une moyenne $\frac{1}{\mu}$
- . S : nombre de serveurs disponibles
- . S : capacité du système, incluant l'espace d'attente (=0) et le nombre de serveurs (=S)

Lorsque nous avons des arrivées par batches, les batches peuvent avoir des tailles différentes ou la même taille. Nous dirons alors qu'un client est de classe j s'il appartient à un batch de taille j . Notons $g_j, j = 1, 2, \dots, S$ la distribution de probabilité de la taille des batches, les probabilités de queues (somme des probabilités à partir d'un état) correspondantes sont données par :

$$q_j = g_j + g_{j+1} + \dots (j = 1, 2, \dots, S) \quad (3.12)$$

Le service est assuré par S serveurs identiques, chaque serveur pouvant servir un client du batch à la fois. Le système fonctionne comme suit : à chaque fois qu'un batch de taille j ($j = 1, 2, \dots, S$) trouve à son arrivée n serveurs ($n=0, 1, \dots, S$) occupés, un nombre $\min(j, S-n)$ des clients du batch, choisis au hasard, occupent un nombre égal de serveurs inactifs ainsi ils rentrent en service. Les clients restants, s'il y en a, sont rejetés par le système (bloqués) et ne reviennent pas plus tard. Par conséquent, aucune file d'attente n'est autorisée à être formée et les clients bloqués sont considérés comme perdus pour le système. Les clients appartenant à des batches différents sont supposés avoir des temps de service indépendants.

Dans [Fak90], D. Fakinos examine également le comportement à l'équilibre de la $M^{[X]}/G/S/S$, où le système $M^{[X]}/M/S/S$ est un cas particulier de ce modèle et c'est celui que nous avons utilisé dans nos études.

Pour la $M^{[X]}/M/S/S$, la distribution des probabilités à l'équilibre (p_n), $n = 0, 1, \dots, S$, du nombre de serveurs occupés est donnée par :

$$p_n = \frac{\sum_{\substack{\nu_1, \dots, \nu_n \in N \\ \nu_1 + \dots + n\nu_n = n}} \prod_{j=1}^n \frac{(q_j \rho / j)^{\nu_j}}{\nu_j!}}{\sum_{\substack{\nu_1, \dots, \nu_n \in N \\ \nu_1 + \dots + S\nu_S \leq S}} \prod_{j=1}^S \frac{(q_j \rho / j)^{\nu_j}}{\nu_j!}} \quad (3.13)$$

où ν_j , pour $j = 1, 2, \dots, S$, représente le nombre de groupes de clients (ou batch de clients) de taille j (voir Tableau 3.2 pour le reste de la terminologie).

A partir de l'équation de distribution des probabilités (3.13), nous pouvons calculer la probabilité de blocage comme suit :

$$p^b = p_S = \frac{\sum_{\substack{\nu_1, \dots, \nu_S \in N \\ \nu_1 + \dots + S\nu_S = S}} \prod_{j=1}^S \frac{(q_j \rho / j)^{\nu_j}}{\nu_j!}}{\sum_{\substack{\nu_1, \dots, \nu_S \in N \\ \nu_1 + \dots + S\nu_S \leq S}} \prod_{j=1}^S \frac{(q_j \rho / j)^{\nu_j}}{\nu_j!}} \quad (3.14)$$

Nous pouvons déduire X_j , le débit moyen pour les clients de classe j , comme suit :

$$\begin{cases} X_1 = \lambda \times g_1 \times (1 - p^b) \\ X_j = \lambda \times g_j \times \left(\sum_{l=0}^{S-j} j \times p_l + \sum_{l=S-j+1}^{S-1} (S-l) \times p_l \right); \text{ pour } j > 1 \end{cases} \quad (3.15)$$

Le débit de sortie du système est donc calculé comme suit :

$$X = \sum_{j=1}^S X_j \quad (3.16)$$

3.4.3 La file G/G/1

- . G (générale) : Processus d'arrivée selon une loi générale, de paramètre λ
- . G (générale) : temps de service de loi générale, avec une moyenne $\frac{1}{\mu}$
- . 1 : un seul serveur disponible

La $G/G/1$ considère que les arrivées de clients suivent une loi de probabilité générale (G) (arbitraire) avec un taux λ . Toutes les arrivées sont en compétition pour une seule ressource (un serveur) et sont stockées temporairement (en mémoire tampon ou en file d'attente) selon la discipline FCFS, et ont un temps de service $E[S] = \frac{1}{\mu}$ dont la distribution est également générale. Nous exigeons une charge du système $\rho = \frac{\lambda}{\mu} < 1$, pour la stabilité.

Notons que la $G/G/1$, considérée dans le chapitre 7, ne possède pas de formes close pour le calcul du temps d'attente moyen. En effet, nous le calculons en nous basant sur l'approximation de Allen-Cunneen [Kin61] :

$$E[W] \leq E[S] \times \frac{\rho}{1 - \rho} \times \frac{C^2[T] + C^2[S]}{2} \quad (3.17)$$

où nous définissons le carré du coefficient de variation d'une variable aléatoire \mathbb{X} par $C^2[\mathbb{X}] = \frac{\text{Var}[\mathbb{X}]}{E[\mathbb{X}]^2}$, T par la variable aléatoire modélisant le temps d'inter-arrivées des clients dans la file et S par la variable aléatoire du temps de service.

Quant au temps de séjour moyen, il est calculé comme suit :

$$E[R] = E[W] + E[S] \quad (3.18)$$

En se basant sur la loi de congestion de Kingsman, les auteurs de [PHL18] obtiennent le temps d'attente du p ème percentile comme suit :

$$W^{(p)} = \max \left\{ 0, E[S] \times \frac{1}{1 - \rho} \times \frac{C^2[T] + C^2[S]}{2} \times \ln \left(\frac{\rho}{1 - \frac{p}{100}} \right) \right\} \quad (3.19)$$

Le temps de séjour du *pième* percentile est calculé par l'équation suivante :

$$R^{(p)} = W^{(p)} + E[\mathbb{S}] \quad (3.20)$$

3.4.4 La file $G^{[X]}/G/1$

- . $G^{[X]}$ (générale) : Processus d'arrivée par batchs selon une loi générale, de paramètre λ
- . G (générale) : temps de service de loi générale, avec une moyenne $\frac{1}{\mu}$
- . 1 : un seul serveur disponible

Etant donné que nous avons des arrivées par batchs, afin de modéliser correctement le fonthaul (voir chapitre 7), nous allons nous intéresser au cas où tous les batchs ont la même taille et arrivent à des instants précis (temps discret). Notons que la $G^{[X]}/G/1$ ne possède pas de formes close pour le calcul du temps d'attente moyen. Dans SF07, Schleyer et Furmans montrent qu'il est possible de calculer la distribution des temps d'attente d'une $G^{[X]}/G/1$ en se basant sur la factorisation de Wiener–Hopf. En effet, ils utilisent d'abord l'équation de Lindley Lin52, sous sa forme discrète, pour déterminer la distribution du temps d'attente des batchs qui arrivent. Ensuite, ils résolvent cette équation grâce aux algorithmes de Grassmann et Jain GJ89, basés sur la factorisation de Wiener–Hopf.

3.5 Conclusion

En complément des aspects technologiques et architecturaux présentés dans le chapitre précédent, ce chapitre est consacré à la description des outils de modélisation et de dimensionnement utilisés dans cette thèse.

Nous avons choisi de présenter ici les principaux outils mathématiques sur lesquels nous nous sommes appuyés pour proposer des solutions de modélisation et de dimensionnement. En particulier, nous rappelons dans ce chapitre certaines notions des chaînes de Markov et des files d'attente. Nous avons essentiellement décrit les modèles de file d'attente multi-serveurs et/ou avec services et arrivées générales ou groupées ($M/G/S/S$, $M^{[X]}/G/S/S$, $G/G/1$ et $G^{[X]}/G/1$). Le choix de ces techniques est guidée par plusieurs constatations.

En premier, les modèles multi-serveurs peuvent très bien refléter la nature des trames radio LTE et le principe de transmission multi-porteuses. En effet, avec les techniques basées sur l'OFDMA, plusieurs blocs de ressources radio pourront être exploités en parallèle à chaque trame. D'un autre côté, les arrivées groupées peuvent très bien modéliser un trafic dense et variable. Ces aspects seront expliqués et exploités dans les chapitres suivants.

Par ailleurs, ce choix de présenter et d'utiliser ces outils est aussi guidé par notre souhait de rechercher des formes closes et à défaut des approximations permettant d'estimer en temps réel (ou presque), les performances des différentes techniques de coopération inter-cellulaires (eICIC/ ABS

et JT-CoMP) et d'en déduire rapidement la meilleure configuration et le meilleur dimensionnement (paramétrage) possibles, selon la charge et la distribution du trafic. En effet, seules des techniques à formes close permettent d'analyser rapidement l'état du réseau et du trafic et de proposer des solutions rapides de configuration et de dimensionnement.

Notons que tout en présentant ces modèles, nous avons également expliqué comment les principaux paramètres de performances peuvent être extraits de ces modèles, en particulier les délais, les débits et les taux de perte.

Les chapitres suivants vont décrire et expliquer plus en détail le choix de ces techniques, comment elles sont utilisées pour modéliser les techniques de coopération radio et donner les résultats d'optimisation obtenus. Les avantages et les limites de ces techniques seront également discutés.

Chapitre 4

Modélisation de eICIC/ABS et allocation optimale de ressources dans les C-RAN/ H-CRAN

4.1 Problématique et contexte

Dans ce chapitre, nous nous intéressons à la technique de remédiation aux interférences eICIC/ABS (se référer à la section [2.3.2.1](#)). Lorsque eICIC/ABS est appliqué, une partie des ressources radio (sous-trames) est déléguée aux micro cellules. Pendant les sous-trames allouées, la macro cellule reste silencieuse ainsi la micro cellule peut servir ses utilisateurs avec des interférences amoindries. Avec cette approche, l'un des défis les plus importants consiste à déterminer le moment d'activation et de désactivation de la micro cell et de déterminer le nombre de sous-trames à lui déléguer si elle est déployée. Notre objectif consiste donc à proposer des outils de dimensionnement très rapides (presque à temps réel) qui permettent de déterminer le moment d'activation de la micro et le nombre de ressources à lui déléguer.

D'excellentes contributions ont été fournies dans [\[GMR⁺12\]](#), [\[And13\]](#), [\[PWL⁺15\]](#) et [\[WHG⁺14\]](#) pour mieux comprendre les réseaux HetNets. Afin d'étudier les exigences de ces réseaux, les auteurs dans [\[GMR⁺12\]](#) discutent des modèles théoriques comprenant l'analyse de la couverture et du débit, la distribution du SINR, l'association de cellules et les modèles spatiaux cellulaires. Dans la littérature, plusieurs travaux investissent dans les problèmes de gestion d'interférences dans les HetNets en utilisant eICIC/ABS. L'objectif principal est de déterminer le nombre optimal de ressources radio que la macro devrait déléguer à ses micro cells associées tout en assurant une capacité maximale. Dans [\[CWW⁺13\]](#), les auteurs analysent l'interférence en utilisant la géométrie stochastique pour dériver le nombre d'ABS requis pour garantir le niveau de QoS déterminé de l'utilisateur. Pour ce faire, les auteurs ont utilisé un processus stochastique à 2 dimensions de type PPP (Poisson Point Process) afin de modéliser la position des stations de base et des utilisateurs. La

distribution des SINR est donc dérivée à partir de ce modèle. L'objectif des auteurs est de déterminer le nombre d'ABS optimal qui maximise le débit des utilisateurs non victimes (qui n'interfèrent pas). Leur problème d'optimisation est formulé comme suit :

$$\begin{aligned}
& \max_{N_A} C_{NV}(N_A) \\
& \text{s.t.} \quad C_V(N_A) \geq C_{V,min}, \\
& \quad \quad 0 \leq N_A \leq N_S
\end{aligned} \tag{4.1}$$

où N_S représente le nombre de sous-trames dans une trame, N_A représente le nombre d'ABS dans une trame, C_{NV} signifie le débit des UE non victimes, C_V signifie le débit des UE victime (qui interfèrent) et $C_{V,min}$ représente le débit minimal requis par les UE victimes, c'est un paramètre choisi ou fixé en entrée. Ainsi, en fixant $C_{V,min}$, les auteurs font varier N_A de 0 à N_S en retenant la première valeur de N_A permettant d'assurer $C_{V,min}$. Cette valeur est considérée comme la solution optimale.

Dans [SP14], les auteurs proposent deux algorithmes pour ajuster dynamiquement le nombre d'ABS. Le premier algorithme, basé sur la charge instantanée exige seulement l'information sur le nombre d'UE dans la macro et la micro. Dans cet algorithme, trois types de sous-trame sont définis : normale, ABS et optionnelle. Sur chaque sous-trame la puissance de transmission est calculée, si elle est égale à la puissance de transmission maximale alors cette sous-trame est définie comme sous-trame normale. Si elle est égale à la puissance de transmission basse alors c'est un ABS. Sinon, elle est définie comme sous-trame optionnelle et dans ce cas si le pourcentage d'UE associés à la micro est supérieur au pourcentage d'ABS qui lui sont délégués et que le pourcentage d'UE associés à la macro est inférieur au pourcentage de ses sous-trames alors cette sous-trame optionnelle est définie comme ABS sinon elle est définie comme sous-trame normale. Quant au second algorithme, il est basé sur la moyenne des métriques PF (proportional fair) dans la macro et la micro. Nous rappelons que la métrique PF pour un UE est calculé comme le rapport entre le débit estimé pour l'UE sur un PRB à l'instant t et le débit moyen dans la cell à l'instant t. L'idée est d'utiliser les mesures PF moyennes dans la macro cell et dans la micro cell comme des indicateurs afin de savoir si le débit de chaque cellule est amélioré ou détérioré, c'est-à-dire comme une estimation de la nécessité d'attribuer plus ou moins d'ABS par période de mise à jour d'ABS (appelée période ABS et définie par les auteurs comme étant 8 sous-trames). Au début de la période ABS, il est décidé s'il faut augmenter, diminuer ou garder le même nombre d'ABS que celui utilisé dans la période ABS précédente. En particulier, un ABS supplémentaire est attribué si les conditions radio dans la micro se détériorent (c.-à-d. la métrique de PF moyenne de la micro augmente) et en même temps les conditions radio de la macro s'améliorent (c'est-à-dire que la mesure de FP moyenne de la macro est en baisse). Il est évident que si ces conditions sont vérifiées, le nombre d'ABS est limité par la période d'ABS et par le fait qu'au minimum une sous-trame normale doit être configurée pour la remontée des mesures. Toutefois, si les conditions radio de la macro cell se détériorent, l'algorithme définit un ABS comme sous-trame

normale dans la prochaine période ABS. Si l'algorithme n'est pas dans l'un de ces deux cas alors le nombre d'ABS garde la même valeur que dans la période ABS précédente.

Dans [ZCJL14], un framework analytique basé sur la théorie des jeux a été proposé pour permettre aux femto cellules de remédier aux interférences avec les macro cellules voisines. Les auteurs dérivent les probabilités de couverture des UE de la macro et de la femto en se basant sur le modèle stochastique PPP (Poisson Point Process). Ils considèrent le cas où le propriétaire de la femto cell est capable de gérer les interférences en fonction de la récompense offerte par l'opérateur de la macro. Pour ce faire, ils formulent le problème de gestion d'interférences en tant que jeu de tarification tel que l'opérateur de la macro et le propriétaire de la femto sont les deux joueurs. L'utilité ou le gain de l'opérateur de la macro consiste en deux paramètres : les frais que l'opérateur perçoit des UE de la macro pour la fourniture du service et le coût que l'opérateur paie. Quant à l'utilité du propriétaire de la femto cell, elle consiste en trois paramètres : les frais initiaux qu'il doit payer à l'opérateur, l'avantage de transmission (qui représente le produit entre le prix unitaire du débit minimal facturé par l'opérateur à la femto cell, le rapport entre le nombre d'ABS et nombre de sous-trames et le débit moyen dans un PRB d'un canal partagé) et la récompense de l'opérateur. En se basant sur ces paramètres, les auteurs obtiennent l'équilibre de Nash sous une forme close, qui révèle les paramètres de gestion d'interférences de la femto cell et le paiement dû à l'opérateur de la macro.

Dans [LZH⁺18], les auteurs ont proposé un framework basé sur la théorie des jeux pour gérer les interférences entre la macro et les micro cells. Dans ce framework, chaque micro cell gère les interférences en mettant en sourdine des sous-trames qu'elle dédie aux utilisateurs de la macro cellule à usage exclusif (sans interférences). En retour, la station de base macro annonce le prix de ces sous-trames. Pour ce faire, les auteurs formulent le problème comme un jeu de Stackelberg qui considère la macro et les micro cellules comme leader et suiveurs respectivement. Plus précisément, dans le framework proposé, la macro cell détermine le prix optimal qui permet de maximiser son utilité et le broadcast à toutes les micro cellules. L'utilité des micro cells consiste en deux paramètres : les bénéfices rapportés de la macro cell et la perte de débit des UE de la micro due à la réduction des ressources. Quant à l'utilité de la macro, elle consiste à la différence entre le profit réalisé des micro cells et le paiement dû aux micro cells pour la proportion de ressources qui lui sont attribuées. En se basant sur la valeur du prix, chaque micro cell décide d'optimiser la proportion d'ABS afin de maximiser son utilité. En utilisant l'optimisation convexe et l'induction rétrograde (backward induction, une méthode, principalement utilisée en théorie des jeux, qui consiste à déduire les effets d'une action en analysant ses conséquences éventuelles), les auteurs déterminent l'unique point du jeu Stackelberg qui correspond aux stratégies optimales pour la macro et les micro cellules.

Dans [LZSW19], une autre approche basée sur la théorie des jeux est proposée pour gérer les interférences entre la macro cell et les micro cellules. Les auteurs formulent ce problème d'optimisation comme un jeu de Stackelberg avec un leader (la macro) et plusieurs suiveurs (les micro). Ensuite, ils prouvent l'existence de l'équilibre de Stackelberg et obtiennent les meilleures stratégies pour la

macro cell et les micro cellules en utilisant l'induction rétrograde.

Un système à logique floue (fuzzy logic system) est adopté dans [DSG14] pour calculer le nombre d'ABS en se basant sur l'historique des performances d'autres systèmes. Ainsi, en suivant les règles de type "si... alors", ce système peut simultanément travailler avec des données numériques et des informations linguistiques en utilisant un mapping entre les données en entrée et en sortie. Les entrées incluent la densité des UE, le débit des macro cellules et la qualité du canal des micro cellules. Les sorties consistent en le nombre d'ABS.

Dans [ZJWY16], la configuration ABS est formulée en tant que problème de consensus de forme générale, puis résolue à l'aide de l'approche ADMM (un algorithme qui permet de résoudre les problèmes d'optimisation en les décomposant en sous-problèmes dont chacun est plus simple à résoudre).

Dans [WH19], les auteurs ont mis au point un mécanisme permettant d'ajuster conjointement les paramètres de l'ABS et du mécanisme DRX (Discontinuous Reception : une méthode utilisée dans les communications mobiles pour conserver la batterie des UE) afin d'améliorer les performances dans les HetNets. ABS et DRX semblent être deux méthodes indépendantes pour la gestion des interférences et de la puissance dans LTE-A, respectivement. Toutefois, ils partagent certaines similitudes en termes de comportement des UE. En effet, les UE des macro cellules ne peuvent pas être servis dans un ABS, ainsi les UE peuvent désactiver leurs émetteurs-récepteurs et se mettre dans un état de sommeil. Le mécanisme de gestion d'interférences et de la puissance proposé synchronise d'abord les cycles ABS et DRX. Basé sur la qualité du canal et la demande de trafic de chaque UE, il estime la quantité de ressources requises par l'UE. Il calcule ensuite un nombre approprié d'ABS pour équilibrer les charges dans les macro cellules et les micro cellules et décide des paramètres DRX en conséquence.

Après une analyse approfondie de la littérature concernée, il s'est avéré que les modèles de files d'attente avec des formes closes manquaient pour permettre une analyse dynamique des performances de eICIC. De tels modèles peuvent permettre une configuration en temps réel du mécanisme eICIC en fonction de la charge du trafic et des conditions radio des utilisateurs. Cela peut aider, par exemple, à déterminer de manière dynamique la répartition appropriée des ressources disponibles entre macro et micro-cellules. En outre, la plupart des études disponibles portent principalement sur la capacité radio théorique maximale et ne fournissent pas de résultats au niveau du trafic et des flux de données. Notre objectif consiste donc à proposer des outils et approches de dimensionnement en temps réel pour analyser (et prévoir) l'attribution des ressources entre la macro et la micro cellules selon la charge du trafic, ce qui permet d'offrir une meilleure stabilité du réseau en maximisant le débit global. Pour ce faire, nos contributions dans ce chapitre sont les suivantes :

- Proposer des chaînes de Markov pour la modélisation de eICIC/ ABS ;
- Proposer des modèles de files d'attente avec formes closes pour évaluer les performances de eICIC/ ABS ;

- Comparer les résultats numériques des modèles de files d'attente avec les résultats des chaînes de Markov où des conditions radio et des distributions d'utilisateurs réalistes ont été considérées ;
- Proposer des outils de dimensionnement qui offrent aux opérateurs les décisions sur l'attribution des ressources entre la macro et la micro cellule pour atteindre la capacité optimale du système de façon rapide et adaptative.

Nous pouvons résumer notre démarche pour les contributions sus-citées par le schéma illustré dans la FIGURE 4.1.

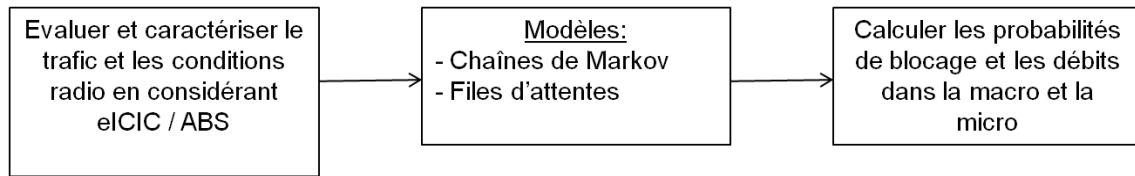


FIGURE 4.1 – Résumé de notre démarche pour la modélisation de eICIC/ ABS

Comme illustré dans la FIGURE 4.1, en entrée des chaînes de Markov et des modèles de file d'attente proposés, nous utilisons des conditions radio et des distributions d'utilisateurs réalistes. Ces paramètres sont exploités par ces approches afin de considérer les mêmes conditions radio des UE lors du calcul des probabilités de blocage et des débits dans la macro et la micro.

4.2 Formulation du problème

Nous focalisons nos études sur le downlink et considérons un H-CRAN composé d'une cellule centrale et ses 6 cellules voisines du premier tier (cellules qui sont directement autour d'elle), énumérées de 2 à 7 comme illustré dans la FIGURE 4.2.

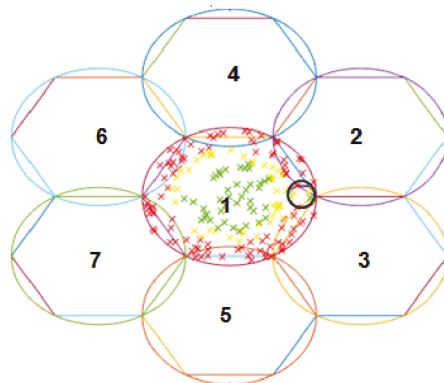


FIGURE 4.2 – H-CRAN cluster : 7 macro cells et 1 micro cell

Toutes ces cellules sont des macro cells. Dans chaque macro cell, il y a un nombre total de S ressources (ou RBG) pouvant être servis dans une trame. S est calculée à partir de l'équation (2.4). Rappelons que la valeur de S correspond au rapport entre le nombre total de PRB dans la trame (voir Tableau 2.1) et le nombre de PRB dans un RBG (voir Tableau 2.3). Comme déjà expliqué dans la section 2.3.1.2.a, ces paramètres dépendent de la bande passante considérée.

Ce cluster inclue également une micro cellule déployée dans la macro centrale. Nous effectuons nos analyses sur les UE dans la région centrale, couverte conjointement par la macro centrale et la micro qui lui est associée où le mécanisme eICIC/ ABS est appliqué. En présence de la micro cell, les S RBG sont partagés entre la macro et la micro cell selon le paramètre n_{ABS} ($0 \leq n_{ABS} \leq 10$: qui représente le nombre d'ABS alloués à la micro cell) tel que :

$$S_M = S - (S \times \frac{n_{ABS}}{10}) \quad (4.2)$$

$$S_m = S \times \frac{n_{ABS}}{10} \quad (4.3)$$

où S_M et S_m représentent respectivement le nombre de RBG attribué à la macro et à la micro. Par exemple, pour $n_{ABS} = 1$ et une largeur de bande passante de 20 MHz qui implique $S = \frac{100 \times 2 \times 10}{4} = 500$ ressources par trame (voir l'équation (2.4)) alors $S_M = 450$ et $S_m = 50$.

De plus, nous considérons les interférences de toutes les autres cellules du cluster afin de déterminer les conditions radio des UE. Les conditions radio d'un UE sont exprimées par le nombre de RBG requis pour transmettre un paquet. Nous classifions les UE selon le nombre de RBG qu'ils demandent et nous dirons que les UE de la classe i requièrent i RBG. Nous considérons le cas général où les UE sont aléatoirement distribués dans la région centrale et nous supposons que les clients arrivent selon un processus aléatoire avec un taux $\lambda^{clients} = \lambda_M^{clients} + \lambda_m^{clients}$, par durée de trame, tel que $\lambda_M^{clients}$ (resp. $\lambda_m^{clients}$) est le taux d'arrivée des clients dans la macro cell (resp. la micro cell). Nous notons $\lambda_{M,i}^{clients}$ (resp. $\lambda_{m,i}^{clients}$) les taux d'arrivée de clients dans la classe i de la macro (resp. la micro), et J_M (respectivement J_m) le nombre de classes dans la macro cellule (resp. micro cellule).

Un processus d'arrivée poissonien est l'hypothèse la plus couramment utilisée en ingénierie du trafic pour la répartition des arrivées des clients. Cette hypothèse (également présente dans le réseau téléphonique classique) est raisonnable car nous considérons un nombre important d'utilisateurs qui génèrent des requêtes et se comportent de manière indépendante.

Notre problème, visant à prévoir l'attribution des ressources entre la macro et la micro cellule et le nombre d'UE à attribuer à la micro $\lambda_m^{clients}$ afin de maximiser le débit global ($X_M + X_m$), peut être

formulé globalement comme suit :

$$\begin{aligned}
& \max_{S_m, S_M, \lambda_m^{clients}, \lambda_M^{clients}} (X_M + X_m) \\
& \text{s.t.} \quad S_m + S_M = S, \\
& \quad \lambda_M^{clients} + \lambda_m^{clients} = \lambda^{clients}, \\
& \quad S_M > 0, \\
& \quad S_m \geq 0
\end{aligned} \tag{4.4}$$

où :

- X_M est le débit de sortie moyen de la macro, exprimé en kbits/trame ;
- X_m est le débit de sortie moyen de la micro, exprimé en kbits/trame.

Ainsi, contrairement au modèle d'optimisation présenté à l'équation (4.1), ayant pour objectif de déterminer le nombre d'ABS optimal permettant de maximiser le débit des UE qui n'interfèrent pas tout en assurant un débit minimal (fixé) pour les UE qui interfèrent, notre modèle d'optimisation est plus global et vise à déterminer le nombre d'ABS optimal permettant de maximiser le débit de la macro et de la micro.

4.3 Chaînes de Markov pour la modélisation de eICIC/ ABS

Nous résumons dans le Tableau 4.1 tous les paramètres nécessaires pour la compréhension de cette section.

Dans ces modèles, chaque batch qui arrive représente le nombre de RBG nécessaires pour la transmission d'un paquet du UE (appelé aussi client) selon ses conditions radio. La FIGURE 4.3 représente un exemple d'un client avec une taille de batch égal à 2 RBG.

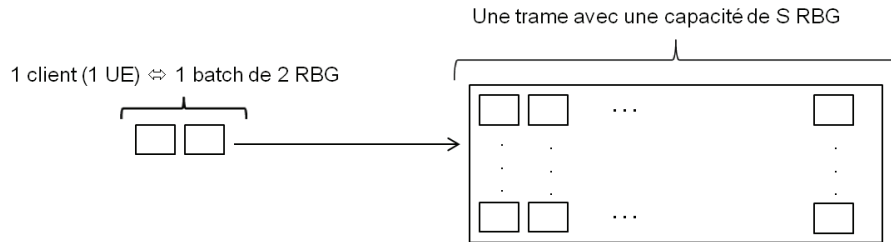


FIGURE 4.3 – Exemple d'arrivée d'un client, avec une taille de batch égal à 2 RBG, dans une trame

Nous notons $A_{M,i}(n)$ (resp. $A_{m,i}(n)$) le nombre de clients, appartenant à la classe i de la macro cell (resp. micro cell), qui arrivent à la trame n , $i \times A_{M,i}(n)$ (resp. $i \times A_{m,i}(n)$) le nombre de RBG requis à la trame n par les clients de la classe i de la macro cell (resp. micro cell). Ainsi, $\sum_{i=1}^{J_M} i \times A_{M,i}(n)$ (resp. $\sum_{i=1}^{J_m} i \times A_{m,i}(n)$) représente le nombre total de RBG requis dans la macro cell (resp. la

Paramètre	Explication
$\lambda^{clients}$	Taux d'arrivée des clients dans la région centrale (macro et micro)
$\lambda_M^{clients}$	Taux d'arrivée des clients dans la macro (toutes classes confondues)
$\lambda_m^{clients}$	Taux d'arrivée des clients dans la micro (toutes classes confondues)
$\lambda_{M,i}^{clients}$	Taux d'arrivée des clients dans la classe i de la macro
$\lambda_{m,i}^{clients}$	Taux d'arrivée des clients dans la classe i de la micro
λ	Taux d'arrivée dans la région centrale (macro et micro) exprimé en kbits/trame
λ_M	Taux d'arrivée dans la macro (toutes classes confondues) exprimé en kbits/trame
λ_m	Taux d'arrivée des clients dans la micro (toutes classes confondues) exprimé en kbits/trame
$\lambda_{M,i}$	Taux d'arrivée des clients dans la classe i de la macro exprimé en kbits/trame
$\lambda_{m,i}$	Taux d'arrivée des clients dans la classe i de la micro exprimé en kbits/trame
J_M	Nombre de classes dans la macro
J_m	Nombre de classes dans la micro
$A_{M,i}(n)$	Nombre de clients, appartenant à la classe i de la macro cell, qui arrivent à la trame n
$A_{m,i}(n)$	Nombre de clients, appartenant à la classe i de la micro cell, qui arrivent à la trame n
$i \times A_{M,i}(n)$	Nombre de RBG requis à la trame n par les clients de la classe i de la macro
$i \times A_{m,i}(n)$	Nombre de RBG requis à la trame n par les clients de la classe i de la micro
S	Capacité de service (nombre de RBG pouvant être servis dans une trame)
S_M	Capacité de service de la macro
S_m	Capacité de service de la micro
n_{ABS}	Nombre d'ABS (sous-trames) attribués à la micro
$Y^M(n)$	Nombre de RBG occupant la macro
$Y^m(n)$	Nombre de RBG occupant la micro
X_M^{RBG}	Débit de sortie moyen de la macro exprimé en RBG/trame
X_m^{RBG}	Débit de sortie moyen de la micro exprimé en RBG/trame
r_M	Taille moyenne des batchs de la macro
r_m	Taille moyenne des batchs de la micro
$g_{M,j}$	Distribution de proba des batchs de taille j dans la macro
$g_{m,j}$	Distribution de proba des batchs de taille j dans la micro
$X_M^{clients}$	Débit de sortie moyen de la macro exprimé en clients/trame
$X_m^{clients}$	Débit de sortie moyen de la micro exprimé en clients/trame
$CR_{M,i}$	Taux de codage moyen des clients de la classe i de la macro
$CR_{m,i}$	Taux de codage moyen des clients de la classe i de la micro
$N_{M,i}$	Nombre moyen des points de constellation des clients de la classe i de la macro
$N_{m,i}$	Nombre moyen des points de constellation des clients de la classe i de la micro
p_M^b	Probabilité de blocage dans la macro
p_m^b	Probabilité de blocage dans la micro
X_M^{bits}	Débit de sortie moyen de la macro exprimé en bits/trame
X_m^{bits}	Débit de sortie moyen de la micro exprimé en bits/trame
X^{bits}	Débit de sortie global exprimé en bits/trame
X_M	Débit de sortie moyen de la macro exprimé en kbits/trame
X_m	Débit de sortie moyen de la micro exprimé en kbits/trame
X	Débit de sortie global exprimé en kbits/trame

Tableau 4.1 – Paramètres des chaînes de Markov pour la modélisation de eICIC/ABS

micro cell). Dans chaque macro cell, il y a un nombre total de S RBG pouvant être servis dans une trame. S représente donc la capacité de service. Comme déjà expliqué dans la section précédente, en présence de la micro cell, ces RBG sont partagés entre la macro et la micro cell selon le paramètre n_{ABS} (voir l'équation (4.2) et l'équation (4.3)). Durant chaque trame, nous devons considérer (i) les arrivées de clients dans la macro cell, (ii) les arrivées de clients dans la micro cell et (iii) la capacité de service de la macro de S_M RBG au plus pour les clients de la macro (resp. la capacité de service de la micro de S_m RBG pour les clients de la micro cell).

La technique eICIC peut alors être modélisée par deux chaînes de Markov à temps discret, où l'unité de temps est la durée de la trame : une pour la macro et l'autre pour la micro cell. Soit $Y^M(n)$ (resp. $Y^m(n)$) représentant le nombre de RBG occupant la macro (resp. la micro) à la trame n . A la fin de chaque trame n , nous considérons :

- le service des clients arrivés à la trame $n - 1$
- et l'arrivée des clients à la trame actuelle (c'est à dire à la trame n).

Les équations d'évolution définissant les chaînes de Markov sont données comme suit pour la macro :

$$Y^M(n) = \min(S_M, [Y^M(n-1) - S_M]^+ + \sum_{i=1}^{J_M} i \times A_{M,i}(n)) \quad (4.5)$$

où $Y^M(0) = \min(S_M, \sum_{i=1}^{J_M} i \times A_{M,i}(0))$
et pour la micro :

$$Y^m(n) = \min(S_m, [Y^m(n-1) - S_m]^+ + \sum_{i=1}^{J_m} i \times A_{m,i}(n)) \quad (4.6)$$

et $Y^m(0) = \min(S_m, \sum_{i=1}^{J_m} i \times A_{m,i}(0))$.

Etant donné que les chaînes de Markov proposées considèrent des arrivées par batch et relativement un nombre important de serveurs (500 serveurs pour 20 MHz, 1000 serveurs pour 40 MHz, ...), ceci génère un nombre important de transitions. Effectivement, la complexité de la matrice en termes de nombre de transitions est exprimée comme le produit du nombre d'états, la variabilité des arrivées et le nombre de tailles des batchs, par exemple, pour une bande passante de 20 MHz où la macro est déployée seule, nous avons 500 états et pour chaque état si nous considérons $\lambda^{clients} = 100$, le nombre d'arrivées varient entre 70 et 130. En effet, mathématiquement c'est comme une loi normale : pour un niveau de confiance de 99%, l'intervalle des valeurs est $[\lambda - 3 \times \sigma, \lambda + 3 \times \sigma]$ (σ est l'écart type, $\sigma = \sqrt{\lambda}$). Ce qui fait $130 - 70 = 60$ cas d'arrivées possibles. Si nous supposons que le nombre de tailles de batchs possibles est 2. Ce qui nous fait, pour cet exemple $500 \times 60 \times 2 = 60000$ transitions. Ainsi, les chaînes de Markov que nous proposons sont difficiles à analyser numériquement. En considérant T trames, nous définissons $X_M^{RBG}(T)$ et $X_m^{RBG}(T)$, le débit moyen de sortie

pour la macro et la micro respectivement exprimés en RBG/trame, comme suit :

$$X_M^{RBG}(T) = \frac{\sum_{t=1}^T Y^M(t)}{T} \quad (4.7)$$

$$X_m^{RBG}(T) = \frac{\sum_{t=1}^T Y^m(t)}{T} \quad (4.8)$$

Le débit de sortie de la macro (resp. la micro) en termes de clients/trame représente donc le rapport entre $X_M^{RBG}(T)$ (resp. $X_m^{RBG}(T)$) et la taille moyenne des batchs dans la macro, r_M (resp. la micro, r_m). Nous choisissons évidemment une valeur de T suffisamment grande pour la convergence de $X_M^{RBG}(T)$ et $X_m^{RBG}(T)$ et nous répétons les simulations afin d'assurer un intervalle de confiance à 95%. Ainsi, le débit de sortie de la macro et de la micro en termes de clients/trame sont calculés comme suit :

$$X_M^{clients} = \lim_{T \rightarrow \infty} \frac{X_M^{RBG}(T)}{r_M} \quad (4.9)$$

$$X_m^{clients} = \lim_{T \rightarrow \infty} \frac{X_m^{RBG}(T)}{r_m} \quad (4.10)$$

où $r_M = \sum_{j=1}^{J_M} j \times g_{M,j}$ avec $g_{M,j} = \frac{\lambda_{M,j}^{clients}}{\lambda_M^{clients}}$ (resp. $r_m = \sum_{j=1}^{J_m} j \times g_{m,j}$ avec $g_{m,j} = \frac{\lambda_{m,j}^{clients}}{\lambda_m^{clients}}$) est la taille moyenne des batchs dans la macro (resp. la micro). Les distributions des tailles de batchs vérifient $\sum_{i=1}^{J_M} g_{M,i} = \sum_{i=1}^{J_m} g_{m,i} = 1$.

Afin de déduire le débit de sortie binaire, nous devons considérer la distribution des tailles des batchs et les paramètres radio (c'est à dire le nombre de PRB dans un RBG, le nombre d'OFDM symboles par PRB, le nombre de sous-porteuses par PRB, le taux de codage moyen par classe, le nombre moyen de points de constellation par classe), nous le calculons comme suit :

$$X_M^{bits} = X_M^{clients} \times \left(\sum_{i=1}^{J_M} i \times g_{M,i} \times N_{PRB} \times 7 \times 12 \times CR_{M,i} \times \log_2(N_{M,i}) \right) \quad (4.11)$$

$$X_m^{bits} = X_m^{clients} \times \left(\sum_{i=1}^{J_m} i \times g_{m,i} \times N_{PRB} \times 7 \times 12 \times CR_{m,i} \times \log_2(N_{m,i}) \right) \quad (4.12)$$

où $CR_{M,i}$ et $N_{M,i}$ (resp. $CR_{m,i}$ et $N_{m,i}$) représentent le taux de codage moyen et le nombre moyen de points de constellation dans la classe i de la macro (resp. de la micro).

Le débit de sortie global en termes de kbits/trame est $X = \frac{X_M^{bits} + X_m^{bits}}{1024}$.

Quant à la probabilité de blocage pour la macro, elle peut être calculée comme suit :

$$p_M^b = \frac{\lambda_M^{clients} - X_M^{clients}}{\lambda_M^{clients}} \quad (4.13)$$

et celle pour la micro :

$$p_m^b = \frac{\lambda_m^{clients} - X_m^{clients}}{\lambda_m^{clients}} \quad (4.14)$$

Les probabilités de blocage de la macro (resp. micro) sont aussi calculées pour chaque classe i en multipliant les débits par la probabilité $g_{M,i}$ (resp. $g_{m,i}$).

La résolution numérique et la simulation en Matlab de la chaîne de Markov (macro ou micro) prennent un certain temps d'exécution. Nous avons observé en ce qui concerne la simulation de la chaîne de Markov, et pour un taux d'arrivée de clients $\lambda^{clients}$ compris entre 50 clients/trame et 450 clients/trame, un temps de simulation compris entre 3 minutes et 25 minutes. Par conséquent, les simulations des chaînes de Markov ne peuvent pas être utilisées comme outils de configuration en temps réel pour la technique eICIC/ ABS. Dès lors, nous explorons dans ce qui suit des modèles de files d'attente qui permettraient de remédier à ces faiblesses.

4.4 Modèles de file d'attente pour la modélisation de eICIC/ ABS

Dans cette section, nous identifions les modèles de files d'attente qui pourraient refléter le mécanisme eICIC/ ABS et qui donnent de préférence des formes close plus rapides à exploiter. Résumons les spécificités de nos chaînes de Markov :

- Chaînes de Markov à temps discret où l'unité de temps est la trame ;
- Les clients arrivent indépendamment selon un processus de Poisson avec un taux $\lambda^{clients}$, par trame
 ⇒ Arrivées poissonniennes ;
- Chaque client qui arrive requiert, selon ses conditions radio et la taille de son paquet, un certain nombre de RBG, le client est donc vu comme un batch (composé de RBG)
 ⇒ Arrivées par batch ;
- Nous avons S RBG pouvant être servis dans une trame, ce qui correspond à une capacité de service S par trame
 ⇒ S serveurs ;
- Chaque client va occuper simultanément, par trame, autant de RBG qu'il demande (un nombre déterministe mais qui change d'un client à un autre). De plus, nous supposons une acceptation partielle dans les chaîne de Markov tel que lorsqu'un client demande un nombre de RBG supérieur au nombre de RBG disponibles, une partie de ce client sera accepté et le reste sera rejeté
 ⇒ Loi de service générale et absence de files d'attente ;
- Les clients sont servis selon leur ordre d'arrivée
 ⇒ Discipline de service FCFS.

Nous souhaitons modéliser le système en utilisant des files d'attente en considérant autant que possible ces spécificités. Par ailleurs, en nous basant sur l'analyse ci-dessus des chaînes de Markov, nous nous intéressons donc aux modèles de files d'attente multi-serveurs avec pertes. Pour cette catégorie, nous distinguons intuitivement deux files d'attente : la $M/G/S/S$ et la $M^{[X]}/G/S/S$. La $M/G/S/S$ considère des arrivées exponentielles et la $M^{[X]}/G/S/S$ des arrivées exponentielles par batch (voir chapitre 3 pour plus de détails sur ces deux modèles).

- Réflexion sur la $M/G/S/S$

Le point faible d'une modélisation avec la $M/G/S/S$ pourrait être le fait de ne pas considérer des arrivées par batch. Ceci peut donc faire de la $M/G/S/S$ une mauvaise approximation. Toutefois, le point fort de ce modèle c'est sa forme close simple (voir l'équation (3.10)). En effet, son implémentation permet de calculer la probabilité de blocage et donc le débit correspondant en temps réel. Elle est largement utilisée par les réseaux téléphoniques et cellulaires de type GSM et LTE [DMV05] [SGM12]. Donc si l'erreur relative n'est pas très importante, la $M/G/S/S$ pourrait être utilisée comme outil de dimensionnement très performant pour l'allocation de ressources lorsque eICIC/ ABS est appliqué.

- Réflexion sur la $M^{[X]}/G/S/S$

La $M^{[X]}/G/S/S$ considère toutes les caractéristiques de la chaîne de Markov (celle de la macro et celle de la micro) sauf que nous avons utilisé la $M^{[X]}/M/S/S$ qui est à temps continu et qui a une forme close dans ce cas. Toutefois, la complexité de sa forme close (voir l'équation (3.14)) dépend du nombre de classes générées. Effectivement, sa complexité est d'ordre de $\theta(n^{J_M})$ pour la macro et d'ordre $\theta(n^{J_m})$ pour la micro.

C'est pourquoi nous avons défini comme objectif d'explorer ces deux modèles de files d'attente pour la modélisation de eICIC/ ABS.

4.4.1 Modèle de file d'attente : $M/G/S/S$ pour la modélisation de eICIC/ ABS

Nous considérons la file $M/G/S/S$ avec des arrivées exponentielles de paramètre $\lambda^{clients} = \lambda_M^{clients} + \lambda_m^{clients}$, et un temps de service déterministe. Dans ce modèle, chaque client occupe un serveur. Nous supposons que l'unité de temps est la trame. Le nombre de RBG pouvant être servis dans une trame est représenté par S serveurs qui seront partagés entre la macro et la micro (lorsque eICIC/ ABS est appliqué) comme illustré dans la FIGURE 4.4.

Ainsi, nous avons deux $M/G/S/S$, une pour la macro avec S_M serveurs et l'autre pour la micro avec S_m serveurs. Par ailleurs, afin de compenser l'effet des tailles variables des clients tout en utilisant un temps de service déterministe, nous proposons de déterminer une valeur moyenne pondérée du temps de service, en suivant la répartition des tailles des batchs dans la serving cell (macro ou micro). Par conséquent, si dans le mécanisme eICIC/ ABS, un client occupe j ressources (ou RBG) pendant une trame, dans le modèle de file d'attente $M/G/S/S$ proposé, le même client occupera un RBG pendant j trames consécutives. Ainsi, le temps de service de la macro est $\frac{1}{\mu_M} = \sum_{i=1}^{J_M} i \times g_{M,i}$,

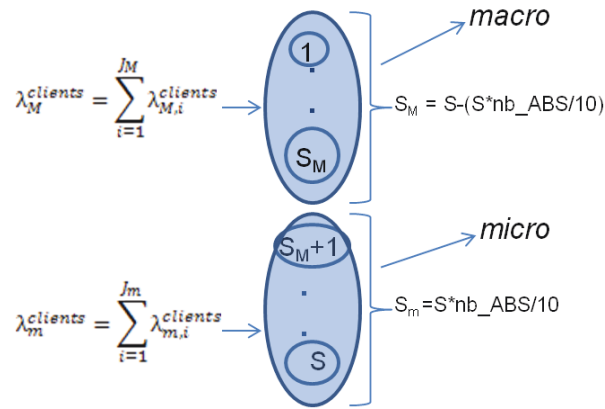


FIGURE 4.4 – Arrivée des clients et partage de ressources entre la macro et la micro

et de la micro $\frac{1}{\mu_m} = \sum_{i=1}^{J_m} i \times g_{m,i}$.

La FIGURE 4.5 illustre un exemple où nous supposons qu'un client de classe 2 arrive dans la macro. Nous supposons également qu'il y a assez de serveurs disponibles.

Dans la chaîne de Markov de la macro, ce client arrivant à la trame t sera servi à la trame suivante

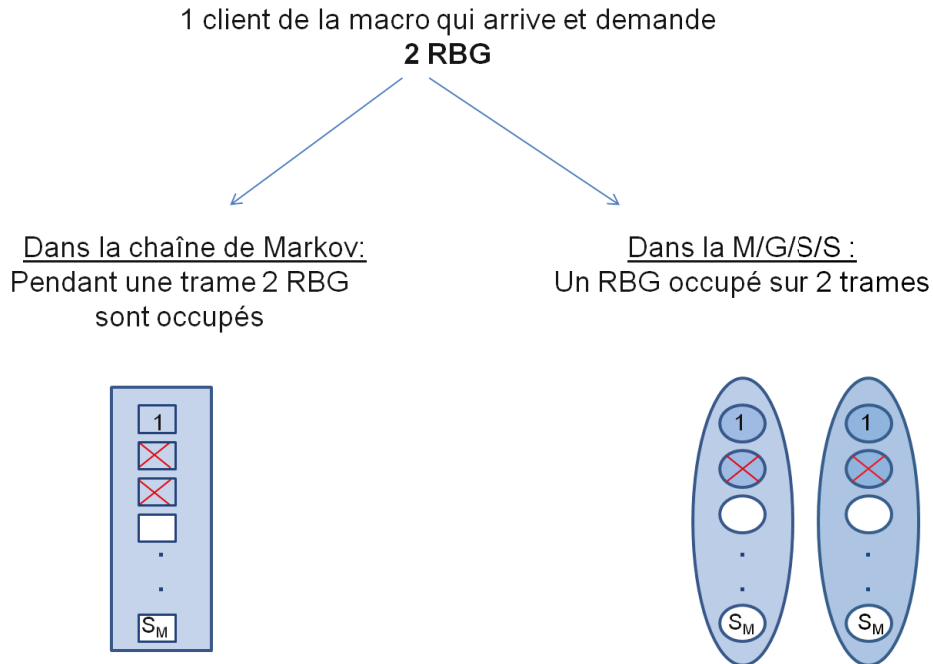


FIGURE 4.5 – Exemple de service d'un client dans la chaîne de Markov et dans la $M/G/S/S$

(la trame $t + 1$) par contre dans la $M/G/S/S$ (qui est à temps continu) il rentre en service dès son arrivée et occupe le serveur deux fois plus longtemps.

La file $M/G/S/S$ permet d'utiliser la formule d'Erlang B (voir le chapitre [3](#)) afin de déterminer la probabilité de blocage des clients dans la serving cell puis de calculer le débit dans chaque cell (macro et micro) ainsi que le débit total dans la région centrale du cluster considérée (macro et micro). Soit p_M^b (resp. p_m^b) la probabilité de blocage dans la macro cell (resp. la micro cell) exprimée comme suit en utilisant la formule d'Erlang B :

$$p_M^b = \frac{\frac{\rho_M^{S_M}}{S_M!}}{\sum_{n=0}^{S_M} \frac{\rho_M^n}{n!}} \quad (4.15)$$

où $\rho_M = \frac{\lambda_M^{clients}}{\mu_M}$, $\frac{1}{\mu_M} = \sum_{i=1}^{J_M} i \times g_{M,i}$

$$p_m^b = \frac{\frac{\rho_m^{S_m}}{S_m!}}{\sum_{n=0}^{S_m} \frac{\rho_m^n}{n!}} \quad (4.16)$$

où $\rho_m = \frac{\lambda_m^{clients}}{\mu_m}$, $\frac{1}{\mu_m} = \sum_{i=1}^{J_m} i \times g_{m,i}$

Le débit de sortie en termes de clients/ trame, $X_M^{clients}$, pour la macro cell est calculé par :

$$X_M^{clients} = \lambda_M^{clients} \times (1 - p_M^b) \quad (4.17)$$

et celui pour la micro cell est calculé par :

$$X_m^{clients} = \lambda_m^{clients} \times (1 - p_m^b) \quad (4.18)$$

Les débits correspondants en termes de bits/trame sont calculés respectivement par :

$$X_{M,j}^{bits} = X_M^{clients} \times j \times g_{M,j} \times N_{PRB} \times 7 \times 12 \times CR_{M,j} \times \log_2(N_{M,j}) \quad (4.19)$$

et donc :

$$X_M^{bits} = \sum_{j=1}^{J_M} X_{M,j}^{bits} \quad (4.20)$$

et pour la micro :

$$X_{m,j}^{bits} = X_m^{clients} \times j \times g_{m,j} \times N_{PRB} \times 7 \times 12 \times CR_{m,j} \times \log_2(N_{m,j}) \quad (4.21)$$

ce qui fait :

$$X_m^{bits} = \sum_{j=1}^{J_m} X_{m,j}^{bits} \quad (4.22)$$

Le débit de sortie global en termes de kbits/trame est :

$$X = \frac{X_M^{bits} + X_m^{bits}}{1024} \quad (4.23)$$

4.4.2 Modèle de file d'attente : $M^{[X]}/G/S/S$ pour la modélisation de eICIC/ABS

Nous proposons une autre approche d'approximation, la file $M^{[X]}/G/S/S$, qui possède une forme close en tenant compte des arrivées par batch et des pertes avec plusieurs classes de services. Le raisonnement, expliqué dans la section 3.4.2, est appliqué pour la macro et la micro cell. Toutefois, pour utiliser la même terminologie que celle de la section précédente, nous considérons que chaque batch de RBG représente un client qui arrive et nous dirons qu'un client est de classe i si la taille de son batch est égale à i , et nous appellerons les composants d'un batch d'un client (UE) par : unités du batch.

Nous distinguons deux files $M^{[X]}/G/S/S$, une pour la macro et l'autre pour la micro. Pour la macro cell (resp. la micro cell), le service est assuré par S_M (resp. S_m) serveurs identiques, chaque serveur pouvant servir une unité du batch à la fois. Le système fonctionne comme suit : à chaque fois qu'un batch de taille j trouve à son arrivée n serveurs ($n=0, 1, \dots, S_M$) occupés, un nombre $\min(j, S_M-n)$ (resp. $\min(j, S_m-n)$) d'unités de batch, choisis au hasard, occupent les serveurs inactifs ainsi ils rentrent en service simultanément. Les unités de batch restantes, s'il y en a, sont rejetées par le système (bloquées) et ne reviennent pas plus tard. Par conséquent, aucune file d'attente n'est autorisée à être formée et les unités de batch bloquées sont considérées comme perdues pour le système. La durée de service de chaque RBG correspond à la durée de la trame.

La FIGURE 4.6 illustre un exemple d'un client ayant une taille de batch égal 2 qui arrive dans la macro. Nous supposons qu'il y a assez de serveurs disponibles.

Dans la chaîne de Markov ce client arrivant à la trame t est accepté dans cette trame (étant donné qu'il y a assez de ressources disponibles) mais il ne sera servi qu'à la trame $t+1$. Toutefois, dans la $M^{[X]}/G/S/S$, étant donné que ce modèle est à temps continu, aussitôt arrivé il est servi en occupant 2 serveurs simultanément s'ils sont disponibles (car nous avons considéré la $M^{[X]}/M/S/S$ et sa forme close).

L'équation (3.14) est utilisée, pour calculer la probabilité de blocage dans la macro p_M^b , avec les hypothèses suivantes : $S = S_M$, $\rho = \rho_M = \frac{\lambda_M^{clients}}{\mu_M}$, $\lambda^{clients} = \lambda_M^{clients}$, $g_j = g_{M,j} = \frac{\lambda_{M,j}^{clients}}{\lambda_M^{clients}}$ et $\mu = \mu_M = 1$ (correspondant à un RBG par trame). Le même raisonnement est effectué sur la micro

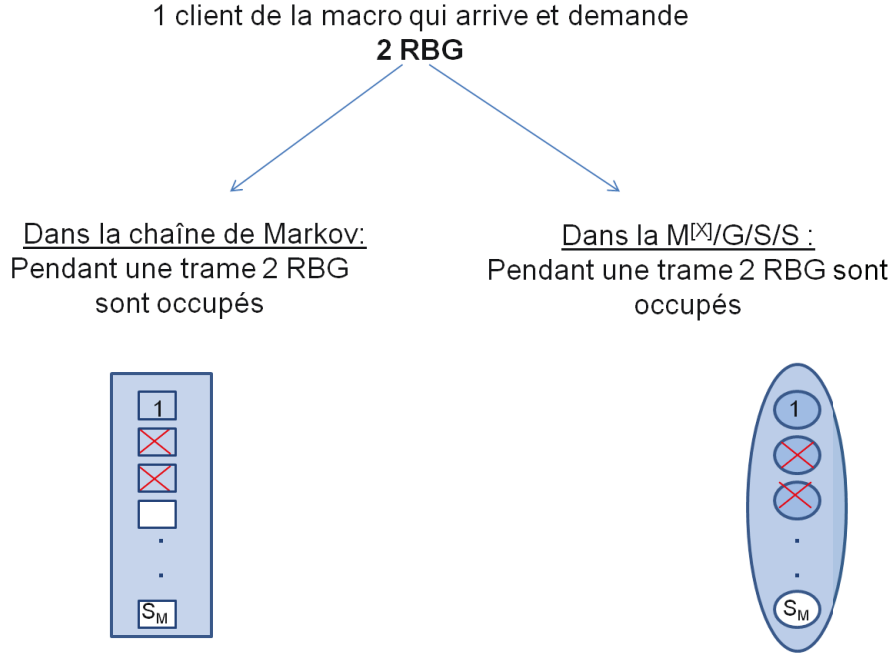


FIGURE 4.6 – Exemple de service d'un client dans la chaîne de Markov et dans la $M^{[X]}/G/S/S$

cell afin de déterminer la probabilité de blocage dans la micro, p_m^b , en considérant dans l'équation (3.14) : $S = S_m$, $\rho = \rho_m = \frac{\lambda_m^{clients}}{\mu_m}$, $\lambda^{clients} = \lambda_m^{clients}$, $g_j = g_{m,j} = \frac{\lambda_{m,j}^{clients}}{\lambda_m^{clients}}$ et $\mu = \mu_m = 1$.

En nous basant sur l'équation (3.14) et sur ce paramétrage de la macro et de la micro, le débit de sortie en termes de RBG/ trame, $X_{M,j}^{RBG}$, pour la classe j de la macro cell est calculé par :

$$\begin{cases} X_{M,1}^{RBG} = \lambda_{M,1}^{clients} \times (1 - p_M^b) \\ X_{M,j}^{RBG} = \lambda_{M,j}^{clients} \times \left(\sum_{l=0}^{S_M-j} j \times p_l + \sum_{l=S_M-j+1}^{S_M-1} (S_M - l) \times p_l \right); \text{ pour } 1 < j \leq J_M \end{cases} \quad (4.24)$$

et celui pour la micro cell est calculé par :

$$\begin{cases} X_{m,1}^{RBG} = \lambda_{m,1}^{clients} \times (1 - p_m^b) \\ X_{m,j}^{RBG} = \lambda_{m,j}^{clients} \times \left(\sum_{l=0}^{S_m-j} j \times p_l + \sum_{l=S_m-j+1}^{S_m-1} (S_m - l) \times p_l \right); \text{ pour } 1 < j \leq J_m \end{cases} \quad (4.25)$$

Nous déduisons ainsi les débits correspondants en termes de clients/ trame, $X_{M,j}^{clients}$ pour la macro et $X_{m,j}^{clients}$ pour la micro, en utilisant l'équation (4.26) et l'équation (4.27), respectivement.

$$X_{M,j}^{clients} = \frac{X_{M,j}^{RBG}}{j}; \text{ pour } 1 \leq j \leq J_M \quad (4.26)$$

$$X_{m,j}^{clients} = \frac{X_{m,j}^{RBG}}{j}; \text{ pour } 1 \leq j \leq J_m \quad (4.27)$$

Les débits correspondants en termes de bits/ trame sont calculés respectivement par :

$$X_{M,j}^{bits} = X_{M,j}^{RBG} \times N_{PRB} \times 7 \times 12 \times CR_{M,j} \times \log_2(N_{M,j}); \text{ pour } 1 \leq j \leq J_M \quad (4.28)$$

$$X_{m,j}^{bits} = X_{m,j}^{RBG} \times N_{PRB} \times 7 \times 12 \times CR_{m,j} \times \log_2(N_{m,j}); \text{ pour } 1 \leq j \leq J_m \quad (4.29)$$

Les débits de sortie binaires dans la macro X_M^{bits} , la micro X_m^{bits} et le débit global en termes de kbits/trame X sont calculés respectivement par l'équation (4.20), l'équation (4.22) et l'équation (4.23).

4.5 Analyse des performances

Dans cette section nous analysons les performances des trois approches proposées. Nous décrirons dans la sous-section 4.5.1 l'algorithme de génération et de calcul des conditions radio des UE pour eICIC/ ABS. La sous-section 4.5.2 (resp. la sous-section 4.5.3) présentera une étude comparative entre les performances de la $M/G/S/S$ (resp. $M^{[X]}/G/S/S$) et celles des chaînes de Markov simulées sous deux scénarios : macro seule et macro déployée avec une micro. Ensuite, dans la sous-section 4.5.4 nous comparons la précision des 3 approches proposées (chaînes de Markov, $M^{[X]}/G/S/S$ et $M/G/S/S$). Enfin, dans la sous-section 4.5.5, nous expliquerons l'influence du déploiement de la micro et du nombre d'ABS à lui attribuer.

Les résultats obtenus sont représentés, à la fois par des courbes et des tableaux. L'intérêt des tableaux (mis en Annexe) est d'apporter des informations supplémentaires sur la précision des modèles proposés et sur les probabilités de blocage.

4.5.1 Algorithme de génération et de calcul des conditions radio des UE pour eICIC/ ABS

Afin d'évaluer et de caractériser le trafic et les conditions radio des UEs (voir FIGURE 4.1), nous nous basons sur les explications et les hypothèses énumérées dans la section 2.3.1.2.a (au niveau du paragraphe "adaptation du lien") pour proposer l'algorithme 1. Cet algorithme explique comment nous calculons, pour chaque UE noté "u", avec une taille de paquet (size(u)) en bits, le nombre de RBG qu'il requiert selon ses conditions radio lorsque eICIC/ ABS est appliqué.

Algorithme 1 Génération des utilisateurs et détermination du nombre de RBG requis par chaque UE dans le cas de eICIC/ ABS

Inputs : une macro cell où une micro cell est déployée, le taux d'arrivée des clients (noté $\lambda^{clients}$), la taille de paquet de chaque UE ($size(u)$)

Résultats : nombre de RBGs(u) pour chaque UE " u ", nombre de classes, la distribution des probabilités des tailles des batchs, les conditions radio par classe (dans la macro et la micro)

- 1: Génération de Nb UE aléatoirement et uniformément dans la macro cell centrale selon une loi de poisson de paramètre $\lambda^{clients}$
 - 2: Pour $u=1$ à Nb
 - 2.1. Déterminer la serving cell (celle qui offre la puissance de réception maximale) :
$$PRx_M(u) = Ptx_M + G - PL_M(u)$$
$$PRx_m(u) = Ptx_m + G - PL_m(u)$$
$$PRx_s(u) = \max(PRx_M(u), PRx_m(u))$$
 - 2.2. Déterminer le SINR(u) avec eICIC/ABS :
$$SINR(u) = 10 \log_{10} \left(\frac{PRx_s(u)}{\sum_{i=2}^7 PRx_i(u) + No} \right)$$
 - 2.3. Dédire la valeur de $Bits_{MCS}(u)$ correspondant au SINR(u) (à partir du Tableau [2.2](#))
 - 2.4. Déterminer le nombre d'OFDM symboles requis :
$$N_{Symb}(u) = \frac{size(u)}{Bits_{MCS}(u)}$$
 - 2.5. Déterminer le nombre PRB requis :
$$N_{Subch}(u) = \frac{N_{Symb}(u)}{7 \times 12}$$
 - 2.6. Déterminer le nombre RBG requis correspondant :
$$N_{RBGs}(u) = \frac{N_{Subch}(u)}{N_{PRB}}$$
 - 3: Fin
 - 4: Calcul du nombre de classes dans la macro et dans la micro
 - 5: Calcul du taux d'arrivées par classe (macro et micro)
 - 6: Calcul de la distribution des probabilités des tailles des batchs (macro et micro)
 - 7: Calcul du taux de codage moyen et le nombre de points de constellation moyen par classe (macro et micro)
-

Les paramètres utilisés dans cet algorithme sont expliqués dans le Tableau [4.2](#)

L'algorithme génère d'abord des UE de façon aléatoire et uniforme [SBS+16](#) dans la région centrale (macro et micro) selon une loi de poisson de paramètre $\lambda^{clients}$, représentant le taux d'arrivée des clients dans la trame. Ensuite, l'algorithme calcule le nombre de RBG requis par chaque UE u .

Il est à noter que, comme chaque UE u génère une requête de taille $size(u)$, $\lambda^{clients}$ représente, donc, également le taux d'arrivée des requêtes par trame.

M, m, s, i	macro, micro, serving cell et cellules interférantes
$PRx_M(u)$	puissance reçue de la macro (in watt) par le UE "u"
Ptx_M	puissance transmise par la macro (in watt)
G	gain d'antenne par Hz
N_0	bruit thermique
PL	modèle de pathloss (en decibels) : Acc17 $PL_M(u) = 128.1 + 37.6 \times \log_{10}(R)$ où R est en kilomètres (cas macro) $PL_m(u) = 38 + 30 \times \log_{10}(R)$ où R est en mètres (cas micro)
N_{PRB}	nombre de PRB au sein d'un RBG

Tableau 4.2 – Paramètres pour le calcul du nombre de RBG requis par chaque UE

Application numérique de l'algorithme 1 : Déroulons cet algorithme en considérant une bande passante de 20 MHz ($\Rightarrow N_{PRB}=4$) afin de déterminer le nombre de RBG requis par 3 UE : u_1 , u_2 et u_3 ayant respectivement les tailles suivantes : $size(u_1) = 1024$ bits, $size(u_2) = 2048$ bits et $size(u_3) = 3200$ bits. Ne pouvant pas connaître le MCS de ces UE sans simulations, nous déroulons sur les différentes valeurs $Bits_{MCS}$ possibles (2, 4 ou 6 voir le Tableau 2.2).

Pour u_1 tel que $size(u_1) = 1024$ bits :

- si $Bits_{MCS}(u_1) = 2$ donc $N_{Symb}(u_1) = \frac{1024}{2} = 512$ et par conséquent $N_{Subch}(u_1) = \frac{512}{7 \times 12} = 6$ ce qui fait que $N_{RBG}(u_1) = \frac{6}{4} = 2$
- si $Bits_{MCS}(u_1) = 4$ donc $N_{Symb}(u_1) = \frac{1024}{4} = 256$ et par conséquent $N_{Subch}(u_1) = \frac{256}{7 \times 12} = 3$ ce qui fait que $N_{RBG}(u_1) = \frac{3}{4} = 1$
- si $Bits_{MCS}(u_1) = 6$ donc $N_{Symb}(u_1) = \frac{1024}{6} = 171$ et par conséquent $N_{Subch}(u_1) = \frac{171}{7 \times 12} = 2$ ce qui fait que $N_{RBG}(u_1) = \frac{2}{4} = 1$

Ainsi, pour un UE u avec une taille de paquet $size(u) = 1024$ bits, le nombre de RBG requis $N_{RBG}(u)$ peut être égal à 1 ou à 2.

Pour u_2 tel que $size(u_2) = 2048$ bits :

- si $Bits_{MCS}(u_2) = 2$ donc $N_{Symb}(u_2) = \frac{2048}{2} = 1024$ et par conséquent $N_{Subch}(u_2) = \frac{1024}{7 \times 12} = 13$ ce qui fait que $N_{RBG}(u_2) = \frac{13}{4} = 4$
- si $Bits_{MCS}(u_2) = 4$ donc $N_{Symb}(u_2) = \frac{2048}{4} = 512$ et par conséquent $N_{Subch}(u_2) = \frac{512}{7 \times 12} = 7$ ce qui fait que $N_{RBG}(u_2) = \frac{7}{4} = 2$
- si $Bits_{MCS}(u_2) = 6$ donc $N_{Symb}(u_2) = \frac{2048}{6} = 342$ et par conséquent $N_{Subch}(u_2) = \frac{342}{7 \times 12} = 5$ ce qui fait que $N_{RBG}(u_2) = \frac{5}{4} = 2$

Ainsi, pour un UE u avec une taille de paquet $size(u) = 2048$ bits, le nombre de RBG requis $N_{RBG}(u)$ peut être égal à 2 ou à 4.

Pour u_3 tel que $size(u_3) = 3200$ bits :

- si $Bits_{MCS}(u_3) = 2$ donc $N_{Symb}(u_3) = \frac{3200}{2} = 1600$ et par conséquent $N_{Subch}(u_3) = \frac{1600}{7 \times 12} = 20$ ce qui fait que $N_{RBG}(u_3) = \frac{20}{4} = 5$
- si $Bits_{MCS}(u_3) = 4$ donc $N_{Symb}(u_3) = \frac{3200}{4} = 800$ et par conséquent $N_{Subch}(u_3) = \frac{800}{7 \times 12} = 10$ ce qui fait que $N_{RBG}(u_3) = \frac{10}{4} = 3$
- si $Bits_{MCS}(u_3) = 6$ donc $N_{Symb}(u_3) = \frac{3200}{6} = 534$ et par conséquent $N_{Subch}(u_3) = \frac{534}{7 \times 12} = 7$ ce qui fait que $N_{RBG}(u_3) = \frac{7}{4} = 2$

Ainsi, pour un UE u avec une taille de paquet $size(u) = 3200$ bits, le nombre de RBG requis $N_{RBG}(u) = 2, 3$ ou 5 .

Par conséquent, les tailles des batches diffèrent selon la taille des paquets des utilisateurs ($size(u)$) et leurs conditions radio ($Bits_{MCS}(u)$).

Remarque : Comme le lecteur peut le remarquer, nous effectuons des arrondis à l'entier supérieur pour déterminer le nombre de RBG.

Une fois que le nombre de RBG requis par chaque UE est calculé, l'algorithme regroupe les UE demandant le même nombre de RBG dans la même classe (nous notons que le nombre de RBG requis correspond à l'indice de la classe) et déduit les taux d'arrivée de clients dans chaque classe, notés $\lambda_{M,i}^{clients}$ (resp. $\lambda_{m,i}^{clients}$) pour la classe i de la macro (resp. la micro), et la distribution de la probabilité des batches de taille i , noté $g_{M,i}$ (resp. $g_{m,i}$) pour la macro (resp. la micro).

Rappelons que J_M (respectivement J_m) est le nombre de classes dans la macro cellule (resp. micro cellule). Ainsi, les taux d'arrivée totales dans la macro cell et dans la micro cell respectivement sont calculés par l'algorithme [1](#) comme suit :

$$\lambda_M^{clients} = \sum_{i=1}^{J_M} \lambda_{M,i}^{clients} \quad \text{et} \quad \lambda_m = \sum_{i=1}^{J_m} \lambda_{m,i} \quad (4.30)$$

$$\lambda_m^{clients} = \sum_{i=1}^{J_m} \lambda_{m,i}^{clients} \quad \text{et} \quad \lambda_M = \sum_{i=1}^{J_M} \lambda_{M,i} \quad (4.31)$$

où $\lambda_M^{clients}$ et $\lambda_m^{clients}$ sont exprimés en termes de clients/trame et λ_M et λ_m sont exprimés en termes de kbits/trame.

De plus, connaissant le SINR de chaque UE, l'algorithme déduit en se basant sur le Tableau [2.2](#) le taux de codage et le nombre de points de constellation pour chaque UE puis calcule le taux de codage moyen et le nombre de points de constellation moyen par classe. Nous répétons évidemment un nombre de simulations suffisamment grand pour assurer un intervalle de confiance à 95%.

Ces paramètres (le taux d'arrivée par classe, la distribution des probabilités des tailles des batches, le taux de codage moyen par classe et le nombre de points de constellation moyen par classe) sont ensuite exploités par les modèles que nous proposons ie. les chaînes de Markov et les files d'attente afin de considérer les mêmes conditions radio des UE lors du calcul des probabilités de blocage et des débits dans la macro et la micro. Toute cette démarche est détaillée dans la FIGURE [4.7](#)

L'intérêt de notre approche est de simuler un environnement réaliste en utilisant un algorithme simple et performant, visant à générer des utilisateurs selon des conditions radio réelles et nous

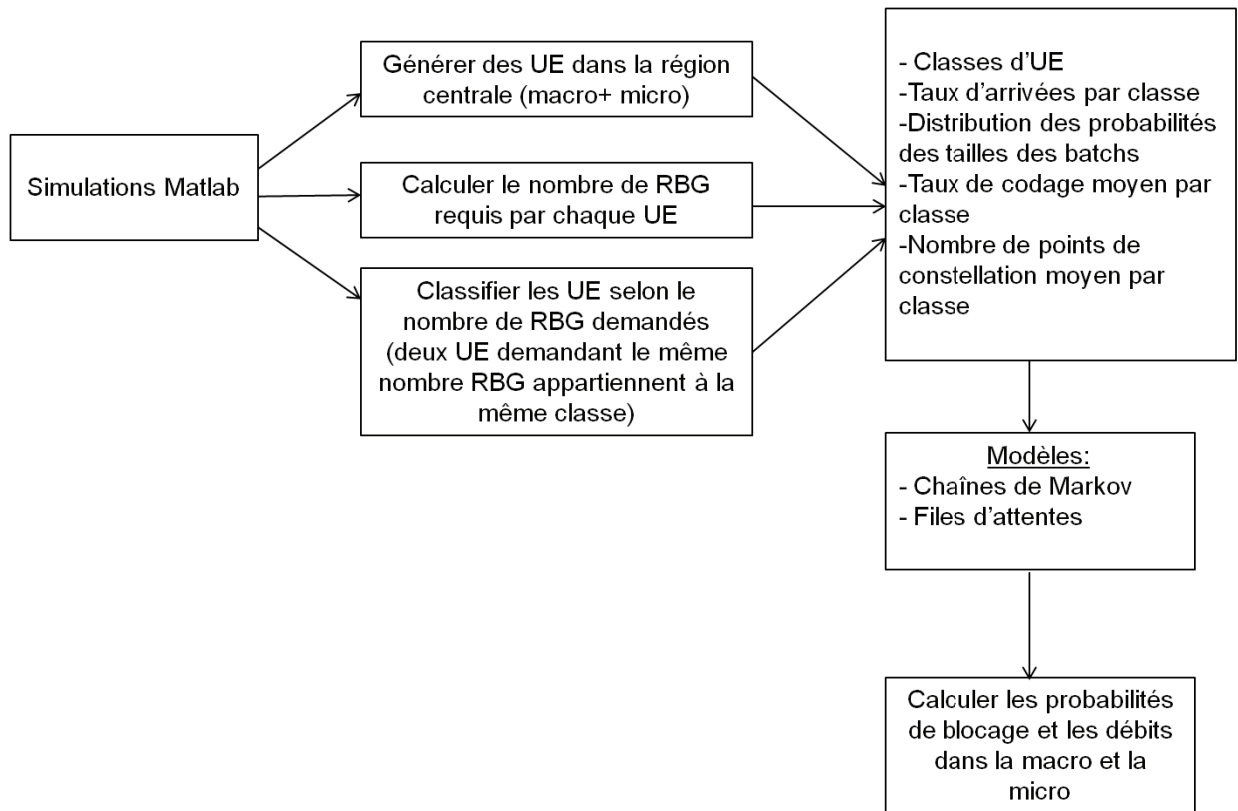


FIGURE 4.7 – Explication détaillée sur notre démarche pour la modélisation de eICIC/ ABS

permettant de calculer des paramètres précis avec les chaînes de Markov et les modèles de files d'attente proposés.

4.5.2 Etude comparative entre les performances de la M/G/S/S et celles des chaînes de Markov simulées

A présent, nous présentons et analysons les résultats obtenus par la $M/G/S/S$ et par les simulations de la chaîne de Markov proposée (de la macro et de la micro). Nous nous intéressons aux débits de sortie de la macro et de la micro cell. Plusieurs simulations ont été effectuées afin d'assurer un intervalle de confiance à 95%. Les paramètres utilisés pour ces évaluations sont listés dans le Tableau 4.3 [Acc11].

Les distributions des UE et les conditions radio générées par les simulations Matlab sont utilisées

Paramètre	Valeur
Fréquence porteuse	2 GHz
Bande passante	20 MHz
Rayon de la macro cell	3500 m
Rayon de la micro cell	600 m
Puissance de transmission de la macro cell	40 dBm
Puissance de transmission de la micro cell	30 dBm
Gain d'antenne de la macro cell	18 dBi
Gain d'antenne de la micro cell	6 dBi
Taille du client u	$size(u)$
Nombre d'ABS	n_{ABS}
S	500 RBGs/trame
N_{PRB}	4 PRB
Distribution de la localisation des UE	Uniforme
Nombre d'UE actifs par trame	Poisson

Tableau 4.3 – Paramètres de simulation

pour extraire les paramètres d'entrée du modèle d'approximation ($M/G/S/S$). Celles-ci incluent les classes d'utilisateurs, les taux d'arrivée par classe d'utilisateurs et la taille moyenne des batchs par classe. Cela garantit que des scénarios similaires sont simulés et résolus par les deux approches (à savoir les simulations Matlab de la chaîne de Markov et la résolution de la $M/G/S/S$). Rappelons que dans le trafic sans fil, la taille des paquets commence à partir de 112 bits (par exemple, ACK, etc.) sans charge utile de données, jusqu'à 12240 bits, ce qui correspond à la limite de fragmentation [Adi10]. Il est clair que le fait de changer la taille des paquets des clients va influencer le nombre de RBG demandés (voir algorithme 1) et par conséquent sur les classes des clients. Nous allons donc considérer différentes tailles de paquets afin de valider nos études sous différentes conditions radio. Ces tailles de paquets peuvent correspondre à n'importe quel type de trafic (voix, vidéo,

vidéo streaming, ...) selon la qualité demandée.

La FIGURE 4.2 illustre un échantillon de génération d'utilisateurs (ou clients). Les clients en rouge, jaune et vert représentent respectivement les clients avec de mauvaises ($Bits_{MCS} = 2$), bonnes ($Bits_{MCS} = 4$) et très bonnes conditions radio ($Bits_{MCS} = 6$).

Nous avons étudié différents scénarios (avec et sans micro pour trois tailles de paquets distinctes : $size(u) = 1024$ bits, $size(u) = 2048$ bits et $size(u) = 3200$ bits) afin de valider nos modèles de file d'attente pour la modélisation de la technique de remédiation aux interférences eICIC/ ABS.

De plus, afin de comparer les performances sous différentes charges (faible, moyenne et forte charge), nous avons fait varier $\lambda^{clients}$ de 50 clients/trame à 450 clients/trame avec un pas de 50. Nous avons donc considéré $\lambda^{clients} = 50, 100, 150, \dots, 450$ clients/trame pour les trois tailles de paquets considérées ($size(u) = 1024, 2048$ et 3200 bits). Par ailleurs, comme nous nous intéressons aux débits de sortie exprimés en kbits/trame (considérant les conditions radio des utilisateurs), nous exprimons également (dans les figures/ tableaux) les taux d'arrivée (ou débits d'entrée) en kbits/trame. Ainsi, pour convertir le débit d'entrée des clients $\lambda^{clients}$ en débit d'entrée λ exprimé en kbits/trame, nous basons sur les deux équations suivantes :

$$\lambda_j = \frac{\lambda_j^{bits}}{1024} = \frac{\lambda^{clients} \times j \times g_j \times N_{PRB} \times 7 \times 12 \times CR_j \times \log_2(N_j)}{1024} \quad (4.32)$$

et donc :

$$\lambda = \sum_{j=1}^J \lambda_j \quad (4.33)$$

Etant donné que pour chaque $size(u)$ nous avons un nombre de RBG requis différent et des distributions par classe différentes, et d'après l'équation (4.32), pour le même taux d'arrivée des clients les taux d'arrivée en kbits/trame différent pour chaque $size(u)$. Ainsi pour le même taux d'arrivée des clients ($\lambda^{clients}$), le fait d'augmenter la valeur de $size(u)$ va engendrer l'augmentation du taux d'arrivée (le débit d'entrée) exprimé en kbits/trame (λ). Par exemple, pour $\lambda^{clients} = 450$ clients/trame, nous obtenons : pour $size(u) = 1024$ bits le taux d'arrivée en kbits/trame $\lambda = 406.5579$ kbits/trame, pour $size(u) = 2048$ bits, $\lambda = 814.0782$ kbits/trame et pour $size(u) = 3200$ bits, $\lambda = 941.1578$ kbits/trame.

Scénario a : sans micro cell Dans ce scénario, la micro cell n'est pas déployée ce qui fait que $\lambda^{clients} = \lambda_M^{clients}$.

- Lorsque tous les clients ont une taille de paquet $size(u) = 1024$ bits, deux classes sont générées où les clients requièrent respectivement 1 RBG et 2 RBG (voir l'application numérique de l'algorithme 1) avec un taux d'arrivées de clients $\lambda_{M,1}^{clients} = 0.5 \times \lambda_M^{clients}$ et $\lambda_{M,2}^{clients} = 0.5 \times \lambda_M^{clients}$, et tel que $g_{M,1} = 0.5$ et $g_{M,2} = 0.5$.

Pour la M/G/S/S, nous avons $\lambda_M^{clients} = \lambda_{M,1}^{clients} + \lambda_{M,2}^{clients}$ et $\frac{1}{\mu_M} = \sum_{i=1}^2 i \times g_{M,i}$ pour calculer la probabilité de blocage (voir équation (4.15)).

- Lorsque la taille du paquet $size(u) = 2048$ bits, deux classes sont générées ou les clients requièrent respectivement 2 RBG et 4 RBG (voir l'application numérique de l'algorithme [1](#)) avec un taux d'arrivées de clients $\lambda_{M,2}^{clients} = 0.5 \times \lambda_M^{clients}$ et $\lambda_{M,4}^{clients} = 0.5 \times \lambda_M^{clients}$, et tel que $g_{M,2} = 0.5$ et $g_{M,4} = 0.5$.
Pour la M/G/S/S, nous avons $\lambda_M^{clients} = \lambda_{M,2}^{clients} + \lambda_{M,4}^{clients}$ et $\frac{1}{\mu_M} = 2 \times g_{M,2} + 4 \times g_{M,4}$ pour calculer la probabilité de blocage (voir équation [\(4.15\)](#)).
- Lorsque la taille du paquet $size(u) = 3200$ bits, trois classes sont générées ou les clients requièrent respectivement 2 RBG, 3 RBG et 5 RBG (voir l'application numérique de l'algorithme [1](#)) avec un taux d'arrivées de clients $\lambda_{M,2}^{clients} = 0.27 \times \lambda_M^{clients}$, $\lambda_{M,3}^{clients} = 0.22 \times \lambda_M^{clients}$ et $\lambda_{M,5}^{clients} = 0.51 \times \lambda_M^{clients}$ dans la classe 5 et tel que $g_{M,2} = 0.27$, $g_{M,3} = 0.22$, $g_{M,5} = 0.51$.
Pour la M/G/S/S, nous avons $\lambda_M^{clients} = \lambda_{M,2}^{clients} + \lambda_{M,3}^{clients} + \lambda_{M,5}^{clients}$ et $\frac{1}{\mu_M} = 2 \times g_{M,2} + 3 \times g_{M,3} + 5 \times g_{M,5}$ pour calculer la probabilité de blocage (voir équation [\(4.15\)](#)).

Lorsque la micro n'est pas déployée, la [FIGURE 4.8](#), la [FIGURE 4.9](#) et la [FIGURE 4.10](#) montrent la précision de la forme close de la M/G/S/S, en comparant ses résultats de performance (en termes de débit) avec ceux de la simulation des chaînes de Markov en considérant les trois tailles de paquets sus-citées. Les résultats détaillés (probabilités de blocage et débits) correspondant à ces figures sont énumérés respectivement dans le [Tableau 2](#), le [Tableau 3](#) et le [Tableau 4](#) dans l'Annexe [8.5.2.4](#)

Les résultats de ces tableaux montrent que la probabilité de blocage, et par conséquent le dé-

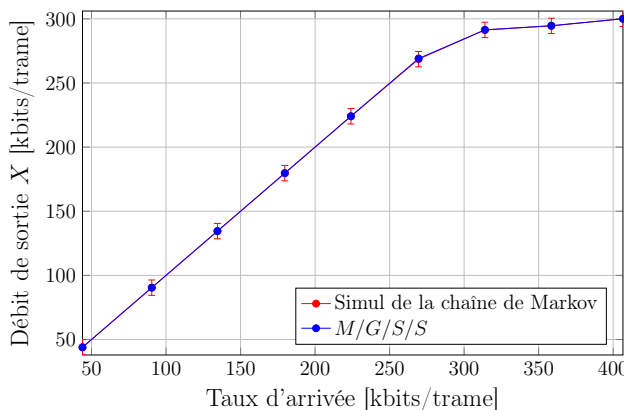


FIGURE 4.8 – Comparaison entre les simulations Matlab de la chaîne de Markov et la M/G/S/S en terme de débit (macro sans micro, $size(u) = 1024$ bits)

bit correspondant, obtenus en utilisant la forme close de la M/G/S/S sont très proches de ceux obtenus par simulation de la chaîne de Markov. En effet, nous remarquons que pour chaque taux d'arrivée des clients, la probabilité de blocage (resp. le débit correspondant) appartient à l'intervalle de probabilités de blocage (resp. le débit correspondant) obtenu par simulation où un intervalle de confiance à 95% est assuré. Ce qui engendre une erreur relative $< 1\%$ quelque soit le taux d'arrivée des clients et pour les trois tailles de paquets considérées.

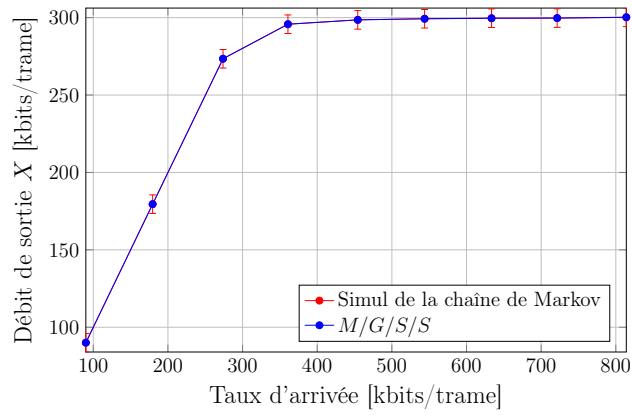


FIGURE 4.9 – Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de débit (macro sans micro, $size(u)=2048$ bits)

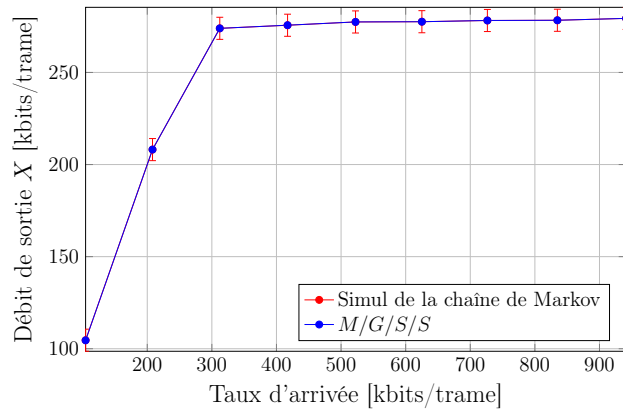


FIGURE 4.10 – Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de débit (macro sans micro, $size(u)=3200$ bits)

Remarque : Impact de $size(u)$ sur le débit de sortie

En comparant les débits de sortie pour $size(u)=1024$ bits, 2048 bits et 3200 bits, nous remarquons que lorsque la macro est faiblement/moyennement chargée, le débit de sortie augmente suite à l'augmentation de $size(u)$. Ceci est dû, d'une part, au fait que le débit d'entrée augmente avec l'augmentation de $size(u)$ (se référer à l'équation (4.32) et au paragraphe au dessous de cette équation), et d'autre part, au fait que la macro dispose d'assez de ressources pour servir ces clients. Toutefois, lorsque la macro devient fortement chargée, la valeur de $size(u)$ n'a pas d'impact sur le débit de sortie étant donné qu'il est limité par le nombre total de RBG disponibles. En effet, quelque soit le $size(u)$ considéré et par conséquent le taux d'arrivé, nous notons des débits de sortie limités par 300 kbits/trame.

Scénario b : avec une micro cell Dans ce scénario, la micro cell, localisée en périphérie de la macro cell centrale est activée (voir FIGURE 4.2).

- Lorsque la taille du paquet $size(u)= 1024$ bits, deux classes sont générées dans la macro et la micro où les clients requièrent 1 RBG et 2 RBG avec un taux d'arrivées de clients $\lambda_{M,1}^{clients} = 0.5 \times \lambda^{clients}$ (resp. $\lambda_{m,1}^{clients} = 0.02 \times \lambda^{clients}$) dans la classe 1 de la macro (resp. la micro) et $\lambda_{M,2}^{clients} = 0.47 \times \lambda^{clients}$ (resp. $\lambda_{m,2}^{clients} = 0.01 \times \lambda^{clients}$) dans la classe 2 de la macro (resp. la micro) et tel que $g_{M,1} = 0.52$, $g_{M,2} = 0.48$, $g_{m,1} = 0.67$ et $g_{m,2} = 0.33$.

Pour la M/G/S/S, nous avons $\lambda_M^{clients} = \lambda_{M,1}^{clients} + \lambda_{M,2}^{clients}$, $\lambda_m^{clients} = \lambda_{m,1}^{clients} + \lambda_{m,2}^{clients}$, $\frac{1}{\mu_M} = \sum_{i=1}^2 i \times g_{M,i}$ et $\frac{1}{\mu_m} = \sum_{i=1}^2 i \times g_{m,i}$ pour calculer la probabilité de blocage (voir équation (4.15) et (4.16)).

- Lorsque la taille du paquet $size(u)= 2048$ bits, deux classes sont générées dans la macro et la micro où les clients requièrent 2 RBG et 4 RBG avec un taux d'arrivées de clients $\lambda_{M,2}^{clients} = 0.49 \times \lambda^{clients}$ (resp. $\lambda_{m,2}^{clients} = 0.02 \times \lambda^{clients}$) dans la classe 2 de la macro (resp. la micro) et $\lambda_{M,4}^{clients} = 0.48 \times \lambda^{clients}$ (resp. $\lambda_{m,4}^{clients} = 0.01 \times \lambda^{clients}$) dans la classe 4 de la macro (resp. la micro) et tel que $g_{M,2} = 0.51$, $g_{M,4} = 0.49$, $g_{m,2} = 0.67$ et $g_{m,4} = 0.33$.

Pour la M/G/S/S, nous avons $\lambda_M^{clients} = \lambda_{M,2}^{clients} + \lambda_{M,4}^{clients}$, $\lambda_m^{clients} = \lambda_{m,2}^{clients} + \lambda_{m,4}^{clients}$, $\frac{1}{\mu_M} = 2 \times g_{M,2} + 4 \times g_{M,4}$ et $\frac{1}{\mu_m} = 2 \times g_{m,2} + 4 \times g_{m,4}$ pour calculer la probabilité de blocage (voir équation (4.15) et (4.16)).

- Lorsque la taille du paquet des clients $size(u)= 3200$ bits, trois classes sont générées dans la macro et la micro où les clients requièrent 2 RBG, 3 RBG et 5 RBG avec un taux d'arrivées de clients $\lambda_{M,2}^{clients} = 0.27 \times \lambda^{clients}$ (resp. $\lambda_{m,2}^{clients} = 0.01 \times \lambda^{clients}$) dans la classe 2 de la macro (resp. la micro), $\lambda_{M,3}^{clients} = 0.22 \times \lambda^{clients}$ (resp. $\lambda_{m,3}^{clients} = 0.01 \times \lambda^{clients}$) dans la classe 3 de la macro (resp. la micro) et $\lambda_{M,5}^{clients} = 0.47 \times \lambda^{clients}$ (resp. $\lambda_{m,5}^{clients} = 0.02 \times \lambda^{clients}$) dans la classe 5 de la macro (resp. la micro) et tel que $g_{M,2} = 0.28$, $g_{M,3} = 0.23$, $g_{M,5} = 0.49$, $g_{m,2} = 0.25$,

$$g_{m,3} = 0.25, g_{m,5} = 0.5.$$

Pour la M/G/S/S, nous avons $\lambda_M^{clients} = \lambda_{M,2}^{clients} + \lambda_{M,3}^{clients} + \lambda_{M,5}^{clients}$ et $\lambda_m^{clients} = \lambda_{m,2}^{clients} + \lambda_{m,3}^{clients} + \lambda_{m,5}^{clients}$, $\frac{1}{\mu_M} = 2 \times g_{M,2} + 3 \times g_{M,3} + 5 \times g_{M,5}$ et $\frac{1}{\mu_m} = 2 \times g_{m,2} + 3 \times g_{m,3} + 5 \times g_{m,5}$ pour calculer la probabilité de blocage (voir équation (4.15) et (4.16)).

Lorsque la micro est déployée, la FIGURE 4.11, la FIGURE 4.12 et la FIGURE 4.13 illustrent les débits globaux obtenus par la simulation des chaînes de Markov et par la M/G/S/S où respectivement $size(u) = 1024$ bits, 2048 bits et 3200 bits. Par ailleurs, plus de détails sont donnés dans le Tableau 5, le Tableau 6 et le Tableau 7 dans l'Annexe 8.5.2.4 où les probabilités de blocage et les débits globaux (macro et micro) obtenus par la simulation des chaînes de Markov et la résolution de la M/G/S/S sont présentés.

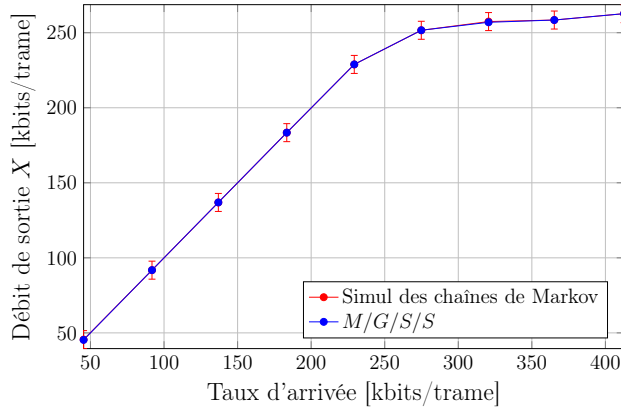


FIGURE 4.11 – Comparaison entre les simulations Matlab de la chaîne de Markov et la M/G/S/S en terme de débit global (macro avec micro $n_{ABS} = 3$, $size(u) = 1024$ bits)

D'après les résultats énumérés dans ces tableaux, nous remarquons que pour chaque taux d'arrivée des clients, la probabilité de blocage appartient à l'intervalle de probabilités de blocage obtenu par simulation où un intervalle de confiance à 95% est assuré. Ainsi, le débit global obtenu suite à la résolution de la M/G/S/S correspond aux simulations Matlab des chaînes de Markov. En faisant varier le taux d'arrivée des clients par trame, l'erreur relative entre les deux approches (chaînes de Markov et M/G/S/S) reste inférieure à 1%, pour les trois tailles de paquets que nous avons considérées (1024 bits, 2048 bits et 3200 bits).

En résumé, dans cette première contribution, nous avons proposé deux modèles pour modéliser eICIC/ABS. Le premier modèle consiste en une chaîne de Markov qui reflète le comportement du système avec des arrivées par batches, correspondant aux utilisateurs avec des conditions radio variables et demandant plus ou moins de ressources pour transmettre leurs données. Le second modèle

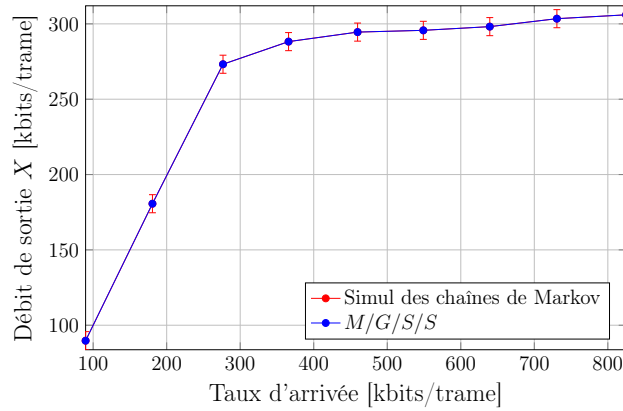


FIGURE 4.12 – Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de débit global (macro avec micro $n_{ABS} = 1$, $size(u)=2048$ bits)

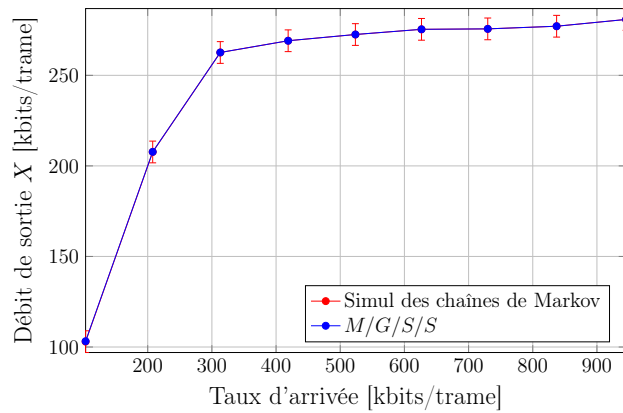


FIGURE 4.13 – Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de débit global (macro avec micro $n_{ABS} = 1$, $size(u)=3200$ bits)

est une approximation utilisant le modèle de file d'attente $M/G/S/S$ qui offre une forme close. Nous avons validé le modèle de file d'attente avec la chaîne de Markov simulée. La pertinence des modèles réside dans le fait que les paramètres d'entrée sont basés sur des distributions réelles et des conditions radio obtenues par des simulations d'un environnement urbain réaliste.

Les modèles proposés peuvent ainsi être utilisés comme outil de dimensionnement pour la technique de remédiation aux interférences eICIC/ ABS. En particulier, la forme close de la $M/G/S/S$, donnant des résultats immédiats, peut être utilisée en temps réel pour adapter le mécanisme eICIC/ ABS en fonction de la charge des utilisateurs et des conditions radio afin d'atteindre le débit global optimal. Cette contribution fera l'objet de nos études dans la section 4.6 mais avant de passer à cette étude, nous considérons dans la section suivante un modèle de file d'attente multi-serveurs avec des arrivées par batches : La $M^{[X]}/G/S/S$, ce modèle reflète donc mieux les arrivées par batch des clients.

4.5.3 Etude comparative entre les performances de la $M^{[X]}/G/S/S$ et celles des chaînes de Markov simulées

A présent, analysons les résultats de performances en termes de débit et de probabilité de blocage par classe de service (ce qui est différent par rapport à la $M/G/S/S$). Des comparaisons avec les simulations Matlab précédentes ont été effectuées afin d'estimer la précision de la $M^{[X]}/G/S/S$. Les deux scénarios avec et sans micro sont considérés, pour les tailles de paquets $size(u) = 1024$ bits et $size(u) = 2048$ bits. Les paramètres d'entrée du modèle de file d'attente ($M^{[X]}/G/S/S$), dans ces deux scénarios, sont ajustés comme suit :

Scénario a : sans micro cell Dans ce scénario, la micro cell n'est pas déployée.

- Lorsque tous les clients ont une taille de paquet $size(u) = 1024$ bits (générant deux classes de 1 RBG et 2 RBG avec un taux d'arrivée $\lambda_{M,1}^{clients}$ et $\lambda_{M,2}^{clients}$ respectivement), pour la $M^{[X]}/G/S/S$, nous avons $\lambda_M^{clients} = \lambda_{M,1}^{clients} + \lambda_{M,2}^{clients}$, $g_{M,1} = g_{M,2} = 0.5$ et $\frac{1}{\mu_M} = 1$ pour calculer la probabilité de blocage (voir équation (4.15)).
- Lorsque la taille du paquet $size(u) = 2048$ bits (générant deux classes de 2 RBG et 4 RBG avec un taux d'arrivée $\lambda_{M,2}^{clients}$ et $\lambda_{M,4}^{clients}$ respectivement), pour la $M^{[X]}/G/S/S$, nous avons $\lambda_M^{clients} = \lambda_{M,2}^{clients} + \lambda_{M,4}^{clients}$, $g_{M,2} = g_{M,4} = 0.5$ et $\frac{1}{\mu_M} = 1$ pour calculer la probabilité de blocage (voir équation (4.15)).

La FIGURE 4.14 et la FIGURE 4.15 comparent respectivement les débits de sortie de la classe 1, $X_{M,1}$, et les débits de sortie de la classe 2, $X_{M,2}$, obtenus par les simulations de la chaîne de Markov et par la résolution de la $M^{[X]}/G/S/S$ lorsque la taille des paquets $size(u) = 1024$ bits.

D'après ces figures, nous remarquons que les deux approches (chaîne de Markov et $M^{[X]}/G/S/S$)

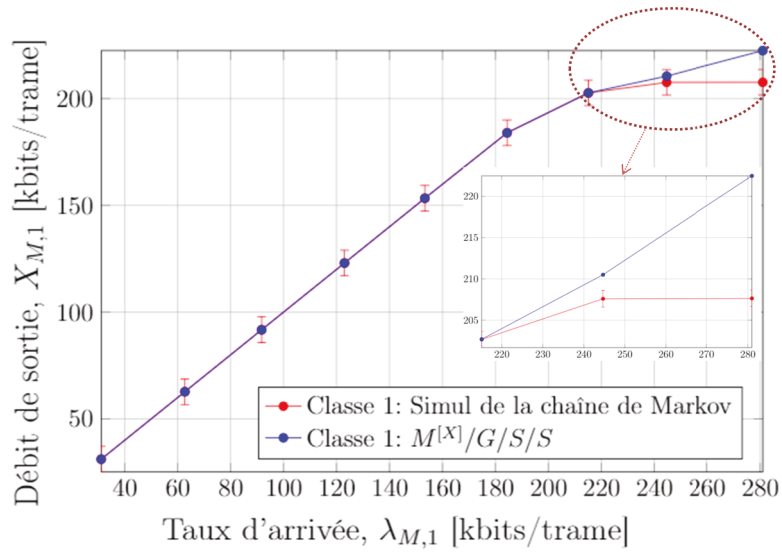


FIGURE 4.14 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de de débit $X_{M,1}$ (macro sans micro, $size(u) = 1024$ bits)

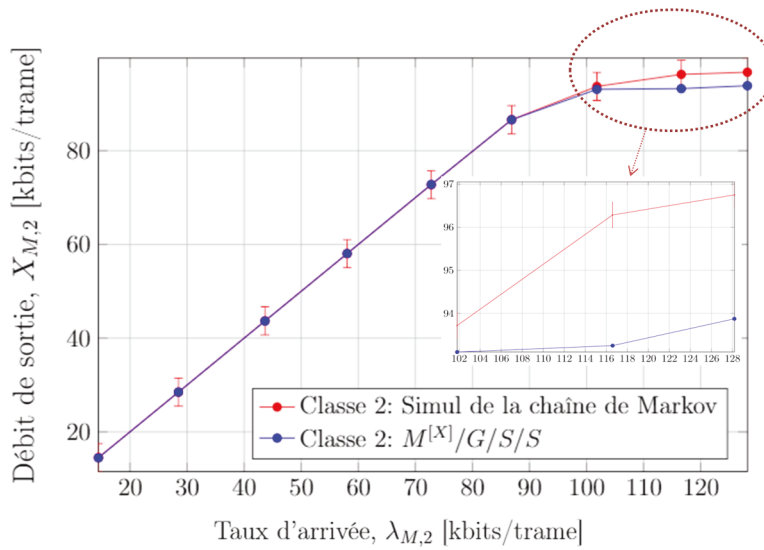


FIGURE 4.15 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de de débit $X_{M,2}$ (macro sans micro, $size(u) = 1024$ bits)

donnent les mêmes débits globaux (à faible charge) puis des débits globaux proches (à moyenne et forte charge). Ceci est dû au fait que la probabilité de blocage est nulle à faible charge dans les deux approches, et ensuite (à moyenne et forte charge) elles sont proches dans les deux modèles. Afin de voir plus en détails cet écart, le Tableau 8 dans l'Annexe 8.5.2.4 montre davantage la précision de la forme close de la $M^{[X]}/G/S/S$, en comparant la probabilité de blocage par classe $p_{M,j}^b$ et le débit par classe $X_{M,j}$, exprimé en termes de kbits/trame, lorsque la micro n'est pas déployée et la taille des paquets $size(u) = 1024$ bits.

Les résultats de ce tableau montrent que les probabilités de blocage par classe, et par conséquent le débit correspondant, obtenus en utilisant la $M^{[X]}/G/S/S$ sont proches de ceux obtenus par les simulations. En variant les taux d'arrivée de clients de 50 à 350 par trame, l'erreur relative entre les débits reste inférieure à 1% pour les deux classes. Pour des taux plus élevés, l'erreur relative reste inférieure à 7% pour la classe 1 et inférieure à 4% pour la classe 2. Ceci est expliqué par le fait que les départs dans la $M^{[X]}/G/S/S$ sont asynchrones, les clients de la classe 2 (composés de 2 unités de batch) quittent le système de file d'attente plus rapidement que dans le système réel simulé, offrant à la $M^{[X]}/G/S/S$ la possibilité de servir plus de batchs de la classe 1. Notons que l'erreur relative en terme de débit global reste inférieure à 4% pour les taux d'arrivée de clients supérieurs à 350 clients/trame et elle est inférieure à 1% pour les taux d'arrivée de clients inférieurs à ce taux.

La FIGURE 4.16 et la FIGURE 4.17 comparent respectivement les débits de sortie de la classe 2, $X_{M,2}$, et les débits de sortie de la classe 4, $X_{M,4}$, obtenus par les simulations de la chaîne de Markov et par la résolution de la $M^{[X]}/G/S/S$ lorsque la taille des paquets $size(u) = 2048$ bits.

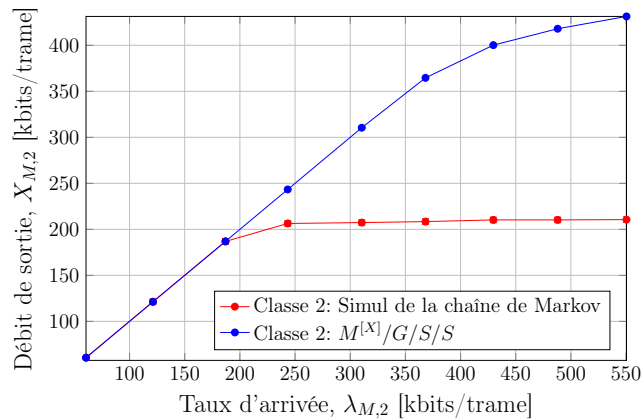


FIGURE 4.16 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de débit de sortie dans la classe 2 (macro sans micro, $size(u) = 2048$ bits)

D'après ces figures, nous remarquons un écart entre les deux approches (chaîne de Markov et

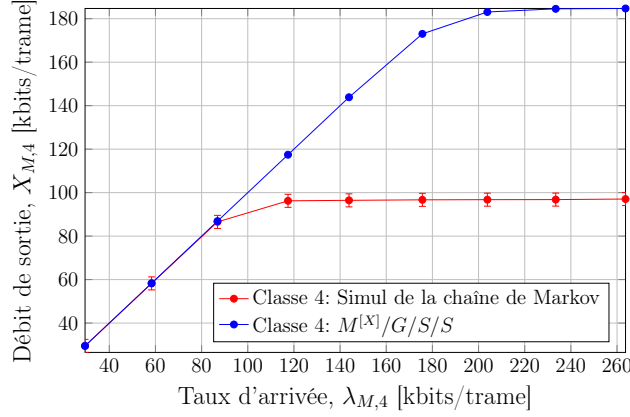


FIGURE 4.17 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de débit de sortie dans la classe 4 (macro sans micro, $size(u) = 2048$ bits)

$M^{[X]}/G/S/S$) lorsque les classes sont chargées. Afin de voir plus en détails cet écart, le Tableau 9 dans l'Annexe 8.5.2.4 montre davantage la précision de la forme close de la $M^{[X]}/G/S/S$, en comparant la probabilité de blocage par classe $p_{M,j}^b$ et le débit par classe $X_{M,j}$, exprimé en termes de kbits/trame, lorsque la micro n'est pas déployée et la taille des paquets $size(u) = 2048$ bits.

D'après ce tableau, nous remarquons qu'à faible charge c'est à dire lorsque le taux d'arrivée des clients est inférieur à 150 clients/trame, les probabilités de blocage obtenues suite à la résolution de la $M^{[X]}/G/S/S$ sont proches de celles obtenues par la simulation de la chaîne de Markov. Toutefois, lorsque le taux d'arrivée des clients est supérieur à 150 clients/trame, les deux modèles se comportent différemment et donc offrent des probabilités de blocage et des débits de sortie différents. Plus précisément, la $M^{[X]}/G/S/S$ offre des probabilités de blocage par classe inférieures à celle de la simulations correspondant à des débits par classe supérieurs. Rappelons qu'au niveau de la chaîne de Markov, un client de classe j a un temps de service correspondant à une unité (c'est à dire la trame) car chaque élément du batch occupe 1 RGB pendant une trame : $t_{s_j} = \frac{1}{\mu_j} = 1$. Dans ce système, les départs sont synchronisés car au bout d'une trame tous les éléments du batch d'un client sont servis. Toutefois, pour la $M^{[X]}/G/S/S$, un client de classe j , ou plus précisément chaque élément du batch de taille j occupe un RGB (ou un serveur), et les temps entre les fins de services sont en moyenne de $\frac{1}{j}$. Ainsi, comme le système est asynchrone, il se vide plus rapidement, engendrant ainsi moins de perte, et plus de clients servis.

Scénario b : avec micro cell Dans ce scénario, la micro cell est déployée.

- Lorsque tous les clients ont une taille de paquet $size(u) = 1024$ bits (générant deux classes, dans la macro et la micro, de 1 RGB et 2 RGB avec un taux d'arrivée dans la classe 1 de la macro (resp. la micro) $\lambda_{M,1}$ (resp. $\lambda_{m,1}$) et un taux d'arrivées de clients dans la classe 2 de la macro (resp. la micro) $\lambda_{M,2}$ (resp. $\lambda_{m,2}$)), pour la $M^{[X]}/G/S/S$, nous avons $g_{M,1} = 0.52$,

$g_{M,2} = 0.48$, $\frac{1}{\mu_M} = 1$, $g_{m,1} = 0.67$, $g_{m,2} = 0.33$ et $\frac{1}{\mu_m} = 1$ pour calculer la probabilité de blocage (voir équation (4.15) et (4.16)).

- Lorsque la taille du paquet $size(u) = 2048$ bits (générant deux classes, dans la macro et la micro, de 2 RBG et 4 RBG avec un taux d'arrivée dans la classe 2 de la macro (resp. la micro) $\lambda_{M,2}$ (resp. $\lambda_{m,2}$) et un taux d'arrivées de clients dans la classe 4 de la macro (resp. la micro) $\lambda_{M,4}$ (resp. $\lambda_{m,4}$) respectivement), pour la $M^{[X]}/G/S/S$, nous avons $g_{M,2} = 0.51$, $g_{M,4} = 0.49$, $g_{m,2} = 0.67$, $g_{m,4} = 0.33$ et $\frac{1}{\mu_M} = 1$ pour calculer la probabilité de blocage (voir équation (voir équation (4.15) et (4.16)).

Lorsque la micro est déployée (avec $n_{ABS} = 3$, taille de paquets $size(u) = 1024$ bits), la FIGURE 4.18 et la FIGURE 4.19 illustrent, respectivement, les résultats obtenus dans la macro classe 1 et la macro classe 2. Quant à la micro, pour les taux d'arrivée considérés et avec $n_{ABS} = 3$, la probabilité de blocage obtenue par les deux approches est nulle engendrant ainsi les mêmes débits globaux (se référer au Tableau 10 et au Tableau 11, dans l'Annexe 8.5.2.4, pour plus de détails sur la précision des deux approches et des probabilités de blocage).

Les résultats détaillés dans le Tableau 10 révèlent que pour la classe 1 de la macro l'erreur re-

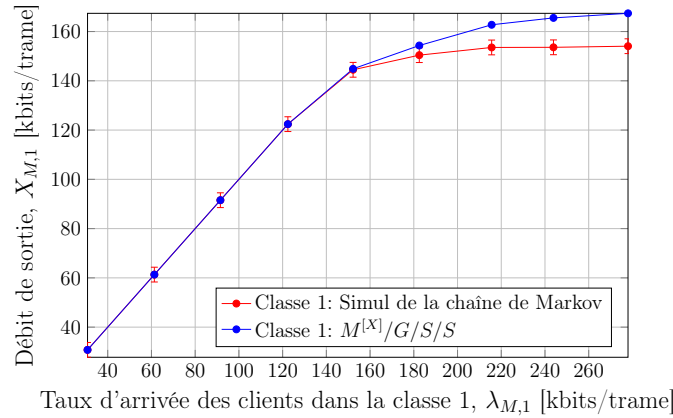


FIGURE 4.18 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de débit de sortie dans la classe 1 de la macro (macro avec micro, $n_{ABS} = 3$, $size(u) = 1024$ bits)

lative, entre les débits, est inférieur à 1% lorsque le taux d'arrivée des clients est inférieur à 250 clients/trame (correspondant à $\lambda_{M,1} = 152.2261$ kbits/trame). Lorsque la charge est comprise entre 250 clients/trame (correspondant à $\lambda_{M,1} = 152.2261$ kbits/trame) et 400 clients/trame (correspondant à $\lambda_{M,1} = 243.8834$ kbits/trame), l'erreur relative est inférieure à 6%, elle atteint 9% lorsque le taux d'arrivée des clients est à 400 clients/trame (correspondant à $\lambda_{M,1} = 243.8834$ kbits/trame) et 13% lorsque nous avons 450 clients/trame (correspondant à $\lambda_{M,1} = 277.9834$ kbits/trame).

Pour la classe 2 de la macro, à faible charge (entre 50 et 250 clients/trame correspondant à $\lambda_{M,2}$ entre 13.4903 et 67.5475 kbits/trame) l'erreur relative est inférieur à 1%, à moyenne charge (entre 250 et 300 clients/trame correspondant à $\lambda_{M,2}$ entre 67.5475 et 81.3593 kbits/trame) nous notons

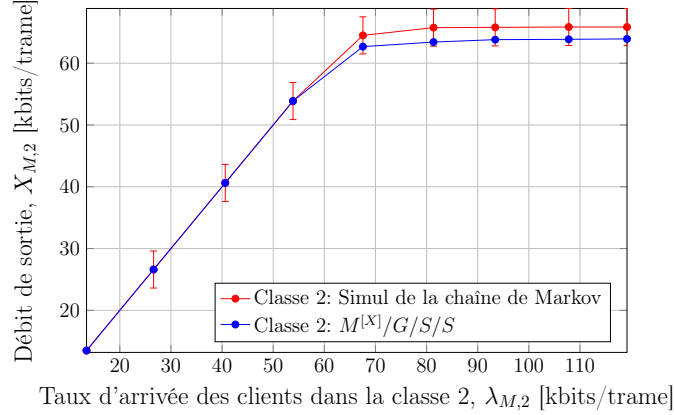


FIGURE 4.19 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de débit de sortie dans la classe 2 de la macro (macro avec micro, $n_{ABS} = 3$, $size(u) = 1024$ bits)

une erreur relative inférieure à 5% et à forte charge (entre 350 et 450 clients/trame correspondant à $\lambda_{M,2}$ entre 93.4219 et 119.2055 kbits/trame) une erreur relative inférieure à 4%.

Le Tableau 12 et le Tableau 13 donnés dans l'Annexe 8.5.2.4 énumèrent, respectivement, les résultats obtenus dans la macro et la micro pour la taille de paquets $size(u) = 2048$ bits avec $n_{ABS} = 1$. Avec la taille de paquets $size(u) = 2048$ bits qui engendre deux classes de clients demandant 2 RBG et 4 RBG, nous remarquons d'après les résultats sur les débits par classe dans la macro, énumérés dans le Tableau 12, que les deux approches (simulations de la chaîne de Markov et la résolution de la $M^{[X]}/G/S/S$) offrent des résultats proches en termes de probabilités de blocage et de débit de sortie par classe (erreur relative inférieur à 1% pour les deux classes), lorsque le taux d'arrivée des clients est faible (probabilité de blocage nulle ou presque). Toutefois, lorsque le taux d'arrivée des clients devient important (à partir de 200 clients/trame correspondant à $\lambda_{M,2}=243.4543$ kbits/trame) engendrant ainsi des probabilités de blocage par classe, comme les deux approches se comportent différemment, nous notons que la $M^{[X]}/G/S/S$ donne des probabilités de blocage plus faibles dues au fait que les départs sont asynchrones dans la $M^{[X]}/G/S/S$, ceci permet de servir plus de clients, d'où une probabilité plus faible et par conséquent un débit plus élevé.

Quant aux débits par classe dans la micro, la $M^{[X]}/G/S/S$ et les simulations offrent des résultats proches avec une erreur relative inférieure à 4% pour les deux classes quelque soit le taux d'arrivée des clients. Ceci est expliqué par le fait que la micro, grâce aux ressources qui lui sont attribuées ($n_{ABS} = 1$), arrive à servir ses clients d'où une probabilité de blocage nulle ou presque pour les deux approches.

Ainsi, dans cette deuxième contribution sur la modélisation de la technique eICIC/ ABS, nous avons considéré un modèle de file d'attente $M^{[X]}/G/S/S$ multi-serveurs avec des arrivées par batch afin de mieux refléter les conditions radio des clients. Ce modèle est validé sur une partie des

scénarios considérés. En effet, nous avons vu qu’il reflète bien le comportement réel, quelque soit la charge du trafic, lorsque nous avons deux classes de clients demandant 1 RBG et 2 RBG. Par ailleurs lorsque nous avons des tailles de paquets plus importantes, par exemple $size(u) = 2048$ bits, engendrant ainsi deux classes où les clients requièrent 2 RBG et 4 RBG, les résultats obtenus par le modèle de file d’attente $M^{[X]}/G/S/S$ sont validées par la simulation lorsque le taux d’arrivée des clients est faible. Toutefois, lorsque le taux d’arrivée des clients augmente, la $M^{[X]}/G/S/S$ se comporte différemment du comportement réel offrant des probabilités de blocage plus faibles et donc des débits plus élevés avec des erreurs relatives dépassant 15%, ainsi pour ces cas le modèle de file d’attente $M^{[X]}/G/S/S$ n’est pas validé.

4.5.4 Etude comparative entre les trois approches proposées : chaînes de Markov, $M/G/S/S$ et $M^{[X]}/G/S/S$

4.5.4.1 Comparaison des débits par classe en fonction du taux d’arrivée

Dans la FIGURE 4.20 et la FIGURE 4.21 (voir le Tableau 14 en Annexe 8.5.2.4 pour plus de détails), nous comparons respectivement les débits de sortie de la classe 1, $X_{M,1}$, et les débits de sortie de la classe 2, $X_{M,2}$, obtenus par les trois approches : simulations Matlab, $M/G/S/S$ et $M^{[X]}/G/S/S$, exprimés en termes de kbits/trame, lorsque la macro est déployée sans micro, avec une taille de paquet $size(u) = 1024$ bits. Nous calculons $X_{M,j}$ dans la $M/G/S/S$ par l’équation (??) suivante :

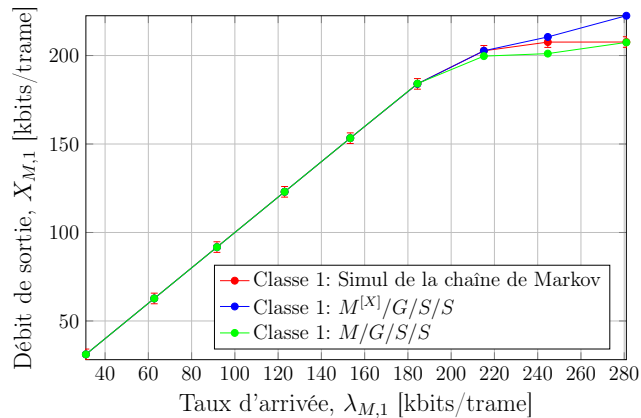


FIGURE 4.20 – Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 1024$ bits)

D’après ces résultats, lorsque le taux d’arrivée des clients est faible ($\lambda_{M,1}=153.3447$ kbits/trame), les modèles de files d’attente que nous proposons ($M^{[X]}/G/S/S$ et la $M/G/S/S$) offrent les mêmes débits par classe et appartenant à l’intervalle obtenu par la simulation des chaînes de Markov.

Lorsque le système est moyennement chargé ($\lambda_{M,1}$ varie entre 184.4042 et 244.6795 kbits/trame),

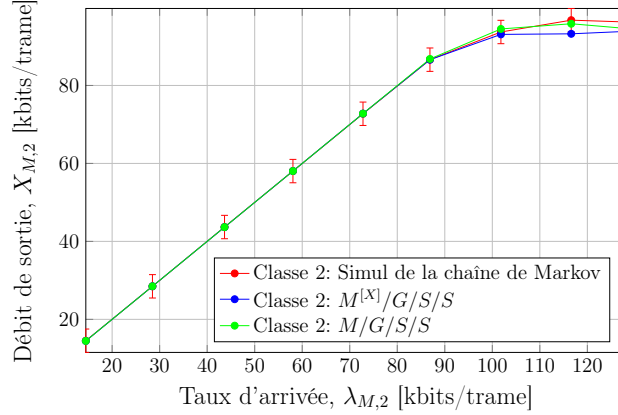


FIGURE 4.21 – Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 1024$ bits)

le débit de la classe 1 obtenu par la $M^{[X]}/G/S/S$ est plus proche de la simulation que celui obtenu par la $M/G/S/S$. De même pour le débit de la classe 2, lorsque le taux d'arrivée des clients $\lambda_{M,2}$ varie entre 86.8772 et 101.8105 kbits/trame, la $M^{[X]}/G/S/S$ offre également des débits plus proches de la simulation que la $M/G/S/S$. Enfin, lorsque le taux d'arrivée est supérieur aux valeurs sus-citées, la $M/G/S/S$ offre des débits par classe plus proches de la simulation que ceux obtenus par la $M^{[X]}/G/S/S$. Notons que pour $size(u) = 1024$ bits, les deux modèles de files d'attente que nous proposons représentent de bonnes approximations.

Dans la FIGURE 4.22 et la FIGURE 4.23 (voir le Tableau 15 en Annexe 8.5.2.4 pour plus de détails), nous comparons respectivement les débits de sortie de la classe 2, $X_{M,2}$, et les débits de sortie de la classe 4, $X_{M,4}$, obtenus par les trois approches : simulations Matlab, $M/G/S/S$ et $M^{[X]}/G/S/S$, exprimés en termes de kbits/trame, lorsque la macro est déployée sans micro, avec une taille de paquet $size(u) = 2048$ bits.

D'après ces résultats où $size(u) = 2048$ bits, lorsque le système est faiblement chargé la $M/G/S/S$ et la $M^{[X]}/G/S/S$ offrent les mêmes débits par classe appartenant à l'intervalle obtenu par simulation de la chaîne de Markov. Toutefois, lorsque le système devient moyennement/ fortement chargé la $M/G/S/S$ offre des débits par classe plus proches de ceux obtenus par simulation. Comme vu dans la section précédente, pour $size(u) = 2048$ bits, la $M^{[X]}/G/S/S$ ne donne pas de bons résultats étant donné que nous avons des tailles de batches qui vont jusqu'à 4 (plus la taille des batches est grande, plus la forme close de la $M^{[X]}/G/S/S$ est mauvaise).

En conclusion de cette étude sur la comparaison des débits globaux obtenus par les trois approches : simulations Matlab, $M/G/S/S$ et $M^{[X]}/G/S/S$, nous retenons ce qui suit. Globalement, la $M/G/S/S$ donne de meilleurs résultats car nous utilisons la formule de la probabilité de blocage de l'Erlang B pour la probabilité de perte, qui est insensible à la distribution des temps de service

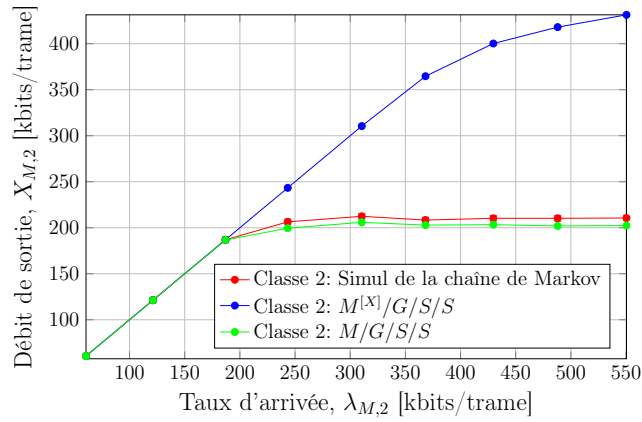


FIGURE 4.22 – Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 2048$ bits)

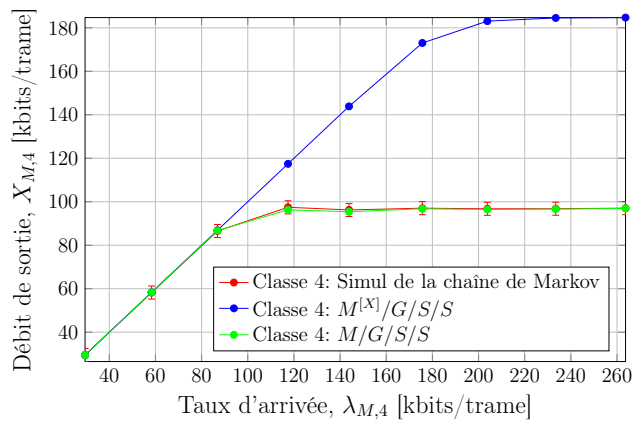


FIGURE 4.23 – Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 2048$ bits)

(nous avons considéré des temps de service constants). Alors que la $M^{[X]}/G/S/S$ considère des temps de service exponentiels, et des arrivées exponentielles par batch. Ainsi, même si ce système a un processus d'arrivées plus proche de la chaîne de Markov, que l'autre modèle de file d'attente, la $M/G/S/S$ qui considère des temps de services constants permet d'obtenir par l'Erlang B des probabilités de perte globalement plus proches de celles de la chaîne de Markov.

4.5.4.2 Comparaison des temps d'exécution des trois approches proposées : chaînes de Markov, $M/G/S/S$ et $M^{[X]}/G/S/S$

Suite aux résultats obtenus, nous pouvons donc utiliser les modèles $M/G/S/S$ et $M^{[X]}/G/S/S$ pour modéliser le mécanisme eICIC/ ABS. Ayant pour objectif de proposer des outils de dimensionnement très rapides, nous nous intéressons à présent à la comparaison du temps de simulation des chaînes de Markov avec ceux de la $M^{[X]}/G/S/S$ et de la $M/G/S/S$.

Nous précisons que le pc sur lequel les exécutions ont été faites a les caractéristiques suivantes :

- Processeur : Intel (R) Core (TM) i7-6600U CPU@ 2.60 GHz 2.70 GHz
- Mémoire installée (RAM) : 16 Go

Dans le Tableau 4.4, nous résumons les temps de simulation des chaînes de Markov et de la résolution de la $M/G/S/S$ et de la $M^{[X]}/G/S/S$ en fonction du taux d'arrivée des clients, lorsque la micro est déployée et nous supposons que la taille des paquets des clients $size(u) = 1024$ bits. Dans la première colonne, nous représentons le temps d'exécution de l'algorithme 1 et le temps de simulation de la chaîne de Markov que nous notons Simul (Algo 1+ CM) où Simul(CM) signifie simulation de la chaîne de Markov. Dans les autres colonnes, nous ne prenons pas en compte le temps de simulation de l'algorithme 1, permettant de déterminer la distribution des utilisateurs et leurs conditions radio, étant donné que ces paramètres sont communs aux trois approches et nécessaires pour la simulation des chaînes de Markov et pour la résolution de la $M/G/S/S$ et de la $M^{[X]}/G/S/S$.

D'après le Tableau 4.4, nous remarquons que le temps de simulation des chaînes de markov et de l'algorithme 1 dépendent du taux d'arrivée des clients. En effet, le fait d'augmenter le taux d'arrivée des clients implique l'augmentation du nombre de clients à générer ($Nb UE$, voir l'algorithme 1) ce qui augmente le nombre d'utilisateurs à traiter dans l'algorithme 1 et dans la simulation des chaînes de Markov. Nous rappelons que dans la simulation des chaînes de Markov, nous calculons le débit par utilisateur puis le débit par classe. En ce qui concerne les modèles de files d'attente, nous remarquons que leurs temps de résolution sont indépendants du taux d'arrivée des clients (voir l'équation (4.15) et l'équation (4.16) pour la $M/G/S/S$ ainsi que l'équation (4.24) et l'équation (4.25) pour la $M^{[X]}/G/S/S$).

De plus, nous remarquons que la $M/G/S/S$ est plus rapide que la simulation. Effectivement, pour la $M/G/S/S$ (qui nécessite uniquement 2s de résolution), nous notons une différence très impor-

$\lambda^{clients}$	Simul (Algo 1 + CM)	Simul (CM)	$M^{[X]}/G/S/S$	$M/G/S/S$
50	13 min	3 min	10 min	2 s
100	19 min	6 min	10 min	2 s
150	30 min	7 min	10 min	2 s
200	41 min	9 min	10 min	2 s
250	54 min	11 min	10 min	2 s
300	1h 9 min	14 min	10 min	2 s
350	1h 24 min	17 min	10 min	2 s
400	1h 45 min	20 min	10 min	2 s
450	2h 25 min	25 min	10 min	2 s

Tableau 4.4 – Temps d'exécution des trois approches (macro avec micro, $size(u)=1024$ bits)

tante quelque soit le taux d'arrivée des clients (résolution en "temps réel"). Par ailleurs, le temps de résolution de la $M^{[X]}/G/S/S$ est estimé à 10 min ce qui apporte une amélioration pour tout $\lambda^{clients} \geq 250$ clients/trame. La lenteur de la $M^{[X]}/G/S/S$ par rapport à la $M/G/S/S$ est due à la complexité de l'équation (4.24) et de l'équation (4.25) qui est d'ordre de $\theta(n^{J_M})$ et $\theta(n^{J_m})$, respectivement.

4.5.5 Influence de la micro et du nombre d'ABS

Afin d'expliquer l'influence du déploiement de la micro, nous comparons dans la FIGURE 4.26 le débit de sortie global obtenu, en fonction du taux d'arrivée des clients, lorsque la macro est déployée seule et lorsque la macro est déployée avec la micro pour une taille de paquet $size(u) = 1024$ bits. Lorsque la micro est déployée, nous considérons deux scénarios : $n_{ABS} = 1$ et $n_{ABS} = 3$.

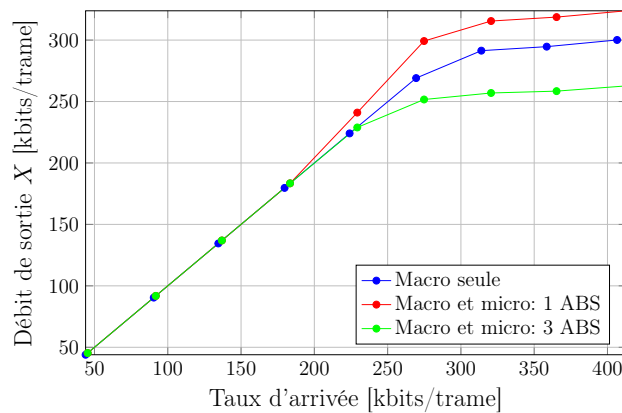


FIGURE 4.24 – Influence du déploiement de la micro

D'après la FIGURE 4.26, lorsque le taux d'arrivée des clients est inférieur à 250 kbits/trame, le

déploiement de la micro va permettre d'améliorer légèrement le débit global. Ceci est dû au fait que pour de telles charges, le nombre d'UE servis dans la micro est faible et donc peu d'utilisateurs vont avoir de meilleures conditions radio. Toutefois, les résultats révèlent que lorsque le taux d'arrivée est supérieur à 250 kbits/trame, si nous déployons une micro il sera indispensable de lui attribuer le nombre d'ABS adéquat. En effet, pour de tels taux d'arrivée de clients, si nous déployons la micro en lui attribuant 3 ABS, le débit global atteint sera inférieur au débit global obtenu sans son déploiement. Par ailleurs, le fait de déployer la micro et lui attribuer 1 ABS va permettre d'améliorer le débit global. Le déploiement de la micro et du nombre d'ABS influencent donc le débit global d'où la nécessité d'utiliser un outil de dimensionnement qui permet de déterminer le moment d'activation/désactivation de la micro et le nombre d'ABS à lui attribuer afin d'atteindre un débit global optimal.

Afin d'expliquer davantage l'influence du nombre d'ABS, considérons une taille de paquet $size(u) = 1024$ bits et deux taux d'arrivées des clients, que nous notons $taux_1$ et $taux_2$ où $taux_1 = 200$ clients/trame et $taux_2 = 450$ clients/trame.

Nous représentons dans la FIGURE 4.25 le débit de sortie global, exprimé en kbits/trame, en fonction du nombre d'ABS attribué à la micro pour le $taux_1$ et le $taux_2$.

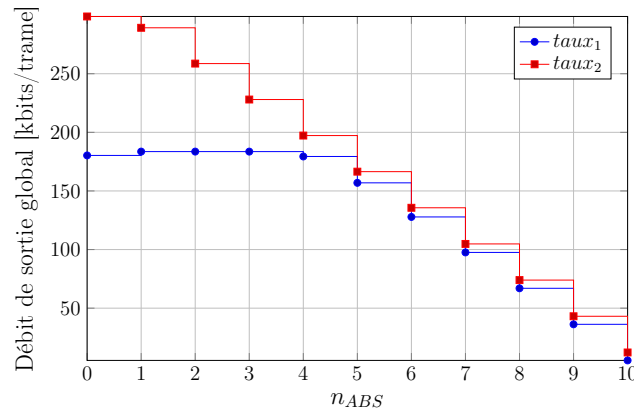


FIGURE 4.25 – Débit de sortie global en fonction du n_{ABS} ($size(u) = 1024$ bits)

D'après la FIGURE 4.25, nous remarquons que le débit de sortie global change en fonction de la valeur du nombre d'ABS attribué à la micro (n_{ABS}) où $n_{ABS} = 0$ correspond au cas où la micro n'est pas déployée.

Pour le $taux_1$, il s'avère qu'il serait intéressant en termes de débit de sortie global de déployer la micro si on lui attribue 1 ABS, 2 ABS ou 3 ABS. En effet, lorsque la micro n'est pas déployée nous notons un débit de sortie global égal à 180.2923 kbits/trame (contre 183.5657 kbits/trame pour $n_{ABS} = 1, 2$ et 3). La micro va donc améliorer les conditions radio des UE qui lui sont rattachés sans pénaliser les UEs rattachés à la macro. Toutefois, le débit de sortie global décroît lorsque

$n_{ABS} \geq 4$. Effectivement, le déploiement de la micro en sur-dimensionnant le nombre d'ABS à lui attribuer pénalise les UE de la macro, ce qui engendre un débit de sortie plus faible par rapport au scénario où elle n'est pas déployée ou par rapport au scénario où 1, 2, 3 ABS lui sont attribués.

Pour le $taux_2$, il est préférable de ne pas déployer la micro pour atteindre un débit de sortie global optimal, $X = 298.8392$ kbits/trame. Le fait de déployer la micro en lui attribuant même 1 seul ABS va pénaliser les clients de la macro. En effet, pour cette charge le taux d'arrivée des clients dans la macro est $\lambda_M = 400.8145$ kbits/trame et dans la micro est $\lambda_m = 12.2083$ kbits/trame, ce qui correspond à un taux d'arrivée des clients en termes de RBG/trame dans la macro $\lambda_M^{RBG} = 648$ RBG/trame et dans la micro $\lambda_m^{RBG} = 18$ RBG/trame. L'attribution même d'un seul ABS à la micro, correspondant à 50 RBG (ou serveurs) pour servir $\lambda_m^{RBG} = 18$ RBG/trame, est donc vu comme un sur-dimensionnement des ressources de la micro pénalisant les clients de la macro qui est complètement chargée.

Ainsi, lorsque la micro cell est déployée et que le mécanisme eICIC/ ABS est appliqué, pour chaque taux d'arrivée des clients, les performances vont évidemment dépendre du nombre d'ABS attribués à la micro et des conditions radio. Le débit optimal ne peut être atteint qu'avec une configuration d'ABS optimal.

Afin d'atteindre cette objectif, nous proposons dans ce qui suit un algorithme basé sur les modèles $M/G/S/S$ et $M^{[X]}/G/S/S$ pour décider aisément, selon la charge du trafic, sur l'activation/ désactivation de la micro ainsi que la distribution des ressources (sous-trames) et des clients entre la macro et la micro lorsque cette dernière est déployée.

4.6 Optimisation de eICIC/ ABS basée sur les modèles de files d'attente

Dans cette section, nous proposons un algorithme basé sur les modèles de files d'attente ($M/G/S/S$ et $M^{[X]}/G/S/S$) qui permet de décider dynamiquement sur l'activation/ désactivation des micro cells ainsi que le nombre de ressources que la macro devrait leur déléguer. Le principal objectif est de maximiser le débit global du système.

Lorsque la micro cell est déployée, un nombre limité d'utilisateurs qui lui sont associés devraient être bloqués (ou perdus) (c'est à dire la probabilité de blocage $\leq \epsilon$). Nous supposons, sans perte de généralité, que les conditions radio sont toujours bonnes pour les UE associés à des micro cells. En effet, un UE n'est associé à une micro cell que si cette dernière permet d'améliorer ses conditions radio. Sinon, l'UE restera associé à la macro cellule. Par conséquent, nous pouvons supposer qu'une seule classe de service est disponible dans la micro cell, où tous les UE ont de très bonnes conditions radio et demandent un RBG (c'est-à-dire $\lambda_m^{clients} = \lambda_{m,1}^{clients}$).

Dans ce contexte, nous formulons un problème d'optimisation. Les variables de décisions sont les suivantes :

- $\lambda_m^{clients}$: taux d'arrivée des clients dans la micro cell,
- $\lambda_M^{clients}$: taux d'arrivée des clients dans la macro cell,
- $S_m = S \times \frac{n_{ABS}}{10}$: nombre de ressources attribuées à la micro cell et
- $S_M = S - S_m$: nombre de ressources attribuées à la macro cell.

Notons que nous avons délibérément défini le taux d'arrivée des clients dans la micro cellule en tant que variable, car ce taux (qui fait partie du taux d'arrivée global) doit également être optimisé avec la configuration d'ABS.

Notre problème d'optimisation est exprimé comme suit :

$$\begin{aligned}
& \max_{S_m, S_M, \lambda_m^{clients}, \lambda_M^{clients}} f(S_m, S_M, \lambda_m^{clients}, \lambda_M^{clients}) \\
& \text{s.t.} \quad S_m + S_M = S, \\
& \quad \lambda_m^{clients} + \sum_{j=1}^{J_M} \lambda_{M,j}^{clients} = \lambda^{clients}, \\
& \quad \sum_{j=1}^{J_M} \lambda_{M,j}^{clients} = \lambda_M^{clients}, \\
& \quad S_M > 0, \\
& \quad S_m \geq 0, \\
& \quad p_m^b \leq \epsilon
\end{aligned} \tag{4.34}$$

où $f(S_m, S_M, \lambda_m^{clients}, \lambda_M^{clients})$ représente le débit global exprimé en termes de kbits/ trame et calculé comme suit :

$$f(S_m, S_M, \lambda_m^{clients}, \lambda_M^{clients}) = X = \sum_{j=1}^{J_M} X_{M,j} + X_{m,1} = \frac{\sum_{j=1}^{J_M} X_{M,j}^{bits} + X_{m,1}^{bits}}{1024} \tag{4.35}$$

tel que :

$X_{M,j}^{bits}$ et $X_{m,j}^{bits}$ sont calculés respectivement en utilisant l'équation (4.28) et l'équation (4.29) pour la $M^{[X]}/G/S/S$ et en utilisant l'équation (4.19) et l'équation (4.21) pour la $M/G/S/S$.

Suite à la combinaison de variables continues ($\lambda_m^{clients}$) et de variables discrètes (S_m et S_M) et à cause de la non-linéarité de la fonction objective ($X_{M,j}$ pour $j=1, \dots, J_M$ et $X_{m,1}$) et de la contrainte (p_m^b), le problème de maximisation correspond à un MINLP [Flo95] connu comme un problème NP-hard [VG18]. Une façon simple pour résoudre ce problème est de fixer S_m (et par conséquent S_M) ensuite, pour chaque valeur nous déterminons le $\lambda_m^{clients}$ optimal qui maximise le débit global. Enfin, le S_m (et S_M) et $\lambda_m^{clients}$ optimaux qui offrent le débit global optimal sont retenus.

Notre problème correspond maintenant à un problème NLP [BSS13] [NW06] à cause de la contrainte et de la fonction objective qui ne sont pas linéaires. L'une des solutions pour résoudre un problème

de minimisation NLP consiste à utiliser Global Optimization Toolbox de Matlab. Cette toolbox propose des fonctions permettant de rechercher des solutions globales pour les problèmes contenant plusieurs minima ou maxima. L'un des solveurs de cette toolbox, et utilisé dans le cadre de cette thèse, est Global Search. Le principe de fonctionnement et les différentes étapes de Global Search sont représentées dans [Mat20].

Toutefois, pour nos études, par exemple pour un pourcentage de clients maximal à rattacher à la micro égal à $3\% \times \lambda^{clients}$, pour déterminer la configuration optimale, la résolution par Global Search prend 2h lorsque Global Search se base sur la forme close de la $M/G/S/S$ et 10h lorsqu'il se base sur la forme close de la $M^{[X]}/G/S/S$. De plus, nous notons que plus le pourcentage de clients maximal à rattacher à la micro est grand, plus la durée de résolution par Global Search prend du temps (pouvant atteindre des dizaines d'heures). Afin de remédier à cette faiblesse, nous proposons une heuristique (voir l'algorithme 2) permettant de rechercher la valeur optimale dans un ensemble réduit (pour déterminer $\lambda_m^{clients}$ optimal, nous nous limitons à tester certaines valeurs).

4.6.1 Description de l'algorithme d'optimisation

Afin de réduire le temps de résolution, nous proposons l'algorithme 2 où :

- (1) D'abord, le modèle d'optimisation pour eICIC/ ABS détermine $X^* = \frac{\sum_{j=1}^{J_M} X_{M,j}^{bits}}{1024}$ le débit global lorsque la micro n'est pas déployée tel que $X_{M,j}^{bits}$ est calculé à partir de l'équation (4.28) pour la $M^{[X]}/G/S/S$ et à partir de l'équation (4.19) pour la $M/G/S/S$.
- (2) Nous notons i_{max} le pourcentage initial d'utilisateurs générés dans la micro cell, il correspond donc au pourcentage maximal de clients à lui rattacher. Le modèle d'optimisation va ensuite varier de $i \neq 0$ ($i = 0$ correspond au cas où la micro n'est pas déployée) à i_{max} avec un pas de boucle noté 'step' (plus les valeurs de 'i' et de 'step' sont petites, meilleure est la précision du modèle d'optimisation). Ensuite, pour chaque pourcentage de clients à associer à la micro cell (chaque i), le modèle d'optimisation détermine les taux d'arrivée de clients dans la macro et la micro. Puis, il détermine le nombre minimal de ressources $S_m(i)$ à attribuer à la micro qui offre un débit global optimal $X(i)$ avec une garantie de service pour les clients de la micro égale à $(1 - \epsilon)\%$ autrement dit vérifiant $p_m^b(i) \leq \epsilon$. Pour ce faire, pour chaque i , l'algorithme fait varier le nombre d'ABS à attribuer à la micro $n_{ABS}(i)$ ($1 \leq n_{ABS}(i) \leq 10$). Ensuite, pour chaque ABS, notre algorithme détermine le nombre de ressources correspondant ($S_m^{**}(i) = S \times \frac{n_{ABS}(i)}{10}$) et ne conserve que le/les $S_m^{**}(i)$ qui garantissent un service de $(1 - \epsilon)\%$ pour les clients de la micro. Pour les $S_m^{**}(i)$ garantissant cette contrainte de service dans la micro, l'algorithme calcule le débit global correspondant puis détermine le max de ces débits globaux, $X(i)$, et retient le/les $S_m^{**}(i)$ offrant le débit maximal, que nous notons $S_m^*(i)$. Si plusieurs $S_m^*(i)$ sont retenus alors l'algorithme sélectionne le $S_m^*(i)$ minimal, qu'on note $S_m(i)$. Par conséquent, pour i , $S_m(i)$ représente le nombre de ressources optimal qui assure $p_m^b(i) \leq \epsilon$ et un débit global maximal $X(i)$.

- Après cela dans (5), l'algorithme détermine le/les i maximisant le débit global tel que $X = \max(X(i))$, s'il y a plusieurs i qui vérifient cela, alors prendre $S_m = \min(S_m(i))$ et retenir le i correspondant.
- Dans (6), l'algorithme en déduit les taux d'arrivée des clients optimaux. Le taux d'arrivée des clients optimal dans la micro est calculé à partir de la valeur de i optimale déterminée à l'étape (5). Notez que lorsque $i < i_{max}$, les clients restants dans la micro cell ($i_{max} \times \lambda^{clients} - i \times \lambda^{clients}$) seront ajoutés au taux d'arrivée des clients de la macro de la classe J_M ($\lambda_{M,J_M}^{clients}$), de sorte que le taux d'arrivée total $\lambda^{clients}$ reste constant. Le débit optimal obtenu en (5) correspond au scénario dans lequel la micro est déployée.
- Ainsi, dans (7), il est comparé au débit global lorsque la micro n'est pas déployée (déterminé dans (1)).

Algorithme 2 Détermination du taux d'arrivée des clients et du nombre de serveurs à attribuer à la micro cell pour atteindre le débit global optimal

Inputs : $i \neq 0, i_{max}, \lambda^{clients}, J_M, \lambda_{M,j}^{clients}$ ($j = 1, \dots, J_M$), $\epsilon, S, step$

Résultats : $S_m, S_M, \lambda_m^{clients}, \lambda_M^{clients}, X$

- 1: Calculer le débit global dans le cas ou la macro est déployée sans micro (X^*)
 - 2: **Tant que** $i \leq i_{max}$ **faire**
 - 2.1. Déterminer les taux d'arrivée des clients :

$$\lambda_m^{clients} = i \times \lambda^{clients}$$

$$\lambda_{M,J_M}^{clients} = \lambda_{M,J_M}^{clients} + (i_{max} \times \lambda^{clients} - \lambda_m^{clients})$$

$$\lambda_M^{clients} = \sum_{j=1}^{J_M-1} \lambda_{M,j}^{clients} + \lambda_{M,J_M}^{clients}$$
 - 2.2. Déterminer le/les $S_m^{**}(i)$ tel que $p_m^b(i) \leq \epsilon$
 - 2.3. Sélectionner le/les $S_m^*(i)$ parmi le/les $S_m^{**}(i)$ offrant le débit global maximal $X(i)$
 - 2.4. Déterminer le $S_m(i)$ minimal tel que $S_m(i) = \min(S_m^*(i))$
 - 3: $i = i + step$
 - 4: **Fin Tant que**
 - 5: Détermine le/les i maximisant le débit global tel que $X = \max(X(i))$, s'il y a plusieurs i qui vérifient cela, alors prendre $S_m = \min(S_m(i))$ et retenir le i correspondant
 - 6: Déduire :

$$\lambda_m^{clients} = i \times \lambda^{clients}$$

$$\lambda_{M,J_M}^{clients} = \lambda_{M,J_M}^{clients} + (i_{max} \times \lambda^{clients} - \lambda_m^{clients})$$

$$\lambda_M^{clients} = \sum_{j=1}^{J_M-1} \lambda_{M,j}^{clients} + \lambda_{M,J_M}^{clients}$$
 - 7: Déterminer le débit global optimal $X = \max(X^*, X)$

Si $X = X^*$ **alors**
 $S_m = 0; S_M = S; \lambda_m^{clients} = 0; \lambda_M^{clients} = \lambda^{clients}$
Fin Si
-

Selon la charge du trafic, le modèle d'optimisation basé sur l'un des modèles de file d'attente ($M/G/S/S$ et $M^{[X]}/G/S/S$) permet également de décider sur le moment d'activation/ désactivation

de la micro cell afin d’atteindre le débit optimal global.

Par ailleurs, grâce à l’heuristique proposée, le temps de résolution de notre algorithme d’optimisation pour déterminer la configuration optimale est nettement réduit par rapport à Global Search. En effet, par exemple pour un pourcentage de clients maximal à rattacher à la micro égal à $3\% \times \lambda^{clients}$, notre algorithme d’optimisation prend 25s lorsqu’il se base sur la $M/G/S/S$ (contre 2h pour Global Search) et 2h lorsqu’il se base sur la $M^{[X]}/G/S/S$ (contre 10h pour Global Search).

4.6.2 Application numérique du modèle d’optimisation de eICIC/ABS

Dans ce qui suit, nous comparons les résultats obtenus par l’algorithme [2](#) (en appliquant la $M/G/S/S$ et la $M^{[X]}/G/S/S$) et l’algorithme Global Search. Nous notons dans le Tableau [4.5](#), le pourcentage optimal d’utilisateurs acceptés pour être servis par la micro, $\lambda_m^{clients}$, et le nombre d’ABS optimal, n_{ABS} , à attribuer aux UE de la micro, afin d’atteindre le débit global optimal (X). Nous calculons par la suite le gain apporté par la micro, $X-X^*$, si son déploiement offre le débit optimal. Rappelons que le nombre optimal de serveurs attribués à la micro cell est $S_m = S \times \frac{n_{ABS}}{10}$ où $S = 500$ pour une bande passante de 20 MHz.

Nous considérons une distribution de clients réaliste dans la macro et la micro avec une distribution uniforme et une taille de paquet $size(u) = 1024$ bits, nous obtenons alors : $\lambda_{M,1}^{clients} = 50\% \times \lambda^{clients}$ des clients sont dans la région centrale (classe 1) de la macro, $i_{max} \times \lambda^{clients} = 3\% \times \lambda^{clients}$ des clients se situent au plus dans la région centrale (classe 1) de la micro (avec de bonnes conditions radio), $\lambda_{M,2}^{clients} = 47\% \times \lambda^{clients}$ des clients sont dans la région edge (classe 2) de la macro (avec de mauvaises conditions radio).

Nous limitons l’espace de recherche du nombre optimal de serveurs en limitant le nombre d’ABS possibles à 5 (la moitié du nombre de sous-trames disponibles dans une trame). Nous supposons que $i = 1\%$, $step = 1\%$ et $\epsilon = 0.02$ le but de cette dernière condition est de garantir que les UE de la micro soient servis à 98 %.

Notons que des valeurs réalistes d’efficacité du MCS de l’utilisateur par classe (extraites des simulations Matlab) sont utilisées : $CR_{M,1} \times \log_2(N_{M,1}) = 3.7746$, $CR_{M,2} \times \log_2(N_{M,2}) = 0.86$ et $CR_{m,1} \times \log_2(N_{m,1}) = 3.334$ (ces valeurs sont utilisées dans le scénario macro déployée sans micro et dans le scénario macro déployée avec une micro).

Les mêmes valeurs optimales sont obtenues par l’algorithme Global Search et par l’algorithme [2](#) que nous appliquons sur les deux modèles de file d’attente. Par conséquent, les résultats ne sont affichés qu’une seule fois dans le Tableau [4.5](#).

Pour chaque taux d’arrivée (ou densité d’utilisateur en kbits/ trame), un nombre optimal de n_{ABS} et un nombre optimal de UE associés à la micro peuvent être sélectionnés afin de maximiser le débit global en kbits/ trame (comme expliqué précédemment dans le problème d’optimisation).

— Lorsque le taux d’arrivée varie entre 46 et 275 kbits/trame, la macro et la micro sont peu

$\lambda^{clients}$	λ [kbits/trame]	n_{ABS} optimal	$\lambda_m^{clients}$ optimal	X optimal	$X-X^*$
50	46	1	$3\% \times \lambda^{clients}$	46	1
100	92	1	$3\% \times \lambda^{clients}$	92	2
200	183	1	$3\% \times \lambda^{clients}$	183	3
300	275	1	$3\% \times \lambda^{clients}$	271	1
400	361	0	$0\% \times \lambda^{clients}$	298	0
500	451	0	$0\% \times \lambda^{clients}$	299	0
600	541	0	$0\% \times \lambda^{clients}$	299	0
700	631	0	$0\% \times \lambda^{clients}$	300	0
800	734	1	$3\% \times \lambda^{clients}$	302	2
900	826	1	$3\% \times \lambda^{clients}$	306	6
1000	917	1	$3\% \times \lambda^{clients}$	309	9
1200	1101	1	$3\% \times \lambda^{clients}$	315	15
2500	2228	2	$3\% \times \lambda^{clients}$	328	28
2750	2450	2	$3\% \times \lambda^{clients}$	335	35
3000	2672	3	$3\% \times \lambda^{clients}$	340	40
4000	3563	3	$3\% \times \lambda^{clients}$	346	46

Tableau 4.5 – n_{ABS} optimal, $\lambda_m^{clients}$ optimal, débit global optimal en kbits/trame et gain ($X-X^*$) en kbits/trame selon le taux d'arrivée des clients obtenus par l'algorithme Global Search et l'algorithme

2

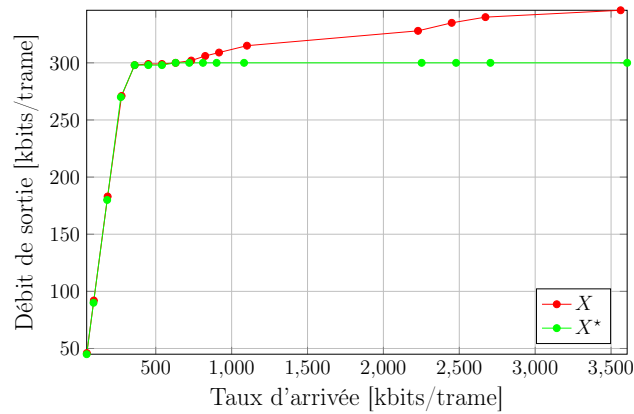


FIGURE 4.26 – Comparaison entre X (le débit global optimal) et X^* (le débit global sans déploiement de la micro) où $\epsilon = 0.02$

chargées. Les algorithmes proposent d'activer la micro cell et de servir tous les UE de la micro. Par conséquent, la macro alloue le nombre nécessaire d'ABS à la micro, car elle possède suffisamment de ressources pour servir ses propres UE. Notons que 1 ABS pour la micro est suffisant pour servir ses clients et pour garantir une probabilité de blocage dans la micro $p_m^b \leq 0.02$, en raison du faible pourcentage de clients dans la micro (3%). L'amélioration apportée suite au déploiement de la micro est faible et varie entre 1, 2 et 3 kbits/trame ceci est expliqué par le fait que le taux d'arrivée des clients dans la macro n'est pas assez important et donc le nombre de clients rattachés à la micro est faible. Par conséquent, le déploiement de la micro va certes améliorer le débit global (par rapport au scénario où elle n'est pas déployée) mais pas de façon remarquable. Ainsi, pour de telles charges et étant donné les gains qui sont clairement faibles, un compromis entre l'amélioration du débit global apportée suite au déploiement de la micro et par exemple sa consommation énergétique est à envisager.

- Lorsque le taux d'arrivée est supérieur à 361 et inférieur à 631 kbits/ trame, les modèles d'optimisation proposent de ne pas déployer la micro. Effectivement, pour ces taux d'arrivée de clients, la macro cell est chargée, contrairement à la micro. L'attribution d'un ABS à la micro va pénaliser les UE de la macro, ce qui diminue le débit global.
- Lorsque le taux d'arrivée est supérieur à 734 kbits/ trame et inférieur à 1101 kbits/trame, la macro est très chargée alors que la micro est modérément chargée. Le déploiement de la micro cell permettra d'améliorer le débit global, mais les algorithmes d'optimisation proposent de déléguer à la micro le minimum d'ABS dont elle a besoin pour atteindre le débit global optimal et en attribuant le nombre maximal de clients de sorte que $p_m^b \leq 0.02$. Par ailleurs, nous remarquons que pour de telles charges le déploiement de la micro permettrait d'apporter un gain (par rapport au scénario où elle n'est pas déployée) (15 kbits/trame pour $\lambda^{clients} = 1200$ clients/trame). Ceci est expliqué par le fait que pour de telles charges la micro si elle est déployée elle sera bien chargée et donc les clients qui lui seront rattachés seront servis avec de meilleures conditions radio ce qui engendre une amélioration assez importante du débit global.

Comme nous venons de voir pour ces taux d'arrivées, le nombre d'ABS optimal est soit 0 soit 1 parce que la micro est faiblement ou moyennement chargée. Toutefois, pour les charges supérieures (2500 clients/trame, 2750 clients/trame, 3000 clients/trame et 4000 clients/trame), le nombre d'ABS optimal est 2, 2, 3 et 3 ABS et le gain apporté suite au déploiement de la micro est de 28, 35, 40 et 46 kbits/trame, respectivement. L'analyse approfondie des probabilités de blocage dans la micro révèle que les configurations optimales d'ABS pour ces taux d'arrivée de clients sont dues à la contrainte $p_m^b \leq 0.02$. En effet, pour chaque taux d'arrivée de clients, le n_{ABS} optimal retourné par notre modèle d'optimisation représente le nombre d'ABS permettant de vérifier que la probabilité de blocage dans la micro est inférieur à $\epsilon = 0.02$ et assurant un débit global optimal.

Plus précisément, nous notons que pour $\lambda^{clients} = 2500$ clients/trame (resp. 2750 clients/trame),

l'attribution d'un ABS ($n_{ABS} = 1$) engendre une probabilité de blocage dans la micro $p_m^b = 0.36$ (resp. $p_m^b = 0.41$) éliminant ainsi la configuration $n_{ABS} = 1$. Par ailleurs, l'attribution de 2 ABS ($n_{ABS} = 2$) pour $\lambda^{clients} = 2500$ clients/trame (resp. 2750 clients/trame) implique que $p_m^b = 9E-04$ (resp. $p_m^b = 0.0072$).

Pour $\lambda^{clients} = 3000$ clients/trame, l'attribution d'un ABS ($n_{ABS} = 1$) et de deux ABS ($n_{ABS} = 2$) engendrent respectivement une probabilité de blocage dans la micro $p_m^b = 0.46$ et $p_m^b = 0.027$ éliminant ainsi la configuration $n_{ABS} = 1$ et $n_{ABS} = 2$. Par ailleurs, l'attribution de 3 ABS ($n_{ABS} = 3$) implique que $p_m^b = 1.96E-09$.

Pour $\lambda^{clients} = 4000$ clients/trame, l'attribution d'un ABS ($n_{ABS} = 1$) et de deux ABS ($n_{ABS} = 2$) engendrent respectivement une probabilité de blocage dans la micro $p_m^b = 0.589$ et $p_m^b = 0.1963$ éliminant ainsi la configuration $n_{ABS} = 1$ et $n_{ABS} = 2$. Par ailleurs, l'attribution de 3 ABS ($n_{ABS} = 3$) implique que $p_m^b = 0.001$.

Dans ce qui suit, nous nous intéressons à étudier l'influence d'épsilon sur notre modèle d'optimisation.

4.6.3 Influence d'épsilon sur le modèle d'optimisation de eICIC/ ABS

L'analyse approfondie des résultats obtenus par notre outil d'optimisation (voir Tableau 4.5) nous révèle que, selon le taux d'arrivée des clients, $\epsilon = 0.02$ influence sur le nombre d'ABS optimal. En effet, nous avons observé que l'augmentation du taux d'arrivée des clients engendre une augmentation du nombre d'ABS pour garantir la contrainte sur la probabilité de perte. Nous rappelons qu'une configuration donnée n'est pas retenue, même si elle offre un débit global optimal, si la probabilité de blocage correspondante est supérieure à la valeur d'épsilon choisie. Nous avons donc voulu généraliser et analyser davantage l'influence de la valeur d'épsilon sur le nombre d'ABS optimal.

Pour montrer plus concrètement l'influence de la valeur d'épsilon sur le nombre d'ABS optimal en fonction du taux d'arrivée des clients, dans la FIGURE 4.27, nous choisissons quatre valeurs d'épsilon : $\epsilon = 0.1$, $\epsilon = 0.02$, $\epsilon = 0.01$ et $\epsilon = 0.001$.

Dans le Tableau 4.6, nous montrons l'influence d'épsilon sur le gain apporté suite au déploiement de la micro (en nous basant sur les configurations optimales du nombre d'ABS donnée dans la FIGURE 4.27) en fonction du taux d'arrivée des clients.

Les résultats du Tableau 4.6 montrent que pour chaque taux d'arrivée des clients la réduction de la valeur d'épsilon engendre la réduction du gain de déploiement de la micro. Pour ces taux d'arrivée de clients élevés, la probabilité de blocage dans la micro est de plus en plus susceptible de ne pas vérifier la contrainte $p_m^b \leq \epsilon$ de notre problème d'optimisation (voir l'équation 4.34). Pour chaque taux d'arrivée des clients, l'augmentation du nombre d'ABS optimal implique la réduction du gain de déploiement de la micro. Par exemple, pour $\lambda^{clients} = 5000$ clients/trame, lorsque $\epsilon = 0.1$ le

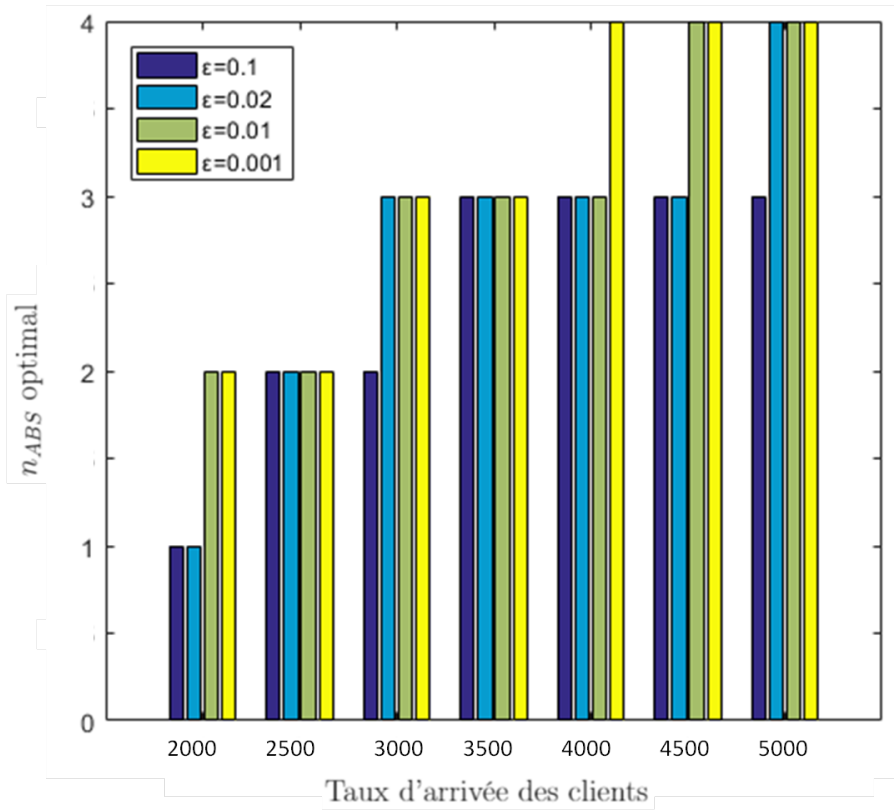


FIGURE 4.27 – Influence d'epsilon sur le nombre d'ABS optimal

$\lambda^{clients}$	Gain : $X - X^*$ [kbits/trame]			
	$\epsilon = 0.1$	$\epsilon = 0.02$	$\epsilon = 0.01$	$\epsilon = 0.001$
2000	17	17	11	11
2500	27	27	27	27
3000	42	40	40	40
3500	45	45	45	45
4000	46	46	46	21
4500	60	60	32	32
5000	68	48	48	48

Tableau 4.6 – Influence d'epsilon sur le gain apporté suite au déploiement optimale de la micro

déploiement de la micro en lui attribuant 3 ABS permettrait d’atteindre un gain en termes de débit global égal à 68 kbits/trame. Toutefois, pour la même charge et pour $\epsilon = 0.001$ le nombre optimal d’ABS permettant d’atteindre le débit optimal est de 4 avec un gain de 48 kbits/trame.

Ces résultats prouvent donc l’influence d’epsilon sur la configuration optimale (n_{ABS} optimal et λ_m optimal) permettant d’atteindre un débit global optimal.

La réduction des gains pour des epsilons faibles et des taux d’arrivée élevés est expliquée par le fait que l’algorithme d’optimisation attribue plus d’ABS à la micro afin d’assurer la probabilité de blocage. Toutefois, ceci engendre un sur-dimensionnement des ressources de la micro ce qui pénalise les utilisateurs de la macro et donc implique une réduction du débit global optimal et par conséquent une réduction du gain.

4.7 Conclusion

Ce chapitre est dédié à la modélisation de la technique de remédiation aux interférences eICIC/ABS. Nous y proposons nos deux premières contributions qui consistent en des modèles d’approximation pour la coordination de la macro et de la micro basées sur la théorie des files d’attente.

Le premier modèle est une approximation basée sur le modèle multi-serveurs $M/G/S/S$ avec des arrivées simples. Le second modèle est une approximation basée sur le modèle multi-serveurs avec des arrivées groupées (ou par batch) $M^{[X]}/G/S/S$. Le second modèle, plus complexe, permet en plus une différenciation de service. Les modèles proposés sont validés et comparés aux résultats de simulation reflétant le comportement réel du système. Tous les deux ont été analysés et étudiés en fonction de plusieurs paramètres.

La pertinence des modèles réside dans le fait que les paramètres d’entrée sont basés sur des distributions d’utilisateurs et des conditions radio réelles issus de la simulation des utilisateurs.

Les modèles de file d’attente proposés sont relativement simples à analyser et rapides à exploiter. Ils nous ont permis de proposer un algorithme d’optimisation capable d’adapter rapidement les paramètres eICIC/ABS en fonction de la charge des utilisateurs et de leurs conditions radio. Ce résultat valorise les modèles proposés et prouve la pertinence de les utiliser sur un système réel.

Les travaux présentés dans ce chapitre ont fait l’objet de deux publications internationales [TCTJ+18b] et [TCTJA19b] et d’une publication nationale [TCTJ+18a].

En termes de perspectives, il serait intéressant de comparer ces résultats avec des résultats obtenus avec des simulateurs à événements discrets tels que NS3 ou LTE-Sim et utilisant des distributions d’arrivées reflétant mieux les services réels (voix, données, vidéo). De plus, comme discuté dans le chapitre, un compromis entre l’amélioration apportée par le déploiement et l’activation des micro cellules et le coût énergétique serait intéressant à prendre en compte dans les algorithmes d’optimisation. Ceci peut être apporté dans l’algorithme [2].

Dans le chapitre suivant, nous chercherons à étendre ces résultats de modélisation à une autre technique de remédiation aux interférences : la technique JT-CoMP qui propose une approche dif-

férente basée sur l'exploitation des ressources radio de manière coordonnée par des cellules voisines.

Chapitre 5

Modélisation de JT-CoMP et allocation optimale de ressources dans les C-RAN

5.1 Problématique et contexte

Avec JT-CoMP, pour chaque utilisateur CoMP, les cellules coopératives transmettent la même information en utilisant la même ressource radio (fréquence et temps) comme expliqué dans la section [2.3.2.2.b](#). JT-CoMP a été démontré comme une approche efficace pour améliorer le débit des utilisateurs en périphérie, en particulier dans les environnements à fortes interférences mais ce mécanisme engendre une consommation accrue des ressources [Abb16a](#). Il est bien connu que les techniques CoMP ne sont avantageuses que pour les utilisateurs edge de la cellule [3GP08](#). Il a été démontré qu'à forte charge et dans un environnement à faibles interférences, ces techniques peuvent également être néfastes pour les utilisateurs centraux de la cellule, en raison de l'utilisation inefficace de ressources radio [3GP08](#). Ainsi, un compromis entre gain des conditions radio des UE CoMP et perte de disponibilité de ressources est indispensable. Notre objectif consiste donc à proposer des outils de dimensionnement à temps réel qui permettent une allocation ressources optimale, selon la charge du trafic, entre les utilisateurs CoMP et non-CoMP afin d'assurer un débit global optimal.

Une simple stratégie pour attribuer des ressources radio dans les systèmes CoMP, connue sous le nom d'allocation statique, consiste comme son nom l'indique à allouer une bande de fréquence dédiée aux utilisateurs CoMP, comme proposé par le 3GPP dans [3GP09a](#) [3GP09b](#). Ainsi, x % des ressources disponibles sont dédiées aux utilisateurs non-CoMP tandis que les ressources restantes sont attribuées aux utilisateurs CoMP. Néanmoins, cette approche engendrerait un gaspillage de ressources car le nombre d'utilisateurs dans les régions CoMP et non CoMP varie selon la charge du trafic. Dans [LCP⁺10](#), les auteurs proposent un algorithme de programmation d'équité proportionnelle conjointe (joint proportional fairness scheduling algorithm) qui traite les utilisateurs CoMP et non-CoMP de façon équitable sans aucun partitionnement de la bande fréquentielle. Dans [ABS16](#), les auteurs proposent deux stratégies d'allocation de ressources avec priorisation d'une des deux

catégorie des utilisateurs (priorité aux CoMP UE ou priorité aux non-CoMP UE) puis ils comparent ces stratégies avec la stratégie d'allocation statique [3GP09a] [3GP09b] et la stratégie d'allocation itérative [KBE13]. Ces comparaisons révèlent que la mobilité des UE améliore le débit moyen dans le cluster pour toutes les stratégies sauf lorsque les CoMP UE sont prioritaires. Sous cette stratégie, la mobilité conduit à une dégradation du débit. Cependant, donner la priorité aux utilisateurs non-CoMP en présence de la mobilité surpasse toutes les autres stratégies contrairement au scénario sans mobilité. Cela est dû au fait que les UE edge sont plus susceptibles d'être servis avec de bonnes conditions radio sans effectuer d'opérations de coopération, en minimisant ainsi le gaspillage de ressources. Cependant, cette stratégie diminue la condition de stabilité du système lorsque les utilisateurs sont statiques, car la condition de servir un utilisateur CoMP est plus restrictive que celle de servir un utilisateur non-CoMP en raison du fait que cela nécessite la disponibilité de toutes les cellules coopératives, et donc à forte charge les utilisateurs CoMP statiques ne seront jamais servis. Ainsi, nous concluons que d'après ces études : dans le scénario de gain de coordination faible, il n'est pas intéressant de perdre des ressources pour servir un utilisateur CoMP si cet utilisateur est en mouvement et peut bénéficier de meilleures conditions radio. Toutefois, cette étude nécessite une prédiction précise de la mobilité et un trafic qui tolère les délais.

Afin de répondre aux faiblesses rencontrées dans la littérature, nous nous intéressons dans ce chapitre à modéliser mathématiquement JT-CoMP dans le but de déterminer soigneusement et en temps réel le nombre optimal d'utilisateurs CoMP ainsi que le nombre optimal de ressources à dédier pour la coopération tout en assurant un débit global optimal.

Pour ce faire, nos contributions dans ce chapitre sont les suivantes :

- Proposer des chaînes de Markov pour la modélisation de JT-CoMP ;
- Proposer des modèles de files d'attente avec formes closes pour évaluer les performances de JT-CoMP ;
- Comparer les résultats numériques des modèles de files d'attente avec les résultats des chaînes de Markov où des conditions radio et des distributions d'utilisateurs réalistes ont été considérées ;
- Proposer des outils de dimensionnement qui offrent aux opérateurs les décisions sur l'attribution des ressources entre les utilisateurs CoMP et non-CoMP ainsi que le nombre d'UE à servir en coopération pour atteindre le débit global optimal du système de façon rapide et adaptative.

Nous pouvons résumer notre démarche pour les contributions sus-citées par le schéma illustré dans la FIGURE 5.1

Comme illustré dans la FIGURE 5.1, en entrée des chaînes de Markov et des modèles de file d'attente proposés, nous utilisons des conditions radio et des distributions d'utilisateurs réalistes. Ces paramètres sont exploités par ces approches afin de considérer les mêmes conditions radio des UE lors du calcul des probabilités de blocage et des débits dans les régions centrales des macro cellules

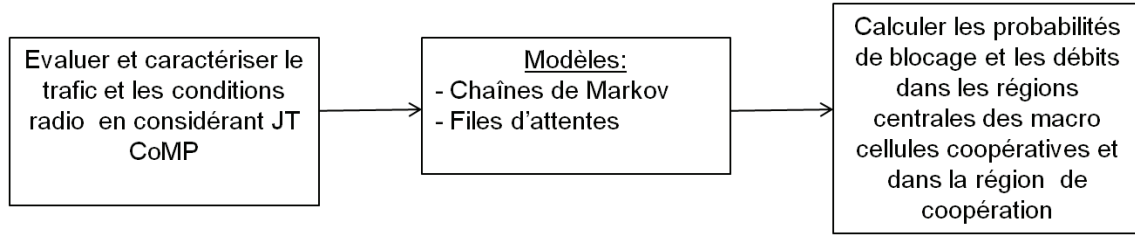


FIGURE 5.1 – Résumé de notre démarche pour la modélisation de JT-CoMP

coopératives et dans la région de coopération.

5.2 Formulation du problème

Nous considérons un C-RAN cluster composé de 2 cellules centrales et de ses 8 cellules voisines du premier tier (cellules qui sont directement autour d’elles), numérotés de 3 à 10 comme illustrée dans la FIGURE 5.2.

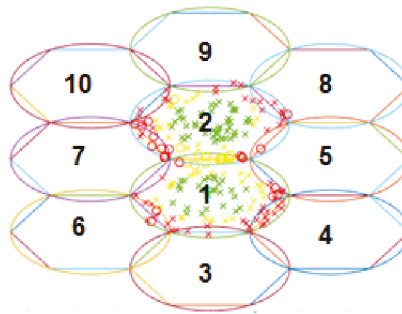


FIGURE 5.2 – C-RAN cluster : 10 macro cells

Toutes ces cellules sont des macro cells. Dans chaque macro cell, il y a un nombre total de S ressources (ou RBG) pouvant être servis dans une trame. Rappelons que la valeur de S correspond au rapport entre le nombre total de PRB dans la trame (voir Tableau 2.1) et le nombre de PRB dans un RBG (voir Tableau 2.3). Comme déjà expliqué dans la section 2.3.1.2.a, ces paramètres dépendent de la bande passante considérée.

Nous focalisons notre analyse sur les utilisateurs dans la région centrale, couverte par les deux macro cells centrales (macro M1 et macro M2). Nous supposons qu’uniquement ces deux macro cells coopèrent. En présence d’utilisateurs CoMP, les S ressources disponibles dans la macro M1 et la macro M2 sont partagées entre utilisateurs CoMP et non-CoMP. De plus, lorsque JT-CoMP est appliqué les cellules coopératives doivent réserver le même nombre de ressources, que nous notons S_c , pour servir conjointement les utilisateurs CoMP. Nous notons S_{M1} (resp. S_{M2}) le nombre de ressources disponibles pour servir les utilisateurs non-CoMP de la macro M1 (resp. de la macro M2) ce qui fait, $S_{M1} = S_{M2}$. Par simplification, nous définissons $S_M = S_{M1} = S_{M2}$ le nombre de

ressources disponibles pour servir les utilisateurs non-CoMP. Par conséquent, dans chaque macro, nous avons : $S = S_M + S_c$.

De plus, nous considérons les interférences de toutes les autres cellules du cluster afin de déterminer les conditions radio des UE. Les conditions radio d'un UE sont exprimées par le nombre de RBG requis pour transmettre un paquet. Nous classifions les UE selon le nombre de RBG qu'ils demandent et nous dirons que les UE de la classe i requièrent i RBG. Nous considérons le cas général où les UE sont aléatoirement distribués dans la région centrale et nous supposons que les clients arrivent selon un processus aléatoire avec un taux $\lambda^{clients} = \lambda_{M1}^{clients} + \lambda_{M2}^{clients} + \lambda_c^{clients}$, par durée de trame, tel que :

- $\lambda_{M1}^{clients}$ est le taux d'arrivée des clients dans la région centrale de la macro M1
- $\lambda_{M2}^{clients}$ est le taux d'arrivée des clients dans la région centrale de la macro M2
- $\lambda_c^{clients}$ est le taux d'arrivée des clients dans la région de coopération entre la macro M1 et de la macro M2

Nous notons :

- $\lambda_{M1,i}^{clients}$: le taux d'arrivée de clients dans la classe i de la région centrale de la macro M1
- $\lambda_{M2,i}^{clients}$: le taux d'arrivée de clients dans la classe i de la région centrale de la macro M2
- $\lambda_{c,i}^{clients}$: le taux d'arrivée de clients dans la classe i de la région de coopération entre la macro M1 et de la macro M2
- J_{M1} : le nombre de classes dans la région centrale de la macro M1
- J_{M2} : le nombre de classes dans la région centrale de la macro M2
- J_c : le nombre de classes dans la région de coopération entre la macro M1 et de la macro M2

Un processus d'arrivée poissonien est l'hypothèse la plus couramment utilisée en ingénierie du trafic pour la répartition des arrivées des clients. Cette hypothèse (également présente dans le réseau téléphonique classique) est raisonnable car nous considérons un nombre important d'utilisateurs qui génèrent des requêtes et se comportent de manière indépendante.

Notre problème, visant à prévoir l'attribution des ressources entre les UE CoMP et les UE non-CoMP et le nombre d'UE à servir en coopération ($\lambda_c^{clients}$) afin de maximiser le débit global $X = (X_{M1} + X_{M2} + X_c)$, peut être formulé globalement comme suit :

$$\begin{aligned}
& \max_{S_c, S_M, \lambda_c^{clients}, \lambda_M^{clients}} (X_{M1} + X_{M2} + X_c) \\
& \text{s.t.} \quad S_c + S_{M1} = S, \\
& \quad \quad S_c + S_{M2} = S, \\
& \quad \quad \lambda_{M1}^{clients} + \lambda_{M2}^{clients} + \lambda_c^{clients} = \lambda^{clients}, \\
& \quad \quad S_M > 0, \\
& \quad \quad S_c \geq 0
\end{aligned} \tag{5.1}$$

où :

- X_{M1} est le débit de sortie moyen de la macro M1 région de coopération non comprise, exprimé en kbits/trame ;
- X_{M2} est le débit de sortie moyen de la macro M2 région de coopération non comprise, exprimé en kbits/trame ;
- X_c est le débit de sortie moyen de la région de coopération entre la macro M1 et la macro M2, exprimé en kbits/trame.

5.3 Chaîne de Markov pour la modélisation de JT-CoMP

Nous résumons dans un premier temps tous les paramètres nécessaires pour la compréhension de cette section dans le Tableau [5.1](#).

Dans ces modèles, chaque batch qui arrive représente le nombre de RBG nécessaires pour la transmission d'un paquet du UE (appelé aussi client) selon ses conditions radio comme illustrée dans la FIGURE [4.3](#). Cette figure représente un exemple d'un client avec une taille de batch égal à 2 RBG.

Nous notons :

- $A_{M1,i}(n)$: le nombre de clients, appartenant à la classe i de la région centrale de la macro M1, qui arrivent à la trame n
- $A_{M2,i}(n)$: le nombre de clients, appartenant à la classe i de la région centrale de la macro M2, qui arrivent à la trame n
- $A_{c,i}(n)$: le nombre de clients, appartenant à la classe i de la région de coopération, qui arrivent à la trame n
- $i \times A_{M1,i}(n)$: le nombre de RBG requis à la trame n par les clients de la classe i de la macro M1
- $i \times A_{M2,i}(n)$: le nombre de RBG requis à la trame n par les clients de la classe i de la macro M2
- $i \times A_{c,i}(n)$: le nombre de RBG requis à la trame n par les clients de la classe i de la région de coopération

Ainsi :

- $\sum_{i=1}^{J_{M1}} i \times A_{M1,i}(n)$ est le nombre total de RBG requis dans la macro M1
- $\sum_{i=1}^{J_{M2}} i \times A_{M2,i}(n)$ est le nombre total de RBG requis dans la macro M2
- $\sum_{i=1}^{J_c} i \times A_{c,i}(n)$ est le nombre total de RBG requis dans la région de coopération

Paramètre	Explication
$\lambda^{clients}$	Taux d'arrivée des clients dans la macro M1 et la macro M2
$\lambda_{M1}^{clients}$	Taux d'arrivée des clients dans la région centrale de la macro M1 (toutes classes confondues)
$\lambda_{M2}^{clients}$	Taux d'arrivée des clients dans la région centrale de la macro M2 (toutes classes confondues)
$\lambda_c^{clients}$	Taux d'arrivée des clients dans la région de coopération entre la macro M1 et la macro M2 (toutes classes confondues)
$\lambda_{M1,i}^{clients}$	Taux d'arrivée des clients dans la classe i de la macro M1 (région de coopération non comprise)
$\lambda_{M2,i}^{clients}$	Taux d'arrivée des clients dans la classe i de la macro M2 (région de coopération non comprise)
$\lambda_c^{clients}$	Taux d'arrivée des clients dans la région de coopération (entre les macro M1 et M2)
λ	Taux d'arrivée dans la macro M1 et la macro M2 exprimé en kbits/trame
λ_{M1}	Taux d'arrivée dans la macro M1 (toutes classes confondues, région de coopération non comprise) exprimé en kbits/trame
λ_{M2}	Taux d'arrivée dans la macro M2 (toutes classes confondues, région de coopération non comprise) exprimé en kbits/trame
λ_c	Taux d'arrivée des clients dans la région de coopération (toutes classes confondues) exprimé en kbits/trame
$\lambda_{M1,i}$	Taux d'arrivée des clients dans la classe i de la macro M1 (région de coopération non comprise) exprimé en kbits/trame
$\lambda_{M2,i}$	Taux d'arrivée des clients dans la classe i de la macro M2 (région de coopération non comprise) exprimé en kbits/trame
$\lambda_{c,i}$	Taux d'arrivée des clients dans la classe i de la région de coopération (entre M1 et M2) exprimé en kbits/trame
J_{M1}	Nombre de classes dans la macro M1 (région de coopération non comprise)
J_{M2}	Nombre de classes dans la macro M2 (région de coopération non comprise)
J_c	Nombre de classes dans la région de coopération (entre M1 et M2)
$A_{M1,i}(n)$	Nombre de clients, appartenant à la classe i de la macro M1, qui arrivent à la trame n (région de coopération non comprise)
$A_{M2,i}(n)$	Nombre de clients, appartenant à la classe i de la macro M2, qui arrivent à la trame n (région de coopération non comprise)
$A_{c,i}(n)$	Nombre de clients, appartenant à la classe i de la région de coopération (entre M1 et M2), qui arrivent à la trame n
$i \times A_{M1,i}(n)$	Nombre de RBG requis à la trame n par les clients de la classe i de la macro M1 (région de coopération non comprise)
$i \times A_{M2,i}(n)$	Nombre de RBG requis à la trame n par les clients de la classe i de la macro M2 (région de coopération non comprise)
$i \times A_{c,i}(n)$	Nombre de RBG requis à la trame n par les clients de la classe i de la région de coopération
S	Capacité de service (nombre de RBG pouvant être servis dans une trame)
S_M	Capacité de service des UE non-CoMP de la macro M1 ou de la macro M2
S_c	Capacité de service des UE CoMP entre la macro M1 et la macro M2
$Y^{M1}(n)$	Nombre de RBG occupant la macro M1 (région de coopération non comprise)
$Y^{M2}(n)$	Nombre de RBG occupant la macro M2 (région de coopération non comprise)
$Y^c(n)$	Nombre de RBG occupant la région de coopération

X_{M1}^{RBG}	Débit de sortie moyen de la macro M1 exprimé en RBG/trame (région de coopération non comprise)
X_{M2}^{RBG}	Débit de sortie moyen de la macro M2 exprimé en RBG/trame (région de coopération non comprise)
X_c^{RBG}	Débit de sortie moyen de la région de coopération exprimé en RBG/trame
r_{M1}	Taille moyenne des batchs de la macro M1 (région de coopération non comprise)
r_{M2}	Taille moyenne des batchs de la macro M2 (région de coopération non comprise)
r_c	Taille moyenne des batchs de la région de coopération
g_{M1}	Distribution de proba de la taille des batchs dans la macro M1 (région de coopération non comprise)
g_{M2}	Distribution de proba de la taille des batchs dans la macro M2 (région de coopération non comprise)
g_c	Distribution de proba de la taille des batchs dans la région de coopération
$X_{M1}^{clients}$	Débit de sortie moyen de la macro M1 exprimé en clients/trame (région de coopération non comprise)
$X_{M2}^{clients}$	Débit de sortie moyen de la macro M2 exprimé en clients/trame (région de coopération non comprise)
$X_c^{clients}$	Débit de sortie moyen de la région de coopération exprimé en clients/trame
$CR_{M1,i}$	Taux de codage moyen des clients de la classe i de la macro M1 (région de coopération non comprise)
$CR_{M2,i}$	Taux de codage moyen des clients de la classe i de la macro M2 (région de coopération non comprise)
$CR_{c,i}$	Taux de codage moyen des clients de la classe i de la région de coopération
$N_{M1,i}$	Nombre moyen des points de constellation des clients de la classe i de la macro M1 (région de coopération non comprise)
$N_{M2,i}$	Nombre moyen des points de constellation des clients de la classe i de la macro M2 (région de coopération non comprise)
$N_{c,i}$	Nombre moyen des points de constellation des clients de la classe i de la région de coopération
p_{M1}^b	Probabilité de blocage dans la macro M1 (région de coopération non comprise)
p_{M2}^b	Probabilité de blocage dans la macro M2 (région de coopération non comprise)
p_c^b	Probabilité de blocage dans la région de coopération
X_{M1}^{bits}	Débit de sortie moyen de la macro M1 exprimé en bits/trame région de coopération non comprise
X_{M2}^{bits}	Débit de sortie moyen de la macro M2 exprimé en bits/trame région de coopération non comprise
X_c^{bits}	Débit de sortie moyen de la région de coopération exprimé en bits/trame
X^{bits}	Débit de sortie global exprimé en bits/trame
X_{M1}	Débit de sortie moyen de la macro M1 exprimé en kbits/trame région de coopération non comprise
X_{M2}	Débit de sortie moyen de la macro M2 exprimé en kbits/trame région de coopération non comprise
X_c	Débit de sortie moyen de la région de coopération exprimé en kbits/trame
X	Débit de sortie global exprimé en kbits/trame

Tableau 5.1 – Paramètres des chaînes de Markov pour la modélisation de JT-CoMP

Dans chaque macro cell, il y a un nombre total de S RBG pouvant être servis dans une trame. S représente donc la capacité de service. Comme déjà expliqué dans la section précédente, en présence des UE CoMP, la macro M1 et la macro M2 réservent le même nombre de ressources S_c pour servir ces UE et le reste des ressources disponibles vont être utilisées pour servir les UE non-CoMP. Durant chaque trame, nous devons considérer (i) les arrivées de clients non-CoMP, (ii) les arrivées des clients CoMP (iii) la capacité de service des clients non-CoMP de chaque macro coopérative, S_M RBG et (iv) la capacité de service des clients CoMP, S_c RBG.

La technique JT-CoMP peut alors être représentée par trois chaînes de Markov à temps discret, où l'unité de temps est la durée de la trame : une pour chaque macro et une pour la région de coopération. Soit $Y^{M1}(n)$ (resp. $Y^{M2}(n)$ et $Y^c(n)$) le nombre de RBG occupant la macro M1 (resp. la macro M2 et la région de coopération) à la trame n . A la fin de chaque trame n , le modèle de chaîne de Markov sert les clients arrivés à la trame $n - 1$ et accepte les clients arrivés à la trame courante (c'est à dire à la trame n). Ainsi, $Y^{M1}(n)$, $Y^{M2}(n)$ et $Y^c(n)$ pour $n \geq 0$ représentent des chaînes de Markov à temps discret. Les équations d'évolution définissant les chaînes de Markov sont données comme suit :

Pour la macro M1 :

$$Y^{M1}(n) = \min(S_M, [Y^{M1}(n-1) - S_M]^+ + \sum_{j=1}^{J_{M1}} j \times A_{M1,j}(n)) \quad (5.2)$$

où $Y^{M1}(0) = \min(S_M, \sum_{j=1}^{J_{M1}} j \times A_{M1,j}(0))$,

Pour la macro M2 :

$$Y^{M2}(n) = \min(S_M, [Y^{M2}(n-1) - S_M]^+ + \sum_{j=1}^{J_{M2}} j \times A_{M2,j}(n)) \quad (5.3)$$

où $Y^{M2}(0) = \min(S_M, \sum_{j=1}^{J_{M2}} j \times A_{M2,j}(0))$

Et pour la région de coopération :

$$Y^c(n) = \min(S_c, [Y^c(n-1) - S_c]^+ + \sum_{j=1}^{J_c} j \times A_{c,j}(n)) \quad (5.4)$$

où $Y^c(0) = \min(S_c, \sum_{j=1}^{J_c} j \times A_{c,j}(0))$.

Etant donné que ces chaînes de Markov considèrent des arrivées par batch et un nombre important de serveurs, le nombre de transitions générées est très élevé et il est donc difficile de les résoudre numériquement. Par conséquent, nous avons décidé de simuler les chaînes de Markov sur un nombre "T" de trames afin de calculer les probabilités de blocage et les débits de sorties.

En considérant T trames, nous déduisons $X_{M1}^{RBG}(T)$, $X_{M2}^{RBG}(T)$ et $X_c^{RBG}(T)$, le débit moyen de sortie pour la macro M1, la macro M2 et la région de coopération respectivement exprimés en RBG/trame, comme suit :

$$X_{M1}^{RBG}(T) = \frac{\sum_{t=1}^T Y^{M1}(t)}{T} \quad (5.5)$$

$$X_{M2}^{RBG}(T) = \frac{\sum_{t=1}^T Y^{M2}(t)}{T} \quad (5.6)$$

$$X_c^{RBG}(T) = \frac{\sum_{t=1}^T Y^c(t)}{T} \quad (5.7)$$

Le débit de sortie de la macro M1 (resp. la macro M2 et la région de coopération) en termes de clients/trame représente donc le rapport entre $X_{M1}^{RBG}(T)$ (resp. $X_{M2}^{RBG}(T)$ et $X_c^{RBG}(T)$) et la taille moyenne des batchs dans la macro M1, r_{M1} (resp. la macro M2, r_{M2} et la région de coopération r_c). Nous choisissons évidemment une valeur de T suffisamment grande pour la convergence de $X_{M1}^{RBG}(T)$, $X_{M2}^{RBG}(T)$ et $X_c^{RBG}(T)$ et nous répétons les simulations afin d'assurer un intervalle de confiance à 95%. Ainsi, les débits de sortie de la macro M1, de la macro M2 et de la région de coopération en termes de clients/trame sont calculés comme suit :

$$X_{M1}^{clients} = \lim_{T \rightarrow \infty} \frac{X_{M1}^{RBG}(T)}{r_{M1}} \quad (5.8)$$

$$X_{M2}^{clients} = \lim_{T \rightarrow \infty} \frac{X_{M2}^{RBG}(T)}{r_{M2}} \quad (5.9)$$

$$X_c^{clients} = \lim_{T \rightarrow \infty} \frac{X_c^{RBG}(T)}{r_c} \quad (5.10)$$

où $r_{M1} = \sum_{j=1}^{J_{M1}} j \times g_{M1,j}$ avec $g_{M1,j} = \frac{\lambda_{M1,j}^{clients}}{\lambda_{M1}^{clients}}$ (resp. $r_{M2} = \sum_{j=1}^{J_{M2}} j \times g_{M2,j}$ avec $g_{M2,j} = \frac{\lambda_{M2,j}^{clients}}{\lambda_{M2}^{clients}}$ et $r_c = \sum_{j=1}^{J_c} j \times g_{c,j}$ avec $g_{c,j} = \frac{\lambda_{c,j}^{clients}}{\lambda_c^{clients}}$) est la taille moyenne des batchs dans la macro M1 (resp. la macro M2 et la région de coopération) qui est supposée être finie. Les distributions des tailles des batchs vérifient $\sum_{j=1}^{J_{M1}} g_{M1,j} = \sum_{j=1}^{J_{M2}} g_{M2,j} = \sum_{j=1}^{J_c} g_{c,j} = 1$.

Afin de déduire le débit de sortie binaire, nous devons considérer la distribution des tailles des batchs et les paramètres radio (c'est à dire le nombre de PRB dans un RBG, le nombre d'OFDM symboles par PRB, le nombre de sous-porteuses par PRB, le taux de codage moyen par classe, le nombre moyen de points de constellation par classe), nous le calculons comme suit :

$$X_{M1}^{bits} = \lim_{T \rightarrow \infty} X_{M1}^{clients}(T) \times \left(\sum_{i=1}^{J_{M1}} i \times g_{M1,i} \times N_{PRB} \times 7 \times 12 \times CR_{M1_i} \times \log_2(N_{M1,i}) \right) \quad (5.11)$$

$$X_{M2}^{bits} = \lim_{T \rightarrow \infty} X_{M2}^{clients}(T) \times \left(\sum_{i=1}^{J_{M2}} i \times g_{M2,i} \times N_{PRB} \times 7 \times 12 \times CR_{M2_i} \times \log_2(N_{M2,i}) \right) \quad (5.12)$$

$$X_c^{bits} = \lim_{T \rightarrow \infty} X_c^{clients}(T) \times \left(\sum_{i=1}^{J_c} i \times g_{c,i} \times N_{PRB} \times 7 \times 12 \times CR_{c_i} \times \log_2(N_{c,i}) \right) \quad (5.13)$$

où $CR_{M1,i}$ et $N_{M1,i}$ (resp. $CR_{M2,i}$ et $N_{M2,i}$ et $CR_{c,i}$ et $N_{c,i}$) représentent le taux de codage moyen et le nombre moyen de points de constellation dans la classe i de la macro $m1$ (resp. de la macro $M2$ et de la région de coopération).

Le débit de sortie global en termes de kbits/trame est $X = \frac{X_{M1}^{bits} + X_{M2}^{bits} + X_c^{bits}}{1024}$.

Quant à la probabilité de blocage pour la macro $M1$, elle peut être calculée comme suit :

$$p_{M1}^b = \frac{\lambda_{M1}^{clients} - X_{M1}^{clients}}{\lambda_{M1}^{clients}} \quad (5.14)$$

pour la macro $M2$:

$$p_{M2}^b = \frac{\lambda_{M2}^{clients} - X_{M2}^{clients}}{\lambda_{M2}^{clients}} \quad (5.15)$$

et celle pour la région de coopération :

$$p_c^b = \frac{\lambda_c^{clients} - X_c^{clients}}{\lambda_c^{clients}} \quad (5.16)$$

Les probabilités de blocage de la macro $M1$ (resp. la macro $M2$ et de la région de coopération) sont aussi calculées pour chaque classe i en considérant les débits par classe dans les formules de probabilités de blocage ci-dessus.

La résolution numérique (simulations Matlab) de ces chaînes de Markov pour un taux d'arrivée de clients $\lambda^{clients} \leq 100$ clients/trame prend 1/2 heure au minimum (plus le taux d'arrivée des clients est supérieur, plus le temps de simulation est grand) afin d'atteindre un intervalle de confiance à 95%. Par conséquent, les chaînes de Markov ne peuvent pas être utilisées comme outils de configuration en temps réel pour la technique JT-CoMP. Dès lors, nous explorons dans ce qui suit des modèles de files d'attente qui permettraient de remédier à ces faiblesses.

5.4 Modèles de file d'attente pour la modélisation de JT-CoMP

Dans cette section, nous identifions les modèles de files d'attente qui pourraient refléter le mécanisme JT-CoMP et qui donnent de préférence des formes close plus rapides à exploiter. Résumons les spécificités de nos chaînes de Markov :

- chaînes de Markov à temps discret où l'unité de temps est la trame
- les clients arrivent indépendamment selon un processus de Poisson avec un taux $\lambda^{clients}$, par trame
 \Rightarrow arrivées poissonniennes
- chaque client qui arrive requiert, selon ses conditions radio et la taille de son paquet, un certain nombre de RBG, le client est donc vu comme un batch (composé de RBG)

- ⇒ arrivées par batch
- nous avons S RBG pouvant être servis dans une trame, ce qui correspond à une capacité de service S par trame
 - ⇒ S serveurs
- chaque client d'une classe donnée va occuper simultanément, par trame, autant de RBG qu'il demande. De plus, nous supposons une acceptation partielle dans les chaînes de Markov tel que lorsqu'un client demande un nombre de RBG supérieur au nombre de RBG disponibles, une partie de ce client sera accepté et le reste sera rejeté
 - ⇒ loi de service générale plus précisément déterministe et absence de files d'attente
- les clients sont servis selon leur ordre d'arrivée
 - ⇒ discipline de service FCFS

Nous souhaitons modéliser le système en utilisant des files d'attente en considérant autant que possible ces spécificités. La première différence entre ces approches c'est que les chaînes de Markov sont à temps discret alors que les modèles de files d'attente sont à temps continu

Par ailleurs, en nous basant sur l'analyse ci-dessus des chaînes de Markov, nous nous intéressons donc aux modèles de files d'attente multi-serveurs avec pertes. Pour cette catégorie, nous distinguons intuitivement deux files d'attente : la $M/G/S/S$ et la $M^{[X]}/G/S/S$. La $M/G/S/S$ considère des arrivées exponentielles et la $M^{[X]}/G/S/S$ des arrivées exponentielles par batch (voir chapitre 3 pour plus de détails sur ces deux modèles et le chapitre 4 pour la comparaison de ces deux modèles).

5.4.1 Modèle de file d'attente : $M/G/S/S$ pour la modélisation de JT-CoMP

Nous considérons la $M/G/S/S$ avec des arrivées exponentielles de paramètre $\lambda^{clients}$, et un temps de service déterministe. Dans ce modèle, chaque client occupe un serveur. Nous supposons que l'unité de temps est la trame. Dans chaque macro, le nombre de RBG pouvant être servis dans une trame sera représenté par S serveurs qui vont être partagés entre les utilisateurs CoMP et les utilisateurs non-CoMP. De plus, lorsque JT-CoMP est appliqué, la macro M1 et la macro M2 doivent réserver le même nombre de ressources, qu'on note S_c , pour servir conjointement les utilisateurs CoMP (voir la FIGURE 5.3).

Ainsi, nous avons trois $M/G/S/S$, une pour la région de coopération avec S_c serveurs et deux autres pour la macro M1 et la macro M2 avec S_M serveurs. Par ailleurs, afin de compenser l'effet de tailles variables des clients tout en utilisant un temps de service déterministe, nous proposons de déterminer une valeur moyenne pondérée du temps de service, en suivant la répartition des tailles des batchs dans la région de coopération ou dans la région de non-coopération (région centrale de la macro M1 ou de la macro M2). Par conséquent, si dans le mécanisme JT-CoMP, un client occupe j ressources (ou RBG) pendant une trame, dans le modèle de file d'attente $M/G/S/S$ proposé, le même client occupera un RBG pendant j trames consécutives. Ainsi, le temps de service de la macro

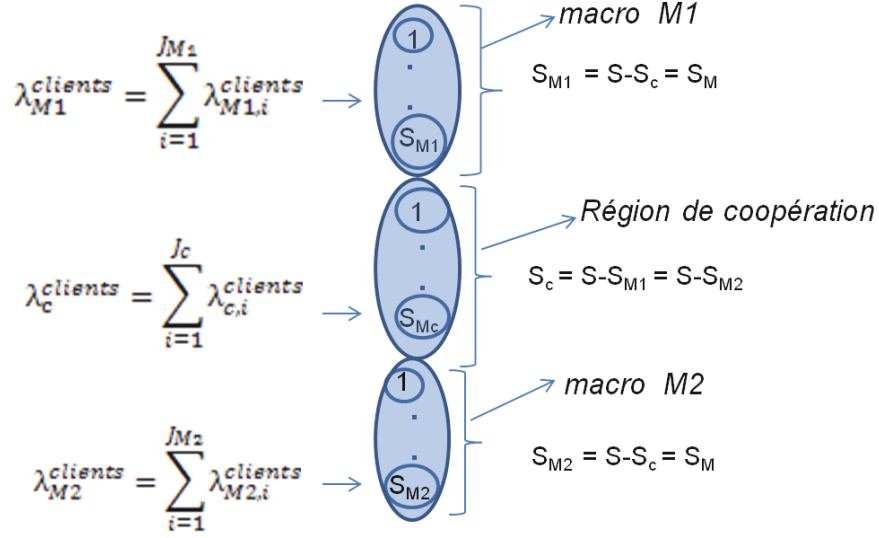


FIGURE 5.3 – Arrivée des clients et partage de ressources entre les utilisateurs CoMP et non-CoMP

M1 (resp. la macro M2) est $\frac{1}{\mu_{M1}} = \sum_{i=1}^{J_{M1}} i \times g_{M1,i}$ (resp. $\frac{1}{\mu_{M2}} = \sum_{i=1}^{J_{M2}} i \times g_{M2,i}$) et dans la région de coopération $\frac{1}{\mu_c} = \sum_{i=1}^{J_c} i \times g_{c,i}$.

La file $M/G/S/S$ permet d'utiliser la formule d'Erlang B (voir le chapitre 3) afin de déterminer la probabilité de blocage des clients dans la serving cell puis de calculer le débit des utilisateurs non-CoMP (dans la macro M1 et la macro M2) et le débit des utilisateurs CoMP (dans la région de coopération) ainsi que le débit global dans la région centrale du cluster considérée.

Soit p_{M1}^b (resp. p_{M2}^b et p_c^b) la probabilité de blocage dans la macro M1 (resp. la macro M2 et la région de coopération) exprimée comme suit en utilisant la formule d'Erlang B :

$$p_{M1}^b = \frac{\frac{\rho_{M1}^{S_{M1}}}{S_{M1}!}}{\sum_{n=0}^{S_{M1}} \frac{\rho_{M1}^n}{n!}} \quad (5.17)$$

où $\rho_{M1} = \frac{\lambda_{M1}}{\mu_{M1}}$, $\frac{1}{\mu_{M1}} = \sum_{i=1}^{J_{M1}} i \times g_{M1,i}$

$$p_{M2}^b = \frac{\frac{\rho_{M2}^{S_{M2}}}{S_{M2}!}}{\sum_{n=0}^{S_{M2}} \frac{\rho_{M2}^n}{n!}} \quad (5.18)$$

où $\rho_{M2} = \frac{\lambda_{M2}}{\mu_{M2}}$, $\frac{1}{\mu_{M2}} = \sum_{i=1}^{J_{M2}} i \times g_{M2,i}$

$$p_c^b = \frac{\frac{\rho_c^{S_c}}{S_c!}}{\sum_{n=0}^{S_c} \frac{\rho_c^n}{n!}} \quad (5.19)$$

où $\rho_c = \frac{\lambda_c}{\mu_c}$, $\frac{1}{\mu_c} = \sum_{i=1}^{J_c} i \times g_{c,i}$

Le débit de sortie en termes de clients/trame, $X_{M1}^{clients}$, pour la macro M1 est calculé par :

$$X_{M1}^{clients} = \lambda_{M1}^{clients} \times (1 - p_{M1}^b) \quad (5.20)$$

celui pour la macro M2 par :

$$X_{M2}^{clients} = \lambda_{M2}^{clients} \times (1 - p_{M2}^b) \quad (5.21)$$

et celui pour la région de coopération est calculé par :

$$X_c^{clients} = \lambda_c^{clients} \times (1 - p_c^b) \quad (5.22)$$

Les débits correspondants en termes de bits/trame sont calculés respectivement par :

$$X_{M1,j}^{bits} = X_{M1}^{clients} \times j \times g_{M1,j} \times N_{PRB} \times 7 \times 12 \times CR_{M1,j} \times \log_2(N_{M1,j}) \quad (5.23)$$

et donc :

$$X_{M1}^{bits} = \sum_{j=1}^{J_{M1}} X_{M1,j}^{bits} \quad (5.24)$$

pour la macro M2, nous avons :

$$X_{M2,j}^{bits} = X_{M2}^{clients} \times j \times g_{M2,j} \times N_{PRB} \times 7 \times 12 \times CR_{M2,j} \times \log_2(N_{M2,j}) \quad (5.25)$$

et donc :

$$X_{M2}^{bits} = \sum_{j=1}^{J_{M2}} X_{M2,j}^{bits} \quad (5.26)$$

et pour la région de coopération :

$$X_{c,j}^{bits} = X_c^{clients} \times j \times g_{c,j} \times N_{PRB} \times 7 \times 12 \times CR_{c,j} \times \log_2(N_{c,j}) \quad (5.27)$$

ce qui fait :

$$X_c^{bits} = \sum_{j=1}^{J_c} X_{c,j}^{bits} \quad (5.28)$$

Le débit de sortie global en termes de kbits/trame est :

$$X = \frac{X_{M1}^{bits} + X_{M2}^{bits} + X_c^{bits}}{1024} \quad (5.29)$$

5.4.2 Modèle de file d'attente : $M^{[X]}/G/S/S$ pour la modélisation de JT-CoMP

Nous proposons une approche d'approximation, la $M^{[X]}/G/S/S$, utilisant une forme close en tenant compte des arrivées par batch et des pertes avec plusieurs classes de services. Le raisonnement, expliqué dans la section 3.4.2, est appliqué pour la macro M1, la macro M2 et la région de coopération. Toutefois, pour utiliser la même terminologie que celle de la section précédente, nous considérons que chaque batch de RBG représente un client qui arrive et nous dirons qu'un client est de classe i si la taille de son batch est égale à i , et nous appellerons les composants d'un batch d'un client (UE) par : unités du batch.

Nous distinguons trois files $M^{[X]}/G/S/S$, une pour la région de coopération et deux pour la macro M1 et la macro M2. Le service des non-CoMP UE (resp. des CoMP UE) est assuré par S_M (resp. S_c) serveurs identiques, chaque serveur pouvant servir une unité du batch à la fois. Le système fonctionne comme suit : à chaque fois qu'un batch de taille j trouve à son arrivée n serveurs ($n=0, 1, \dots, S_M$) occupés, un nombre $\min(j, S_M-n)$ (resp. $\min(j, S_c-n)$) d'unités de batch, choisis au hasard, occupe un nombre égal de serveurs inactifs ainsi il rentre en service. Les unités de batch restantes, s'il y en a, sont rejetées par le système (bloquées) et ne reviennent pas plus tard. Par conséquent, aucune file d'attente n'est autorisée à être formée et les unités de batch bloquées sont considérées comme perdues pour le système.

Nous utilisons l'équation (3.14) pour calculer la probabilité de blocage dans la macro M1, p_{M1}^b , dans la macro M2, p_{M2}^b , et dans la région de coopération, p_c^b , en considérant les hypothèses suivantes :

- Dans la macro M1 : $S = S_M$, $\rho = \rho_{M1} = \frac{\lambda_{M1}^{clients}}{\mu_{M1}}$, $\lambda^{clients} = \lambda_{M1}^{clients}$, $g_j = g_{M1,j} = \frac{\lambda_{M1,j}^{clients}}{\lambda_{M1}^{clients}}$ et $\mu = \mu_{M1} = 1$ (correspondant à un RBG par trame) ;
- Dans la macro M2 : $S = S_M$, $\rho = \rho_{M2} = \frac{\lambda_{M2}^{clients}}{\mu_{M2}}$, $\lambda^{clients} = \lambda_{M2}^{clients}$, $g_j = g_{M2,j} = \frac{\lambda_{M2,j}^{clients}}{\lambda_{M2}^{clients}}$ et $\mu = \mu_{M2} = 1$;
- Dans la région de coopération : $S = S_c$, $\rho = \rho_c = \frac{\lambda_c^{clients}}{\mu_c}$, $\lambda^{clients} = \lambda_c^{clients}$, $g_j = g_{c,j} = \frac{\lambda_{c,j}^{clients}}{\lambda_c^{clients}}$ et $\mu = \mu_c = 1$.

En nous basant sur l'équation (3.14) et sur ce paramétrage, les débits de sortie en termes de RBG/trame, $X_{M1,j}^{RBG}$, pour toute classe j entre 1 et J_{M1} de la macro M1 sont calculés par :

$$\begin{cases} X_{M1,1}^{RBG} = \lambda_{M1,1}^{clients} \times (1 - p_{M1}^b) \\ X_{M1,j}^{RBG} = \lambda_{M1,j}^{clients} \times \left(\sum_{l=0}^{S_{M1}-j} j \times p_l + \sum_{l=S_{M1}-j+1}^{S_{M1}-1} (S_{M1} - l) \times p_l \right); \text{ pour } 1 < j \leq J_{M1} \end{cases} \quad (5.30)$$

ceux de la macro M2 sont calculés par :

$$\begin{cases} X_{M2,1}^{RBG} = \lambda_{M2,1}^{clients} \times (1 - p_{M2}^b) \\ X_{M2,j}^{RBG} = \lambda_{M2,j}^{clients} \times \left(\sum_{l=0}^{S_{M2}-j} j \times p_l + \sum_{l=S_{M2}-j+1}^{S_{M2}-1} (S_{M2} - l) \times p_l \right); \text{ pour } 1 < j \leq J_{M2} \end{cases} \quad (5.31)$$

et ceux de la région de coopération sont calculés par :

$$\begin{cases} X_{c,1}^{RBG} = \lambda_{c,1}^{clients} \times (1 - p_c^b) \\ X_{c,j}^{RBG} = \lambda_{c,j}^{clients} \times \left(\sum_{l=0}^{S_c-j} j \times p_l + \sum_{l=S_c-j+1}^{S_c-1} (S_c - l) \times p_l \right); \text{ pour } 1 < j \leq J_c \end{cases} \quad (5.32)$$

Ainsi, les débits correspondants en termes de clients/trame, $X_{M1,j}^{clients}$ pour la macro M1, $X_{M2,j}^{clients}$ pour la macro M2 et $X_{c,j}^{clients}$ pour la région de coopération, sont calculés par l'équation (5.33), l'équation (5.34) et l'équation (5.35), respectivement.

$$X_{M1,j}^{clients} = \frac{X_{M1,j}^{RBG}}{j}; \text{ pour } 1 \leq j \leq J_{M1} \quad (5.33)$$

$$X_{M2,j}^{clients} = \frac{X_{M2,j}^{RBG}}{j}; \text{ pour } 1 \leq j \leq J_{M2} \quad (5.34)$$

$$X_{c,j}^{clients} = \frac{X_{c,j}^{RBG}}{j}; \text{ pour } 1 \leq j \leq J_c \quad (5.35)$$

Les débits correspondants en termes de bits/trame sont calculés respectivement par :

$$X_{M1,j}^{bits} = X_{M1,j}^{RBG} \times N_{PRB} \times 7 \times 12 \times CR_{M1,j} \times \log_2(N_{M1,j}); \text{ pour } 1 \leq j \leq J_{M1} \quad (5.36)$$

$$X_{M2,j}^{bits} = X_{M2,j}^{RBG} \times N_{PRB} \times 7 \times 12 \times CR_{M2,j} \times \log_2(N_{M2,j}); \text{ pour } 1 \leq j \leq J_{M2} \quad (5.37)$$

$$X_{c,j}^{bits} = X_{c,j}^{RBG} \times N_{PRB} \times 7 \times 12 \times CR_{c,j} \times \log_2(N_{c,j}); \text{ pour } 1 \leq j \leq J_c \quad (5.38)$$

Les débits de sortie binaires dans la macro M1, X_{M1}^{bits} , dans la macro M2, X_{M2}^{bits} , dans la région de coopération, X_c^{bits} et le débit global en termes de kbits/trame X sont calculés respectivement par l'équation (5.24), l'équation (5.26), l'équation (5.28) et l'équation (5.29).

5.5 Analyse des performance

5.5.1 Algorithme de génération et de calcul des conditions radio des UE pour JT-CoMP

Afin de calculer pour chaque utilisateur u , avec une taille ($size(u)$) en bits, le nombre de RBG par trame qu'il requiert selon ses conditions radio lorsque JT-CoMP est appliqué, nous proposons l'algorithme [3](#). Cet algorithme va donc nous permettre d'évaluer et de caractériser le trafic et les conditions radio des UEs (voir [FIGURE 5.2](#)).

Algorithme 3 Calculer le nombre de RBG requis par chaque UE dans le cas de JT-CoMP

Inputs : Cluster de cellules, le taux d'arrivée des clients (noté $\lambda^{clients}$), la taille de paquet de chaque UE (size(u))

Résultats : RBGs(u) pour chaque UE "u", nombre de classes, la distribution des probabilités des tailles des batchs, nombre de classes, les conditions radio par classe (dans la macro M1, la macro M2 et la région de coopération)

1: Génération de Nb UE aléatoirement et uniformément dans le cluster selon une loi de poisson de paramètre $\lambda^{clients}$

2: Pour u=1 à Nb

2.1. Calculer la puissance de réception de la macro M1 et de la macro M2 :

$$PRx_{M1}(u) = Ptx_{M1}(u) + G - PL_{M1}(u)$$

$$PRx_{M2}(u) = Ptx_{M2}(u) + G - PL_{M2}(u)$$

2.2. Calculer le SINR(u) sans coopération et définir la serving cell :

Si $PRx_{M1}(u) \geq PRx_{M2}(u)$ **alors**

$$SINR(u) = 10\log_{10}\left(\frac{PRx_{M1}(u)}{\sum_{i=2}^{10} PRx_i(u) + N_o}\right)$$

$$PRx_s(u) = PRx_{M1}(u)$$

sinon

$$SINR(u) = 10\log_{10}\left(\frac{PRx_{M2}(u)}{\sum_{i=1, i \neq 2}^{10} PRx_i(u) + N_o}\right)$$

$$PRx_s(u) = PRx_{M2}(u)$$

Fin Si

2.3. Dédire le MCS(u) correspondant au SINR(u) sans coopération

2.4. Calculer le SINR(u) avec coopération et définir la serving cell et la cooperative cell :

Si $PRx_{M1}(u) \geq PRx_{M2}(u)$ **alors**

$$PRx_s(u) = PRx_{M1}(u)$$

$$PRx_c(u) = PRx_{M2}(u)$$

sinon

$$PRx_s(u) = PRx_{M2}(u)$$

$$PRx_c(u) = PRx_{M1}(u)$$

2.5. Calculer le $SINR_c(u)$ avec coopération :

$$SINR_c(u) = 10\log_{10}\left(\frac{PRx_s(u) + PRx_c(u)}{\sum_{i=3}^{10} PRx_i(u) + N_o}\right)$$

Fin Si

2.6. Dédire le $MCS_c(u)$ correspondant au $SINR_c(u)$

$$2.7. MCS(u) = \max(MCS_c(u), MCS(u))$$

2.8. Déterminer le nombre d'OFDM symboles requis :

$$N_{Symb}(u) = \frac{size(u)}{MCS(u)}$$

2.9. Déterminer le nombre de PRB correspondant :

$$N_{Subch}(u) = \frac{N_{Symb}(u)}{7 \times 12}$$

2.10. Déterminer le nombre de RBG correspondant :

$$N_{RBG}(u) = \frac{N_{Subch}(u)}{N_{PRB}}$$

3: Fin

4: Calcul du nombre de classes dans la macro M1, la macro M2 et la région de coopération

5: Calcul du taux d'arrivées par classe (dans la macro M1, la macro M2 et dans la région de coopération)

6: Calcul de la distribution des probabilités des tailles des batchs (dans la macro M1, la macro M2 et dans la région de coopération)

7: Calcul du taux de codage moyen et le nombre de points de constellation moyen par classe (dans la macro M1, la macro M2 et dans la région de coopération)

où $PRx_{M1}(u)$, $PRx_{M2}(u)$, $PRx_s(u)$, $PRx_c(u)$ et $PRx_i(u)$ représentent, respectivement, la puissance de réception (en watt) de la macro M1, la macro M2, la serving cell, la cell coopérative et les cellules interférentes. Notons que i représente les cellules macro interférentes :

- Si le UE est servi uniquement par la macro M1 (sans coopération avec la M2), alors toutes les autres macro cell du cluster représentent des cellules interférentes. Dans ce cas, $i = 2, 3, \dots, 10$ (voir Figure 5.2).
- Si le UE est servi uniquement par la macro M2 (sans coopération avec la M2), alors toutes les autres cellules du cluster représentent des cellules interférentes. Dans ce cas, $i = 1, 3, \dots, 10$ (voir Figure 5.2).
- Si la macro M1 et la macro M2 coopèrent pour servir un UE, alors toutes les autres cellules du cluster représentent donc des cellules interférentes. Dans ce cas, $i = 3, 4, \dots, 10$ (voir Figure 5.2).

$PL_{M1}(u)$ (resp. $PL_{M2}(u)$) exprime le pathloss entre le UE u et la macro M1 (resp. la macro M2) en db. Pour le pathloss des macro, nous considérons $PL(u)$ dans un environnement urbain comme expliqué dans le Tableau 4.2.

Une fois que le nombre de RBG requis par chaque UE est calculé, l'algorithme regroupe les UE demandant le même nombre de RBG dans la même classe (nous notons que le nombre de RBG requis correspond à l'indice de la classe) et déduit les taux d'arrivée de clients dans chaque classe, notés $\lambda_{M1,i}^{clients}$ (resp. $\lambda_{M2,i}^{clients}$ et $\lambda_{c,i}^{clients}$) pour la classe i de la macro M1 (resp. la macro M2 et la région de coopération), et la distribution de la probabilité des batchs de taille i , noté $g_{M1,i}$ (resp. $g_{M2,i}$ et $g_{c,i}$) pour la macro M1 (resp. la macro M2 et la région de coopération).

Rappelons que J_{M1} (respectivement J_{M2} et J_c) est le nombre de classes dans la macro M1 (resp. la macro M2 et la région de coopération). Ainsi, les taux d'arrivées totales dans la macro M1, dans la macro M2 et dans la région de coopération sont calculés respectivement par l'algorithme 3 comme suit :

$$\lambda_{M1}^{clients} = \sum_{i=1}^{J_{M1}} \lambda_{M1,i}^{clients} \quad \text{et} \quad \lambda_{M1} = \sum_{i=1}^{J_{M1}} \lambda_{M1,i} \quad (5.39)$$

$$\lambda_{M2}^{clients} = \sum_{i=1}^{J_{M2}} \lambda_{M2,i}^{clients} \quad \text{et} \quad \lambda_{M2} = \sum_{i=1}^{J_{M2}} \lambda_{M2,i} \quad (5.40)$$

$$\lambda_c^{clients} = \sum_{i=1}^{J_c} \lambda_{c,i}^{clients} \quad \text{et} \quad \lambda_c = \sum_{i=1}^{J_c} \lambda_{c,i} \quad (5.41)$$

où $\lambda_{M1}^{clients}$, $\lambda_{M2}^{clients}$ et $\lambda_c^{clients}$ sont exprimés en termes de clients/trame et λ_{M1} , λ_{M2} et λ_c sont exprimés en termes de kbits/trame.

De plus, connaissant le SINR de chaque UE, l'algorithme déduit en se basant sur le Tableau 2.2 le taux de codage et le nombre de points de constellation pour chaque UE puis calcule le taux de codage moyen et le nombre de points de constellation moyen par classe. Nous répétons évidemment un nombre de simulations suffisamment grand pour assurer un intervalle de confiance à 95%.

Ces paramètres (le taux d'arrivée par classe, la distribution des probabilités des tailles des batches, le taux de codage moyen par classe et le nombre de points de constellation moyen par classe) sont ensuite exploités par les modèles que nous proposons ie. les chaînes de Markov et les files d'attente afin de considérer les mêmes conditions radio des UE lors du calcul des probabilités de blocage et des débits dans la macro et la micro. Toute cette démarche est détaillée dans la FIGURE 5.4

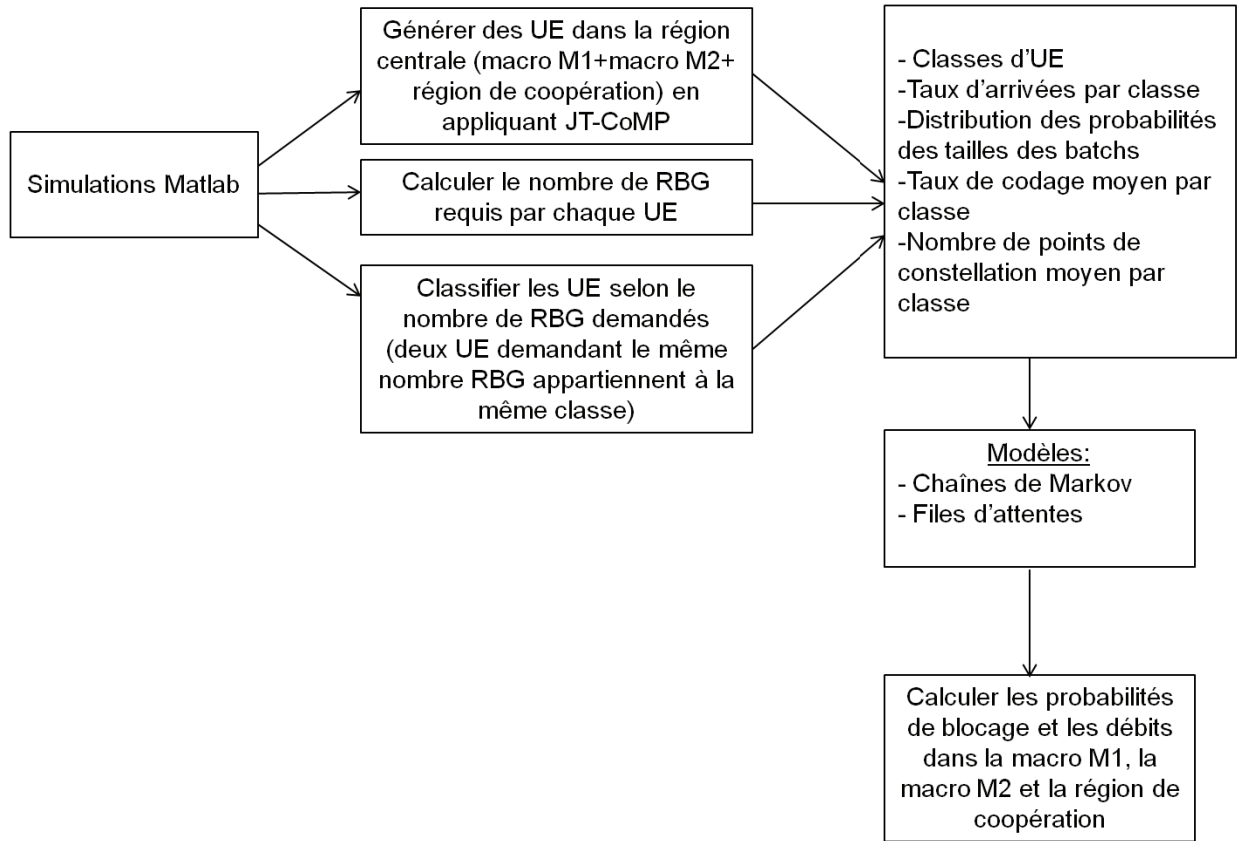


FIGURE 5.4 – Explication détaillée sur notre démarche pour la modélisation de JT-CoMP

L'intérêt de notre approche est de simuler un environnement réaliste en utilisant un algorithme simple et performant, visant à générer des utilisateurs selon des conditions radio réelles et nous permettant de calculer des paramètres précis avec les chaînes de Markov et les modèles de files d'attente proposés.

5.5.2 Etude comparative entre les trois approches proposées : chaînes de Markov, $M/G/S/S$ et $M^{[X]}/G/S/S$

Dans cette section, nous analysons et comparons les résultats de performances obtenus par les trois approches que nous proposons (chaîne de Markov, $M/G/S/S$ et $M^{[X]}/G/S/S$) en termes de débit et de probabilité de blocage par classe. Des comparaisons avec des simulations Matlab (basées

sur les équations d'évolution des chaînes de Markov) ont été effectuées afin d'estimer la précision des modèles de files d'attente ($M/G/S/S$ et $M^{[X]}/G/S/S$) pour la modélisation de JT-CoMP. Les paramètres d'entrée du modèle de file d'attente sont obtenus des simulations Matlab reproduisant le comportement réel de la technique JT-CoMP (listés dans le Tableau 4.3). Tous ces résultats sont présentés en assurant un intervalle de confiance à 95%.

La figure 5.2 présente un échantillon de génération d'utilisateurs en utilisant le simulateur Matlab tel que $size(u) = 1024$ bits $\forall u$, $\lambda^{clients} = 100$ clients/trame et le nombre de serveurs pour servir les CoMP UE est égal à 109 (proportionnel au nombre de CoMP UE générés). Nous présentons les non-CoMP UE par (\times) et les CoMP UE par (o). Cet échantillon de génération de UE montre que les clients sont classés dans deux classes dans la macro M1, la macro M2 et la région de coopération. Les clients avec de mauvaises conditions radio (en rouge) sont des clients edge demandant 2 RBG et le reste des clients avec de bonnes (en jaune) et très bonnes (en vert) conditions radio représentent les clients centraux demandant 1 RBG.

En considérant un nombre important de trames, nous validons les modèles de file d'attente. Les paramètres obtenus des simulations Matlab et utilisés dans les modèles de file d'attente sont listés dans le Tableau 5.2. Le nombre de serveurs S_c pour la région de coopération est calculé proportionnellement au nombre d'utilisateurs en coopération générés.

$\lambda^{clients}$	100	200	300	400	500	600	700	800	900
$\lambda_{M1,1}^{clients}$	24	50	74	101	131	162	191	216	248
$\lambda_{M1,2}^{clients}$	15	30	45	62	80	99	118	132	148
$\lambda_{M2,1}^{clients}$	24	49	73	102	132	162	193	216	243
$\lambda_{M2,2}^{clients}$	15	30	46	62	81	100	118	133	150
$\lambda_{c,1}^{clients}$	10	19	29	34	36	36	38	48	54
$\lambda_{c,2}^{clients}$	12	22	33	39	40	41	42	55	62
S_c	109	107	107	109	108	107	108	107	108

Tableau 5.2 – Paramètres extraits de la simulation des chaînes de Markov

La FIGURE 5.5 et la FIGURE 5.6 (resp. la FIGURE 5.7 et la FIGURE 5.8) comparent les débits par classe (classe 1 et classe 2 respectivement) dans la macro M1 (resp. la macro M2) obtenus par les trois approches. De plus, la FIGURE 5.9 et la FIGURE 5.10 comparant les débits par classe dans la région de coopération (dans la classe 1 et la classe 2 respectivement) obtenus par les trois approches. Les résultats détaillés (probabilités de blocage et débits) correspondant à ces figures sont énumérés respectivement dans le Tableau 16, le Tableau 17, le Tableau 18, le Tableau 19, le Tableau 20 et le Tableau 21, dans l'Annexe.

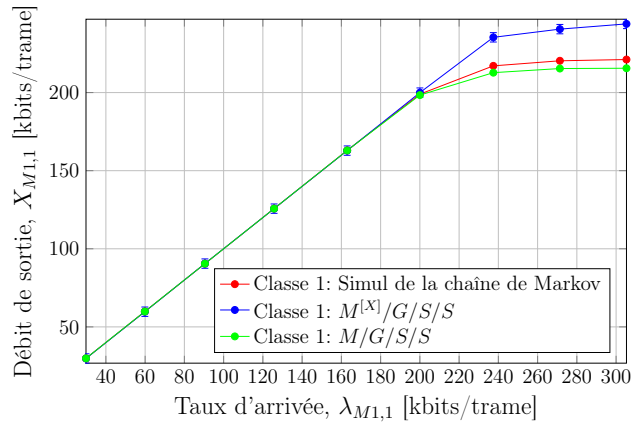


FIGURE 5.5 – Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 1 de la macro M1 ($size(u) = 1024$ bits)

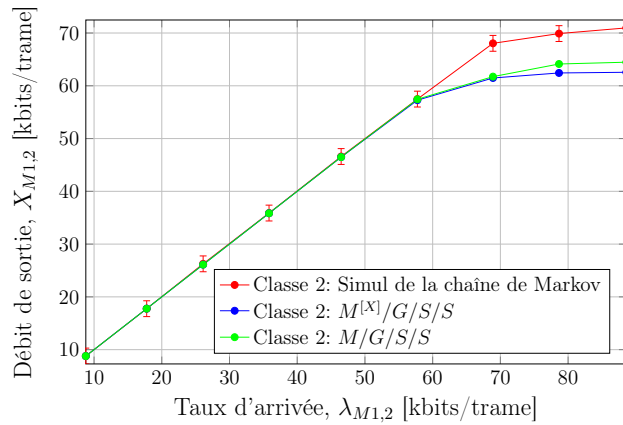


FIGURE 5.6 – Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 2 de la macro M1 ($size(u) = 1024$ bits)

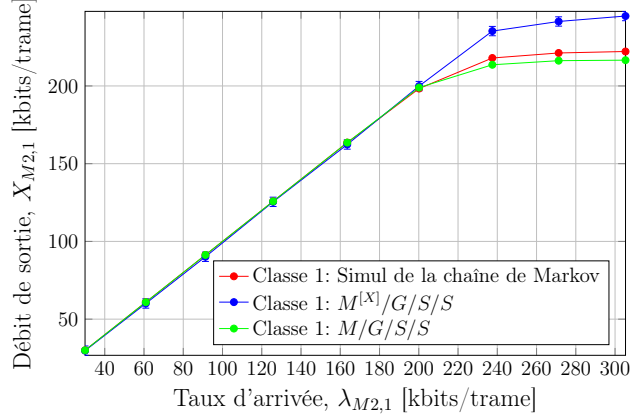


FIGURE 5.7 – Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 1 de la macro M2 ($size(u) = 1024$ bits)

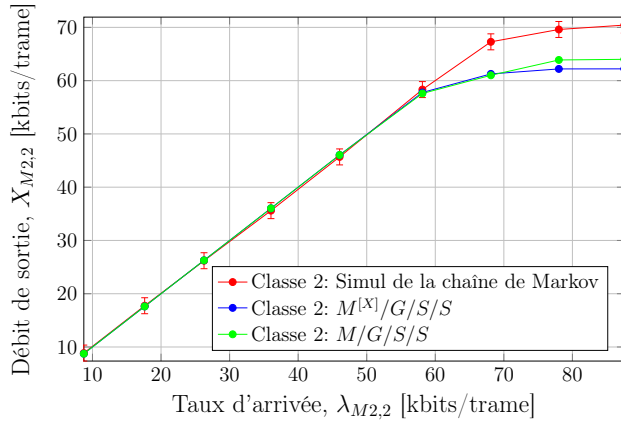


FIGURE 5.8 – Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 2 de la macro M2 ($size(u) = 1024$ bits)

Suite à ces résultats, nous pouvons conclure que les formes closes des deux modèles de files d'attente que nous proposons ($M^{[X]}/G/S/S$ et $M/G/S/S$) représentent de bonnes approximations. Nous pouvons ainsi les utiliser pour modéliser la technique de remédiation aux interférences JT-CoMP.

Par conséquent, nous proposons dans ce qui suit un algorithme basé sur ces formes closes afin de configurer correctement le nombre de serveurs à allouer aux utilisateurs CoMP, selon la charge du trafic, en maximisant le débit global. Dans ce but, nous avons développé un outil de gestion des

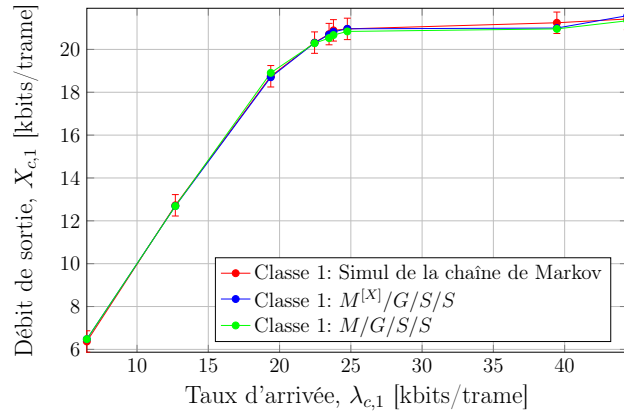


FIGURE 5.9 – Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 1 de la région de coopération ($size(u) = 1024$ bits)

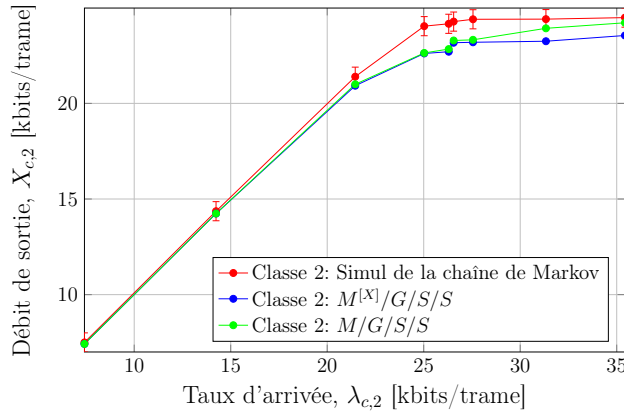


FIGURE 5.10 – Comparaison entre les simulations Matlab, la $M^{[X]}/G/S/S$ et la $M/G/S/S$ en terme de débit de sortie dans la classe 2 de la région de coopération ($size(u) = 1024$ bits)

ressources dynamique pour faciliter la prise de décision concernant la répartition des ressources entre les utilisateurs CoMP et les utilisateurs non-CoMP ainsi que la détermination du nombre d'UE à servir en coopération, afin d'atteindre un débit global optimal.

5.6 Optimisation de JT-CoMP basée sur les modèles de files d'attente

Dans la section précédente, S_c était proportionnel au nombre des utilisateurs en coopération générés (voir le Tableau 5.2), cette section vise à déterminer de manière dynamique le nombre optimal de clients à servir avec JT-CoMP et la distribution optimale des ressources radio entre les UE CoMP et non-CoMP afin d'atteindre le débit global optimal.

Lorsque les UE sont servis dans la région de coopération, les utilisateurs en coopération devraient être perdus avec une probabilité de blocage $\leq \epsilon$. Nous supposons, sans perte de généralité, que les conditions radio sont toujours bonnes pour les utilisateurs en coopération. En effet, un utilisateur en coopération, s'il n'est pas perdu, il ne sera servi que si la coopération permet d'améliorer ses conditions radio. Sinon, l'UE restera un non-CoMP UE et il sera associé à l'une des macro cellules. Par conséquent, nous pouvons supposer qu'une seule classe de service est disponible dans la région de coopération, où tous les UE demandent un RBG (c'est-à-dire $\lambda_c^{clients} = \lambda_{c,1}^{clients}$).

Dans ce but, nous formulons un problème d'optimisation dont les variables de décisions sont les suivantes :

- $\lambda_c^{clients}$: taux d'arrivée des clients dans la région de coopération
- $\lambda_{M1}^{clients}$: taux d'arrivée des clients dans la macro M1
- $\lambda_{M2}^{clients}$: taux d'arrivée des clients dans la macro M2
- S_c : nombre de ressources à attribuer aux CoMP UE
- $S_M = S_{M1} = S_{M2} = S - S_c$: nombre de ressources à attribuer aux non-CoMP UE

Notons que nous avons délibérément défini le taux d'arrivée des clients dans la région de coopération en tant que variable, ce taux (qui fait partie du taux d'arrivée global) peut considérablement impacter le débit global car dans JT-CoMP les clients de la région de coopération utilisent les ressources des cellules coopératives (macro M1 et macro M2) simultanément. Ainsi, nous l'optimisons avec la configuration des ressources.

Notre problème d'optimisation est formulé comme suit :

$$\begin{aligned}
& \max_{S_M, S_c, \lambda_c^{clients}, \lambda_{M1}^{clients}, \lambda_{M2}^{clients}} f(S_M, S_c, \lambda_c^{clients}, \lambda_{M1}^{clients}, \lambda_{M2}^{clients}) \\
& \text{s.t.} \quad S_M + S_c = S, \\
& \quad \lambda_c^{clients} + \lambda_{M1}^{clients} + \lambda_{M2}^{clients} = \lambda^{clients}, \\
& \quad S > 0, \\
& \quad p_c^b \leq \epsilon
\end{aligned} \tag{5.42}$$

où $f(S_M, S_c, \lambda_c^{clients}, \lambda_{M1}^{clients}, \lambda_{M2}^{clients})$ représente le débit global exprimé en termes de kbits/ trame et calculé comme suit :

$$f(S_M, S_c, \lambda_c^{clients}, \lambda_{M1}^{clients}, \lambda_{M2}^{clients}) = \sum_{j=1}^{J_{M1}} X_{M1,j} + \sum_{j=1}^{J_{M2}} X_{M2,j} + X_{c,1} = \frac{\sum_{j=1}^{J_{M1}} X_{M1,j}^{bits} + \sum_{j=1}^{J_{M2}} X_{M2,j}^{bits} + X_{c,1}^{bits}}{1024} \tag{5.43}$$

tel que :

$X_{M1,j}$, $X_{M2,j}$ et $X_{c,j}$ sont calculés respectivement en utilisant l'équation (5.23), l'équation (5.25) et l'équation (5.27) avec la $M/G/S/S$ et en utilisant l'équation (5.36), l'équation (5.37) et l'équation (5.38) avec la $M^{[X]}/G/S/S$.

Suite à la combinaison de variables continues ($\lambda_c^{clients}$) et de variables discrètes (S_c et S_M) et à cause de la non-linéarité de la fonction objective ($X_{M1,j}$ pour $j=1, \dots, J_{M1}$, $X_{M2,j}$ pour $j=1, \dots, J_{M2}$ et $X_{c,1}$) et de la contrainte (p_c^b), le problème de maximisation correspond à un MINLP connu comme un problème NP-hard. Une façon simple pour résoudre ce problème est :

- de fixer d'abord S_c (et par conséquent S_M) ;
- ensuite, pour chaque valeur, de déterminer le $\lambda_c^{clients}$ optimal qui maximise le débit global et garantit une probabilité de blocage dans la région de coopération $p_c^b \leq \epsilon$;
- enfin, de retenir le S_c (et S_M) et $\lambda_c^{clients}$ optimaux qui offrent le débit global optimal.

Notre problème correspond maintenant à un problème NLP à cause de la contrainte et de la fonction objective qui ne sont pas linéaires [BSS13] [NW06]. Ainsi, nous utilisons l'algorithme Global Search [Mat20] défini dans Matlab pour résoudre notre problème d'optimisation NLP.

Toutefois, pour nos études, par exemple pour un pourcentage de clients maximal en coopération égal à $10\% \times \lambda^{clients}$, pour déterminer la configuration optimale, la résolution par Global Search prend 7h lorsque Global Search se base sur la forme close de la $M/G/S/S$ et 1 jour et 14h lorsqu'il se base sur la forme close de la $M^{[X]}/G/S/S$. De plus, nous notons que plus le pourcentage de clients maximal en coopération est grand, plus la durée de résolution par Global Search prend du temps (pouvant atteindre des jours). Afin de remédier à cette faiblesse, nous proposons une heuristique permettant

de rechercher la valeur optimale dans un ensemble réduit (pour déterminer $\lambda_c^{clients}$ optimal, nous nous limitons à tester certaines valeurs). Nous proposons alors l'algorithme [4](#)

Algorithme 4 Comment déterminer le taux d'arrivée des clients et le nombre de serveurs dans la région de coopération pour atteindre le débit global optimal

Init : $i = 0$

Inputs : i_{max} , $step$, $\lambda^{clients}$, J_{M1} , J_{M2} , J_c , $\lambda_{M1,j}^{clients}$ ($j = 1, \dots, J_{M1}$), $\lambda_{M2,j}^{clients}$ ($j = 1, \dots, J_{M2}$), $\lambda_{c,j}^{clients}$ ($j = 1, \dots, J_c$), ϵ , S

Résultats : S_M , S_c , $\lambda_c^{clients}$, $\lambda_{M1}^{clients}$, $\lambda_{M2}^{clients}$, X

1: **Tant que** $i \leq i_{max}$ **faire**

1.1. Déterminer les taux d'arrivée des clients :

$$\lambda_c^{clients} = i \times \lambda^{clients}$$

$$\lambda_{M1,J_{M1}}^{clients} = \lambda_{M1,J_{M1}}^{clients} + \frac{(i_{max} \times \lambda^{clients} - \lambda_c^{clients})}{2}$$

$$\lambda_{M2,J_{M2}}^{clients} = \lambda_{M2,J_{M2}}^{clients} + \frac{(i_{max} \times \lambda^{clients} - \lambda_c^{clients})}{2}$$

$$\lambda_{M1}^{clients} = \sum_{j=1}^{J_{M1}-1} \lambda_{M1,j}^{clients} + \lambda_{M1,J_{M1}}^{clients}$$

$$\lambda_{M2}^{clients} = \sum_{j=1}^{J_{M2}-1} \lambda_{M2,j}^{clients} + \lambda_{M2,J_{M2}}^{clients}$$

1.2. Déterminer le/les $S_c^{**}(i)$ tel que $0 \leq S_c^{**}(i) \leq \frac{S}{2}$ et $p_c^b(i) \leq \epsilon$

1.3. Sélectionner le/les $S_c^*(i)$ parmi le/les $S_c^{**}(i)$ offrant le débit global maximal $X(i)$

1.4. Déterminer le $S_c(i)$ minimal tel que $S_c(i) = \min(S_c^*(i))$

2: $i = i + step$

3: **Fin Tant que**

4: Déterminer le/les i maximisant le débit global tel que $X = \max(X(i))$, s'il y a plusieurs i qui vérifient cela, alors prendre $S_c = \min(S_c(i))$ et retenir le i correspondant

5: Dédire :

$$S_M = S - S_c$$

$$\lambda_c^{clients} = i \times \lambda^{clients}$$

$$\lambda_{M1,J_{M1}}^{clients} = \lambda_{M1,J_{M1}}^{clients} + \frac{(i_{max} \times \lambda^{clients} - \lambda_c^{clients})}{2}$$

$$\lambda_{M2,J_{M2}}^{clients} = \lambda_{M2,J_{M2}}^{clients} + \frac{(i_{max} \times \lambda^{clients} - \lambda_c^{clients})}{2}$$

$$\lambda_{M1}^{clients} = \sum_{j=1}^{J_{M1}-1} \lambda_{M1,j}^{clients} + \lambda_{M1,J_{M1}}^{clients}$$

$$\lambda_{M2}^{clients} = \sum_{j=1}^{J_{M2}-1} \lambda_{M2,j}^{clients} + \lambda_{M2,J_{M2}}^{clients}$$

L'objectif de l'algorithme [4](#) est de déterminer le taux d'arrivée des clients et le nombre de serveurs dans la région de coopération permettant d'atteindre un débit global optimal. Pour ce faire, il suit le raisonnement suivant :

- (1) Nous notons i_{max} le pourcentage initial d'utilisateurs générés dans la région de coopération, il correspond donc au pourcentage maximal d'utilisateurs en coopération. Le modèle d'optimisation va ensuite faire varier i de 0 jusqu'à i_{max} avec un pas de boucle noté 'step' (plus les valeurs de 'i' et de 'step' sont petites, meilleure est la précision du modèle d'optimisation). Ensuite, pour chaque pourcentage de clients à associer à la région de coopération (chaque i),

le modèle d'optimisation détermine le taux d'arrivée des clients. Puis, il détermine le nombre minimal de ressources $S_c(i)$ à attribuer à la micro qui offre un débit global optimal $X(i)$ avec une garantie de service pour les clients de la micro égale à $(1 - \epsilon)\%$ autrement dit vérifiant $p_c^b(i) \leq \epsilon$. Pour ce faire, pour chaque i , l'algorithme fait varier le nombre de ressources, noté $S_c^{**}(i)$ à attribuer à la région de coopération de 0 à $\frac{S}{2}$ et ne conserve que le/les $S_c^{**}(i)$ qui garantissent un service de $(1 - \epsilon)\%$ pour les clients de la région de coopération. Pour les $S_c^{**}(i)$ garantissant cette contrainte de service dans la région de coopération, il calcule le débit global correspondant puis détermine le max de ces débits globaux, $X(i)$, et retient le/les $S_c^{**}(i)$ offrant le débit maximal, que nous notons $S_c^*(i)$. Si plusieurs $S_c^*(i)$ sont retenus alors l'algorithme sélectionne le $S_c^*(i)$ minimal, que nous notons $S_c(i)$. Par conséquent, pour i , $S_c(i)$ représente le nombre de ressources optimal qui assure $p_c^b(i) \leq \epsilon$ et un débit global maximal $X(i)$.

- Après cela dans (4), l'algorithme détermine le/les i maximisant le débit global tel que $X = \max(X(i))$, s'il y a plusieurs i qui vérifient cela, alors prendre $S_c = \min(S_c(i))$ et retenir le i correspondant.
- Dans (5), il en déduit les taux d'arrivée des clients optimaux. Le taux d'arrivée des clients optimal dans la micro est calculé à partir de la valeur de i optimale déterminée à l'étape (4). Notez que lorsque $i < i_{max}$, les clients non servis dans la région de coopération ($i_{max} \times \lambda^{clients} - i \times \lambda^{clients}$) seront dispatchés et ajoutés (équitablement dans le cas d'une distribution uniforme) au taux d'arrivée des clients de la macro M1 de la classe J_{M1} ($\lambda_{M1, J_{M1}}^{clients}$) et au taux d'arrivée des clients de la macro M2 de la classe J_{M2} ($\lambda_{M2, J_{M2}}^{clients}$), de sorte que le taux d'arrivée total $\lambda^{clients}$ reste constant.

Par ailleurs, il est à noter que grâce à l'heuristique proposée, le temps de résolution de notre algorithme d'optimisation pour déterminer la configuration optimale est nettement réduit par rapport à Global Search. En effet, par exemple pour un pourcentage de clients maximal en coopération égal à $10\% \times \lambda^{clients}$, notre algorithme d'optimisation prend 2min lorsqu'il se base sur la $M/G/S/S$ (contre 7h pour Global Search) et 10h lorsqu'il se base sur la $M^{[X]}/G/S/S$ (contre 1j et 14h pour Global Search).

5.6.1 Application numérique du modèle d'optimisation de JT-CoMP

Dans ce qui suit, nous comparons les résultats obtenus par l'algorithme [4](#) (en appliquant la $M/G/S/S$ et la $M^{[X]}/G/S/S$) et l'algorithme Global Search. Nous notons dans le [Tableau 5.3](#), le nombre de serveurs optimal en coopération, S_c , le taux d'acceptation des clients qui demandent la coopération (qui revient au rapport entre le taux d'arrivée des clients optimal servis en coopération ($\lambda_c^{clients}$) et le taux d'arrivées des clients dans la région de coopération ($\lambda_{c_{max}}^{clients}$)) et le débit global optimal (X).

Nous considérons une distribution de clients réaliste dans la macro M1, la macro M2 et la région

de coopération avec une distribution uniforme et une taille de paquet $size(u) = 1024$ bits, nous obtenons alors : $\lambda_{M1,1}^{clients} = \lambda_{M2,1}^{clients} = 24\% \times \lambda^{clients}$ des clients sont dans la classe 1 de la macro M1 et la macro M2, $\lambda_{cmax}^{clients} = 10\% \times \lambda^{clients}$ des clients sont dans la classe 1 de la région de coopération (avec de bonnes conditions radio), $\lambda_{M1,2}^{clients} = \lambda_{M2,2}^{clients} = 21\% \times \lambda^{clients}$ des clients sont dans la classe 2 (edge UE) de la macro M1 et la macro M2 (avec de mauvaises conditions radio).

Nous limitons l'espace de recherche du nombre optimal de serveurs en limitant le nombre maximal de serveurs attribués à la région de coopération à 250 (la moitié du nombre de serveurs disponibles dans une trame) pour une bande passante de 20 MHz. Nous supposons que la valeur initiale de $i = 1\%$, $step = 1\%$ et $\epsilon = 0.02$ le but de cette dernière condition est de garantir que les UE de la région de coopération soit servis à 98 %.

Notons que des valeurs réalistes d'efficacité du MCS de l'utilisateur par classe (extraites des simulations Matlab) sont utilisées : $CR_{M1,1} \times \log_2(N_{M1,1}) = CR_{M2,1} \times \log_2(N_{M2,1}) = 3.7746$, $CR_{M1,2} \times \log_2(N_{M1,2}) = CR_{M2,2} \times \log_2(N_{M2,2}) = 0.88$ et $CR_{c,1} \times \log_2(N_{c,1}) = 1.96$.

$\lambda^{clients}$	λ [kbits/trame]	S_c optimal	$\lambda_c^{clients}$ optimal	X optimal
10	9	17	$10\% \times \lambda^{clients}$	9
20	18	22	$10\% \times \lambda^{clients}$	18
30	27	26	$10\% \times \lambda^{clients}$	27
100	90	46	$10\% \times \lambda^{clients}$	90
150	135	55	$10\% \times \lambda^{clients}$	135
200	180	67	$10\% \times \lambda^{clients}$	180
380	342	100	$10\% \times \lambda^{clients}$	342
500	451	84	$10\% \times \lambda^{clients}$	451
600	541	71	$10\% \times \lambda^{clients}$	537
700	626	0	$0\% \times \lambda^{clients}$	577
800	716	0	$0\% \times \lambda^{clients}$	584
900	805	0	$0\% \times \lambda^{clients}$	586
1000	895	0	$0\% \times \lambda^{clients}$	586
2000	1790	0	$0\% \times \lambda^{clients}$	588
3000	2684	0	$0\% \times \lambda^{clients}$	588
4000	3579	0	$0\% \times \lambda^{clients}$	588

Tableau 5.3 – S_c optimal, $\lambda_c^{clients}$ optimal et débit global optimal en fonction du taux d'arrivée des clients ($\lambda_{cmax}^{clients} = 10\% \times \lambda^{clients}$)

Les mêmes valeurs optimales sont obtenues par l'algorithme Global Search et par l'algorithme 4 que nous appliquons sur les deux modèles de file d'attente. Par conséquent, les résultats ne sont affichés qu'une seule fois dans le Tableau 5.3.

Les résultats dans le Tableau 5.3 montrent que pour chaque taux d'arrivée des clients (ou densité en kbits/trame), un nombre optimal S_c et un nombre optimal d'utilisateurs en coopération peuvent être sélectionnés afin de maximiser le débit global en termes de kbits/trame.

- Lorsque le taux d’arrivée varie entre 9 et 342 kbits/trame, le cluster est légèrement chargé. Notre modèle d’optimisation propose de servir tous les UE qui demandent d’être servis en coopération. Notons également que pour ces charges, le nombre optimal de serveurs dans la région de coopération augmente adéquatement avec le taux d’arrivée des clients. Ceci est dû au fait que le débit des utilisateurs en coopération est amélioré grâce à la coopération vu qu’il y a assez de ressources dans la macro M1 et la macro M2 pour que chacune serve ses propres non-CoMP UE.
- Lorsque le taux d’arrivée des clients est supérieur à 342 et inférieur à 626 kbits/trame, contrairement au scénario où le système est légèrement chargé, l’augmentation du taux d’arrivée des clients ne va pas conduire à l’augmentation du nombre optimal de serveurs dans la région de coopération. En effet, le nombre de serveurs décroît avec l’augmentation de la charge. Toutefois, les résultats montrent 100% d’acceptation des clients qui demandent la coopération. Ceci est expliqué par le fait que le nombre de ces clients est petit ($\lambda_{cmax}^{clients} = 10\% \times \lambda^{clients}$) par rapport au taux d’arrivée de clients total. Notons que ces résultats peuvent être complètement différents pour une autre valeur $\lambda_{cmax}^{clients}$ (voir Tableau 5.5).
- Enfin, lorsque le système est fortement chargé, le modèle d’optimisation propose de ne pas servir les utilisateurs en coopération avec JT-CoMP, ces clients sont dispatchés équitablement (suite à la distribution uniforme) dans la classe 2 de la macro M1 et de la macro M2. Ce résultat est dû au fait que servir un client avec JT-CoMP nécessite l’occupation d’un serveur par macro cell ceci va donc pénaliser le service de deux clients qui offrent un meilleur débit global.

5.6.2 Influence d’epsilon sur le modèle d’optimisation de JT-CoMP

Dans la FIGURE 5.11, nous montrons l’influence de la valeur d’epsilon sur le nombre de serveurs à mettre en coopération en fonction du taux d’arrivée des clients, nous choisissons quatre valeurs d’epsilon : $\epsilon = 0.1$, $\epsilon = 0.02$, $\epsilon = 0.01$ et $\epsilon = 0.001$.

Pour $\lambda^{clients} = 600$ clients/trame (541 kbits/trame) et $\epsilon = 0.01$, 0.02 et 0.1 le débit global optimal est 537 kbits/trame obtenu en fixant le nombre de serveurs en coopération à 75, 71 et 69, respectivement. Ceci est expliqué par le fait que pour un même débit de sortie optimal, lorsqu’on augmente la valeur d’epsilon, le nombre de serveurs respectant la contrainte $p_c^b \leq \epsilon$ décroît. Pour chacun de ces epsilons, le nombre de serveurs optimal est donc le plus petit nombre de serveurs permettant à la fois d’atteindre le débit global optimal et de respecter la contrainte $p_c^b \leq \epsilon$. Pour $\epsilon = 0.001$, le débit global optimal est assuré lorsque JT-CoMP n’est pas appliqué.

Dans le Tableau 5.4, nous montrons l’influence d’epsilon sur le gain apporté suite à l’application de JT-CoMP (en nous basant sur les configurations optimales du nombre de serveurs S_c donné dans la

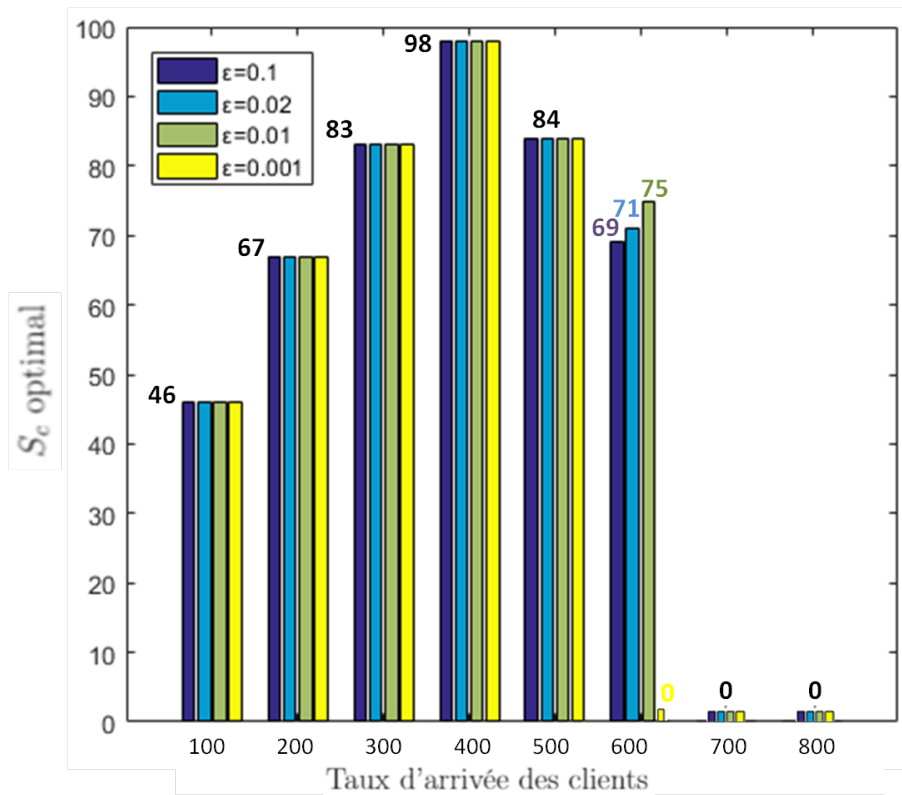


FIGURE 5.11 – Influence d'epsilon sur le nombre de serveurs optimal dans la région de coopération

FIGURE 5.11) en fonction du taux d'arrivée des clients. Ce gain, exprimé en bits/trame, correspond donc à la différence entre le débit global optimal et le débit global obtenu sans coopération (sans JT-CoMP).

$\lambda^{clients}$	Gain [bits/trame]			
	$\epsilon = 0.1$	$\epsilon = 0.02$	$\epsilon = 0.01$	$\epsilon = 0.001$
100	672	672	672	672
200	1344	1344	1344	1344
300	2016	2016	2016	2016
400	2688	2688	2688	2688
500	3360	3360	3360	3360
600	1903	1805	1186	0
700	0	0	0	0
800	0	0	0	0

Tableau 5.4 – Influence d'epsilon sur le gain apporté suite à l'application de JT-CoMP

Les résultats du Tableau 5.4 montrent que pour les taux d'arrivée de clients inférieurs à 600 clients/trame, les valeurs d'epsilon considérées n'influent pas sur le gain. Ceci est dû au fait que, pour un taux d'arrivée de clients, la configuration optimale de S_c est la même quelque soit la valeur d'epsilon considérée ($\epsilon = 0.1$ ou 0.02 ou 0.01 ou 0.001).

Pour $\lambda^{clients} = 600$ clients/trame, les valeurs d'epsilon considérées influent sur le gain. Pour $\epsilon = 0.1$, 0.02 et 0.01 , le gain est non nul et est différent pour chacun de ces epsilons. Ceci est dû au fait que le débit global optimal, pour ces trois valeurs d'epsilon, est obtenu, respectivement, lorsque $S_c = 69$, $S_c = 71$ et $S_c = 75$. Toutefois, pour $\epsilon = 0.001$ le gain est nul car la configuration optimale est obtenue lorsque $S_c = 0$.

Pour les taux d'arrivée des clients supérieurs ($\lambda^{clients} = 700$ et 800 clients/trame), comme indiqué dans la FIGURE 5.11, le nombre optimal $S_c = 0$ engendrant ainsi un gain nul.

Ces résultats prouvent donc l'influence d'epsilon sur la configuration optimale permettant d'atteindre un débit global optimal.

5.6.3 Influence de la distribution des clients sur le modèle d'optimisation de JT-CoMP

Dans le Tableau 5.5, nous considérons une distribution non-uniforme des clients entre la macro M1, la macro M2 et la région de coopération où la majorité des clients sont dans la région de coopération avec de bonnes conditions radio, nous avons alors : $\lambda_{M1,1}^{clients} = \lambda_{M2,1}^{clients} = 24\% \times \lambda^{clients}$ des clients sont dans la classe 1 de la macro M1 et la macro M2, $\lambda_{cmax}^{clients} = 40\% \times \lambda$ des clients sont dans la classe 1 de la région de coopération (avec de bonnes conditions radio), $\lambda_{M1,2}^{clients} = \lambda_{M2,2}^{clients} = 6\% \times \lambda^{clients}$ des clients sont dans la classe 2 de la macro M1 et la macro M2 (avec de mauvaises

conditions radio). Les autres paramètres restent les mêmes.

$\lambda^{clients}$	λ [kbits/trame]	S_c optimal	$\lambda_c^{clients}$ optimal	X optimal
10	9	29	$39\% \times \lambda^{clients}$	9
20	18	40	$39\% \times \lambda^{clients}$	18
30	27	49	$39\% \times \lambda^{clients}$	27
100	92	100	$39\% \times \lambda^{clients}$	92
150	138	133	$39\% \times \lambda^{clients}$	138
200	184	160	$39\% \times \lambda^{clients}$	184
500	460	250	$39\% \times \lambda^{clients}$	460
590	543	250	$39\% \times \lambda^{clients}$	540
600	552	249	$39\% \times \lambda^{clients}$	548
610	561	246	$38\% \times \lambda^{clients}$	554
630	579	241	$36\% \times \lambda^{clients}$	561
640	589	245	$36\% \times \lambda^{clients}$	563
700	626	0	$0\% \times \lambda^{clients}$	577
800	716	0	$0\% \times \lambda^{clients}$	584

Tableau 5.5 – S_c optimal, $\lambda_c^{clients}$ optimal et débit global optimal en fonction du taux d’arrivée des clients ($\lambda_{cmax}^{clients} = 40\% \times \lambda^{clients}$)

D’après les résultats du Tableau 5.5, nous remarquons qu’en changeant uniquement la distribution des clients, où $\lambda_c^{clients} = 40\% \times \lambda^{clients}$, nous distinguons 4 valeurs du taux d’acceptation optimal $\frac{\lambda_c^{clients}}{\lambda_{cmax}^{clients}}$ pour les taux d’arrivée considérés.

- Lorsque le cluster est légèrement chargé ($\lambda^{clients} \leq 600$ clients/trame), le modèle d’optimisation alloue le minimum nécessaire de ressources jusqu’à ce qu’il atteigne 250, le nombre maximal de serveurs dans la région de coopération ($S_c \leq \frac{S}{2}$). Ceci est expliqué par le fait que la charge augmente mais elle ne cause pas la perte de clients (débit de sortie = débit d’entrée, lorsque $\lambda^{clients} \leq 500$ clients/trame) ou négligeable (débit de sortie \simeq débit d’entrée, lorsque $\lambda^{clients} = 590$ clients/trame) dans le cluster. Pour ces charges, le taux d’arrivée des clients CoMP optimal est de $39\% \times \lambda^{clients}$.
- Lorsque $\lambda^{clients} \geq 610$ clients/trame, le cluster commence à être chargé, ainsi il diminue le nombre de serveurs en coopération et le taux d’arrivée des clients CoMP prend les valeurs décroissantes suivantes : $38\% \times \lambda^{clients}$, $36\% \times \lambda^{clients}$, jusqu’à $0\% \times \lambda^{clients}$. Effectivement, lorsque nous passons à forte charge nous augmentons le nombre de clients des macro cellules (macro M1 et macro M2) dans la classe 1 ayant de meilleures conditions radio que ceux servis dans la région de coopération ($CR_{M1,1} \times \log_2(N_{M1,1}) > CR_{c,1} \times \log_2(N_{c,1})$). Ainsi, il est préférable pour les macro cellules d’avoir plus de ressources pour leurs propres clients ; au lieu que les macro cellules coopèrent pour servir ensemble des clients qui ont de moins bonnes conditions radio ; ce qui permet d’atteindre un débit global optimal.

Si nous comparons les résultats du Tableau 5.3 et du Tableau 5.5, nous remarquons clairement que les valeurs de S_c du Tableau 5.5 sont supérieures à celles du Tableau 5.3, car nous avons un taux d'arrivée dans la région de coopération plus élevé.

5.7 Conclusion

Ce chapitre se focalise sur la technique de remédiation aux interférences JT-CoMP pour la coordination des transmissions entre macro cellules.

Après avoir formulé le problème sous forme d'une chaîne de Markov, nous avons proposé deux modèles d'approximation basés sur la théorie des files d'attente. Avec ces modèles nous cherchons à obtenir des approximations pertinentes et rapides à résoudre pour un usage quasi temp réel.

Les deux modèles sont basés sur les files multi-serveurs. Le premier, consiste en une file de type $M/G/S/S$ avec des arrivées simples. Le second modèle consiste en une file $M^{[X]}/G/S/S$ avec des arrivées par batch. Des batchs de tailles différentes permettent en effet de considérer des flux de natures différentes et donc d'introduire la notion de différenciation de service.

Les précisions des modèles proposés ont été validées et comparées entre elles ainsi qu'aux simulations du comportement réel reproduit par la chaîne de Markov. Les paramètres d'entrée de ces modèles (distributions des utilisateurs et conditions radio) ont été extraits de simulations reflétant un comportement réel.

Les deux modèles proposés ont réussi à reproduire le comportement réel du système pour différentes configurations testées.

Par ailleurs, les résultats montrent que le débit global dépend fortement du nombre d'utilisateurs pouvant bénéficier de la coopération inter-cellulaires et du nombre de ressources attribuées aux utilisateurs CoMP.

Ces approximations qui tirent profit de la rapidité et de la simplicité des formes closes obtenues par les modèles de files d'attente, peuvent donc être utilisées afin de déterminer le débit optimal du système. Nous avons proposé dans ce sens un algorithme basé sur ces approximations et dont l'objectif est de décider rapidement de la répartition des ressources disponibles entre les utilisateurs CoMP et les utilisateurs non-CoMP, tout en tenant compte de la charge et des conditions radio.

Les travaux décrits dans ce chapitre ont fait l'objet d'une publication internationale [TCTJA19a]. Dans le cadre des travaux futures, il serait intéressant de comparer nos résultats avec des résultats obtenus des environnements de simulations tels que NS3 ou LTE-Sim utilisant des distributions d'arrivées générales.

Dans le chapitre suivant, nous présenterons une étude comparative des techniques eICIC/ABS et JT-CoMP, en nous basant sur les approximations et les algorithmes d'allocation optimale des ressources proposés jusqu'ici. Cette étude vise à déterminer, pour une configuration donnée, la technique de remédiation aux interférences (eICIC/ ABS ou JT-CoMP) qui permettrait d'atteindre un débit global plus élevé ainsi que la distribution correspondante des ressources entre les cellules

coordonnées. Notons également que ces études nous permettront de définir la charge exercée sur le Fronthaul de l'architecture C-RAN. Une étude dédiée sera présentée dans le dernier chapitre de cette thèse.

Chapitre 6

Etude comparative entre eICIC/ABS et JT-CoMP

6.1 Problématique et contexte

Plusieurs défis restent à résoudre pour que les C-RAN/H-CRAN puissent voir le jour, notamment les problèmes de prise de décisions nécessaires de manière rapide, coopérative, coordonnée et distribuée, permettant de résoudre en particulier les problèmes radio tels que la gestion des interférences. Comme nous l'avons déjà expliqué dans les chapitres précédents, les deux approches les plus prometteuses pour remédier aux interférences sont eICIC/ABS et JT-CoMP (voir la section [2.3.2.1](#) et la section [2.3.2.2.b](#)).

Rappelons que dans les deux chapitres précédents nous avons proposé des outils de dimensionnement pour la technique eICIC/ABS et pour la technique JT-CoMP. L'outil de dimensionnement pour eICIC/ABS (voir la section [4.6](#) du chapitre [4](#)), basé sur les modèles de files d'attente ($M/G/S/S$ et $M^{[X]}/G/S/S$), permet selon la charge du trafic de décider dynamiquement sur l'activation/désactivation des micro cells, sur le nombre d'UE à rattacher à la micro ainsi que sur le nombre optimal de ressources que la macro devrait leur déléguer afin d'assurer un débit global optimal. Quant à l'outil de dimensionnement pour JT-CoMP (voir la section [5.6](#) du chapitre [5](#)), en se basant sur ces modèles de files d'attente, il vise selon la charge du trafic à déterminer de manière dynamique le nombre optimal de clients CoMP et la distribution optimale des ressources radio entre les UE CoMP et non-CoMP afin d'atteindre le débit global optimal.

L'objectif de ce chapitre consiste à déterminer quelle est la technique de remédiation aux interférences, eICIC/ABS ou JT-CoMP, qui permettrait d'assurer un débit global optimal. Autrement dit, pour une charge de trafic donnée serait-il intéressant d'effectuer eICIC/ABS ou JT-CoMP afin d'atteindre le débit global optimal? Nous allons donc faire une étude comparative en considérant des conditions radio réalistes et donner pour la technique optimale la configuration optimale sur l'allocation de ressources et des utilisateurs entre les cellules coordonnées.

Dans cette optique, les auteurs dans [SDR17] présentent une étude comparative entre eICIC/ABS et JT-CoMP. Il a été conclu que JT-CoMP peut améliorer le SINR et la capacité du système par rapport à eICIC/ABS. Une autre étude comparative de ces deux techniques a été faite dans [DMBP13]. Dans ce dernier, les auteurs comparent les performances (en termes d'efficacité spectrale) de JT-CoMP et de eICIC/ABS sous différentes tailles de clusters. Il a été démontré par simulations que JT-CoMP n'apporte pas d'amélioration ou peu par rapport à eICIC/ABS lorsque la taille du cluster est petite. Toutefois, JT-CoMP apporte des gains importants lorsque la taille du cluster est grande. Dans [LJG13], les auteurs proposent une combinaison de JT-CoMP et de eICIC/ABS adaptative. L'idée est de calculer le débit obtenu lorsque eICIC/ABS est appliqué (la macro est silencieuse) et lorsque JT-CoMP est appliqué (la macro et la micro servent leurs UE simultanément). En comparant les deux débits obtenus, la technique offrant le meilleur débit est maintenue. Les résultats de simulations démontrent que la combinaison adaptative des deux techniques offrent des gains de débits de 23% par rapport à eICIC/ABS et 16% par rapport à JT-CoMP. Dans [LLL+12], les auteurs proposent une coordination combinée entre JT-CoMP et eICIC avec LP-ABS (Low Power ABS) dans un réseau hétérogène. Le concept consiste à ce que pendant les ABS (où la micro sert ses UE et la macro est silencieuse), la macro sert ses UE avec une puissance de transmission faible au lieu qu'il n'y ait aucune transmission (ZP-ABS : Zero Power ABS). Afin de remédier aux interférences existantes sur les LP-ABS, les auteurs proposent d'effectuer du JT-CoMP. Les résultats de simulations montrent que la combinaison de ces deux techniques offrent des gains significatifs, en terme de débit, dans le centre et le edge de la cellule. Les auteurs dans [VMT+13] proposent également de combiner JT-CoMP et eICIC/ABS. Leur proposition consiste à appliquer sur toutes les sous-trames JT-CoMP et eICIC/ABS. Selon le type de la sous-trame ABS ou non-ABS, ils retirent les macro (si c'est un ABS) ou les micro (si c'est un non-ABS) de l'ensemble des cellules coopératives. Les résultats de simulations révèlent que l'application conjointe de JT-CoMP et de eICIC/ABS peut fournir une amélioration de débit global supérieure à 20% et une amélioration de débit supérieure à 40% pour les edge UE par rapport à JT-CoMP et eICIC/ABS appliqués individuellement. Dans [MBFP15], les auteurs proposent une utilisation adaptative de JT-CoMP avec eICIC/ABS. Dans ce dernier article, la sélection des UEs et des cellules pour JT-CoMP a été optimisée afin de minimiser l'incapacité du réseau à satisfaire les demandes des UE.

Comme nous pouvons le constater, les auteurs des travaux sus-cités cherchent à choisir la technique de remédiation aux interférences (eICIC/ABS ou JT-CoMP) ou à combiner les deux afin d'offrir les meilleures performances. Toutefois, les comparaisons et les combinaisons n'ont pas été faites en se basant sur un dimensionnement optimal (sauf dans l'article [MBFP15] où uniquement une optimisation de JT-CoMP a été faite). Il est clair que l'utilisation optimale conjointe des techniques de remédiation aux interférences permet d'exploiter leurs avantages et de réduire leurs inconvénients. Notre objectif consiste donc à déterminer la technique de remédiation (eICIC/ABS ou JT-CoMP) qui selon la charge du trafic permettrait d'obtenir un débit global optimal avec une configuration optimale des ressources. Nos outils peuvent être également exploités afin d'effectuer

une combinaison adaptative entre eICIC/ABS et JT-CoMP. Cette combinaison aura plus de sens étant donné qu'elle sera basée sur l'optimisation de ces deux techniques.

6.2 Description des systèmes à comparer

Nous comparons les deux topologies suivantes et nous focalisons notre analyse sur les UEs dans la région centrale, couverte par les deux macro cells centrales (macro M1 et macro M2) :

- **topologie 1** : un C-RAN cluster composé de 2 cellules centrales et de ses 8 cellules voisines du premier tier, numérotés de 3 à 10 comme illustrée dans la FIGURE 6.1. Toutes ces cellules sont des macro cells.

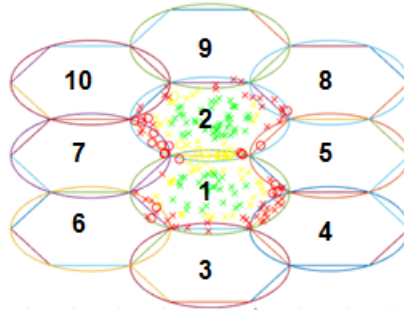


FIGURE 6.1 – C-RAN cluster : 10 macro cells

Dans cette topologie de C-RAN, nous supposons que uniquement les deux macro cells M1 et M2 peuvent coopérer. Elles coopèrent pour servir un UE en appliquant JT-CoMP si et seulement si la coopération améliore les conditions radio de ce UE et par conséquent le débit du UE. Notons que les interférences inter-macro cells sont considérées à travers tout le cluster.

- **topologie 2** : un H-CRAN cluster composé de 2 macro cellules centrales, de ses 8 macro cellules voisines du premier tier, numérotés de 3 à 10 comme illustrée dans la FIGURE 6.2 et de deux micro cellules micro a et micro b (qui n'interfèrent pas entre elles) déployées dans la région de chevauchement de la macro M1 et de la macro M2. Dans cette topologie de H-CRAN, nous supposons que la micro a est rattachée à la macro M1 et que la micro b à la macro M2. La micro a et la macro M1 (resp. la micro b et la macro M2) appliquent eICIC/ABS pour remédier aux interférences. Par ailleurs, les interférences inter-macro sont considérées à travers tout le cluster.

Pour ces deux topologies, nous considérons dans ce qui suit le cas général où les utilisateurs sont générés aléatoirement et uniformément dans la macro M1 et la macro M2 selon une loi de poisson de paramètre $\lambda^{clients}$ tel que :

- Pour la topologie 1 : $\lambda^{clients} = \lambda_{M1}^{clients} + \lambda_{M2}^{clients} + \lambda_c^{clients}$

- Pour la topologie 2 : $\lambda^{clients} = \lambda_{M1}^{clients} + \lambda_{M2}^{clients} + \lambda_{m_a}^{clients} + \lambda_{m_b}^{clients}$

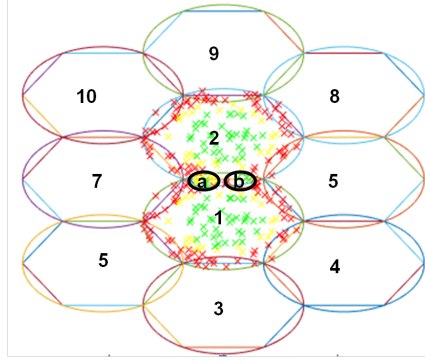


FIGURE 6.2 – H-RAN cluster : 10 macro cells et 2 micro cells

L'étude est bien évidemment faite sur la même génération d'utilisateurs (même emplacement). La serving cell d'un UE est déterminée en fonction de la puissance de réception. Rappelons que pour calculer le nombre de RBG requis par trame par chaque utilisateur u , avec une taille ($size(u)$) en bits, dans le cas de eICIC/ABS et de JT-CoMP, nous utilisons respectivement l'algorithme [1](#) et l'algorithme [3](#).

6.3 Analyse des performances

Nous énumérons dans le Tableau [6.1](#) (resp. le Tableau [6.2](#)), l'ensemble des paramètres extraits des simulations où une taille de paquet $size(u) = 1024$ bits est considérée (se référer à la section [5.3](#) (resp. la section [4.3](#)) pour plus de détails) et $\epsilon = 0.02$ (nous rappelons que la valeur d'épsilon garantie un service pour les clients de la micro, dans le cas de eICIC, ou les utilisateurs CoMP, dans le cas de JT-CoMP, égale à $(1 - \epsilon)\%$). Ces paramètres sont utilisés afin de déterminer la distribution optimale des ressources entre les utilisateurs CoMP UE et les utilisateurs non-CoMP (resp. entre la macro M1 et la micro a et entre la macro M2 et la micro b) pour atteindre le débit global optimal, dans la topologie 1 (resp. la topologie 2) illustrée dans la FIGURE [6.1](#) (resp. la FIGURE [6.2](#)).

Gardons en tête qu'en comparant les paramètres du Tableau [6.1](#) avec ceux du Tableau [6.2](#), nous remarquons que $CR_{m,1} > CR_{c,1}$ et que $N_{m,1} > N_{c,1}$, ainsi un UE servi par l'une des micro (a ou b) en appliquant eICIC/ABS aura de meilleures conditions radio, et par conséquent un meilleur débit, que s'il serait servi en coopération en appliquant JT-CoMP entre la macro M1 et la macro M2.

Etant donné que la résolution de la $M/G/S/S$ se fait en temps réel, dans ce qui suit nous utilisons sa forme close afin de calculer le débit global dans les deux topologies.

Dans la topologie 1 (voir FIGURE [6.1](#)), le débit de sortie binaire dans la macro M1 : $X_{M1}^{bits} = \sum_{j=1}^{J_{M1}} X_{M1,j}^{bits}$, dans la macro M2 : $X_{M2}^{bits} = \sum_{j=1}^{J_{M2}} X_{M2,j}^{bits}$, et celui dans la région de coopération : $X_c^{bits} = X_{c,1}$ tel que les débits de sortie binaire par classe dans la macro M1, la macro M2 et

Paramètre	Explication	Valeur
topologie 1 : C-RAN		
$\lambda_{M1,1}^{clients}$ (resp. $\lambda_{M2,1}^{clients}$)	Taux d'arrivée des clients dans la classe 1 de la macro M1 (resp. macro M2)	$24\% \times \lambda^{clients}$
$\lambda_{M1,2}^{clients}$ (resp. $\lambda_{M2,2}^{clients}$)	Taux d'arrivée des clients dans la classe 2 de la macro M1 (resp. macro M2)	$21\% \times \lambda^{clients}$
$\lambda_{c,max}^{clients}$	Taux d'arrivée maximal des clients dans la classe 1 de la région de coopération	$10\% \times \lambda^{clients}$
$CR_{M1,1}$ (resp. $CR_{M2,1}$)	Taux de codage moyen des clients de la classe 1 de la macro M1 (resp. macro M2)	0.7
$N_{M1,1}$ (resp. $N_{M2,1}$)	Nombre de points de constellation moyen des clients de la classe 1 de la macro M1 (resp. macro M2)	42
$CR_{M1,2}$ (resp. $CR_{M2,2}$)	Taux de codage moyen des clients de la classe 2 de la macro M1 (resp. macro M2)	0.44
$N_{M1,2}$ (resp. $N_{M2,2}$)	Nombre de points de constellation moyen des clients de la classe 2 de la macro M1 (resp. macro M2)	4
$CR_{c,1}$	Taux de codage moyen des clients de la classe 1 de la région de coopération	0.49
$N_{c,1}$	Nombre de points de constellation moyen des clients de la classe 1 de la région de coopération	16

Tableau 6.1 – Paramètres extraits des simulations de la topologie 1 (voir FIGURE 6.1)

Paramètre	Explication	Valeur
topologie 2 : H-CRAN		
$\lambda_{M1,1}^{clients}$ (resp. $\lambda_{M2,1}^{clients}$)	Taux d'arrivée des clients dans la classe 1 de la macro M1 (resp. macro M2)	$24\% \times \lambda^{clients}$
$\lambda_{M1,2}^{clients}$ (resp. $\lambda_{M2,2}^{clients}$)	Taux d'arrivée des clients dans la classe 2 de la macro M1 (resp. macro M2)	$23\% \times \lambda^{clients}$
$\lambda_{m,max}^{clients}$	Taux d'arrivée maximal des clients dans la classe 1 de la micro (a ou b)	$3\% \times \lambda^{clients}$
$CR_{M1,1}$ (resp. $CR_{M2,1}$)	Taux de codage moyen des clients de la classe 1 de la macro M1 (resp. macro M2)	0.7
$N_{M1,1}$ (resp. $N_{M2,1}$)	Nombre de points de constellation moyen des clients de la classe 1 de la macro M1 (resp. macro M2)	42
$CR_{M1,2}$ (resp. $CR_{M2,2}$)	Taux de codage moyen des clients de la classe 2 de la macro M1 (resp. macro M2)	0.44
$N_{M1,2}$ (resp. $N_{M2,2}$)	Nombre de points de constellation moyen des clients de la classe 2 de la macro M1 (resp. macro M2)	4
$CR_{m,1}$	Taux de codage moyen des clients de la classe 1 de la micro	0.65
$N_{m,1}$	Nombre de points de constellation moyen des clients de la micro	35

Tableau 6.2 – Paramètres extraits des simulations de la topologie 2 (voir FIGURE 6.2)

la région de coopération sont calculés respectivement par l'équation (5.36), l'équation (5.37) et l'équation (5.38). Ensuite, les débits de sortie binaires dans la macro M1, X_{M1}^{bits} , dans la macro M2, X_{M2}^{bits} et dans la région de coopération, X_c^{bits} , sont calculés respectivement par l'équation (5.24), l'équation (5.26) et l'équation (5.28).

Quant à la topologie 2 (voir FIGURE 6.2), le débit global est calculé par l'équation (6.1).

$$X^{bits} = X_{M1}^{bits} + X_{m_a}^{bits} + X_{M2}^{bits} + X_{m_b}^{bits} \quad (6.1)$$

tel que $X_{M1}^{bits} = \sum_{j=1}^{J_{M1}} X_{M1,j}^{bits}$ (resp. $X_{M2}^{bits} = \sum_{j=1}^{J_{M2}} X_{M2,j}^{bits}$) est le débit de sortie binaire dans la macro M1 (resp. macro M2) et $X_{m_a}^{bits} = X_{m_a,1}^{bits}$ (resp. $X_{m_b}^{bits} = X_{m_b,1}^{bits}$) est débit de sortie binaire dans la micro a (resp. la micro b). Les débits de sortie binaires par classe dans la macro M1 et la macro M2 sont calculés par l'équation (4.19), l'équation (4.21) est utilisée pour calculer le débit de sortie binaire de la micro a et de la micro b.

Vu que nous considérons une distribution uniforme, les taux d'arrivée de clients, le taux de codage et le nombre de points de constellation par classe dans la macro M1 et la macro M2 (resp. la micro a et la micro b) sont les mêmes (voir Tableau 6.2). Ainsi, nous avons $X_{M1}^{bits} = X_{M2}^{bits}$ et $X_{m_a}^{bits} = X_{m_b}^{bits} = X_m^{bits}$. Le débit de sortie binaire global pour cette topologie est alors formulé par l'équation (6.2).

$$X^{bits} = 2 \times X_{M1}^{bits} + 2 \times X_m^{bits} \quad (6.2)$$

où X_{M1}^{bits} et X_m^{bits} sont calculés à partir de l'équation (4.20) et l'équation (4.22).

Afin de comparer les deux topologies sous différentes charge, nous résolvons :

- le problème d'optimisation formulé par l'équation (5.42) (tel que $f(S_M, S_c, \lambda_c^{clients}, \lambda_{M1}^{clients}, \lambda_{M2}^{clients}) = X$ où X , le débit global exprimé en kbits/trame, est calculé à partir de l'équation (5.29) pour la topologie 1 appliquant JT-CoMP ;
- le problème d'optimisation formulé par l'équation (4.34) (tel que $f(S_m, S_M, \lambda_m^{clients}, \lambda_M^{clients}) = X$ où X , le débit global exprimé en kbits/trame, est calculé à partir de l'équation (4.23)) pour la topologie 2 appliquant eICIC/ABS.

Ensuite, dans le Tableau 6.3, nous résumons les configurations optimales pour les deux topologies en fonction du taux d'arrivée total des clients dans le centre du cluster considéré. En effet, pour chaque taux d'arrivée de clients, $\lambda^{clients}$, nous y représentons :

- Le taux d'arrivée exprimé en kbits/trame pour les deux topologies. Pour un même taux d'arrivée de clients, nous pouvons avoir des taux d'arrivée (qui représentent la conversion des taux d'arrivée de clients exprimés en clients/trame en kbits/trame) différents. Ceci est dû au fait, que pour chaque topologie, nous avons des conditions radio (taux de codage et nombre de points de constellation) différentes comme listé dans le Tableau 6.1 et le Tableau 6.2. Par exemple, $\lambda^{clients} = 100$ clients/trame représente 90 kbits/trame dans la topologie 1 et 92

kbits/trame dans la topologie 2 ;

- Le débit global optimal X (exprimé en kbits/trame) obtenu pour la topologie 1 et la topologie 2 ;
- La configuration optimale pour la topologie 1 (ie. le nombre de serveurs optimal à déléguer aux CoMP-UE, S_c , et le taux d'arrivée de clients à servir en coopération, $\lambda_c^{clients}$) et pour la topologie 2 (ie. le nombre d'ABS optimal et le taux d'arrivée de clients à servir dans la micro qui est le même dans micro a et dans la micro b).

$\lambda^{clients}$	Taux d'arrivée		X optimal		Configuration optimale			
	Topo 1	Topo 2	Topo 1	Topo 2	Topo 1		Topo 2	
					S_c	$\lambda_c^{clients}$	n_{ABS}	$\lambda_m^{clients}$
10	9	9	9	9	17	$10\% \times \lambda^{clients}$	1	$3\% \times \lambda^{clients}$
20	18	18	18	18	22	$10\% \times \lambda^{clients}$	1	$3\% \times \lambda^{clients}$
30	27	27	27	27	26	$10\% \times \lambda^{clients}$	1	$3\% \times \lambda^{clients}$
100	90	92	90	92	46	$10\% \times \lambda^{clients}$	1	$3\% \times \lambda^{clients}$
150	135	137	135	137	55	$10\% \times \lambda^{clients}$	1	$3\% \times \lambda^{clients}$
200	180	183	180	183	67	$10\% \times \lambda^{clients}$	1	$3\% \times \lambda^{clients}$
380	342	340	342	340	100	$10\% \times \lambda^{clients}$	0	$0\% \times \lambda^{clients}$
500	451	447	451	447	84	$10\% \times \lambda^{clients}$	0	$0\% \times \lambda^{clients}$
600	541	537	537	536	71	$10\% \times \lambda^{clients}$	0	$0\% \times \lambda^{clients}$
700	626	626	577	577	0	$0\% \times \lambda^{clients}$	0	$0\% \times \lambda^{clients}$
800	716	758	584	601	0	$0\% \times \lambda^{clients}$	1	$3\% \times \lambda^{clients}$
900	805	853	586	609	0	$0\% \times \lambda^{clients}$	1	$3\% \times \lambda^{clients}$
1000	895	948	586	616	0	$0\% \times \lambda^{clients}$	1	$3\% \times \lambda^{clients}$
2000	1790	1874	588	630	0	$0\% \times \lambda^{clients}$	1	$2\% \times \lambda^{clients}$
3000	2684	2844	588	627	0	$0\% \times \lambda^{clients}$	3	$3\% \times \lambda^{clients}$
4000	3579	3792	588	692	0	$0\% \times \lambda^{clients}$	3	$3\% \times \lambda^{clients}$

Tableau 6.3 – Comparaison en termes de débit optimal entre la topologie 1 et la topologie 2

D'après les résultats du Tableau 6.3, nous remarquons ce qui suit :

- Lorsque la région centrale est légèrement chargée ($\lambda^{clients} < 200$ clients/trame), eICIC/ABS offre un meilleur débit global optimal avec le déploiement des micro cells en leur attribuant 1 ABS chacune. Ceci est expliqué par le fait que les 3% de clients servis par chaque micro (et donc 6% dans les deux micro) auront de meilleures conditions radio que les 10% de clients servis en coopération ($CR_{m,1} > CR_{c,1}$ et $N_{m,1} > N_{c,1}$, voir le Tableau 6.2), ainsi un meilleur débit global est atteint.
- Lorsque la région centrale devient moyennement chargée ($380 \leq \lambda^{clients} < 700$ clients/trame), le modèle d'optimisation appliquée sur la topologie 2 indique qu'il serait préférable de ne pas

déployer les micro cells. Rappelons que, comme nous l'avons déjà expliqué dans la section 4.6, pour de telles charges, contrairement aux micro cells (servant chacune au maximum 3% du taux d'arrivée global), les macro cells sont chargées. Par conséquent, le déploiement des micro cells en leur attribuant même 1 ABS (le minimum) ne va que pénaliser les clients servis par la macro M1 et la macro M2 diminuant ainsi le débit global optimal. De plus, en comparant les performances entre les configurations optimales des deux topologies (voir Tableau 6.3), JT-CoMP offre le meilleur débit global optimal. Ce résultat est expliqué par le fait que 10% du taux d'arrivée des clients, seront servis avec de meilleures conditions radio dans la région de coopération que s'ils étaient servis dans la classe 2 de la macro M1 et de la macro M2 dans la topologie 2 étant donné que la micro n'est pas déployée. Effectivement, $CR_{c,1} > CR_{M1,2}$ (de même $CR_{c,1} > CR_{M2,2}$) et $N_{c,1} > N_{M1,2}$ (de même $CR_{c,1} > CR_{M2,2}$), comme mentionné dans le Tableau 6.1, offrant ainsi un meilleur débit global optimal.

- Lorsque le taux d'arrivée des clients $\lambda^{clients} = 700$ clients/trame, les modèles d'optimisation appliqués sur ces deux topologies indiquent que le débit global optimal est atteint lorsqu'aucune de ces deux techniques de remédiation aux interférences n'est appliquée.
- Lorsque le taux d'arrivée des clients $\lambda^{clients} \geq 800$ clients/trame, c'est la topologie 2 et donc eICIC/ABS qui offre le débit global optimal. Dans la topologie 1, ce résultat est dû au fait que servir un client avec JT-CoMP nécessite l'occupation d'un serveur par macro cell ceci va donc pénaliser le service de deux clients qui offrent un meilleur débit global. Dans la topologie 2, pour ces taux d'arrivée des clients, les deux macro cellules sont très chargées alors que les deux micro sont modérément chargées. Ainsi, la configuration optimale des micro cellules va permettre d'améliorer le débit global.

D'après cette comparaison, nous pouvons donc conclure que selon la charge du trafic (faible/moyenne charge), la technique de remédiation aux interférences eICIC/ ABS ou JT-CoMP permet d'atteindre le débit global optimal. Par contre, quand le taux d'arrivée des clients est grand (vers des valeurs plus grandes de $\lambda^{clients}$), l'application de eICIC/ABS offre le débit global optimal, et par rapport aux autres cas la différence est plus importante qu'avec JT-CoMP.

6.4 Conclusion

Dans ce chapitre, nous avons présenté une étude comparative entre les mécanismes eICIC/ABS et JT-CoMP. Ces études ont particulièrement concerné l'analyse de l'apport de ces techniques en termes d'amélioration de débit maximal (capacité).

Nous avons pour cela exploité les modèles et algorithmes proposés dans les deux chapitres précédents, avec les approximations basées sur les modèles de files d'attente et qui offrent des solutions rapides en termes de résolution.

Ces études ont été menées sur la zone d'intersection entre deux cellules (au sein d'un cluster). Chacune des techniques apporte une solution de coopération différentes :

- soit l'utilisation d'une sous partie des ressources disponibles pour créer des micro-cellules,
- soit l'utilisation d'une partie des ressources d'une manière conjointe entre les deux macro cellules.

Les études que nous avons menés ont permis de déterminer, pour différentes charges et distributions de trafic, la technique de remédiation aux interférences qui permet d'atteindre un débit global optimal. La distribution optimale des ressources radio entre les cellules coordonnées est également déterminée.

Ces études pourront rapidement être étendues pour considérer des aspects énergétiques et de coût (de déploiement). Il nous semble en effet intéressant de considérer dans cette étude la consommation énergétique. L'objectif d'une telle étude serait d'introduire un compromis entre l'amélioration globale du débit et la consommation énergétique.

Par ailleurs, il serait intéressant d'effectuer une étude comparative sur un cluster se composant de plusieurs tiers. Dans cette étude, nous considérerons un C-RAN où les macro cells peuvent coopérer et un H-CRAN où nous déploierons plusieurs micro cellules couvrant toute la région de coopération. L'objectif de cette étude serait de déterminer quelle topologie offrirait le meilleur compromis entre la consommation énergétique et le débit global optimal. Dans le H-CRAN, nous pourrions considérer des micro cellules qui se chevauchent engendrant ainsi des interférences (même entre elles).

Afin de remédier aux interférences entre les micro cellules, nous proposerons d'appliquer du JT-CoMP entre micro cells pendant les ABS. L'idée est de proposer d'abord une configuration optimale entre eICIC/ABS et JT-CoMP assurant un compromis entre débit global et consommation énergétique. Ensuite, si eICIC/ABS est la technique optimale, dans chaque ABS, nous appliquerons conjointement JT-CoMP et eICIC/ABS en retirant les macro cellules de l'ensemble des cellules coopératives.

D'un autre côté, il paraît évident que la croissance du débit sur la partie radio ainsi que l'utilisation des techniques de coopération ont un impact mais aussi des exigences sur le réseau de transport intra-cellulaire et entre les cellules et les BBU. Ces aspects feront l'objet du chapitre suivant qui concerne la modélisation et le dimensionnement du Fronthaul.

Chapitre 7

Dimensionnement du fronthaul dans les C-RAN/ H-CRAN

7.1 Problématique et contexte

Dans une architecture C-RAN (voir la section [2.3](#) pour plus de détails), les traitements du BBU peuvent être mis en fonction sur du matériel standard et exécutés sur des machines virtuelles, bénéficiant en outre de la virtualisation logicielle et de la NFV (Network Function Virtualization). De plus, le BBU pool peut être partagé entre des clusters de RRH, ce qui réduit les coûts et facilite la coopération, la coordination et le traitement multi-points des cellules.

Dans cette optique, un certain nombre de protocoles de transmission au niveau du FH sont proposés ou sont à l'étude pour le transport d'échantillons I/Q (In-phase/Quadrature) dans une architecture C-RAN totalement centralisée, telle que l'interface CPRI (Common Public Radio Interface) [\[CPR15\]](#), Open Base Station Architecture Initiative (OBSAI) [\[I⁺16\]](#) et Open Radio equipment Interface (ORI) [\[Rya16\]](#). Malgré les avantages de l'architecture C-RAN entièrement centralisée, l'une des principales limitations pouvant ralentir son déploiement réside dans la capacité très élevée [\[CPR15\]](#). Effectivement, le transfert de tout le traitement de la bande de base vers le BBU pool implique la transmission du signal échantillonné. De plus, un transport de débit continu est requis, que le trafic utilisateur soit présent ou non. Par exemple, et d'après [\[Nik15\]](#), le FH devrait acheminer environ 1 Gb/s d'informations pour acheminer le trafic correspondant à un flux d'utilisateur final à 75 Mb/s. De plus, comme une réponse ACK/ NACK est attendue dans les 4 ms pour chaque MAC PDU transmise [\[Nik15\]](#), il existe également une très forte exigence de latence sur le FH.

Afin d'atténuer la contrainte de capacité excessive sur le FH, des solutions telles que la réduction du taux d'échantillonnage du signal, l'application d'une quantification non linéaire, la compression de sous-porteuse de fréquence et la compression de données I/Q ont été proposées dans la littérature, telle que [\[CD11\]](#). Alternativement, la modification de l'architecture fonctionnelle partagée actuelle entre RRH et BBU a été considérée comme l'une des solutions prometteuses pour surmonter ces

exigences de bande passante élevée et de latence réduite. Les organismes de normalisation tels que le 3GPP et le groupe de travail NGFI (IEEE Next Generation Fronthaul Interface) étudient activement une telle architecture fonctionnelle partagée. En effet, en plaçant progressivement plus de traitement sur les RRH, les besoins en capacité de FH peuvent être réduits. Cependant, des RRH plus complexes et coûteuses sont nécessaires et les possibilités d'opérations de coopération et de coordination entre cellules sont réduites.

Une autre question clé est de savoir comment les informations entre la RRH et le BBU pool sont acheminées sur la liaison FH. En effet, selon le niveau du split fonctionnel, les données transportées sur le lien FH peuvent avoir différents formats. Le groupe de travail IEEE NGFI est en train de définir l'architecture des réseaux de transport FH et les mécanismes à la fois pour encapsuler et mapper ce trafic dans des paquets Ethernet [BVC^M+17]. Néanmoins, contrairement à l'architecture C-RAN considérée par 3GPP [3GP17], où chaque RRH est connectée directement au BBU pool, tous les flux de RRH sont maintenant agrégés et multiplexés via le switch Ethernet [E⁺17] où les paquets sont commutés et/ou routés vers/depuis la RRH/le BBU pool comme illustré dans la FIGURE 7.1

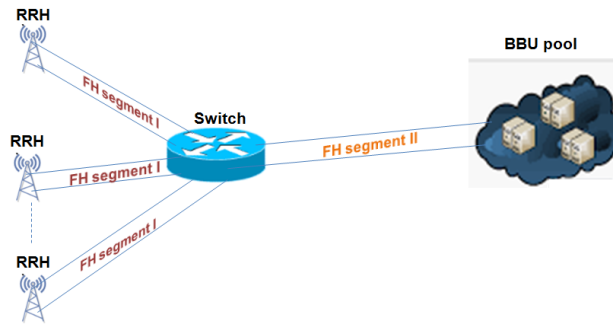


FIGURE 7.1 – Cas d'étude de la topologie du FH du C-RAN

De ce fait, l'utilisation d'Ethernet, entre le BBU pool et les RRH, peut présenter de nombreux avantages, notamment l'utilisation d'équipements moins coûteux, l'utilisation partagée d'infrastructures avec des réseaux d'accès fixes, pour obtenir un multiplexage statistique et des performances optimisées. Cependant, il existe un certain nombre de défis : des exigences de débit binaire ultra élevé provenant du transport de flux radio à bande passante accrue pour plusieurs antennes dans les futurs réseaux de téléphonie mobile, et une latence et une gigue faibles pour répondre aux exigences de délai et aux exigences de traitement commun [GCT⁺15]. Par conséquent, la politique de planification des paquets au niveau du switch Ethernet a un impact sur la latence et l'équité pour toutes les RRH connectées.

Dans ce chapitre, nous limitons notre étude à la liaison montante (UL) du transport FH basé sur Ethernet dans une architecture C-RAN, car le traitement nécessaire à cet effet impose des délais plus stricts que ceux de la liaison descendante (DL) [DDM⁺13]. Nos principales contributions sont :

- Proposer un split alternatif en UL, notée II_U , dans un transport FH basé sur Ethernet ;

- Evaluer les besoins en capacité sur le FH selon le split proposé ;
- Proposer des formules mathématiques lorsque les RRH sont synchronisées afin de déterminer le nombre maximal de RRH à merger qui répond aux contraintes de latence et à la capacité du switch ;
- Analyser le split proposé II_U pour les situations de canal à atténuation rapide (fast fading) et lente (slow fading) et comparer les résultats de performance de ces variantes avec ceux du split physique eCPRI normalisé I_U sous une grande variété de scénarios (différentes bandes passantes et différentes capacités du switch).

7.2 Etat de l’art et revue des splits fonctionnels

L’évolution du FH vers une architecture à base de paquets a été récemment considérée par plusieurs activités de normalisation. Dans [CLYHL19], les auteurs présentent les principes de conception de Ethernet-transported NGFI ainsi que certains scénarios d’application et les résultats des mesures de performances du réseau. Dans [Mai14], la spécification IEEE 1904.3 des encapsulations RoE (Radio over Ethernet) et mappings est présentée. La coopération industrielle (ie. NEC, Nokia, Huawei et Ericsson) impliquée dans la spécification du CPRI a récemment publié une spécification CPRI évoluée (eCPRI) [E+17] pour les splits fonctionnels. Ces splits sont illustrées dans la FIGURE 7.2

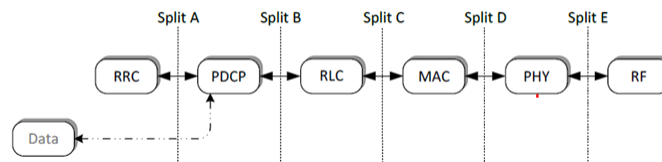


FIGURE 7.2 – Décomposition fonctionnelle au niveau de la couche RAN [E+17]

D’après la FIGURE 7.2, le groupe de travail NGFI a proposé cinq options de splits fonctionnels représentées par des lignes verticales et nommées : Split A, Split B, Split C, Split D et Split E. Les fonctions à droite des lignes verticales seront les fonctions implémentées dans la RRH et les fonctions à gauche seront exécutées dans le BBU pool. Par exemple, le split A dispose d’un RRC central dans le BBU pool et le reste de la pile de protocoles est distribué dans les RRH. Il s’agit simplement d’un exemple montrant comment cette figure doit être interprétée. Les fonctions laissées dans la RRH sont très proches des utilisateurs car elles seront situées dans le mât d’antenne, quant aux fonctions situées dans le BBU pool, elles bénéficieront de la centralisation du traitement et des puissances de traitement élevées dans le BBU pool. Plus il y a de fonctions situées dans la RRH, plus de traitements sont effectués avant que les données ne soient transmises sur le fronthaul, engendrant ainsi un débit binaire amoindri sur le réseau fronthaul.

Dans [CYRN+16], les auteurs analysent le peak rate pour différentes options de splits. Les mêmes auteurs considèrent dans [CNS16] l’impact du multiplexage temporel et fournissent un algorithme

de paquets et une politique de planification des paquets (packet scheduling policy). Dans la littérature, il est admis que le fait d'ajouter plus de fonctionnalités aux RRH (avec des options de partage de niveau supérieur) augmenterait certainement l'efficacité de la bande passante, mais réduirait les avantages du traitement radio centralisé et coopératif en bande de base. Par conséquent, la centralisation flexible ou partielle en effectuant des splits au niveau de la couche physique offre un compromis entre ce que l'on gagne en termes d'assouplissement des exigences sur le FH et ce que l'on perd en termes de fonctionnalités et d'avantages du C-RAN [E+17]. Par conséquent, notre proposition porte sur les splits de la couche physique eCPRI, comme indiqué dans la FIGURE 7.3. Les fonctions de traitement les plus proches des ports d'antenne sont situées dans la partie inférieure et, en montant, le signal subit de plus en plus de traitements avant d'être envoyé sur le fronthaul.

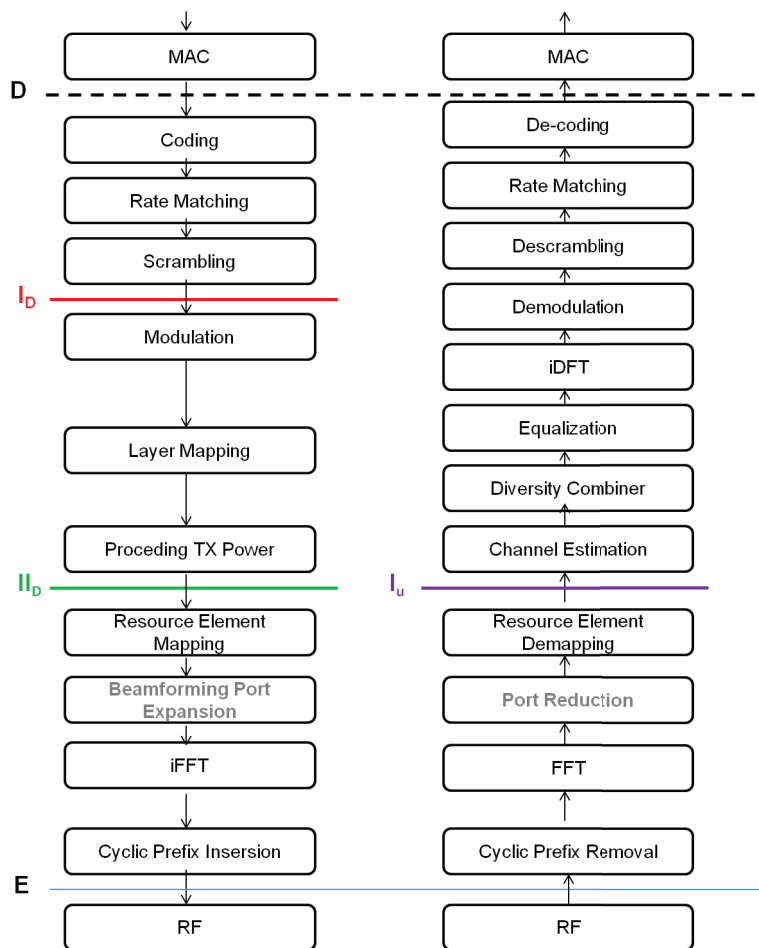


FIGURE 7.3 – Splits eCPRI de la couche physique [E+17]

Pour chaque split de la FIGURE 7.3 (Split E, Split I_U, Split I_D et Split II_D), les fonctions situées au-dessous de la ligne horizontale seront les fonctions implémentées dans la RRH et les fonctions

situées au-dessus seront exécutées dans le BBU pool.

Le split E consiste à transmettre le signal échantillonné. Il nécessite un transport de débit continu, que le trafic utilisateur soit présent ou non. Ceci est considéré comme l'interface de split la plus agressive, où presque toutes les fonctions PHY et des couches supérieures sont centralisées dans le BBU pool. Ce split agressif permet d'obtenir le maximum d'avantages de la centralisation. Cependant, il génère une énorme transmission à débit constant sur le FH qui peut être exprimée comme suit [PHL18] :

$$D_{splitE} = f_s \times 2 \times N_{bits} \times N_{ant} \quad (7.1)$$

où f_s représente la fréquence d'échantillonnage qui dépend de la bande passante considérée, les fréquences d'échantillonnage pour les bandes passantes 1.4, 3, 5, 10, 15, et 20 MHz sont 1.92, 3.84, 7.68, 15.36, 23.04 et 30.72 MHz, respectivement. Le facteur de 2 est dû à la nature complexe des échantillons I/Q. N_{bits} et N_{ant} représentent le nombre de bits par échantillon I/Q et le nombre d'antennes, respectivement.

Une première étape pour remédier au débit énorme exigé par le split E serait de se ramener au domaine fréquentiel (split I_U) suite à l'exécution de la FFT au niveau de la RRH. Ensuite, d'après les spécifications eCPRI, tous les PRB peuvent être démappés. Par conséquent, le débit résultant avec le split I_U dépend maintenant de la fraction η des PRB occupés. Ainsi, le débit exigé par ce split est exprimé comme suit [PHL18] :

$$D_{splitI_U} = N_{sc} \times (t_s)^{-1} \times \eta \times 2 \times N_{bits} \times N_{ant} \quad (7.2)$$

où N_{sc} représente le nombre de sous-porteuses, t_s représente la durée du symbole et η la fraction des ressources blocs occupés, pour le dimensionner au pire cas elle prend la valeur 1 (tous les PRB sont occupés).

Dans la littérature, le modèle de file d'attente M/G/1 est largement adopté pour évaluer les performances du FH en termes de délais [MMS18] [GHLK17]. Dans [CFBF19], les auteurs modélisent le trafic généré par le MIMO massif (entrées et sorties multiples) et mappent la file d'attente au niveau du switch avec des arrivées poissoniennes et un processus de service hyperexponentiel. Suivant la théorie des files d'attente, cela peut être noté par le modèle de file d'attente M/HE/1. Dans [PHL17], les auteurs démontrent que le temps entre les arrivées n'est pas distribué de manière exponentielle (sauf si un nombre suffisant de flux à débit constant et non synchrones sont mergés). Ainsi, ils proposent de modéliser le trafic généré sur le fronthaul en utilisant le modèle de file d'attente G/G/1.

7.3 Proposition d'un nouveau split physique en UL

Afin de faciliter la compréhension de cette section, nous énumérons les paramètres les plus importants, dans le Tableau 7.1.

T_{FH}	Délai total moyen sur le FH
T_1	Délai de propagation sur le FH segment I
T_2	Délai de propagation sur le FH segment II
R_S^{max}	Temps de séjour maximal dans le switch
T_S	Temps de service constant dans le switch
B	Nombre de MTU dans un burst
$W_{S,i}$	Temps d'attente de chaque unité i du batch dans le switch
W_S	Temps d'attente moyen dans le switch
$W_S^{(p)}$	Temps d'attente du p -ième percentile dans le switch
Nb_{RRHs}	Nombre de RRH mergées dans le switch
D_1	Bitrate requis sur le segment I
D_2	Capacité du segment II
R_S	Temps de séjour moyen dans le switch
$R_S^{(p)}$	Temps de séjour du p ième percentile dans le switch
T_{acq}	Temps d'acquisition, le temps nécessaire pour capturer le signal d'entrée par l'eNB
$T_{Rx,eNB}$	Intervalle de temps entre l'arrivée du paquet à l'eNB et la fin de son traitement
$T_{Tx,eNB}$	Temps requis par l'eNB pour transmettre le ACK/NACK à l'UE
R_S	Temps de séjour moyen dans le switch
$R_S^{(p)}$	Temps de séjour du p ième percentile dans le switch
T_{acq}	Temps d'acquisition, le temps nécessaire pour capturer le signal d'entrée par l'eNB
$T_{Rx,eNB}$	Intervalle de temps entre l'arrivée du paquet à l'eNB et la fin de son traitement
$T_{Tx,eNB}$	Temps requis par l'eNB pour transmettre le ACK/NACK à l'UE

Tableau 7.1 – Paramètres utilisés

7.3.1 Analyse de la contrainte de latence sur le fronthaul

En dépit du débit de données, la latence est la principale contrainte sur le fronthaul dans les architectures C-RAN. En LTE, l'HARQ est utilisé comme mécanisme de contrôle des transmissions de paquets où une transmission sans erreur est nécessaire. Plus précisément, chaque unité de données du protocole MAC (PDU) reçue doit faire l'objet d'un accusé de réception positif ou négatif (ACK'ed ou NACK'ed) dans un délai aller-retour appelé HARQ round trip time ($T_{HARQ} = 8$ ms). Cela signifie qu'un UE qui a transmis un élément de données au niveau de la sous-trame N doit savoir s'il doit retransmettre l'ancienne donnée ou transmettre une nouvelle donnée au niveau de la sous-trame N+8 [Nik15].

La décomposition temporelle HARQ en liaison montante est présentée dans la FIGURE 7.4. Au niveau de l'eNB, chaque PDU MAC envoyée par un UE à la N ème sous-trame est acquise au cours de (T_{acq}), traité en réception au cours de ($T_{Rx,eNB}$) et enfin traité pour la transmission de la réponse ACK / NACK pendant ($T_{Tx,eNB}$).

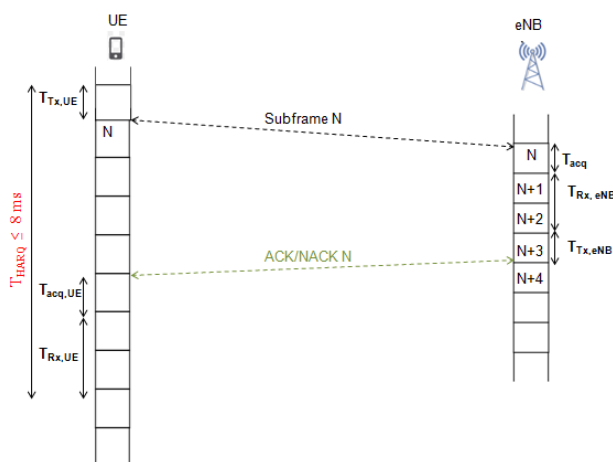


FIGURE 7.4 – Temps HARQ en Uplink

A partir de la FIGURE 7.4, nous déduisons que :

$$T_{acq} + T_{Rx,eNB} + T_{Tx,eNB} \leq \frac{T_{HARQ}}{2} = 4ms \quad (7.3)$$

Nous pouvons également déduire que le temps maximal de réception et de transmission de l'eNB (voir l'équation (7.4)). Pour cela, une limite supérieure de 1 ms pour le temps d'acquisition de la liaison montante est supposée [Nik15]. De plus, étant donné que la plupart des traitements de transmission peuvent être préparés avant la fin de la chaîne de réception, la chaîne de transmission

$(T_{Tx,eNB})$ est indépendante du temps de traitement de la chaîne de réception [CNS16].

$$T_{Rx,eNB} + T_{Tx,eNB} \leq \frac{T_{HARQ}}{2} - T_{acq} = 3ms \quad (7.4)$$

Nous pouvons donc supposer que le temps de traitement maximal de la chaîne de transmission est $T_{Tx,eNB} = 1$ ms [Nik15]. La contrainte peut ensuite être reformulée à l'aide de l'équation (7.5). Notons que cette formulation est basée sur un intervalle de temps de transmission (TTI) égal à 1 ms.

$$T_{Rx,eNB} \leq 3ms - T_{Tx,eNB} = 2ms \quad (7.5)$$

Intéressons nous à présent à la latence exigée par la nouvelle interface de FH NGFI où l'eNB est décomposé en RRH et BBU et les données du UE sont échantillonnées et paquetisées dans la RRH puis transmis, à travers le FH segment I, au switch où différentes RRH sont agrégées et multiplexées puis routées au BBU pool à travers le segment II comme illustré dans la FIGURE 7.1

Dans cette optique, le temps total maximal de traitement de la chaîne de réception en UL dans un C-RAN, $T_{Rx,eNB}$, peut être exprimé par la somme de :

- paquetisation et délai de traitement dans la RRH
- dépaquetisation et délai de traitement dans le BBU pool
- T_{FH} : délai total sur le FH incluant le délai dans le switch

Dans la suite nous notons $T_{(de)pkt}$ la somme des temps de paquetisation/dépaquetisation dans la RRH et le BBU pool et T_{proc} la somme des temps de traitements (baseband processing) au niveau de la RRH et du BBU pool. T_{proc} est exprimé en fonction du nombre de PRB, de l'index du schéma de modulation (MCS) et de la plateforme de virtualisation [Nik15]. Pour des raisons de simplification, nous supposons que toutes les RRH et BBU ont la même plateforme de virtualisation qui considère le temps de traitement le plus long (DOCKER [Nik15]), le nombre maximal de PRB (100 PRB voir Tableau 2.1) et l'index du MCS le plus élevé (27 [Acc11]) afin d'obtenir $T_{proc}(100, 27, Docker)$ maximal (100 et 27 correspondent à 20 MHz and 64 QAM resp.) comme borne supérieure à T_{proc} . Nous reformulons donc l'équation (7.5) comme suit :

$$T_{Rx,eNB} = \max(T_{(de)pkt} + T_{proc}(PRB, MCS, V)) + T_{FH} \leq 2ms \quad (7.6)$$

T_{FH} peut alors être exprimé comme énoncé dans l'équation (7.7) :

$$T_{FH} \leq 2ms - \max(T_{(de)pkt} + T_{proc}(PRB, MCS, V)) \quad (7.7)$$

D'après les recommandations de NGMN (Next Generation Mobile Networks) [All15a], les délais maximaux de paquetisation/dépaquetisation et de traitement dans la RRH et le BBU pool sont au tour de 1.75 ms. Cette valeur est composée de $T_{proc}(100, 27, Docker) = 1562.3 \mu s$ et $T_{(de)pkt} = 187.7$

μs . Ainsi, le délai restant pour le transport sur le FH ne peut excéder $250 \mu s$ (voir l'équation (7.7)). Nous notons alors l'inégalité suivante :

$$T_{FH} = T_1 + R_S + T_2 \leq 250 \mu s \quad (7.8)$$

où T_1 (resp. T_2) représentent les délais sur le FH segment I (resp. FH segment II). T_1 (resp. T_2) peuvent être exprimés comme $\frac{d_1}{c}$ (resp. $\frac{d_2}{c}$) où c réfère à la vitesse de la lumière et d_1 et d_2 à la longueur du FH segment I et du FH segment II respectivement.

Nous supposons que toutes les RRH ont la même distance jusqu'au switch (c'est à dire la même longueur du FH segment I). D'après [All15b], la distance maximale entre la RRH et le BBU pool est supposée être $d_1 + d_2 = 10$ Km. Nous avons ainsi $T_1 + T_2 \leq \frac{10Km}{299792Km/s} \simeq 0.3 \mu s$. Notons R_S l'intervalle de temps entre l'arrivée d'un paquet au niveau du switch et son départ à travers le FH segment II (il représente donc le temps de séjour dans le switch). Par conséquent, la contrainte HARQ est respectée si et seulement si :

$$R_S^{max} = W_S^{max} + T_S \leq 249.7 \mu s \quad (7.9)$$

où R_S^{max} est le temps de séjour maximal dans le switch, W_S^{max} est le temps d'attente maximal et T_S est le temps de service constant dans le switch.

7.3.2 Split II_U : nouveau split physique en UL sur le fronthaul

Afin de réduire les exigences énormes en termes de bande passante de transmission sur le FH, nous proposons le split II_U, un nouveau split en UL s'effectuant entre les fonctions de désembrouillage de canal sans fil et de dé-modulation sans fil au sein de la couche PHY (voir la section 2.3.1.2.a pour plus de détails sur la couche physique) comme illustré dans la FIGURE 7.5. Le split II_U proposé est au même niveau que le split I_D (dans les spécifications eCPRI [E+17]) et l'option 7.3 (dans 3GPP [3GP17]) proposée en DL. Toutefois, le split II_U n'est proposé ni par le groupe de travail NGFI ni par le 3GPP en UL [?]. Dans ce qui suit, nous modélisons, analysons et évaluons cette répartition pour deux scénarios : slow Rayleigh fading (où l'estimation du canal est effectuée par slot) et fast Rayleigh fading (où l'estimation du canal est effectuée par TTI). Rappelons que la durée d'un slot est de 0.5 ms, ce qui correspond à la moitié d'une durée d'un TTI (une sous-trame) [ETS16] (voir FIGURE 2.3). En effet, si le temps d'intervalle est inférieur à la vitesse à laquelle le canal change, alors l'estimation et l'égalisation par slot peuvent être considérées sans erreurs d'estimation résiduelles. En revanche, lorsque le canal change plus fréquemment, il convient d'utiliser une estimation et une égalisation par TTI pour éviter les erreurs d'estimation. Le débit requis sur le FH par le split II_U peut être exprimé comme suit [RWN+17] :

$$D_{splitII_U} = M \times N_{sy} \times N_s \times Nb_{PRB} \times N_{ant} \setminus T \quad (7.10)$$

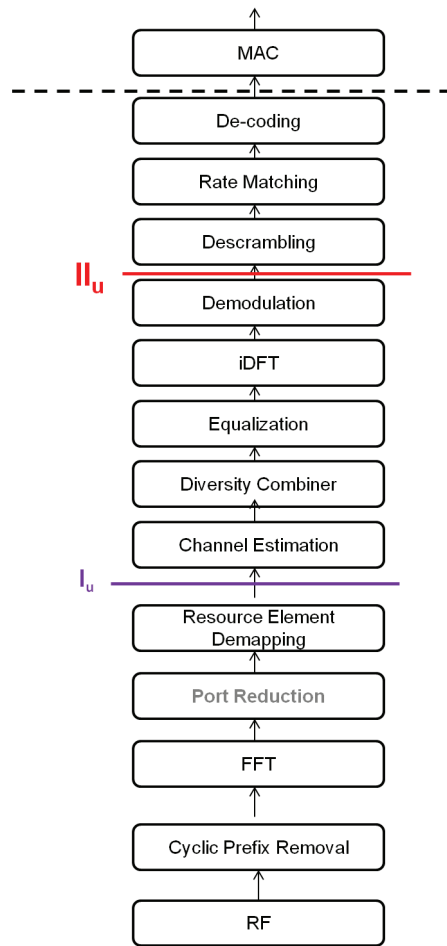


FIGURE 7.5 – Proposed II_U uplink physical layer split

où T est l'intervalle de temps de transmission qui est la durée de transmission du signal radio. Il correspond à une sous-trame pour le fast Rayleigh fading ($T = TTI = 1$ ms) et à un slot pour le slow Rayleigh fading ($T = TTI/2 = 0.5$ ms). M est l'ordre le plus élevé de la modulation du signal radio ($M = 6$ pour 64 QAM). N_{sy} est le nombre de symboles dans l'intervalle de temps de transmission (pour le préfixe cyclique court, $N_{sy} = 14$ OFDM symboles dans un TTI et $N_{sy} = 7$ OFDM symboles dans un slot). N_s est le nombre de sous-porteuses dans un ressource bloc ($N_s = 12$ sous-porteuses) et Nb_{PRB} est le nombre de ressources blocs dans un intervalle de temps de transmission (TTI ou slot) et dépend de la bande passante considérée (voir Tableau 2.1).

À partir des équations (7.1), (7.2) et (7.10), les profils de trafic UL sur le FH peuvent être déduits pour différentes largeurs de bandes passantes LTE avec les splits E, I_U et II_U . Les résultats sont énumérés dans le Tableau 7.2 par TTI pour le fast Rayleigh fading et par slot pour le slow Rayleigh fading. Les profils de trafic sont décrits en termes de taille de burst (en Bytes), de périodicité du burst (en μ s) et de débit binaire résultant (en Mb/s). Notez que la taille des bursts avec le split II_U sous le slow Rayleigh fading correspond à la moitié de la taille du burst avec le split II_U sous fast Rayleigh fading. Cependant, comme ces bursts sont envoyés à chaque slot, ils sont alors envoyés deux fois plus souvent. Gardons en tête qu'avec les options de splits plus élevés, les RRH génèrent des flux de débit plus faibles mais avec plus de burstiness.

	Split E		Split I_U				Split II_U per TTI				Split II_U per slot			
BW [MHz]	10	20	10	20	40	100	10	20	40	100	10	20	40	100
Burst size [Bytes]	30	60	4500	9000	18000	45000	25250	50500	101000	252000	12625	25250	50500	126250
Period [μ s]	0.260416667		66.6				1×10^3				0.5×10^3			
Bit-rate [Mb/s]	921.6	1843.2	540	1080	2160	5400	202	404	808	2020	202	404	808	2020

Tableau 7.2 – Profils de trafic en UL pour les splits physiques, $N_{ant} = 2$ MIMO, $N_{bit} = 15$ bits/sample

D'après le Tableau 7.2, chaque split produit un flux à débit constant (Constant bit rate CBR), avec une taille de burst et une période d'inter-arrivée spécifiques. Par exemple, considérons le flux FH pour un canal LTE à 2×2 MIMO et 20 MHz. Lorsque le split E est appliqué, le flux FH est composé d'un burst périodique de 60 Bytes transmis toutes les 260.416667 ns (durée de la trame CPRI) [CPR15]. Si le split I_U est appliqué, le flux FH est composé d'un burst périodique de 9000 Bytes transmis toutes les 66.6 μ s (durée du symbole OFDM) [ETS16]. Maintenant, lorsque le split II_U est appliqué avec fast Rayleigh fading, le flux FH est composé d'un burst périodique de 50500 Bytes transmis toutes les 1 ms (durée d'un TTI). Enfin, si le split II_U est appliqué avec slow Rayleigh fading, le flux FH est composé d'un burst périodique de 25250 Bytes transmis toutes les 0.5 ms (durée d'un slot) [ETS16].

7.4 Modélisation et analyse des performances

Dans cette section, les résultats de modélisation et de performance du split proposé sont détaillés. Les résultats sont comparés avec le split I_U lorsque les RRH sont synchrones.

Dans ce qui suit, nous supposons que les en-têtes de paquets comprennent :

- Préambule : 8 bytes.
- En-tête Ethernet : 14 bytes.
- Inter-packet Gap (IPG) : 12 bytes
- Cyclic Redundancy Check (CRC) : 4 bytes.
- en-tête eCPRI : 4 bytes.

L'efficacité du schéma de paquétisation (packetization scheme) en termes de délais dépend du nombre de paquets générés à chaque arrivée de burst [PHL17]. Pour plus d'efficacité, il est possible de réduire les overheads des paquets en définissant la taille de la charge utile (payload) sur la valeur MTU (Ethernet Maximum Transfer Unit), c'est-à-dire 1500 Bytes.

Les paquets Ethernet constitueront l'unité desservie par le switch du FH. Soit B la taille des bursts arrivant périodiquement d'une RRH au switch. Chacun de ces bursts sera mis en paquets dans des paquets Ethernet de 1500 Bytes. Il en résulte que $B = \frac{\text{Burst size}}{MTU}$.

Pour chaque split, l'objectif est de déterminer le nombre maximal de secteurs (ie. RRH) pouvant être desservis par le FH tout en veillant à ce que le délai sur le FH ne dépasse pas la contrainte sur le délai exprimée dans l'équation (7.9). Afin de garantir des techniques avancées de coordination et de coopération entre cellules pour les futurs systèmes cellulaires, nous nous intéressons dans ce chapitre aux RRH étroitement synchronisées. Par conséquent, les différentes RRH génèrent des bursts synchronisés avec un temps d'inter-arrivées déterministe. Dans la suite, nous notons l'ensemble de tous les bursts arrivant simultanément au switch par un batch. Le switch reçoit des batchs et assemble/fragmente chaque batch en unités individuelles correspondant à des paquets Ethernet de taille 1500 Bytes. Soit ρ le facteur de charge sur le switch, il est calculé comme suit :

$$\rho = \frac{Nb_{RRHs} \times D_1}{D_2} \quad (7.11)$$

où D_1 représente le débit binaire requis sur le segment I selon le split considéré (voir Tableau 7.2) et D_2 correspond à la capacité du segment II (ce qui revient à la capacité du switch, voir la FIGURE 7.1).

Puisque les arrivées et les départs suivent des lois déterministes, alors, si $\rho < 1$, le switch se videra avant l'arrivée du batch suivant, sinon, il sera surchargé et les délais augmenteront indéfiniment. Pour ce scénario déterministe, le temps d'attente de chaque unité individuelle i du batch (correspondant à un paquet Ethernet de 1500 Bytes) peut facilement être déduit à l'aide de l'équation (7.12).

$$W_{S,i} = (i - 1) \times T_S, \quad 1 \leq i \leq B \times Nb_{RRHs} \quad (7.12)$$

où T_S est calculé comme suit :

$$T_S = \frac{8 \times 1500}{D_2} \quad (7.13)$$

Le temps d'attente moyen peut être exprimé comme suit :

$$W_S = \frac{(Nb_{RRH_s} \times B - 1) \times T_S}{2} \quad (7.14)$$

où Nb_{RRH_s} désigne le nombre de RRH dans le cluster et $Nb_{RRH_s} \times B$ correspond au nombre total d'unités individuelles des batchs entrant dans le switch (à partir de toutes les RRH du cluster).

Rappelons que le délai du p ième percentile est défini comme suit :

$$P(W_{S,i} \leq W_S^{(p)}) = p, \forall i = 1, \dots, Nb_{RRH} \times B \quad (7.15)$$

Ensuite, en raison de la synchronisation des RRH, nous pouvons calculer le délai du p ième percentile en utilisant l'équation suivante :

$$W_S^{(p)} = [p \times Nb_{RRH_s} \times B - 1]^+ \times T_S \quad (7.16)$$

Pour nous concentrer sur les pires délais, nous calculons le temps d'attente des percentiles les plus élevés en définissant p sur 90, 99 et 100. Notons que les temps d'attente ne sont calculés que lorsque $\rho < 1$, c'est-à-dire lorsque le système est stable. Le temps de séjour moyen R_S (resp. le temps de séjour du p ième percentile $R_S^{(p)}$) est ensuite calculé en ajoutant le temps de service constant (T_S) au temps d'attente moyen W_S (resp. au temps d'attente du p ième percentile $W_S^{(p)}$) comme exprimé par les équations (7.17) et (7.18).

$$R_S = W_S + T_S \quad (7.17)$$

$$R_S^{(p)} = W_S^{(p)} + T_S \quad (7.18)$$

Lorsque $p = 100$, le temps de séjour du 100ième percentile correspond au temps de séjour maximal. Ainsi, $R_S^{(100)} = R_S^{max}$.

Dans ce qui suit, nous considérons un switch sur le fronthaul agrégeant un certain nombre de flux, chacun correspondant au trafic UL d'un secteur de cellule (ou RRH). Tous les secteurs du cluster sont synchronisés. Le split proposé est analysé pour des situations de fast et de slow fading. Deux variantes sont proposées pour le split Π_U :

- split Π_U avec slow Rayleigh fading où l'estimation du canal est faite par slot
- split Π_U avec fast Rayleigh fading où l'estimation du canal est faite par TTI

7.5 Résultats numériques

Nous comparons dans cette section les résultats de performance de deux variantes du split II_U à ceux du split physique eCPRI normalisé I_U sous une grande variété de scénarios : différentes capacités de bandes passantes et de liaisons FH (capacités du switch).

7.5.1 Analyse des temps de séjour

Les résultats détaillés où les valeurs moyennes du temps de séjour, ainsi que les 90^{ième}, 99^{ième} et 100^{ième} percentiles sont présentées dans le Tableau 7.4 pour le split I_U , le Tableau 7.5 pour le split II_U par slot et le Tableau 7.6 pour le split II_U par TTI. Différentes capacités de liaison sont considérées allant de 10G à 1T [LdlCdD⁺14b]. Une grande variété de scénarios sont testés, allant des bandes passantes LTE actuelles de 10 MHz - 40 MHz aux bandes passantes futures de 100 MHz. En fonction de la bande passante utilisée (voir le Tableau 7.2), les débits des flux de données pour le split I_U (resp. le split II_U) vont de 540 Mb/s à 5400 Mb/s (resp. de 202 Mb/s à 2020 Mb/s). Le temps d'attente moyen et les percentiles les plus élevés dans le switch sont ensuite calculés à l'aide des équations (7.14) et (7.16). Enfin, le temps de séjour moyen et les percentiles les plus élevés dans le switch sont calculés à l'aide des équations (7.17) et (7.18).

7.5.2 Calcul du nombre maximal de RRH synchrones à merger

En analysant les résultats des tableaux : Tableau 7.4, Tableau 7.5 et Tableau 7.6, le temps de séjour du split II_U par slot est supérieure au temps de séjour du split I_U même si le débit requis par le split II_U par slot est inférieur au débit requis par le split I_U . De plus, le split II_U par TTI engendre la durée de séjour la plus longue. Ceci est dû à la valeur de la taille du burst du split II_U par TTI, qui est plus importante que la valeur de la taille du burst du split II_U par slot et à la valeur de la taille du burst du split I_U (voir le Tableau 7.2). En effet, plus la taille du burst est élevée, plus la variable correspondante B ($B = \frac{\text{Burst size}}{MTU}$) est grande, ce qui implique un temps de séjour moyen et un temps de séjour du $pième$ percentile plus élevés (voir l'équation (7.14) et l'équation (7.16)). Dans ces tableaux, les situations où $\rho > 1$ et $R_S^{(100)} = R_S^{max} > 249.7\mu s$ représentent des scénarios irréalisables. La première implique que la charge du switch dépasse 100 % (le système n'est pas stable), la seconde indique que la contrainte temporelle HARQ n'est pas respectée (voir l'équation (7.9)). Lors de l'analyse des résultats de performance du Tableau 7.4 pour le split I_U , il apparaît que ce split est limité par la contrainte de stabilité, car il nécessite un débit important sur le fronthaul (voir le Tableau 7.2). D'autre part, les performances dans le Tableau 7.5 pour le split II_U par slot et dans le Tableau 7.6 pour le split II_U par TTI montrent que ces deux variantes du split II_U sont limités par la contrainte de temps HARQ qui dépasse 249.7 μs même si la contrainte de stabilité est respectée. La contrainte de stabilité est respectée car le débit requis par le split proposé II_U

est inférieur au débit requis par le split I_U (voir le Tableau 7.2). Cependant, nous notons que pour certains scénarios, le split II_U par slot respecte la contrainte HARQ tandis que le split I_U génère un système non stable.

Par conséquent, un scénario est réalisable si la contrainte de stabilité et la contrainte de temps HARQ sont respectées conformément à l'équation (7.11) et à l'équation (7.18). Dès lors, pour les scénarios réalisables, le nombre maximal de RRH (ou de secteurs) mergées que le FH peut prendre en charge est calculé comme suit :

$$Nb_{RRH}^{max} = \min \left(\frac{D_2}{D_1}, \frac{R_S^{max}}{B \times T_S} \right) \quad (7.19)$$

Par conséquent, le Nb_{RRH}^{max} obtenu par l'équation (7.19) respecte à la fois la contrainte de stabilité et la contrainte de temps HARQ. Un récapitulatif du nombre maximal de RRH possibles pour chaque configuration considérée est fourni dans le Tableau 7.3

BW	D_2	Nb_{RRH}^{max}		
		split I_U	split II_U per slot	split II_U per TTI
10 MHz	10G	18	23	12
	40G	73	92	48
	100G	184	231	122
	200G	369	462	244
20 MHz	40G	36	48	24
	100G	92	122	61
	200G	184	244	122
40 MHz	40G	18	24	12
	100G	46	61	30
	200G	92	122	61
	400G	184	244	122
100 MHz	100G	18	24	12
	200G	36	49	24
	400G	73	99	49
	1T	184	247	123

Tableau 7.3 – Nombre maximal de RRH mergées sous différentes bandes passantes LTE et différentes capacités de switch 40G, 100G, 200G et 1T dans un C-RAN tel que les RRH sont synchrones

Les résultats du Tableau 7.3 montrent que l'application du split II_U par slot permet de merger plus de RRH que le split I_U , ce qui permet de réduire les coûts de déploiement des switchs et des liaisons optiques par rapport au split normalisé I_U . De plus, quand nous considérons le split II_U par TTI, ce dernier ne permet pas certes de merger autant de RRH que le split I_U , mais il peut réduire le débit requis sur le segment I et permet de toute façon de merger un nombre intéressant de RRH.

Par conséquent, le split Π_U par slot et le split Π_U par TTI peuvent être appliqués pour obtenir un temps de latence faible, une bande passante efficace et un déploiement économique sur le FH pour les réseaux cellulaires de nouvelle génération basés sur l'architecture C-RAN.

De plus, grâce à ces résultats, en fonction de la taille des clusters et de la largeur de la bande passante, nous pouvons déterminer la capacité optimale du switch et le split optimal qui réduisent le débit requis sur le segment II et sur le segment I. Par conséquent, en se basant sur nos résultats, il est possible de proposer un dimensionnement du FH avec un temps de latence faible, une bande passante efficace et un déploiement économique sur le FH.

7.6 Conclusion

Le fronthaul peut constituer l'une des composantes les plus coûteuses pour les opérateurs des réseaux de prochaines générations. En effet, pour accompagner l'accroissement des débit et la diversification des usages avec des applications industrielles ou de santé très exigeantes en termes de délais, les contraintes sur cette partie du réseaux sont de plus en plus fortes.

La problématique que nous avons considérée dans ce chapitre est : comment assurer des liens de transport performants et à moindre coût ?

Dans ce chapitre, nous avons considéré le standard eCPRI pour le Fronthaul, dans lequel un switch Ethernet est déployé entre la RRH et le BBU pool. Nous avons proposé dans ce cadre un nouveau split (noté split Π_U) en UL pour les architectures C-RAN, situé sur la partie supérieure de la couche PHY.

Pour concevoir ce nouveau split, nous avons analysé et pris en compte les contraintes de délais basées sur le HARQ et nous avons cherché à maximiser les gains de multiplexage statistique sur le FH.

Afin de permettre la coordination et la coopération entre les cellules avec des mécanismes tels que ceux étudiés dans les chapitres précédents, nous avons considéré des RRH synchrones. Nous avons alors proposé des règles de dimensionnement où pour une bande passante considérée et une capacité de switch donnée, nous pouvons déterminer le nombre maximal de RRH à multiplexer et inversement (et vice versa).

Les résultats obtenus montrent qu'avec le split que nous avons proposé, il est possible de multiplexer plus de RRH tout en réduisant les coûts de déploiement du Fronthaul.

Les travaux de ce chapitre ont fait l'objet de deux publications internationales [\[TCTJA20b\]](#) et [\[TCTJA20a\]](#).

R_S [μs]	3 sectors			18 sectors			19 sectors			37 sectors			38 sectors			46 sectors								
	Mean	90th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th					
10 MHz	10G	6	10.8	10.8	33	58.8	64.8	64.8	8.7	15.6	17.1	17.1	16.8	30	33	33.3	17.25	30.9	33.9	34.2	20.85	37.5	41.1	41.4
	40G	1.5	2.7	2.7	8.25	14.7	16.2	16.2	3.48	6.24	6.84	6.84	6.72	12	13.2	13.32	6.9	12.36	13.56	13.68	8.34	15	16.44	16.56
	100G	0.6	1.08	1.08	3.3	5.88	6.48	6.48	1.74	3.12	3.42	3.42	3.36	6	6.6	6.66	3.45	6.18	6.78	6.84	4.17	7.5	8.22	8.28
	200G	0.3	0.54	0.54	1.65	2.94	3.24	3.24	0.87	1.56	1.71	1.71	1.68	3	3.3	3.33	0.87	1.56	1.71	1.71	0.87	1.56	1.71	1.71
10 MHz	47 sectors			74 sectors			75 sectors			92 sectors			93 sectors											
	Mean	90th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	
	21.3	38.1	42	33.45	60	66	66.6	13.56	24.36	26.76	27	16.62	29.88	32.88	33.12	16.8	30.24	33.24	33.48	16.8	30.24	33.24	33.48	
	8.52	15.24	16.8	13.38	24	26.4	26.64	6.78	12.18	13.4	13.5	8.31	14.94	16.44	16.56	8.4	15.12	16.62	16.74	8.4	15.12	16.62	16.74	
R_S [μs]	3 sectors			18 sectors			19 sectors			37 sectors			38 sectors			46 sectors								
	Mean	90th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	
	2.85	5.01	5.4	16.35	29.31	32.23	32.4	17.25	30.93	34.01	34.2	33.45	60.09	66.08	66.6	33.74	60.09	66.08	66.6	33.74	60.09	66.08	66.6	
	1.14	2.01	2.16	6.54	11.72	12.89	12.96	6.9	12.37	13.60	13.68	13.38	24.04	26.43	26.64	13.74	24.08	27.15	27.36	13.74	24.08	27.15	27.36	
20 MHz	47 sectors			74 sectors			75 sectors			92 sectors			93 sectors											
	Mean	90th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	
	16.98	30.52	33.56	26.7	48.01	52.81	53.28	27.06	48.66	53.52	54	33.18	59.68	65.64	66.24	16.77	30.16	33.18	33.48	16.77	30.16	33.18	33.48	
	8.49	15.26	16.79	13.35	24.01	26.42	26.64	13.53	24.33	26.76	27	16.59	29.84	32.82	33.12	16.59	29.84	32.82	33.12	16.59	29.84	32.82	33.12	
R_S [μs]	3 sectors			18 sectors			19 sectors			37 sectors			38 sectors			46 sectors								
	Mean	90th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	
	5.55	9.87	10.8	32.55	58.47	64.3	64.8	13.74	24.68	27.15	27.36	26.7	48.01	52.81	53.28	27.42	49.31	54.23	54.72	27.42	49.31	54.23	54.72	
	2.22	3.95	4.32	13.02	23.39	25.72	25.92	6.87	12.34	13.57	13.68	13.35	24.01	26.40	26.64	6.86	12.33	13.56	13.68	6.86	12.33	13.56	13.68	
40 MHz	47 sectors			74 sectors			75 sectors			92 sectors			93 sectors											
	Mean	90th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	
	16.95	30.49	33.53	26.67	47.98	52.78	53.28	27.03	48.63	53.49	54	33.15	59.65	65.61	66.24	16.76	30.15	33.16	33.48	16.76	30.15	33.16	33.48	
	8.48	15.24	16.77	13.34	23.99	26.39	26.64	13.52	24.32	26.75	27	16.58	29.8	32.81	33.12	16.58	29.8	32.81	33.12	16.58	29.8	32.81	33.12	
R_S [μs]	3 sectors			18 sectors			19 sectors			37 sectors			38 sectors			46 sectors								
	Mean	90th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	
	5.46	9.78	10.75	32.46	58.38	64.21	64.8	17.13	30.81	33.89	34.2	33.33	59.97	65.96	66.6	17.12	30.8	33.87	34.2	17.12	30.8	33.87	34.2	
	2.73	4.89	5.38	16.23	29.19	32.11	32.4	8.56	15.41	16.94	17.1	16.66	29.98	32.98	33.3	6.67	11.99	13.19	13.32	6.67	11.99	13.19	13.32	
100 MHz	47 sectors			74 sectors			75 sectors			92 sectors			93 sectors											
	Mean	90th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	
	21.17	38.09	41.89	33.32	59.96	65.95	66.6	13.51	24.31	26.74	27	16.57	29.81	32.79	33.12	16.75	30.19	33.15	33.48	16.75	30.19	33.15	33.48	
	8.5	15.23	16.76	13.33	23.98	26.38	26.64	13.51	24.31	26.74	27	16.57	29.81	32.79	33.12	16.75	30.19	33.15	33.48	16.75	30.19	33.15	33.48	

Tableau 7.4 – Valeurs des temps de séjour pour le transport de plusieurs secteurs (RRH) sous différentes bandes passantes et différentes capacités de switches 40G, 100G, 200G et 1T dans une architecture C-RAN avec le split I_U où les RRH sont synchrones.

R_S [µs]	3 sectors			18 sectors			19 sectors			37 sectors			38 sectors			46 sectors								
	Mean	90th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th	Mean	90th	99th	100th					
10 MHz	10G	16.8	29.76	32.4	32.4	97.8	175.56	193.06	194.4	103.2	185.3	203.7	205.2	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	51.45	92.49	101.73	102.6	$R_S^{(100)} > 249.7$	62.25	111.9	123.1	124.2
	40G	4.2	7.44	8.1	8.1	24.45	43.89	48.26	48.6	25.8	46.32	50.94	51.3			20.58	36.99	40.69	41.04		24.9	44.77	49.24	49.68
	100G	1.68	2.98	3.24	3.24	9.78	17.6	19.31	19.44	10.32	18.53	20.37	20.52			10.02	18.5	20.35	20.52		10.29	18.5	20.35	20.52
	200G	0.84	1.49	1.62	1.62	4.89	8.78	9.65	9.72	5.16	8.78	9.65	9.72			92 sectors					93 sectors			
10 MHz	10G	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$
	40G	63.6	114.4	125.8	126.9	100	179.9	197.9	199.8	101.4	182.4	200.6	202.5			124.4	223.7	246.1	248.4		50.28	90.46	99.49	100.5
	100G	25.44	45.75	50.32	50.76	40.02	71.99	79.18	79.92	40.56	72.96	80.25	81			49.74	89.48	98.43	99.36		25.14	45.23	49.75	50.22
	200G	12.72	22.87	25.16	25.38	20.01	35.99	39.59	39.96	20.28	36.48	40.13	40.5			24.87	44.74	49.22	49.68					
20 MHz	10G	7.8	13.92	15.29	15.3	46.05	82.77	91.03	91.8	48.6	87.36	96.08	96.9			94.5	169.9	186.9	188.7		97.05	174.6	192	193.8
	40G	3.12	5.57	6.118	6.12	18.42	33.11	36.41	36.72	19.44	34.94	38.43	38.76			37.8	67.99	74.78	75.48		38.82	69.83	76.81	77.52
	100G	1.56	2.78	3.06	3.06	9.21	16.55	18.21	18.36	9.72	17.47	19.22	19.38			18.9	33.99	37.39	37.74		19.41	34.92	38.4	38.76
	200G	48	86.35	94.98	95.88	75.54	135.9	149.5	151	76.56	137.8	151.5	153			46.95	84.49	92.93	93.84		47.46	85.4	93.94	94.86
40 MHz	10G	15.45	27.69	30.44	30.6	91.95	165.4	181.9	183.6	97.05	174.6	192	193.8			75.54	135.9	149.5	150.9		77.6	139.6	153.6	155
	40G	6.18	11.08	12.18	12.24	36.78	66.16	72.77	73.44	38.82	69.83	76.81	77.52			37.77	67.96	74.76	75.48		38.79	69.8	76.78	77.52
	100G	3.09	5.54	6.09	6.12	18.39	33.08	36.38	36.72	19.41	34.92	38.4	38.76			18.89	33.98	37.38	37.74		19.4	34.9	38.4	38.76
	200G	1.55	2.77	3.05	3.06	9.2	16.54	18.19	18.36	9.71	17.46	19.2	19.38			92 sectors					93 sectors			
40 MHz	10G	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$
	40G	95.94	172.6	189.9	191.8	151	271.8	298.9	301.9	76.53	137.7	151.5	153			93.87	168.9	185.8	187.7		94.89	170.8	187.8	189.7
	100G	47.97	86.32	94.95	95.88	75.51	135.9	149.5	150.9	38.26	68.87	75.75	76.5			46.94	84.47	92.92	93.84		47.45	85.39	93.934	94.86
	200G	23.99	43.16	47.48	47.94	37.76	67.95	74.74	75.48	19 sectors						37 sectors					38 sectors			
100 MHz	10G	15.36	27.6	30.35	30.6	91.86	165.3	181.8	183.6	96.96	174.5	191.9	193.8			94.38	169.9	186.8	188.7		96.93	174.5	191.9	193.8
	40G	7.68	13.8	15.18	15.3	45.93	82.65	90.912	91.8	48.48	87.24	95.96	96.9			47.19	84.93	93.42	94.35		48.47	87.23	95.95	96.9
	100G	3.84	6.9	7.6	7.65	22.96	41.33	45.46	45.9	24.24	43.62	47.98	48.45			18.88	33.97	37.37	37.74		19.39	34.89	38.38	38.76
	200G	1.92	3.45	3.8	3.825	11.48	20.66	22.73	23.0	9.7	17.45	19.19	19.38			92 sectors					93 sectors			
100 MHz	10G	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$
	40G	119.9	215.8	237.3	239.7	94.37	169.8	186.8	188.7	117.3	211.2	232.3	234.6			118.6	213.5	234.8	237.2		117.3	211.2	232.3	234.6
	100G	59.94	107.9	118.7	119.9	37.75	67.94	74.73	75.48	38.26	68.86	75.7	76.5			47.44	85.38	93.92	94.86		47.44	85.38	93.92	94.86
	200G	23.98	43.15	47.47	47.94																			

Tableau 7.5 – Valeurs des temps de séjour pour le transport de plusieurs secteurs (RRH) sous différentes bandes passantes et différentes capacités de switches 40G, 100G, 200G et 1T dans une architecture C-RAN sous le split II_U par slot où les RRH sont synchrones.

R_S [μs]	3 sectors			18 sectors			19 sectors			37 sectors			38 sectors			46 sectors									
	Mean	90th	100th	Mean	90th	100th	Mean	90th	100th	Mean	90th	100th	Mean	90th	100th	Mean	90th	100th							
10 MHz	10G	31.2	55.68	61.19	61.2	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$							
	40G	7.8	13.92	15.29	15.3	46.05	82.77	91.03	91.8	48.6	87.36	96.08	51.3	94.5	169.9	186.9	188.7	97.05	174.6	192	193.8	117.5	211.3	232.4	234.6
	100G	3.12	5.57	6.118	6.12	18.42	33.11	36.41	36.72	19.44	34.95	38.43	20.52	37.8	67.99	74.78	75.48	38.82	69.83	76.8	77.52	46.98	84.52	92.96	93.84
	200G	1.56	2.78	3.059	3.06	9.21	16.6	18.21	18.36	9.72	17.47	19.22	10.26	18.9	33.99	37.39	37.74	19.41	34.91	38.4	38.76	23.49	42.26	46.48	46.92
10 MHz	10G	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$						
	40G	120	215.9	237.5	239.7	75.54	135.9	149.5	150.9	76.56	137.8	151.5	153	93.9	168.9	185.9	187.7	94.92	170.8	187.9	189.7	47.46	85.41	93.94	94.86
	100G	48	86.35	94.98	95.88	37.77	67.96	74.76	75.48	38.28	68.88	75.765	76.5	46.95	84.49	92.9	93.84	47.46	85.41	93.94	94.86				
	200G	24	43.18	47.49	47.94																				
20 MHz	10G	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$						
	40G	95.94	172.6	189.9	191.8	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$						
	100G	47.97	86.32	94.95	95.88	75.51	135.9	149.5	150.9	76.53	137.7	151.5	153	93.87	168.9	185.8	187.7	94.89	170.8	187.8	189.7				
	200G																								
40 MHz	10G	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$						
	40G	30.75	55.23	60.74	61.2	73.5	132.2	145.5	146.9	77.58	139.6	153.5	155	75.51	135.9	149.5	150.9	77.58	139.6	153.6	155	93.9	168.9	185.9	187.7
	100G	6.15	11.05	12.15	12.24	36.75	66.13	72.74	73.44	38.79	69.8	76.77	77.52	37.77	67.96	74.76	75.48	38.79	69.8	76.78	77.52	46.95	84.49	92.93	93.84
	200G	3.075	5.52	6.07	6.12	18.37	33.06	36.37	36.72	19.39	34.9	38.39	38.76	37.76	67.95	74.74	75.48	38.77	69.78	76.76	77.52				
40 MHz	10G	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$						
	40G	95.91	172.6	189.9	191.8	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$						
	100G	47.96	86.31	94.94	95.88	75.5	135.9	149.5	150.9	76.52	137.7	151.5	153	93.86	168.9	185.8	187.7	94.87	170.8	187.8	189.7				
	200G																								
100 MHz	10G	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$						
	40G	30.48	54.82	60.29	60.84	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$						
	100G	15.24	27.41	30.15	30.42	91.29	164.3	180.7	182.5	96.36	173.4	190.8	192.7	93.81	168.9	185.7	187.6	96.3	173.4	190.74	192.7	116.6	209.9	230.9	233.2
	200G	7.62	13.7	15.07	15.21	45.65	82.15	90.36	91.26	48.18	86.71	95.38	96.33	37.52	67.54	74.29	75.04	38.54	69.36	76.3	77.06	46.65	83.96	92.36	93.29
100 MHz	10G	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$						
	40G	119.2	214.5	235.9	238.3	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$	$R_S^{(100)} > 249.7$						
	100G	47.66	85.79	94.37	95.32	75	135	148.6	150	76.06	136.9	150.6	152.1	93.294167.9	184.7	186.6	187.7	94.31	169.7	186.7	188.6				
	200G																								

Tableau 7.6 – Valeurs des temps de séjour pour le transport de plusieurs secteurs (RRH) sous différentes bandes passantes et différentes capacités de switches 40G, 100G, 200G et 1T dans une architecture C-RAN sous le split I_U par TTI où les RRH sont synchrones.

Chapitre 8

Conclusion générale et perspectives

Prévue pour 2020, la nouvelle génération de réseaux cellulaires (5G) devrait relever de nombreux défis, en premier lieu, technologiques, en introduisant de nouvelles solutions (MEC, NFV, SDN, etc) pour une meilleure disponibilité, une meilleure fiabilité et une amélioration des performances, mais également économiques, avec la nécessité de réduire les CAPEX/OPEX qui n'ont cessé de croître ces dernières années. Le C-RAN est de ce fait, défini pour devenir l'architecture d'accès radio standard pour la 5G, apportant une gestion des ressources radio optimisée, une gestion de la mobilité des utilisateurs améliorée, une efficacité énergétique et un coût de déploiement réduit grâce à son architecture centralisée et à sa flexibilité permettant de réutiliser les infrastructures physiques présentes dans le réseau. Dans ce travail de thèse, nous nous sommes intéressés au dimensionnement des RRH et du fronthaul dans une architecture C-RAN. Les objectifs principaux ont été de proposer d'une part des outils de dimensionnement très rapides qui permettent de prévoir l'attribution de ressources de façon optimale au niveau radio tout en remédiant aux interférences et d'autre part des règles de dimensionnement du fronthaul qui permettent de déterminer de façon optimale la capacité des liens en fibre optique, du switch Ethernet et le nombre maximal de RRH à merger dans le même switch tout en respectant la contrainte de latence exigée.

8.1 Contributions sur eICIC/ABS

8.1.1 Modélisation de la technique de remédiation aux interférences eICIC/ABS

La première contribution de cette thèse consistait à modéliser la technique de remédiation aux interférences eICIC/ABS afin d'allouer de façon optimale les ressources dans les C-RAN/ H-CRAN. Pour cela, nous avons d'abord proposé des chaînes de Markov permettant de modéliser eICIC/ABS. Toutefois, à cause des spécificités de ces chaînes de Markov (multi-serveurs et arrivées par batch engendrant un nombre important de transitions) la résolution numérique est difficile et complexe. De plus, la simulation numérique de ces chaînes de Markov nécessitait plusieurs dizaines de minutes. Afin d'y remédier, nous avons proposé ensuite des modèles de files d'attente ($M/G/S/S$ et $M^{[X]}/G/S/S$)

avec des formes closes pour évaluer les performances de eICIC/ ABS.

8.1.2 Dimensionnement et optimisation du débit global pour eICIC/ABS

Les modèles de files d'attente proposés se sont avérés simples et rapides (quelques secondes pour la $M/G/S/S$ et quelques minutes pour la $M^{[X]}/G/S/S$). Forts de ce constat, nous avons proposé un outil de dimensionnement en temps réel pour analyser (et prévoir) l'attribution des ressources entre la macro et la micro cellule et le nombre d'UE à rattacher à la micro, selon la charge du trafic. A travers les expérimentations réalisées, nous avons observé que cet outil apporte de façon rapide et adaptative une meilleure stabilité du réseau en maximisant le débit global.

8.2 Contributions sur JT-CoMP

8.2.1 Modélisation de la technique de remédiation aux interférences JT-CoMP

Dans la deuxième contribution, nous nous sommes intéressés à la modélisation d'une autre technique de remédiation aux interférences inter-cellulaires : JT-CoMP. L'objectif principal était de déterminer précisément et en temps réel le nombre optimal des utilisateurs CoMP ainsi que le nombre optimal de ressources à dédier pour la coopération tout en assurant un débit global optimal. En suivant la même logique de la contribution sur la technique eICIC/ABS, nous avons d'abord proposé des chaînes de Markov puis des modèles de files d'attente ($M/G/S/S$ et $M^{[X]}/G/S/S$) pour modéliser JT-CoMP.

8.2.2 Dimensionnement et optimisation du débit global pour JT-CoMP

En nous basant sur ces modèles de files d'attente simples et rapides, nous avons proposé un outil de dimensionnement rapide et adaptatif qui offre aux opérateurs les décisions optimales sur l'attribution des ressources entre les utilisateurs CoMP et non-CoMP ainsi que le nombre des utilisateurs à servir en coopération pour atteindre le débit global optimal.

8.3 Contribution comparant les performances de eICIC/ABS et de JT-CoMP

Après avoir proposé les outils de dimensionnement pour la technique eICIC/ ABS et pour la technique JT-CoMP, la contribution suivante a été consacrée pour faire une étude comparative entre eICIC/ABS et JT-CoMP. Cette étude consistait à déterminer la technique de remédiation aux interférences, eICIC/ABS ou JT-CoMP, qui permettrait d'assurer un débit global optimal. Autrement dit, pour une charge de trafic donnée serait-il intéressant d'effectuer eICIC/ABS ou JT-CoMP afin d'atteindre le débit global optimal? Lors de cette étude, nous avons déterminé en

fonction de la charge du trafic la technique optimale ainsi que l'allocation optimale de ressources et l'attribution optimale des utilisateurs entre les cellules coordonnées.

8.4 Contribution sur le dimensionnement du fronthaul

Dans la dernière contribution, nous avons eu pour objectif de proposer un dimensionnement du FH avec un temps de latence faible, une bande passante efficace et un déploiement économique sur le FH. Nous avons limité notre étude à la liaison montante (UL) du transport FH basé sur Ethernet dans une architecture C-RAN, car le traitement nécessaire à cet effet impose des délais plus stricts que ceux de la liaison descendante (DL). Nous avons proposé un nouveau split en UL sur le fronthaul dans les architectures C-RAN. Le split proposé est noté split II_U et c'est le split le plus supérieur de la couche physique. La conception de ce nouveau split a pris en compte les contraintes de délais basées sur le HARQ et a cherché à maximiser les gains de multiplexage statistique sur le FH. Le split proposé a considéré des RRH synchrones afin de garantir des techniques avancées de coopération et de coordination entre les RRH dans les réseaux cellulaires futurs. Les résultats révèlent que le split proposé peut être appliqué pour obtenir un réseau FH à faible temps de latence, une bande passante efficace et un déploiement économique sur le FH pour les réseaux cellulaires de la prochaine génération basés sur l'architecture C-RAN.

8.5 Travaux futurs

La nouvelle génération de réseaux mobiles n'a pas encore été standardisée, malgré les nombreux travaux menés pour définir le spectre radiofréquence, les techniques de multiplexage, les technologies réseau, etc., qui seront adoptés dans le C-RAN et plus généralement au sein de la 5G. Par conséquent, un certain nombre de pistes restent encore à explorer avant son lancement officiel en 2020.

8.5.1 Perspectives à court terme

8.5.1.1 Gestion des interférences dans des clusters avec eICIC/ ABS

Concernant notre étude sur la modélisation de la technique de rémediation aux interférences eICIC/ABS (voir chapitre 4), une étude sur la répartition des ressources entre une macro et plusieurs micro serait d'un grand intérêt. Pour cette étude, le raisonnement que nous avons proposé dans le chapitre 4 peut-être appliqué. Dans un tel scénario, où N micro sont déployées sans interférer entre elles (ce qui impliquerait une réutilisation des ressources dans chaque micro) et les clients sont distribués de façon uniforme (ce qui engendre les mêmes taux d'arrivée des clients par micro), le

problème d'optimisation est formulé comme suit :

$$\begin{aligned}
& \max_{S_m, S_M, \lambda_m, \lambda_M} f(S_m, S_M, \lambda_m, \lambda_M) \\
& \text{s.t.} \quad S_m + S_M = S, \\
& \quad \sum_N \lambda_m + \sum_{j=1}^{J_M} \lambda_{M,j} = \lambda, \\
& \quad \sum_{j=1}^{J_M} \lambda_{M,j} = \lambda_M, \\
& \quad S_M > 0, \\
& \quad S_m \geq 0, \\
& \quad p_m^b \leq \epsilon
\end{aligned} \tag{8.1}$$

où $f(S_m, S_M, \lambda_m, \lambda_M)$ est calculé comme suit :

$$f(S_m, S_M, \lambda_m, \lambda_M) = \sum_{j=1}^{J_M} X_{M,j} + N \times X_{m,1} \tag{8.2}$$

Les mêmes méthodes de résolution (algorithme Global Search et l'algorithme [2](#)) restent applicables.

8.5.1.2 Gestion des interférences dans des clusters avec JT-CoMP

Concernant notre étude sur la modélisation de la technique de rémédiation aux interférences JT-CoMP (voir chapitre [5](#)) une étude sur le nombre optimal de cellules coopératives dans un cluster avec de multiples tiers serait d'un grand intérêt. Cette étude peut être faite en appliquant le raisonnement de notre modèle d'optimisation sur 2 puis 3 puis 4 jusqu'à N cellules où N représente la taille du cluster (ou le nombre maximal de cellules coopératives). A la fin de ces applications, nous comparons les débits globaux optimaux obtenus pour chaque nombre de cellules coopératives. Nous gardons la configuration offrant le débit global optimal utilisant un nombre de serveurs à mettre en coopération optimal aussi. Pour une configuration où N macro cellules coopèrent et où les clients sont distribués de façon uniforme, le problème d'optimisation peut être formulé comme suit :

$$\begin{aligned}
& \max_{S_M, S_c, \lambda_c, \lambda_{M1}, \lambda_{M2}, \lambda_{M3}, \dots, \lambda_{MN}} f(S_M, S_c, \lambda_c, \lambda_{M1}, \lambda_{M2}, \lambda_{M3}, \dots, \lambda_{MN}) \\
& \text{s.t.} \quad S_M + S_c = S, \\
& \quad \lambda_c + \lambda_{M1} + \lambda_{M2} + \lambda_{M3} + \dots + \lambda_{MN} = \lambda, \\
& \quad S > 0, \\
& \quad p_c^b \leq \epsilon
\end{aligned} \tag{8.3}$$

où $f(S_M, S_c, \lambda_c, \lambda_{M1}, \lambda_{M2})$ est calculé comme suit :

$$f(S_M, S_c, \lambda_c, \lambda_{M1}, \lambda_{M2}) = \sum_{j=1}^{J_{M1}} X_{M1,j} + \sum_{j=1}^{J_{M2}} X_{M2,j} + \sum_{j=1}^{J_{M3}} X_{M3,j} + \dots + \sum_{j=1}^{J_{MN}} X_{MN,j} + X_{c,1} \quad (8.4)$$

Les mêmes méthodes de résolution (algorithme Global Search et l'algorithme [4](#)) restent applicables.

8.5.1.3 Considérer la mobilité des utilisateurs dans nos études sur eICIC/ ABS et JT-CoMP

Nous avons effectué un raisonnement par trame. Lors de la simulation de la chaîne de Markov, dans chaque trame nous avons généré des clients selon une loi de poisson de paramètre $\lambda^{clients}$. Nous avons évidemment considéré un nombre important de trames pour assurer un intervalle de confiance à 95%. Toutefois, nous n'avons pas pris en considération la mobilité des utilisateurs au sein de la trame qui peut impacter les performances [\[ABS15\]](#). En suivant la logique suivante nos modèles de files d'attente restent valides en considérant la mobilité. En effet, comme expliqué dans nos équations d'évolution (voir l'équation [\(4.5\)](#) et l'équation [\(4.6\)](#) pour eICIC/ABS et l'équation [\(5.2\)](#), l'équation [\(5.3\)](#) et l'équation [\(5.4\)](#) pour JT-CoMP), à la trame n le simulateur considère le service des clients arrivés à la trame $n - 1$ et l'arrivée des clients à la trame actuelle (c'est à dire à la trame n). Par ailleurs, dans nos modèles de files d'attente nous exploitons les taux d'arrivée de clients et leurs conditions radio selon l'évaluation et la caractérisation faites par l'algorithme [1](#) pour eICIC/ABS et l'algorithme [3](#) pour JT-CoMP. Il nous semble donc important de modifier ces algorithmes afin de considérer la proportion d'utilisateurs en mobilité comme un taux de poisson. Comme nous avons validé ces modèles pour différents taux d'arrivée de clients et pour différentes tailles de batch, nos modèles de files d'attente peuvent être considérés pour refléter eICIC/ ABS et JT-CoMP même lors de la mobilité des utilisateurs au sein de la trame. Nous pourrions ainsi envisager d'utiliser la forme close de la $M/G/S/S$ avec un taux global (utilisateurs en mobilité ou pas) ou envisager le modèle $M^{[X]}/G/S/S$ si nous voulons faire une analyse par classe.

8.5.1.4 Considérer d'autres bandes passantes dans nos études sur eICIC/ ABS et JT-CoMP

Ayant pour objectif de valider nos modèles de files d'attente, nous nous sommes focalisés sur la bande passante de 20 MHz. Comme nous l'avons expliqué dans le Tableau [2.1](#) et le Tableau [2.3](#), le nombre de PRB dans une trame et la taille du RBG dépendent de la bande passante considérée. Ainsi, la considération d'une bande passante inférieure/ supérieure à 20 MHz engendrerait la diminution/ augmentation du nombre de RBG (ou serveurs) disponibles dans une trame pour les trois approches proposées. Dès lors, il serait intéressant d'effectuer une étude approfondie sur l'influence

du nombre de serveurs sur la précision de nos modèles de file d'attente. Nous avons effectué une étude initiale qui est représenté en Annexe [8.5.2.4](#).

8.5.1.5 Effectuer une analyse approfondie numérique et théorique de nos modèles de file d'attente

Nous avons comparé, dans la section [4.5.2](#) et la section [4.5.3](#), les résultats de simulation de la chaîne de Markov et les modèles de files d'attente pour des tailles de paquets différentes (1024 bits, 2048 bits et 3200 bits). Nous estimons avoir validé la $M/G/S/S$ numériquement pour les paramètres d'entrée considérés : nombre de serveurs = 500 serveurs et tailles de paquets $size(u) = 1024$ bits, 2048 bits et 3200 bits. Tant que nous n'avons pas réalisé une comparaison des distributions de probabilité des différents systèmes, il est difficile de prévoir la qualité de la $M/G/S/S$ pour d'autres valeurs de paramètres. Il faudrait donc faire une comparaison théorique plus approfondie des distributions de probabilités. Il est clair que le fait d'avoir considéré une moyenne des tailles de batchs pour le temps de service des clients en entrée dans la $M/G/S/S$ au lieu de prendre la distribution de la taille de batchs considérée dans la chaîne de Markov engendre une différence de comportement entre les systèmes qui est peu visible pour des tailles de batch de 1 et de 2, (ou 2 et 4) mais qui peut l'être davantage pour d'autres tailles. Pour la $M^{[X]}/G/S/S$, nous avons observé que les résultats étaient intéressants que pour des batchs de 1 et de 2. Il serait donc intéressant d'étudier la précision des modèles sur un plus grand nombre de valeurs de paramètres d'entrée, et d'arriver aussi à les comparer pour voir dans quels cas un modèle peut générer des débits supérieurs ou inférieurs à la chaîne de Markov. Pour ce fait, nous projetons de proposer une preuve numérique et théorique de nos conclusions afin de généraliser et de déterminer les exceptions sur la qualité des approches proposées.

8.5.1.6 Dimensionnement du fronthaul pour des RRH asynchrones

Concernant notre outil de dimensionnement du fronthaul, dans d'autres réseaux cellulaires, nous pouvons avoir en plus des RRH synchrones des RRH asynchrones. Ces dernières n'auront donc pas besoin, entre elles, des techniques de coopération et de coordination car, par exemple, elles sont lointaines. Comme axe d'amélioration, nous souhaiterions proposer un outil de dimensionnement du FH lorsque les RRH ne sont pas synchrones.

Dans [\[PHL18\]](#), les auteurs proposent d'utiliser la $G/G/1$ (voir la section [3.4.3](#)) pour le dimensionnement du fronthaul lorsque le split I_U est appliqué. Pour ce split le temps d'asynchronisme varie entre 0 et $66.6 \mu s$. Les résultats de temps d'attente obtenus par la $G/G/1$ (l'équation [\(3.17\)](#) et l'équation [\(3.19\)](#)) ont été comparés avec les résultats de simulations pour une bande passante de 20 MHz et une capacité du switch à 200 Gb/s. Cette comparaison révèle que la différence entre ces deux approches est inférieure à $1 \mu s$, ainsi les auteurs ont validé la $G/G/1$ pour le dimensionnement du FH lorsque le split I_U est appliqué.

Afin de modéliser le split Π_U lorsque les RRH sont asynchrones (temps d'asynchronisme entre 0 et $66.6 \mu s$), nous avons simulé ce split en utilisant ciw [ciw19], bibliothèque de simulation des évènements à temps discret pour des réseaux de files d'attente ouverts, sur Python. Notons que l'équation (3.19) ne nous permet pas de calculer le temps d'attente du 100ième percentile, nous nous basons donc sur le temps d'attente du 99ième percentile. Nous comparons dans le Tableau 8.1 les temps de séjour du 99ième percentile, $R_S^{(99)}$, obtenus par simulation et par la G/G/1 en appliquant le split Π_U par slot.

R_S [us]	3 sectors		18 sectors		19 sectors		37 sectors		38 sectors		46 sectors	
	Simul	G/G/1	Simul	G/G/1	Simul	G/G/1	Simul	G/G/1	Simul	G/G/1	Simul	G/G/1
100 MHz	100G	30.36	29.41	181.8	519.22	191.88	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7
	200G	15.18	8.79	90.9	163	95.94	176.83	186.84	546.5	191.88	>249.7	>249.7
	400G	7.59	1.64	45.45	55.83	47.97	60.5	93.42	170.2	95.94	177.48	240.78
R_S [us]	47 sectors		74 sectors		75 sectors		92 sectors		93 sectors			
	Simul	G/G/1	Simul	G/G/1	Simul	G/G/1	Simul	G/G/1	Simul	G/G/1		
100 MHz	100G	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7
	200G	237.36	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7	>249.7
	400G	118.68	249.33	186.84	546.56	189.36	>249.7	232.26	>249.7	234.78	>249.7	>249.7

Tableau 8.1 – Comparaison entre les temps de séjour obtenus par simulation et par la G/G/1 pour une BP= 100 MHz sous différentes capacités de switch 100 G, 200 G et 400 G lorsque le split Π_U par slot est appliqué avec des RRH asynchrones.

Prenons un exemple, si nous déployons un switch avec une capacité de 400 G et nous souhaitons merger 93 secteurs, les résultats du Tableau 8.1 (résultats de simulation) révèlent que la contrainte de latence (l'équation (7.9)) est respectée toutefois, d'après la G/G/1 cette contrainte n'est pas respectée. Nous notons un écart entre les deux approches (simulation et G/G/1) égale à $359 \mu s$. Ainsi, la G/G/1 ne reflète pas le comportement réel, c'est donc une mauvaise approximation pour la modélisation du split Π_U que nous proposons. Ceci est expliqué par le fait que la G/G/1 considère des arrivées simples avec un certain coefficient de variation, alors que dans notre cas les arrivées sont par batch. Ainsi le coefficient de variation que nous avons calculé pour nos arrivées par batch ne représente pas assez finement le processus d'arrivée. Une piste à explorer pour modéliser le split que nous proposons serait de comparer les résultats de simulations avec ceux de la $G^{[X]}/G/1$ qui considère des arrivées par batch (voir la section 3.4.4). Toutefois, la $G^{[X]}/G/1$ ne possède pas de formes close pour le calcul du temps d'attente moyen et le temps d'attente du p ième percentile. Il faudrait ainsi utiliser l'équation de Lindley [Lin52], sous sa forme discrète, pour déterminer la distribution du temps d'attente des batches qui arrivent. Et ensuite, de résoudre cette équation grâce aux algorithmes de Grassmann et Jain [GJ89].

8.5.2 Perspectives à long terme

Dans cette section, nous allons présenter les travaux qui pourraient être réalisés dans les trois prochaines années pour améliorer et/ou exploiter nos outils de dimensionnement de la partie radio et du fronthaul de l'architecture C-RAN/H-CRAN.

8.5.2.1 Stratégie de scheduling des utilisateurs

Il est nécessaire de considérer une stratégie de scheduling des utilisateurs au sein de la trame permettant ainsi de maximiser davantage le débit global. Effectivement, nos outils d'allocation optimale des ressources considèrent des arrivées de clients selon une loi de poisson de paramètre λ . Le processus d'allocation de ressources aux clients a été effectué (dans les simulations et les modèles de files d'attente) selon un ordre FIFO (premier arrivé autrement dit généré, premier servi). Ainsi, nous n'avons pas considéré une stratégie de scheduling permettant de déterminer de façon optimale l'utilisateur à servir à un instant t . Nous citons trois principales stratégies de scheduling : le Round robin, le Max C/I et le Proportional fair. La stratégie Round robin est connue par sa simplicité et consiste à servir des utilisateur indépendamment de leurs conditions radio. Toutefois, la simplicité de cette stratégie se fait au détriment des performances dégradées. Le principe du scheduler Max C/I est le suivant. Il sélectionne l'utilisateur ayant le débit instantané le plus élevé. C'est le plus efficace en terme de débit. Cependant, il néglige les utilisateurs ayant de mauvaises conditions radio qui ne seront jamais servis à forte charge. Enfin, le scheduler Proportional fair sélectionne les utilisateurs ayant le rapport entre le débit instantané et le débit moyen le plus élevé. Ce scheduler représente donc un bon compromis entre l'efficacité et l'équité. Suite à cette première analyse, il faudrait donc choisir adéquatement la stratégie de scheduling à adopter. En considérant l'une de ces stratégies de scheduling, la discipline de service de nos chaînes de Markov ne correspondrait plus à du FIFO, ce serait du processor sharing où les clients seront servis simultanément avec un partage équitable des serveurs.

8.5.2.2 Modélisation d'autres techniques de remédiation aux interférences en UL

Nous pouvons envisager de modéliser d'autres techniques de remédiation aux interférences en UL (ex. Joint reception) afin d'évaluer et de caractériser le trafic de données transitant par le FH en UL tout en remédiant aux interférences sur la partie radio. Cette étude sera notamment nécessaire dans un contexte où par exemple l'opérateur, louant le matériel (fibre optique et switch) hebdomadairement, mensuellement ou autre, souhaiterait un déploiement optimal selon le nombre d'UE actifs pendant la période considérée. De plus, afin de mettre en place un réseau FH robuste qui permettrait de fournir de meilleures performances, une sécurité et une qualité de service requises pour le C-RAN, il est envisageable également d'implémenter, au sein du contrôleur SDN (plan de management), notre outil de dimensionnement dédiées à la gestion du trafic de données transitant par le FH. Par ailleurs, il serait intéressant d'observer l'impact positif ou négatif des différentes techniques de remédiation aux interférences sur le fronthaul.

8.5.2.3 Intégration de nos outils de dimensionnement dans un réseau 5G basé sur le slicing

D'un point de vue technique, le slicing d'un réseau permet aux opérateurs de réseaux mobiles (MNO) de gérer et d'exploiter plusieurs réseaux virtuels sur une infrastructure de réseau physique commune. Il permet le partage virtuel du réseau d'accès radio (RAN), des composants de réseau essentiels de l'EPC (Evolved Packet Core) et des réseaux de commutation et d'agrégation jusqu'aux data centers, où sont hébergés le contenu et les applications auxquels on accède. Une approche basée sur le slicing doit donc assurer la haute rentabilité du réseau mobile 5G pour les MNO. L'article [EJAGS19] énumère les challenges du RAN slicing lors de l'allocation de ressources. Nos outils de dimensionnements peuvent ainsi être implémentés dans un slice afin de gérer l'allocation de ressources et de fournir un réseau 5G adaptable, dynamique et programmable de bout en bout au moyen de structures virtuelles. Un tel réseau mobile 5G entièrement flexible permettra aux MNO d'offrir un éventail beaucoup plus large de services de communication. Il faudrait cependant prendre des modèles de files d'attente qui prendrait en compte plusieurs classes de clients, nécessitant plus ou moins des ressources radios, et correspondant à différents types de slices.

8.5.2.4 Développer des méthodes d'intelligence artificielle

Ces dernières années ont vu l'émergence à la fois de l'internet des objets (IoT), des réseaux 5G, et du big data. Dans [WCS⁺18], il est défini le paradigme de la 5G-IoT (5G Intelligent Internet of Things) qui se caractérise par la convergence des réseaux 5G, de l'internet des objets et de l'intelligence artificielle pour traiter les données réelles dynamiquement permettant une meilleure gestion des ressources de communication et aussi pour offrir une grande variété de services (ville intelligente, domotique, transports intelligents, . . .). Dans ce contexte de la 5G-IoT, il en résulte une architecture réseau hétérogène, où le déploiement des objets est très dynamique, à cause de l'imprévisibilité des événements (conditions météorologiques, catastrophes naturelles, urgences ou pannes). Cette dynamique risque d'engendrer des problèmes de congestions et donc d'augmentation des latences dans les réseaux qu'il est difficile à prévoir par des méthodes classiques de dimensionnement (simulation, théorie des files d'attente). Il nous est donc apparu intéressant de promouvoir des méthodes d'apprentissage et de prédiction permettant de mieux gérer dynamiquement les ressources disponibles (objets IoT, bande passante) en se basant sur nos outils de dimensionnement et ce afin de garantir une meilleure QoS et de maîtriser la consommation énergétique. Ainsi, un apprentissage en ligne, en fonction du trafic réel permet de dimensionner les ressources automatiquement d'une manière dynamique, mais ces algorithmes d'apprentissage par renforcement comme le Q-learning souffrent du problème de convergence à cause de l'initialisation des paramètres (matrice des Qvaleurs) et donc une approche intéressante serait de les initialiser avec des formes closes [DKM⁺11].

Annexe

$\lambda^{clients}$	λ [kbits/trame]	Simulation		M/G/S/S	
		p^b	X [kbits/trame]	p^b	X [kbits/trame]
50	43.9367	0	[43.9274 ; 43.94]	0	43.9367
100	90.4254	0	[90.4181 ; 90.4307]	0	90.4254
150	134.5084	0	[134.4989 ; 134.5115]	0	134.5084
200	179.6997	0	[179.6897 ; 179.7023]	0	179.6997
250	224.0567	0	[224.0499 ; 224.0625]	0	224.0567
300	269.4578	[0.0016 ; 0.0018]	[268.0589 ; 269.0712]	0.0017	269.0671
350	313.9427	[0.0687 ; 0.0742]	[291.3799 ; 291.3925]	0.07	291.3898
400	358.5348	[0.1759 ; 0.18]	[294.6497 ; 294.6523]	0.1763	294.6516
450	406.5579	[0.2658 ; 0.2664]	[300.0642 ; 300.0850]	0.266	300.0842

Tableau 2 – Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit (macro sans micro, $size(u)=1024$ bits)

$\lambda^{clients}$	λ [kbits/trame]	Simulation		M/G/S/S	
		p^b	X [kbits/trame]	p^b	X [kbits/trame]
50	90.0435	0	[89.8799 ; 90.1201]	0	90.0435
100	179.5193	0	[179.3977 ; 179.6201]	0	179.5193
150	273.8341	[0.0012 ; 0.0014]	[273.3869 ; 273.4839]	0.0013	273.4672
200	360.7929	[0.1787 ; 0.1806]	[295.6879 ; 295.8205]	0.1802	295.7614
250	454.3225	[0.3348 ; 0.3372]	[298.4897 ; 298.6203]	0.3368	298.5677
300	543.8816	[0.4469 ; 0.4501]	[299.2699 ; 299.2883]	0.4491	299.2811
350	633.5220	[0.5247 ; 0.5354]	[299.5798 ; 299.7232]	0.5268	299.6071
400	721.4445	[0.5699 ; 0.5897]	[299.6799 ; 299.7364]	0.5861	299.7255
450	814.0782	[0.6299 ; 0.6347]	[299.9989 ; 300.3701]	0.6323	300.2884

Tableau 3 – Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit (macro sans micro, $size(u)=2048$ bits)

$\lambda^{clients}$	λ [kbits/trame]	Simulation		M/G/S/S	
		p^b	X [kbits/trame]	p^b	X [kbits/trame]
50	104.5666	0	[104.5287 ; 104.7201]	0	104.5666
100	208.1388	0	[207.9989 ; 208.2402]	0	208.1388
150	312.2680	[0.1188 ; 0.122]	[273.89 ; 274.0345]	0.12	273.9385
200	417.1863	[0.3279 ; 0.3362]	[275.5894 ; 275.6702]	0.3347	275.6591
250	522.3808	[0.4669 ; 0.4676]	[277.3979 ; 277.4332]	0.4674	277.4294
300	625.0954	[0.5561 ; 0.5563]	[277.5374 ; 277.5608]	0.5562	277.5429
350	726.4672	[0.6204 ; 0.6206]	[278.1956 ; 278.2231]	0.6205	278.2185
400	834.7978	[0.6665 ; 0.667]	[278.3169 ; 278.3321]	0.6666	278.3235
450	941.1578	[0.7029 ; 0.7033]	[279.3534 ; 279.354]	0.7032	279.3535

Tableau 4 – Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit (macro sans micro, $size(u)=3200$ bits)

$\lambda^{clients}$	λ [kbits/trame]	Simulation		M/G/S/S	
		p^b	X [kbits/trame]	p^b	X [kbits/trame]
50	45.3301	0	[45.2989 ; 45.7425]	0	45.3301
100	91.8185	0	[91.5756 ; 91.9998]	0	91.8185
150	136.9178	0	[136.8949 ; 136.9287]	0	136.9178
200	183.4216	0	[183.3976 ; 183.4301]	0	183.4216
250	229.3004	[0.0016 ; 0.0018]	[228.799 ; 228.95]	0.0017	228.893
300	274.8570	[0.0812 ; 0.085]	[251.5789 ; 251.6717]	0.0846	251.6427
350	320.5797	[0.1991 ; 0.1993]	[256.8989 ; 257.9532]	0.1992	256.9455
400	365.2582	[0.2906 ; 0.2935]	[258.449 ; 258.4533]	0.293	258.4528
450	412.6882	[0.3629 ; 0.3648]	[262.6889 ; 262.6984]	0.364	262.6974

Tableau 5 – Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit global (macro avec micro $n_{ABS} = 3$, $size(u)=1024$ bits)

$\lambda^{clients}$	λ [kbits/trame]	Simulation		M/G/S/S	
		p^b	X [kbits/trame]	p^b	X [kbits/trame]
50	89.7369	0	[89.7193 ; 89.7422]	0	89.7369
100	180.6582	0	[180.6129 ; 180.6701]	0	180.6582
150	276.6098	[0.0101 ; 0.0187]	[273.1798 ; 273.2479]	0.0127	273.2067
200	365.7732	[0.1911 ; 0.2078]	[288.2197 ; 288.2263]	0.2206	288.2248
250	459.6505	[0.3440 ; 0.3601]	[294.5397 ; 294.5482]	0.3703	294.5474
300	549.0561	[0.4551 ; 0.4660]	[295.6899 ; 295.7201]	0.4797	295.7127
350	639.4426	[0.5263 ; 0.5363]	[298.1688 ; 298.171]	0.5552	298.1701
400	730.7051	[0.5774 ; 0.5853]	[303.4397 ; 303.4502]	0.6516	303.4455
450	824.8026	[0.6315 ; 0.6236]	[305.98 ; 306.01]	0.7326	306.0027

Tableau 6 – Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit global (macro avec micro $n_{ABS} = 1$, $size(u)=2048$ bits)

$\lambda^{clients}$	λ [kbits/trame]	Simulation		M/G/S/S	
		p^b	X [kbits/trame]	p^b	X [kbits/trame]
50	103.1978	0	[102.8979 ; 103.02]	0	103.1978
100	207.7633	0	[207.4867 ; 207.9001]	0	207.7633
150	313.0927	[0.1489 ; 0.1528]	[262.5999 ; 262.6321]	0.15	262.6299
200	418.8231	[0.3394 ; 0.3564]	[268.9989 ; 269.2002]	0.350	269.1060
250	523.6226	[0.4739 ; 0.4753]	[272.4997 ; 272.5739]	0.4746	272.5617
300	626.6428	[0.5498 ; 0.5561]	[275.3928 ; 275.4234]	0.555	275.41368
350	729.8687	[0.5999 ; 0.6182]	[275.6489 ; 275.66]	0.6178	275.6591
400	837.1789	[0.663 ; 0.6636]	[277.1197 ; 277.1231]	0.6635	277.1200
450	945.4217	[0.6984 ; 0.7024]	[280.8641 ; 280.8649]	0.6987	280.8646

Tableau 7 – Comparaison entre les simulations Matlab de la chaîne de Markov et la $M/G/S/S$ en terme de probabilités de blocage et de débit global (macro avec micro $n_{ABS} = 1$, $size(u)=3200$ bits)

classe 1					
$\lambda^{clients}$	$\lambda_{M,1}$ [kbits/trame]	Simulation		$M^{[X]}/G/S/S$	
		$p_{M,1}^b$	$X_{M,1}$	$p_{M,1}^b$	$X_{M,1}$
50	31.1201	0	[30.9888 ; 31.2639]	0	31.1201
100	62.7241	0	[62.6894 ; 62.7471]	0	62.7241
150	91.7102	0	[91.6894 ; 91.8304]	0	91.7102
200	122.9891	0	[122.9572 ; 123.0101]	0	122.9891
250	153.3447	0	[153.2223 ; 153.3612]	0	153.3447
300	184.4042	[0 ; 0.0014]	[183.9402 ; 184.01]	0.0024	183.9617
350	215.1115	[0.0565 ; 0.0586]	[202.4897 ; 202.8256]	0.0578	202.6792
400	244.6794	[0.1558 ; 0.1733]	[207.5293 ; 207.6461]	0.1397	210.5011
450	280.9952	[0.2550 ; 0.2676]	[207.5836 ; 207.6779]	0.2082	222.4834
classe 2					
$\lambda^{clients}$	$\lambda_{M,2}$ [kbits/trame]	Simulation		$M^{[X]}/G/S/S$	
		$p_{M,2}^b$	$X_{M,2}$	$p_{M,2}^b$	$X_{M,2}$
50	14.4648	0	[14.3940 ; 14.6421]	0	14.4648
100	28.4857	0	[28.4459 ; 28.5036]	0	28.4857
150	43.6525	0	[43.6118 ; 43.7518]	0	43.6525
200	58.0482	0	[58.01 ; 58.0629]	0	58.0482
250	72.7883	0	[72.6711 ; 72.81]	0	72.7883
300	86.8772	[0.003 ; 0.0039]	[86.5895 ; 86.6434]	0.0037	86.6062
350	101.8105	[0.0791 ; 0.0820]	[93.46704 ; 93.9635]	0.0856	93.0985
400	116.6061	[0.1693 ; 0.1757]	[95.8116 ; 96.7567]	0.2003	93.2482
450	128.1944	[0.2482 ; 0.2596]	[96.5781 ; 96.9227]	0.2911	93.8712

Tableau 8 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe (macro sans micro, $size(u) = 1024$ bits)

		classe 2			
		Simulation		$M^{[X]}/G/S/S$	
$\lambda^{clients}$	$\lambda_{M,2}$ [kbits/trame]	$p_{M,2}^b$	$X_{M,2}$	$p_{M,2}^b$	$X_{M,2}$
50	60.5446	0	[60.4798 ; 60.74348]	0	60.5446
100	121.1943	0	[121.0450 ; 121.7214]	0	121.1943
150	186.9437	[0.0025 ; 0.0076]	[186.6958 ; 187.0116]	0	186.9437
200	243.3541	[0.1605 ; 0.1795]	[206.3081 ; 206.4629]	0	243.3541
250	310.4555	[0.3195 ; 0.3424]	[207.0583 ; 207.7579]	2.13E-05	310.4489
300	368.2238	[0.4398 ; 0.4578]	[208.2185 ; 208.4863]	0.0097	364.6392
350	429.7056	[0.5190 ; 0.5353]	[210.1794 ; 210.3095]	0.0687	400.1872
400	488.0057	[0.5728 ; 0.5874]	[210.1937 ; 210.3106]	0.1401	418.0058
450	550.3944	[0.6232 ; 0.6369]	[210.5080 ; 210.6505]	0.2162	431.3723
		classe 4			
		Simulation		$M^{[X]}/G/S/S$	
$\lambda^{clients}$	$\lambda_{M,4}$ [kbits/trame]	$p_{M,4}^b$	$X_{M,4}$	$p_{M,4}^b$	$X_{M,4}$
50	29.4988	0	[29.3463 ; 29.6586]	0	29.4989
100	58.3250	0	[57.9645 ; 58.5247]	0	58.3250
150	86.8903	[0.0020 ; 0.0061]	[86.4101 ; 86.6658]	0	86.8903
200	117.4388	[0.1549 ; 0.1731]	[96.0382 ; 96.4665]	0	117.4388
250	143.8669	[0.3243 ; 0.3415]	[96.1150 ; 96.3516]	3.45E-05	143.8620
300	175.6578	[0.4410 ; 0.4538]	[96.9799 ; 97.1090]	0.0150	173.0266
350	203.8163	[0.5233 ; 0.5356]	[96.7123 ; 96.8045]	0.1017	183.0853
400	233.4387	[0.5785 ; 0.5873]	[96.7284 ; 96.8627]	0.2095	184.5435
450	263.6836	[0.6278 ; 0.6359]	[96.9922 ; 97.1114]	0.3019	184.7030

Tableau 9 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe (macro sans micro, $size(u) = 2048$ bits)

		Macro			
		classe 1			
		Simulation		$M^{[X]}/G/S/S$	
$\lambda^{clients}$	$\lambda_{M,1}$ [kbits/trame]	$p_{M,1}^b$	$X_{M,1}$	$p_{M,1}^b$	$X_{M,1}$
50	30.7664	0	[30.5853 ; 30.7989]	0	30.7664
100	61.3298	0	[61.1884 ; 61.5061]	0	61.3298
150	91.5327	0	[91.3261 ; 91.7175]	0	91.5327
200	122.4051	0	[122.3945 ; 122.6207]	0	122.4051
250	152.2261	[0.0472 ; 0.0498]	[144.2111 ; 144.7311]	0.0483	144.8702
300	182.5146	[0.1747 ; 0.1795]	[150.2571 ; 150.5726]	0.1545	154.3249
350	215.6102	[0.2887 ; 0.2892]	[153.4672 ; 153.6112]	0.2452	162.7426
400	243.8834	[0.3829 ; 0.3897]	[153.2449 ; 153.9462]	0.3212	165.5388
450	277.9834	[0.4576 ; 0.4593]	[153.9791 ; 154.1176]	0.3799	167.3885
		Macro			
		classe 2			
		Simulation		$M^{[X]}/G/S/S$	
$\lambda^{clients}$	$\lambda_{M,2}$ [kbits/trame]	$p_{M,2}^b$	$X_{M,2}$	$p_{M,2}^b$	$X_{M,2}$
50	13.4903	0	[13.3629 ; 13.6821]	0	13.4903
100	26.6079	0	[26.3572 ; 26.8626]	0	26.6079
150	40.6237	0	[40.4858 ; 40.7413]	0	40.6237
200	53.8747	0	[53.4980 ; 54.0762]	0	53.8747
250	67.5475	[0.0431 ; 0.0470]	[64.3704 ; 64.6323]	0.0720	62.6817
300	81.3593	[0.1911 ; 0.1918]	[65.7530 ; 65.7576]	0.2205	63.4195
350	93.4219	[0.2980 ; 0.2989]	[65.7605 ; 65.8357]	0.3169	63.8108
400	107.8061	[0.3891 ; 0.3896]	[65.8207 ; 65.8932]	0.4075	63.8652
450	119.2055	[0.4470 ; 0.4489]	[65.7543 ; 65.9483]	0.4636	63.9307

Tableau 10 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe dans la macro (micro déployée $n_{ABS} = 3$, $size(u) = 1024$ bits)

		micro			
		classe 1			
		Simulation		$M^{[X]}/G/S/S$	
$\lambda^{clients}$	$\lambda_{m,1}$ [kbits/trame]	$p_{m,1}^b$	$X_{m,1}$	$p_{m,1}^b$	$X_{m,1}$
50	0.7497	0	[0.7395 ; 0.7584]	0	0.7497
100	2.3968	0	[2.3185 ; 2.6541]	0	2.3968
150	3.6478	0	[3.5886 ; 3.8449]	0	3.6478
200	5.2353	0	[5.1996 ; 5.4021]	0	5.2353
250	6.4451	0	[6.3172 ; 6.7111]	0	6.4451
300	8.5204	0	[7.9674 ; 8.7464]	0	8.5204
350	8.9936	0	[8.8695 ; 9.1290]	0	8.9936
400	9.7502	0	[9.5139 ; 9.8904]	0	9.7502
450	11.8934	0	[11.7985 ; 12.0104]	0	11.8934
		micro			
		classe 2			
		Simulation		$M^{[X]}/G/S/S$	
$\lambda^{clients}$	$\lambda_{m,2}$ [kbits/trame]	$p_{m,2}^b$	$X_{m,2}$	$p_{m,2}^b$	$X_{m,2}$
50	0.1004	0	[0.0978 ; 0.1010]	0	0.1004
100	0.6498	0	[0.6191 ; 0.6578]	0	0.6498
150	0.9487	0	[0.9278 ; 0.9787]	0	0.9487
200	1.7885	0	[1.6898 ; 1.8045]	0	1.7885
250	2.1575	0	[1.9994 ; 2.2805]	0	2.1575
300	2.5333	0	[2.3908 ; 2.8165]	0	2.5333
350	3.3898	0	[3.1562 ; 3.5969]	0	3.3898
400	3.7707	0	[3.5666 ; 3.9156]	0	3.7707
450	4.1540	0	[3.9170 ; 4.1650]	0	4.1540

Tableau 11 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe dans la micro (micro déployée $n_{ABS} = 3$, $size(u) = 1024$ bits)

		Macro			
		classe 2			
		Simulation		$M^{[X]}/G/S/S$	
$\lambda^{clients}$	$\lambda_{M,2}$ [kbits/trame]	$p_{M,2}^b$	$X_{M,2}$	$p_{M,2}^b$	$X_{M,2}$
50	60.4650	0	[60.2453 ; 60.6540]	0	60.4650
100	121.0458	0	[120.8914 ; 121.2897]	0	121.0458
150	186.6616	[0.0117 ; 0.0141]	[185.5080 ; 185.7843]	0	186.6616
200	243.4543	[0.2014 ; 0.2065]	[194.48134 ; 194.6481]	0	243.4543
250	310.5586	[0.3535 ; 0.3568]	[200.0177 ; 200.2351]	3.16E-04	310.4605
300	367.9585	[0.4703 ; 0.4774]	[196.5643 ; 196.8287]	0.0273	357.9058
350	430.0936	[0.5444 ; 0.5506]	[196.9559 ; 197.0154]	0.1013	386.5238
400	487.6207	[0.6024 ; 0.6035]	[196.9725 ; 197.0212]	0.1785	400.5672
450	550.4117	[0.6426 ; 0.6441]	[196.991 ; 197.0341]	0.2482	413.8250
		Macro			
		classe 4			
		Simulation		$M^{[X]}/G/S/S$	
$\lambda^{clients}$	$\lambda_{M,4}$ [kbits/trame]	$p_{M,4}^b$	$X_{M,4}$	$p_{M,4}^b$	$X_{M,4}$
50	27.2429	0	[27.0622 ; 27.4924]	0	27.2429
100	53.8634	0	[53.7731 ; 53.958]	0	53.8634
150	80.2540	[0.0099 ; 0.0112]	[79.2921 ; 79.5779]	0	80.2540
200	108.0439	[0.1975 ; 0.1997]	[84.1568 ; 84.4079]	0	108.0439
250	132.0894	[0.3598 ; 0.3605]	[84.1767 ; 84.3931]	5.04E-04	132.0894
300	162.8216	[0.4711 ; 0.472]	[84.2038 ; 84.5095]	0.0414	156.0750
350	188.0241	[0.5470 ; 0.5592]	[85.1147 ; 85.2393]	0.1480	160.1949
400	214.4656	[0.5994 ; 0.6088]	[85.2484 ; 85.3465]	0.2530	160.2144
450	242.64	[0.6470 ; 0.6453]	[85.5291 ; 85.6166]	0.3424	159.5587

Tableau 12 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe dans la macro (micro déployée $n_{ABS} = 1$, $size(u) = 2048$ bits)

		micro			
		classe 2			
		Simulation		$M^{[X]}/G/S/S$	
$\lambda^{clients}$	$\lambda_{m,2}$ [kbits/trame]	$p_{m,2}^b$	$X_{m,2}$	$p_{m,2}^b$	$X_{m,2}$
50	1.7060	0	[1.6880 ; 1.7945]	0	1.7060
100	4.3904	0	[4.2867 ; 4.5874]	0	4.3904
150	7.3641	0	[7.277 ; 7.5666]	0	7.3641
200	11.1282	0	[11.0497 ; 11.1637]	0	11.1282
250	13.1059	[0.0289 ; 0.0293]	[12.7819 ; 12.9334]	0	13.1059
300	13.2770	[0.0345 ; 0.0361]	[13.1834 ; 13.2206]	0	13.2770
350	17.1471	[0.0645 ; 0.0652]	[16.6219 ; 16.7063]	3.29E-05	17.1465
400	21.6308	[0.0648 ; 0.0680]	[20.5626 ; 20.8990]	4.76E-04	21.6205
450	24.7943	[0.0855 ; 0.0859]	[23.9785 ; 24.0371]	0.0011	24.7673
		micro			
		classe 4			
		Simulation		$M^{[X]}/G/S/S$	
$\lambda^{clients}$	$\lambda_{m,4}$ [kbits/trame]	$p_{m,4}^b$	$X_{m,4}$	$p_{m,4}^b$	$X_{m,4}$
50	0.3231	0	[0.2885 ; 0.3834]	0	0.3231
100	1.3586	0	[1.3545 ; 1.4545]	0	1.3586
150	2.3302	0	[2.1 ; 2.5322]	0	2.3302
200	3.1469	0	[2.8515 ; 3.2907]	0	3.1469
250	3.8298	0	[3.7418 ; 4.0210]	0	3.8298
300	4.9990	0	[4.8387 ; 5.0981]	0	4.9990
350	4.1778	0	[4.1703 ; 4.2012]	0	4.1778
400	6.9880	[0.0184 ; 0.0192]	[6.7038 ; 6.7950]	9.19E-04	6.9816
450	6.9566	[0.0334 ; 0.0351]	[5.9720 ; 6.7091]	0.0020	6.9424

Tableau 13 – Comparaison entre les simulations Matlab et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage et de débit par classe dans la micro (micro déployée $n_{ABS} = 1$, $size(u) = 2048$ bits)

		classe 1		
$\lambda^{clients}$	$\lambda_{M,1}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
50	31.1201	[30.9888 ; 31.2639]	31.1201	31.1201
100	62.7241	[62.6894 ; 62.7471]	62.7241	62.7241
150	91.7102	[91.6894 ; 91.8304]	91.7102	91.7102
200	122.9891	[122.9572 ; 123.0101]	122.9891	122.9891
250	153.3447	[153.2223 ; 153.3612]	153.3447	153.3447
300	184.4042	[183.9402 ; 184.01]	183.9617	184.1368
350	215.1115	[202.4897 ; 202.8256]	202.6792	199.6584
400	244.6794	[207.5293 ; 207.6461]	210.5011	201.0828
450	280.9952	[207.5836 ; 207.6779]	222.4834	207.4052
		classe 2		
$\lambda^{clients}$	$\lambda_{M,2}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
50	14.4648	[14.3940 ; 14.6421]	14.4648	14.4648
100	28.4857	[28.4459 ; 28.5036]	28.4857	28.4857
150	43.6525	[43.6118 ; 43.7518]	43.6525	43.6525
200	58.0482	[58.01 ; 58.0629]	58.0482	58.0482
250	72.7883	[72.6711 ; 72.81]	72.7883	72.7883
300	86.8772	[86.5895 ; 86.6434]	86.6062	86.8013
350	101.8105	[93.46704 ; 93.9635]	93.0985	94.4967
400	116.6061	[96.5781 ; 96.9227]	93.2482	95.8294
450	128.1944	[95.8116 ; 96.7567]	93.8712	94.6215

Tableau 14 – Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 1024$ bits)

		classe 2		
$\lambda^{clients}$	$\lambda_{M,2}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
50	60.5446	[60.4798 ; 60.74348]	60.5446	60.5446
100	121.1943	[121.0450 ; 121.7214]	121.1943	121.1943
150	186.9437	[186.6958 ; 187.0116]	186.9437	186.6933
200	243.3541	[206.3081 ; 206.4629]	243.3541	199.4904
250	310.4555	[212.0583 ; 212.7579]	310.4489	205.8817
300	368.2238	[208.2185 ; 208.4863]	364.6392	202.8428
350	429.7056	[210.1794 ; 210.3095]	400.1872	203.2980
400	488.0057	[210.1937 ; 210.3106]	418.0058	201.9598
450	550.3944	[210.5080 ; 210.6505]	431.3723	202.3426
		classe 4		
$\lambda^{clients}$	$\lambda_{M,4}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
50	29.4988	[29.3463 ; 29.6586]	29.4989	29.4989
100	58.3250	[57.9645 ; 58.5247]	58.3250	58.3250
150	86.8903	[86.4101 ; 86.6658]	86.8903	86.7739
200	117.4388	[97.1382 ; 97.7665]	117.4388	96.2709
250	143.8669	[96.1150 ; 96.3516]	143.8620	95.4068
300	175.6578	[96.9799 ; 97.1090]	173.0266	96.7643
350	203.8163	[96.7123 ; 96.8045]	183.0853	96.4275
400	233.4387	[96.7284 ; 96.8627]	184.5435	96.6079
450	263.6836	[96.9922 ; 97.1114]	184.7030	96.9385

Tableau 15 – Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de débit par classe (macro sans micro, $size(u) = 2048$ bits)

		classe 1		
$\lambda^{clients}$	$\lambda_{M1,1}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	29.8647	0	0	0
200	59.8573	0	0	0
300	90.4674	0	0	0
400	125.7162	0	0	0
500	162.9559	0	0	0
600	200.0211	[0.0028 ; 0.0098]	2.755E-04	0.0052
700	237.4455	[0.0822 ; 0.0892]	0.0110	0.1040
800	271.3553	[0.1845 ; 0.1915]	0.1133	0.2063
900	305.2871	[0.2721 ; 0.2791]	0.2009	0.2939
		classe 2		
$\lambda^{clients}$	$\lambda_{M1,2}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	8.7834	0	0	0
200	17.7713	0	0	0
300	26.1056	0	0	0
400	35.8378	0	0	0
500	46.4859	0	0	0
600	57.7481	[0 ; 6.03E-05]	0.0052	0.0056
700	68.9060	[0.0088 ; 0.0093]	0.1040	0.1256
800	78.6568	[0.1111 ; 0.1116]	0.2063	0.1847
900	88.5935	[0.1987 ; 0.1992]	0.2939	0.2723

Tableau 16 – Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de probabilité de blocage par classe (macro M1, $size(u) = 1024$ bits)

		classe 1		
$\lambda^{clients}$	$\lambda_{M1,1}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	29.8647	[29.5924 ; 29.9745]	29.8647	29.8647
200	59.8573	[59.4597 ; 59.9781]	59.8573	59.8573
300	90.4674	[90.2726 ; 90.7684]	90.4674	90.4674
400	125.7162	[125.5381 ; 125.8364]	125.7163	125.7163
500	162.9559	[162.7124 ; 162.9794]	162.9559	162.9559
600	200.0211	[197.3074 ; 199.4446]	199.9626	198.9839
700	237.4455	[216.0278 ; 218.1650]	235.3601	212.7398
800	271.3553	[219.2719 ; 221.4091]	240.6107	215.3818
900	305.2871	[220.0814 ; 222.2186]	243.9550	215.5752
		classe 2		
$\lambda^{clients}$	$\lambda_{M1,2}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	8.7834	[8.6574 ; 8.9257]	8.7834	8.7834
200	17.7713	[17.5864 ; 17.9355]	17.7713	17.7713
300	26.1056	[26.0537 ; 26.4665]	26.1056	26.1056
400	35.8378	[35.7984 ; 35.9701]	35.8378	35.8378
500	46.4859	[46.3949 ; 46.7987]	46.4859	46.4859
600	57.7481	[57.2624 ; 57.6962]	57.4486	57.2731
700	68.9060	[67.7643 ; 68.3247]	61.7365	61.4782
800	78.6568	[69.8787 ; 69.911]	62.4320	64.1289
900	88.59351	[70.9456 ; 70.9899]	62.5593	64.4695

Tableau 17 – Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de débit par classe (macro M1, $size(u) = 1024$ bits)

		classe 1		
$\lambda^{clients}$	$\lambda_{M2,1}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	29.9609	0	0	0
200	60.9227	0	0	0
300	91.3065	0	0	0
400	125.7871	0	0	0
500	163.5559	0	0	0
600	200.1487	[0.0031 ; 0.0111]	7.55E-05	0.0059
700	237.4810	[0.0789 ; 0.0869]	0.0129	0.1005
800	271.2595	[0.1805 ; 0.1885]	0.1098	0.2028
900	305.3328	[0.2684 ; 0.2764]	0.1977	0.2907
		classe 2		
$\lambda^{clients}$	$\lambda_{M2,2}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	8.7382	0	0	0
200	17.6117	0	0	0
300	26.2693	0	0	0
400	36.0338	0	0	0
500	46.0449	0	0	0
600	58.1187	[0 ; 0.0108]	0.0059	0.0097
700	68.1080	[0.0084 ; 0.0162]	0.1005	0.1216
800	78.0223	[0.1076 ; 0.1081]	0.2028	0.1812
900	87.5675	[0.1955 ; 0.196]	0.2907	0.2691

Tableau 18 – Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de probabilité de blocage par classe (macro M2, $size(u) = 1024$ bits)

		classe 1		
$\lambda^{clients}$	$\lambda_{M2,1}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	29.9609	[29.4988 ; 29.9745]	29.9609	29.9609
200	60.9227	[59.9781 ; 60.1734]	60.9227	60.9227
300	91.3065	[89.4595 ; 90.7684]	91.3065	91.3065
400	125.7871	[119.0635 ; 125.8364]	125.7871	125.7872
500	163.5559	[149.6971 ; 162.9794]	163.5559	163.5559
600	200.1487	[198.1860 ; 198.4536]	199.9145	198.9669
700	237.4810	[217.8476 ; 218.1152]	235.2929	213.6212
800	271.2595	[221.0783 ; 221.3459]	241.4752	216.2578
900	305.3328	[222.0264 ; 222.2940]	244.9685	216.5827
		classe 2		
$\lambda^{clients}$	$\lambda_{M2,2}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	8.7382	[8.5285 ; 9.1908]	8.7382	8.7382
200	17.6117	[17.3091 ; 18.1823]	17.6117	17.6117
300	26.2693	[25.6669 ; 26.7248]	26.2693	26.2693
400	36.0338	[33.9274 ; 35.2671]	36.0338	36.0338
500	46.0449	[41.9917 ; 43.3755]	46.0448	46.0449
600	58.1187	[58.0634 ; 58.5855]	57.7755	57.5876
700	68.1080	[67.0046 ; 67.5697]	61.2652	61.0050
800	78.0203	[69.5863 ; 69.6253]	62.2006	63.883
900	87.5675	[70.4043 ; 70.4481]	62.2146	64.003

Tableau 19 – Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de débit par classe (macro M2, $size(u) = 1024$ bits)

		classe 1		
$\lambda^{clients}$	$\lambda_{c,1}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	6.4779	0	0	0
200	12.6879	0	0	0
300	19.3832	[0.0010 ; 0.0051]	0.0246	0.0255
400	22.4552	[0.1511 ; 0.1770]	0.0805	0.0964
500	23.47133	[0.0958 ; 0.1347]	0.0976	0.1193
600	23.7801	[0.1036 ; 0.1328]	0.1108	0.1366
700	24.75406	[0.1309 ; 0.1774]	0.1271	0.1581
800	39.4537	[0.1615 ; 0.1685]	0.2397	0.2580
900	44.3954	[0.2331 ; 0.2401]	0.3113	0.3296
		classe 2		
$\lambda^{clients}$	$\lambda_{c,2}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	7.4217	0	0	0
200	14.2397	0	0	0
300	21.4519	[7.22E-04 ; 0.0039]	0.0246	0.0393
400	25.02584	[0.0423 ; 0.0162]	0.0964	0.1197
500	26.3114	[0.0494 ; 0.0799]	0.1193	0.1438
600	26.2965	[0.0963 ; 0.0674]	0.1624	0.1366
700	27.5534	[0.1025 ; 0.1254]	0.1849	0.1581
800	31.3327	[0.1628 ; 0.1633]	0.2580	0.2364
900	35.4258	[0.2344 ; 0.2349]	0.3296	0.308

Tableau 20 – Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de probabilité de blocage par classe (région de coopération, $size(u) = 1024$ bits)

		classe 1		
$\lambda^{clients}$	$\lambda_{c,1}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	6.4779	[6.2938 ; 6.8417]	6.4779	6.4779
200	12.6879	[12.5978 ; 13.2511]	12.6879	12.6879
300	19.3832	[18.7678 ; 19.8455]	18.9062	18.7024
400	22.4552	[20.1092 ; 21.3855]	20.3044	20.2883
500	23.4713	[20.1990 ; 21.4313]	20.6927	20.5294
600	23.7801	[20.3002 ; 21.4787]	20.8243	20.6706
700	24.7540	[20.3677 ; 21.5283]	20.9613	20.8389
800	39.4537	[20.8057 ; 21.6819]	20.9966	20.957
900	44.3954	[21.0502 ; 21.7904]	21.5751	21.3371
		classe 2		
$\lambda^{clients}$	$\lambda_{c,2}$ [kbits/trame]	Simul	$M^{[X]}/G/S/S$	$M/G/S/S$
100	7.4217	[7.1816 ; 7.8202]	7.4217	7.4217
200	14.2397	[13.9306 ; 14.7948]	14.2397	14.2397
300	21.4519	[20.8950 ; 21.8972]	20.9985	20.9239
400	25.0258	[23.4435 ; 24.6333]	22.6379	20.6109
500	26.3114	[23.6143 ; 24.7104]	22.8401	22.7018
600	26.2965	[23.7629 ; 24.7916]	22.1718	23.2895
700	27.5534	[23.9998 ; 24.7963]	23.1955	23.3244
800	31.3327	[23.9799 ; 24.8317]	23.2502	23.9256
900	35.4258	[24.043 ; 24.9279]	23.7500	24.5147

Tableau 21 – Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de débit par classe (région de coopération, $size(u) = 1024$ bits)

Etude initiale : influence du nombre de serveurs sur la précision de nos modèles

Nous avons effectué une étude initiale sur l'influence du nombre de serveurs sur la précision des 3 approches proposées : chaîne de Markov, $M/G/S/S$ et $M^{[X]}/G/S/S$. Le but étant de faire une étude numérique, nous nous permettons alors de considérer même des nombres de serveurs non réalistes.

Dans le Tableau 22, nous comparons la précision de la $M/G/S/S$ et de la $M^{[X]}/G/S/S$, pour une taille de paquet $size(u) = 1024$ bits avec un taux d'arrivée de clients $\lambda^{clients} = 50$ clients/trame, en termes de probabilité de blocage par classe en faisant varier le nombre de serveurs S , lorsque la macro est déployée sans micro.

S	classe 1			classe 2		
	Simul	$M/G/S/S$	$M^{[X]}/G/S/S$	Simul	$M/G/S/S$	$M^{[X]}/G/S/S$
10	[0.8532 ; 0.8651]	0.8676	0.8189	[0.8578 ; 0.8645]	0.8676	0.8946
30	[0.5781 ; 0.5971]	0.6086	0.5265	[0.5845 ; 0.5963]	0.6086	0.6542
50	[0.3119 ; 0.3346]	0.3576	0.2927	[0.3135 ; 0.3309]	0.3576	0.3999
55	[0.2455 ; 0.2676]	0.2944	0.2397	[0.2517 ; 0.2707]	0.2944	0.3346
60	[0.1885 ; 0.2105]	0.2402	0.1947	[0.1951 ; 0.2139]	0.2402	0.2768
65	[0.1269 ; 0.1462]	0.1842	0.1503	[0.1310 ; 0.1485]	0.1842	0.2178
70	[0.0807 ; 0.0973]	0.1334	0.1109	[0.0858 ; 0.1016]	0.1334	0.1635
75	[0.0463 ; 0.0590]	0.0889	0.0768	[0.0456 ; 0.0571]	0.0889	0.1151
80	[0.0228 ; 0.0331]	0.0533	0.0493	[0.0223 ; 0.0309]	0.0533	0.0751
85	[0.0102 ; 0.0167]	0.0270	0.0281	[0.0110 ; 0.0172]	0.0270	0.0435
90	[0.003 ; 0.0075]	0.0116	0.0114	[0.0042 ; 0.0084]	0.0116	0.0226
120	0	0	0	0	0	0
200	0	0	0	0	0	0
300	0	0	0	0	0	0
400	0	0	0	0	0	0
500	0	0	0	0	0	0

Tableau 22 – Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage par classe (macro sans micro), $\lambda^{clients} = 50$ clients/trame

Pour la classe 1, la comparaison des probabilités de blocage obtenues par les trois approches : simulations Matlab, $M/G/S/S$ et $M^{[X]}/G/S/S$ énumérés dans le Tableau 22 révèlent que lorsque le nombre de serveurs S varie entre 10 et 30 serveurs, la $M/G/S/S$ est plus précise que la $M^{[X]}/G/S/S$ offrant ainsi des probabilités de blocage plus proches que celles offertes par la simulation. Toutefois, lorsque le nombre de serveurs S varie entre 50 et 90 serveurs, la $M^{[X]}/G/S/S$ offre des probabilités de blocage plus proches de celles de la simulation que la $M/G/S/S$.

Quant à la classe 2, lorsque le nombre de serveurs S varie entre 10 et 90 serveurs, la $M/G/S/S$ est plus précise que la $M^{[X]}/G/S/S$ offrant des probabilités de blocage plus proches que celles obtenues par la simulation de la chaîne de Markov.

Par ailleurs, lorsque le nombre de serveurs est important, la $M/G/S/S$, la $M^{[X]}/G/S/S$ et les si-

mulations offrent, pour la classe 1 et la classe 2, exactement les mêmes résultats (des probabilités de blocage nulles).

En résumé, à faibles et très fortes charges les deux modèles de files d'attente offrent les mêmes résultats. A moyennes charges, la $M/G/S/S$ est plus précise que la $M^{[X]}/G/S/S$ et à forte charge $M^{[X]}/G/S/S$ est plus proche que la $M/G/S/S$.

Dans le Tableau 23, nous comparons la précision de la $M/G/S/S$ et de la $M^{[X]}/G/S/S$, pour une taille de paquet $size(u) = 1024$ bits avec un taux d'arrivée de clients $\lambda = 200$ clients/trame, en termes de probabilité de blocage par classe en faisant varier le nombre de serveurs S , lorsque la macro est déployée sans micro.

S	classe 1			classe 2		
	Simul	$M/G/S/S$	$M^{[X]}/G/S/S$	Simul	$M/G/S/S$	$M^{[X]}/G/S/S$
50	[0.8265 ; 0.8327]	0.8334	0.7758	[0.8316 ; 0.8352]	0.8334	0.8633
100	[0.6591 ; 0.6677]	0.6688	0.5880	[0.6602 ; 0.6659]	0.6688	0.7099
150	[0.4962 ; 0.5057]	0.5059	0.4242	[0.4937 ; 0.5011]	0.5059	0.5472
200	[0.3245 ; 0.3350]	0.3422	0.2765	[0.3271 ; 0.3360]	0.3422	0.3775
250	[0.1639 ; 0.1752]	0.1839	0.1455	[0.1643 ; 0.1747]	0.1839	0.2085
300	[0.0262 ; 0.0331]	0.0477	0.0410	[0.0268 ; 0.0337]	0.0477	0.0614
350	[0 ; 5.74E-04]	5.46E-04	0.0012	[0 ; 3.65E-04]	5.46E-04	0.0019
400	0	0	0	0	0	0
450	0	0	0	0	0	0
500	0	0	0	0	0	0

Tableau 23 – Comparaison entre les simulations Matlab, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ en terme de probabilités de blocage par classe (macro sans micro), $\lambda = 200$ clients/trame

D'après les résultats listés dans le Tableau 23, nous remarquons également que lorsque le nombre de serveurs est faible, la $M/G/S/S$ est plus précise que la $M^{[X]}/G/S/S$ offrant des probabilités de blocage plus proches que celles obtenues par la simulation de la chaîne de Markov. Lorsqu'il est moyen, la $M^{[X]}/G/S/S$ offre des probabilités de blocage plus proche de celles de la simulation que la $M/G/S/S$. Lorsque le nombre de serveurs est grand, la $M/G/S/S$, la $M^{[X]}/G/S/S$ et les simulations offrent, pour la classe 1 et la classe 2, exactement les mêmes résultats (des probabilités de blocage nulles).

Ainsi, en résumé, lorsque le taux d'arrivée des clients est faible, la $M/G/S/S$ et la $M^{[X]}/G/S/S$ donnent les mêmes débits globaux (probabilités de blocage nulles). A moyenne charge, la $M^{[X]}/G/S/S$ est plus précise que la la $M/G/S/S$. A forte charge, la $M/G/S/S$ est plus précise que la la $M^{[X]}/G/S/S$.

Bibliographie

- [3GP08] 3GPP. Further discussion on inter-cell interference mitigation through limited coordination(r1-083569), 2008.
- [3GP09a] 3GPP. Discussion on comp-su-mimo(r1-090613), 2009.
- [3GP09b] 3GPP. Further discussion of frequency plan scheme on comp-su-mimo(r1-091415), 2009.
- [3GP17] 3GPP. Study on new radio access technology : Radio access architecture and interfaces. 2017.
- [Abb16a] Nivine Abbas. *Conception et performance de sch ? mas de coordination dans les r ? seaux cellulaires*. PhD thesis, Paris, ENST, 2016.
- [Abb16b] Nivine Abbas. *Design and performance of coordination schemes in cellular networks*. PhD thesis, 2016.
- [ABC⁺14a] Jeffrey Andrews, Stefano Buzzi, Wan Choi, Stephen Hanly, Angel Lozano, Anthony Soong, and Jianzhong Charlie Zhang. What will 5g be? *IEEE Journal on selected areas in communications*, 32(6) :1065–1082, 2014.
- [ABC⁺14b] Jeffrey Andrews, Stefano Buzzi, Wan Choi, Stephen Hanly, Angel Lozano, Anthony CK Soong, and Jianzhong Charlie Zhang. What will 5g be? *IEEE Journal on selected areas in communications*, 32(6) :1065–1082, 2014.
- [ABS15] Nivine Abbas, Thomas Bonald, and Berna Sayrac. Opportunistic gains of mobility in cellular data networks. In *2015 13th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, pages 315–322. IEEE, 2015.
- [ABS16] Nivine Abbas, Thomas Bonald, and Berna Sayrac. Mobility-aware scheduler in comp systems. In *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pages 1–7. IEEE, 2016.
- [Acc11] Evolved Universal Terrestrial Radio Access. Physical layer procedures. *Release*, 9 :v9, 2011.
- [Acc17] Evolved Universal Terrestrial Radio Access. Radio frequency (rf) requirements for lte pico node b (release 14), 2017.

- [Adi10] Sasan Adibi. Traffic classification—packet-, flow-, and application-based approaches. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 1(1) :6–15, 2010.
- [All15a] NGMN Alliance. Further study on critical c-ran technologies. *Next Generation Mobile Networks*, 2015.
- [All15b] NGMN Alliance. Ran evolution project backhaul and fronthaul evolution. *Next Generation Mobile Networks (NGMN)*, ver, 1, 2015.
- [And13] Jeffrey G Andrews. Seven ways that hetnets are a cellular paradigm shift. *IEEE Communications Magazine*, 51(3) :136–144, 2013.
- [Bay00] Bruno Baynat. *La théorie des files d’attente : des chaînes de Markov aux réseaux à forme produit*. 2000.
- [Ber07] Claude Berrou. *Codes et turbocodes*. Springer Science & Business Media, 2007.
- [BHW12] Yannick Bouguen, Eric Hardouin, and François-Xavier Wolff. *LTE pour les reseaux 4G*. Editions Eyrolles, 2012.
- [BSS13] Mokhtar S Bazaraa, Hanif D Sherali, and Chitharanjan M Shetty. *Nonlinear programming : theory and algorithms*. John Wiley & Sons, 2013.
- [BVCM⁺17] Jens Bartelt, Nikola Vucic, Daniel Camps-Mur, Eduard Garcia-Villegas, Ilker Demirkol, Albrecht Fehske, Michael Grieger, Anna Tzanakaki, Jesús Gutierrez, Eckhard Grass, et al. 5g transport network requirements for the next generation fronthaul interface. *EURASIP Journal on Wireless Communications and Networking*, 2017(1) :1–12, 2017.
- [CCY⁺14] Aleksandra Checko, Henrik L Christiansen, Ying Yan, Lara Scolari, Georgios Kardaras, Michael S Berger, and Lars Dittmann. Cloud ran for mobile networks—a technology overview. *IEEE Communications surveys & tutorials*, 17(1) :405–426, 2014.
- [CD11] K Chen and R Duan. C-ran the road towards green ran. white paper. *China Mobile Research Institute, Tech. Rep*, 2011.
- [CFBF19] Jay Kant Chaudhary, Jobin Francis, André Noll Barreto, and Gerhard Fettweis. Latency in the uplink of massive mimo cran with packetized fronthaul : Modeling and analysis. In *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–7. IEEE, 2019.
- [Che16] Aleksandra Checko. Cloud radio access network architecture. towards 5g mobile networks. 2016.
- [ciw19] ciwPython. A discrete event simulation library for open queueing networks, 2019. URL : <https://github.com/CiwPython/Ciw>.

- [CLHD⁺14] I Chih-Lin, Jinri Huang, Ran Duan, Chunfeng Cui, Jesse Xiaogen Jiang, and Lei Li. Recent progress on c-ran centralization and cloudification. *IEEE Access*, 2 :1030–1039, 2014.
- [CLRH⁺14] I Chih-Lin, Corbett Rowell, Shuangfeng Han, Zhikun Xu, Gang Li, and Zhengang Pan. Toward green and soft : A 5g perspective. *IEEE Communications Magazine*, 52(2) :66–73, 2014.
- [CLYHL19] I Chih-Lin, Guanding Yu, Shuangfeng Han, and Geoffrey Ye Li. *Green and Software-defined Wireless Networks : From Theory to Practice*. Cambridge University Press, 2019.
- [CNS16] Chia-Yu Chang, Navid Nikaein, and Thrasyvoulos Spyropoulos. Impact of packetization and scheduling on c-ran fronthaul performance. In *2016 IEEE Global Communications Conference (GLOBECOM)*, pages 1–7. IEEE, 2016.
- [COR12] CORDIS. Interworking and joint design of an open access and backhaul network architecture for small cells based on cloud networks, ijoin, 2012. URL : <https://cordis.europa.eu/project/rcn/105819/factsheet/fr>
- [CPR15] Common Public Radio Interface CPRI. Interface specification (v7. 0). Technical report, Technical report, 2015.
- [CWW⁺13] Michal Cierny, Haining Wang, Risto Wichman, Zhi Ding, and Carl Wijting. On number of almost blank subframes in heterogeneous cellular networks. *IEEE Transactions on Wireless Communications*, 12(10) :5061–5073, 2013.
- [CYRN⁺16] Chang Chia-Yu, S Ruggero, N Navid, S Thrasyvoulos, and B Christian. Impact of packetization and functional split on c-ran fronthaul performance. In *Proc. of IEEE ICC*, 2016.
- [DDM⁺13] Uwe Dötsch, Mark Doll, Hans-Peter Mayer, Frank Schaich, Jonathan Segel, and Philippe Sehier. Quantitative analysis of split base station processing and determination of advantageous architectures for lte. *Bell Labs Technical Journal*, 18(1) :105–128, 2013.
- [DKM⁺11] Xavier Dutreilh, Sergey Kirgizov, Olga Melekhova, Jacques Malenfant, Nicolas Rivierre, and Isis Truck. Using reinforcement learning for autonomic resource allocation in clouds : towards a fully automated workflow. In *ICAS 2011, The Seventh International Conference on Autonomic and Autonomous Systems*, pages 67–74, 2011.
- [DMBP13] Alexei Davydov, Gregory Morozov, Ilya Bolotin, and Apostolos Papathanassiou. Evaluation of joint transmission comp in c-ran based lte-a hetnets with large coordination areas. In *2013 IEEE Globecom Workshops (GC Wkshps)*, pages 801–806. IEEE, 2013.

- [DMMS13] Supratim Deb, Pantelis Monogioudis, Jerzy Miernik, and James P Seymour. Algorithms for enhanced inter-cell interference coordination (eicic) in lte hetnets. *IEEE/ACM transactions on networking*, 22(1) :137–150, 2013.
- [DMV05] Hamza Dahmouni, Bertrand Morin, and Sandrine Vaton. Performance modelling of gsm/gprs cells with different radio resource allocation strategies. In *IEEE Wireless Communications and Networking Conference, 2005*, volume 3, pages 1317–1322. IEEE, 2005.
- [DSG14] Ameneh Daeinabi, Kumbesan Sandrasegaran, and Pantha Ghosal. An enhanced intercell interference coordination scheme using fuzzy logic controller in lte-advanced heterogeneous networks. In *2014 International Symposium on Wireless Personal Multimedia Communications (WPMC)*, pages 520–525. IEEE, 2014.
- [E⁺17] AB Ericsson et al. cpri specification v1.0. common public radio interface : cpri interface specification. *Huawei Technologies Co. Ltd, NEC Corporation, Alcatel Lucent, and Nokia Networks*, 2017.
- [EJAGS19] Salah Eddine Elayoubi, Sana Ben Jemaa, Zwi Altman, and Ana Galindo-Serrano. 5g ran slicing for verticals : Enablers and challenges. *IEEE Communications Magazine*, 57(1) :28–34, 2019.
- [EL02] John Eidson and Kang Lee. Ieee 1588 standard for a precision clock synchronization protocol for networked measurement and control systems. In *Sensors for Industry Conference, 2002. 2nd ISA/IEEE*, volume 10. IEEE, 2002.
- [Eri14a] Ericson. 5g : what is it for?, 2014. URL : <https://www.ericsson.com/assets/local/news/2014/10/5g-what-is-it-for.pdf>.
- [Eri14b] AB Ericsson. Huawei technologies co. ltd, nec corporation, nokia networks, alcatel-lucent,“cpri specification v6. 1 (2014-07-01) : Common public radio interface (cpri) ; interface specification,”, 2014.
- [Erl17] Agner Krarup Erlang. Solution of some problems in the theory of probabilities of significance in automatic telephone exchanges. *Post Office Electrical Engineer’s Journal*, 10 :189–197, 1917.
- [ETS] GSORI ETSI. 002-1 v4. 1.1,“. *Open Radio equipment Interface (ORI)*, year=2014.
- [ETS16] TS ETSI. 36.211, " lte ; evolved universal terrestrial radio access (e-utra) ; physical channels and modulation. *Technical Specification Version 13.0.0 Release 13*, 2016.
- [Fak90] D Fakinos. On the m/g/k group-arrival loss system. *European journal of operational research*, 44(1) :75–83, 1990.
- [FGJ⁺08] J-L Ferrant, Mike Gilson, Sebastien Jobert, Michael Mayer, Michel Ouellette, Laurent Montini, Silvana Rodrigues, and Stefano Ruffini. Synchronous ethernet : A

method to transport synchronization. *IEEE Communications Magazine*, 46(9) :126–134, 2008.

- [Flo95] Christodoulos A Floudas. *Nonlinear and mixed-integer optimization : fundamentals and applications*. Oxford University Press, 1995.
- [GCT⁺15] Nathan J Gomes, Philippe Chanclou, Peter Turnbull, Anthony Magee, and Volker Jungnickel. Fronthaul evolution : from cpri to ethernet. *Optical Fiber Technology*, 26 :50–58, 2015.
- [GHLK17] Apurva Gowda, José Alberto Hernández, David Larrabeiti, and Leonid Kazovsky. Delay analysis of mixed fronthaul and backhaul traffic under strict priority queueing discipline in a 5g packet transport network. *Transactions on Emerging Telecommunications Technologies*, 28(6) :e3168, 2017.
- [GJ89] Winfried K Grassmann and Joti L Jain. Numerical solutions of the waiting time distribution and idle time distribution of the arithmetic gi/g/1 queue. *Operations Research*, 37(1) :141–150, 1989.
- [GJD⁺11] Ismail Guvenc, Moo-Ryong Jeong, Ibrahim Demirdogen, Balkan Kecicioglu, and Fujio Watanabe. Range expansion and inter-cell interference coordination (icic) for picocell networks. In *2011 IEEE Vehicular Technology Conference (VTC Fall)*, pages 1–6. IEEE, 2011.
- [GMR⁺12] Amitabha Ghosh, Nitin Mangalvedhe, Rapeepat Ratasuk, Bishwarup Mondal, Mark Cudak, Eugene Visotsky, Timothy A Thomas, Jeffrey G Andrews, Ping Xia, Han Shin Jo, et al. Heterogeneous cellular networks : From theory to practice. *IEEE communications magazine*, 50(6) :54–64, 2012.
- [GRM⁺10] Amitava Ghosh, Rapeepat Ratasuk, Bishwarup Mondal, Nitin Mangalvedhe, and Tim Thomas. Lte-advanced : next-generation wireless broadband technology. *IEEE wireless communications*, 17(3) :10–22, 2010.
- [GSM20] GSMA. Global mobile economy report 2020, 2020. URL : <https://www.gsma.com/mobileeconomy/>.
- [HB13] Mor Harchol-Balter. *Performance modeling and design of computer systems : queueing theory in action*. Cambridge University Press, 2013.
- [HE11] Jonas Hansryd and Jonas Edstam. Microwave capacity evolution. *Ericsson review*, 1 :22–27, 2011.
- [I⁺05] Open Base Station Architecture Initiative et al. Reference point 1 test messages. *Version*, 1 :21, 2005.
- [I⁺16] Open Base Station Architecture Initiative et al. Bts system reference document. *V2. 0*, 2016.

- [Ind17] Cisco Visual Networking Index. Global mobile data traffic forecast update, 2016–2021. *white paper*, 2017.
- [JG07] Michel Joindot and Alain Glavieux. *Introduction aux communications numériques*. Dunod, 2007.
- [JL12] Lei Jiang and Ming Lei. Resource allocation for eicic scheme in heterogeneous networks. In *2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications-(PIMRC)*, pages 448–453. IEEE, 2012.
- [KBE13] Ahlem Khlass, Thomas Bonald, and Salah Eddine Elayoubi. Flow-level performance of intra-site coordination in cellular networks. In *2013 11th International Symposium and Workshops on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, pages 216–223. IEEE, 2013.
- [Ken53] David G Kendall. Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded markov chain. *The Annals of Mathematical Statistics*, pages 338–354, 1953.
- [Kin61] JFC Kingman. The single server queue in heavy traffic. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 57, pages 902–904. Cambridge University Press, 1961.
- [Kle76] Leonard Kleinrock. *Queueing systems, volume 2 : Computer applications*, volume 66. Wiley New York, 1976.
- [LCP⁺10] Jing Liu, Yongyu Chang, Qun Pan, Xin Zhang, and Dacheng Yang. A novel transmission scheme and scheduling algorithm for comp-su-mimo in lte-a system. In *2010 IEEE 71st Vehicular Technology Conference*, pages 1–5. IEEE, 2010.
- [LdlCdD⁺14a] Víctor López, Beatriz de la Cruz, Óscar González de Dios, Ori Gerstel, Norberto Amaya, George Zervas, Dimitra Simeonidou, and Juan Pedro Fernandez-Palacios. Finding the target cost for sliceable bandwidth variable transponders. *Journal of Optical Communications and Networking*, 6(5) :476–485, 2014.
- [LdlCdD⁺14b] Víctor López, Beatriz de la Cruz, Óscar González de Dios, Ori Gerstel, Norberto Amaya, George Zervas, Dimitra Simeonidou, and Juan Pedro Fernandez-Palacios. Finding the target cost for sliceable bandwidth variable transponders. *Journal of Optical Communications and Networking*, 6(5) :476–485, 2014.
- [Lin52] David V Lindley. The theory of queues with a single server. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 48, pages 277–289. Cambridge University Press, 1952.
- [LJG13] Wei Luo, Yusheng Ji, and Aihuang Guo. An adaptive abs-comp scheme in lte-advanced heterogeneous networks. In *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pages 2769–2773. IEEE, 2013.

- [LLL⁺12] Yu-Ngok Ruyue Li, Jian Li, Weimin Li, Yan Xue, and Huaming Wu. Comp and interference coordination in heterogeneous network for lte-advanced. In *2012 IEEE Globecom Workshops*, pages 1107–1111. IEEE, 2012.
- [LSC⁺12] Daewon Lee, Hanbyul Seo, Bruno Clerckx, Eric Hardouin, David Mazzaresse, Satoshi Nagata, and Krishna Sayana. Coordinated multipoint transmission and reception in lte-advanced : deployment scenarios and operational challenges. *IEEE Communications Magazine*, 50(2) :148–155, 2012.
- [LSZ⁺10] Yonghua Lin, Ling Shao, Zhenbo Zhu, Qing Wang, and Ravie K Sabhikhi. Wireless network cloud : Architecture and system requirements. *IBM Journal of Research and Development*, 54(1) :4–1, 2010.
- [LZH⁺18] Li Li, Zhaorong Zhou, Yanjun Hu, Tao Jiang, and Menghan Wei. Pricing framework for almost blank subframe scheme in two-tier heterogeneous networks. *International Journal of Communication Systems*, 31(3) :e3454, 2018.
- [LZSW19] Li Li, Zhaorong Zhou, Sanshan Sun, and Menghan Wei. Distributed optimization of enhanced intercell interference coordination and resource allocation in heterogeneous networks. *International Journal of Communication Systems*, 32(6) :e3915, 2019.
- [M⁺85] David L Mills et al. Network time protocol (ntp). 1985.
- [Mai14] R Maiden. Standard for radio over ethernet encapsulations and mappings, 1914.
- [Mat20] Mathworks. How globalsearch and multistart work, 2020. URL : <https://fr.mathworks.com/help/gads/how-globalsearch-and-multistart-work.html#bsds7nj-1>.
- [MBFP15] Dania Marabissi, Giulio Bartoli, Romano Fantacci, and Marco Pucci. An optimized comp transmission for a heterogeneous network using eicic approach. *IEEE Transactions on Vehicular Technology*, 65(10) :8230–8239, 2015.
- [Med02] Jyotiprasad Medhi. *Stochastic models in queueing theory*. Elsevier, 2002.
- [MMS18] Ghizlane Mountaser, Toktam Mahmoodi, and Osvaldo Simeone. Reliable and low-latency fronthaul for tactile internet applications. *IEEE Journal on Selected Areas in Communications*, 36(11) :2455–2463, 2018.
- [Net13] Mobile Cloud Networking. Mobile cloud networking project, mcn, 2013. URL : <http://mobile-cloud-networking.eu/site/>.
- [Nik15] Navid Nikaein. Processing radio access network functions in the cloud : Critical issues and modeling. In *Proceedings of the 6th International Workshop on Mobile Cloud Computing and Services*, pages 36–43, 2015.
- [NW06] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

- [PHL17] Gabriel Otero Pérez, José Alberto Hernández, and David Larrabeiti López. Delay analysis of fronthaul traffic in 5g transport networks. In *2017 IEEE 17th International Conference on Ubiquitous Wireless Broadband (ICUWB)*, pages 1–5. IEEE, 2017.
- [PHL18] Gabriel Otero Pérez, José Alberto Hernández, and David Larrabeiti. Fronthaul network modeling and dimensioning meeting ultra-low latency requirements for 5g. *IEEE/OSA Journal of Optical Communications and Networking*, 10(6) :573–581, 2018.
- [PLZW15] Mugen Peng, Yong Li, Zhongyuan Zhao, and Chonggang Wang. System architecture and key technologies for 5g heterogeneous cloud radio access networks. *IEEE network*, 29(2) :6–14, 2015.
- [PWL⁺15] Mugen Peng, Chonggang Wang, Jian Li, Hongyu Xiang, and Vincent Lau. Recent advances in underlay heterogeneous networks : Interference control, resource allocation, and self-organization. *IEEE Communications Surveys & Tutorials*, 17(2) :700–729, 2015.
- [Rad13] 3GPP Terrestrial Radio. Coordinated multipoint operation for lte physical layer aspects. *Release*, 11 :v11, 2013.
- [RWN⁺17] Chathurika Ranaweera, Elaine Wong, Ampalavanapillai Nirmalathas, Chamil Jayasundara, and Christina Lim. 5g c-ran architecture : A comparison of multiple optical fronthaul networks. In *2017 International Conference on Optical Network Design and Modeling (ONDM)*, pages 1–6. IEEE, 2017.
- [Rya16] Timothy Ryan. Radio over ethernet, 2016. US Patent 9,301,267.
- [SBS⁺16] Rajarajan Sivaraj, Ioannis Broustis, NK Shankaranarayanan, Vaneet Aggarwal, Rittwik Jana, and Prasant Mohapatra. A qos-enabled holistic optimization framework for lte-advanced heterogeneous networks. In *INFOCOM*, pages 1–9, 2016.
- [SDR17] Selma Sbit, Mohamed Bechir Dadi, and Belgacem Chibani Rhaimi. Comparison of inter cell interference coordination approaches. *International Journal of Electrical and Information Engineering*, 11(7) :871–876, 2017.
- [SF07] Marc Schleyer and Kai Furmans. An analytical method for the calculation of the waiting time distribution of a discrete time g/g/1-queueing system with batch arrivals. *OR Spectrum*, 29(4) :745–763, 2007.
- [SGM12] Konstantin E Samouylov, Irina A Gudkova, and Nataliya D Maslovskaya. A model for analysing impact of frequency reuse on inter-cell interference in lte network. In *2012 IV International Congress on Ultra Modern Telecommunications and Control Systems*, pages 298–301. IEEE, 2012.

- [SP14] Beatriz Soret and Klaus I Pedersen. Centralized and distributed solutions for fast muting adaptation in lte-advanced hetnets. *IEEE Transactions on Vehicular Technology*, 64(1) :147–158, 2014.
- [SUFM16] Takuya Shimojo, Anil Umesh, Daisuke Fujishima, and Atsushi Minokuchi. Special articles on 5g technologies toward 2020 deployment. *NTT DOCOMO Tech. J.*, 17(4) :50–59, 2016.
- [TCE17] Massimo Tornatore, Gee-Kung Chang, and Georgios Ellinas. *Fiber-Wireless Convergence in Next-Generation Communication Networks*. Springer, 2017.
- [TCTJ⁺18a] Hadjer Touati, Hind Castel-Taleb, Badii Jouaber, Sara Akbarzadeh, and Ahlem Khlass. Modeling and performance evaluation of the eicic/abs in h-cran. 2018.
- [TCTJ⁺18b] Hadjer Touati, Hind Castel-Taleb, Badii Jouaber, Sara Akbarzadeh, and Ahlem Khlass. Simulation, modeling and analysis of the eicic/abs in h-cran. In *2018 IFIP/IEEE International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN)*, pages 1–6. IEEE, 2018.
- [TCTJA19a] Hadjer Touati, Hind Castel-Taleb, Badii Jouaber, and Sara Akbarzadeh. Model-based optimization for jt comp in c-ran. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 403–409. IEEE, 2019.
- [TCTJA19b] Hadjer Touati, Hind Castel-Taleb, Badii Jouaber, and Sara Akbarzadeh. Queue-based model approximation for inter-cell coordination with service differentiation. In *2019 IEEE Symposium on Computers and Communications (ISCC)*, pages 1–7. IEEE, 2019.
- [TCTJA20a] Hadjer Touati, Hind Castel-Taleb, Badii Jouaber, and Sara Akbarzadeh. A new lower cost ul split option for ultra-low latency 5g fronthaul. In *2020 IEEE 16th Annual International conference on Wireless Communications Mobile Computing (IWCMC)*, pages 1–7. IEEE, 2020.
- [TCTJA20b] Hadjer Touati, Hind Castel-Taleb, Badii Jouaber, and Sara Akbarzadeh. Split analysis and fronthaul dimensioning in 5g c-ran to guarantee ultra low latency. In *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*, pages 1–4. IEEE, 2020.
- [VG18] Stefan Vigerske and Ambros Gleixner. Scip : Global optimization of mixed-integer nonlinear programs in a branch-and-cut framework. *Optimization Methods and Software*, 33(3) :563–593, 2018.
- [VMT⁺13] Eugene Visotsky, Bishwarup Mondal, Timothy A Thomas, Nitin Mangalvedhe, Amitava Ghosh, and Xiaoyi Wang. Joint scheduling for comp and eicic in heterogeneous network deployments. In *2013 IEEE 77th Vehicular Technology Conference (VTC Spring)*, pages 1–5. IEEE, 2013.

- [WCS⁺18] Dan Wang, Dong Chen, Bin Song, Nadra Guizani, Xiaoyan Yu, and Xiaojiang Du. From iot to 5g i-iot : The next generation iot-based intelligent algorithms and 5g technologies. *IEEE Communications Magazine*, 56(10) :114–120, 2018.
- [WH19] You-Chiun Wang and Chien-Chun Huang. Efficient management of interference and power by jointly configuring abs and drx in lte-a hetnets. *Computer Networks*, 150 :15–27, 2019.
- [WHG⁺14] Cheng-Xiang Wang, Fourat Haider, Xiqi Gao, Xiao-Hu You, Yang Yang, Dongfeng Yuan, Hadi M Aggoune, Harald Haas, Simon Fletcher, and Erol Hepsaydir. Cellular architecture and key technologies for 5g wireless communication networks. *IEEE communications magazine*, 52(2) :122–130, 2014.
- [ZAM⁺18] Ali Zaidi, Fredrik Athley, Jonas Medbo, Ulf Gustavsson, Giuseppe Durisi, and Xiaoming Chen. *5G Physical Layer : principles, models and technology components*. Academic Press, 2018.
- [ZCJL14] Tong Zhou, Yan Chen, Chunxiao Jiang, and KJ Ray Liu. Pricing game for time mute in femto–macro coexistent networks. *IEEE Transactions on Wireless Communications*, 14(4) :2118–2130, 2014.
- [ZJWY16] Hao Zhou, Yusheng Ji, Xiaoyan Wang, and Shigeki Yamada. eicic configuration algorithm with service scalability in heterogeneous cellular networks. *IEEE/ACM Transactions on Networking*, 25(1) :520–535, 2016.
- [ZM07] Jim Zyren and Wes McCoy. Overview of the 3gpp long term evolution physical layer. *Freescale Semiconductor, Inc., white paper*, 7 :2–22, 2007.



Titre : Allocation de Ressources et Partage de Charge adaptées aux architectures de réseaux d'accès basées sur le cloud

Mots clés : Réseau d'accès, Evaluation des performances, Optimisation, C-RAN, Allocation de ressources

Résumé : Au cours des cinq prochaines années, la nouvelle génération des réseaux de téléphonie mobile (5G) devrait faire face à une croissance significative du volume de données, échangées entre des milliards d'objets et d'applications connectés. En outre, l'émergence de nouvelles technologies, telles que l'Internet des objets (IoT), la conduite autonome et la réalité augmentée, imposent des exigences plus élevées en termes de performances et de qualité de service (QoS). Répondre aux exigences tout en réduisant les dépenses d'investissement et d'exploitation (CAPEX / OPEX) sont les objectifs poursuivis par les opérateurs de téléphonie mobile. Par conséquent, les opérateurs télécoms définissent une nouvelle architecture d'accès radio, appelée Cloud Radio Access Network (C-RAN). Le C-RAN propose de découpler la station de base en deux entités: la Remote Radio Head (RRH) sur laquelle seules les opérations de signaux radio électriques sont exécutées, et la Base Band Unit (BBU) où le traitement de bande de base est exécuté de façon centralisée. Ces deux unités sont interconnectées via le Fronthaul (FH). Dans cette thèse, nous modélisons et évaluons les performances liées au dimensionnement des RRH et du FH.

Dans une première partie, nous nous intéressons à la répartition des ressources avec des mécanismes de coopération dans un contexte C-RAN et H-CRAN (Heterogeneous C-RAN). Les ressources offertes par le Cloud devront offrir les capacités de calcul nécessaires pour les traitements requis pour manipuler ces débits et prendre les décisions nécessaires de manière rapide, coopérative, coordonnée et dis-

tribuée, et résoudre en particulier les problèmes radio tels que la gestion des interférences. Dans ce but, nous proposons des outils de dimensionnement en temps réel pour la répartition des ressources avec gestion des interférences, ce qui permet d'offrir une meilleure stabilité du réseau en maximisant le débit global.

D'autre part, étant donné que les réseaux cellulaires de nouvelle génération ciblent des débits binaires plus élevés, des retards moins importants et une coordination améliorée entre les cellules, des solutions à plus haute capacité et à plus faible latence sont nécessaires sur le fronthaul. L'utilisation de réseaux Ethernet à commutation de paquets sur le FH avec une architecture C-RAN est à présent envisagée. Ils permettront de tirer profit des avantages du multiplexage statistique, de la réutilisation des infrastructures et de la réduction des coûts. Différentes options de split de fronthaul sont proposées. Cependant, afin de satisfaire les exigences en termes de latence, la plupart des études préconisent l'utilisation d'options de splits moins élevées, nécessitant une capacité de liaison importante sur le FH. Afin de réduire le coût de déploiement des liaisons et des switchs Ethernet tout en répondant aux exigences strictes en matière de latence des réseaux mobiles de nouvelle génération, nous proposons tout d'abord une nouvelle décomposition des fonctionnalités entre les RRH et le BBU pool. Nous proposons ensuite un outil de dimensionnement en temps réel permettant, en fonction du niveau du split et du type de trafic radio, de concevoir conjointement les règles de dimensionnement.

Title : Load balancing and Cross-layer resource allocation adapted to C-RAN/VRAN architecture

Keywords : Radio Access Network, Performance evaluation, Optimization, C-RAN, Resource allocation

Abstract : Over the next few years, the new fifth generation of mobile networks (5G) will face a significant growth in terms of data volumes and number of connected objects. Furthermore, with the emergence of new applications and services such as Internet of Things (IoT), Autonomous Driving (AD), Intelligent Transportation Systems (ITS), Augmented Reality and Industry 4.0 higher performances are required in terms of reliability and Quality of Service (QoS). Meeting these requirements, while reducing the Capital and Operation Expenditures (CAPEX/OPEX), are the pursued goals of the mobile operators. Consequently, Telcos defined a new radio access architecture, called Cloud/ Centralized Radio Access Network (C-RAN). This latter proposes to split the base station into two parts: the Remote Radio Head (RRH) on which only Radio Frequency (RF) operations are executed, and the BaseBand Unit (BBU) where base band processing is performed. These two units are interconnected through a high-capacity/ low-delay Fronthaul (FH) network. In this thesis, we focus on the C-RAN performances. We propose new modeling approaches allowing the analysis and the tuning of a set of parameters related to cell cooperation and Fronthaul transport.

The first part of the thesis is dedicated to resource sharing with cooperation mechanisms in a C-RAN context and H-CRAN (Heterogeneous C-RAN). The resources offered by the Cloud will have to provide the necessary computing capabilities for the processing

required to manipulate these flows and make the necessary decisions in a fast, cooperative, coordinated and distributed manner, and in particular to solve radio problems such as interference management. For this purpose, we offer real-time dimensioning tools for resource sharing with interference management, which provides better network stability by maximizing overall throughput.

In the other hand, since next generation cellular networks are targeting higher bit rates, lower delays and enhanced inter-cell coordination, higher capacity and lower latency fronthaul solutions are required, among others. The use of Ethernet packet-switched networks for the fronthaul links with a C-RAN architecture are nowadays considered. These will allow leveraging statistical multiplexing gains, infrastructure reuse and cost reduction. Different fronthaul split options are proposed. However, in order to satisfy latency requirements, most of studies advocate the use of lower fronthaul split options, requiring huge fronthaul link capacity. In order to reduce the deployment cost of links and Ethernet switches of the fronthaul while satisfying the stringent latency requirements of next generation mobile networks, we first propose a new decomposition of the functionalities between the radio heads and the BBU pool. Next, we propose a real-time dimensioning tool that allows, according to the split level and the type of radio traffic, to jointly design the sizing rules.