Performances théoriques et analyse des schémas HARQ dans un contexte d'optimisation inter-couches

Aude Le Duc

École Doctorale
d'Informatique,
Télécommunications
et Électronique de Paris

Thèse

présentée pour obtenir le grade de docteur
de Télécom ParisTech

Spécialité : **Électronique et Communications**

Aude LE DUC

Performance Closed-form Derivations and Analysis of Hybrid ARQ Retransmission Schemes in a Cross-layer Context

Soutenue le 12 février 2010 devant le jury composé de

Inbar Fijalkow	Rapporteur
Michele Zorzi	Rapporteur
Pierre Duhamel	Examinateur
Philippe Godlewski	Examinateur
Philippe Ciblat	Directeur de thèse
Christophe Le Martret	Directeur de thèse

Remerciements

Je tiens à exprimer toute ma reconnaissance aux personnes qui m'ont aidée, encouragée et soutenue tout au long de ma thèse. Mille excuses à ceux ou celles que je vais oublier, mais je vais quand même tâcher de faire de mon mieux !

Je tiens tout d'abord à remercier M. Pierre Duhamel du Centre National de la Recherche Scientifique, pour avoir accepté de présider mon jury de thèse.

Mes plus vifs remerciements s'adressent à M. Michele Zorzi, de l'université de Padoue, pour le temps précieux qu'il m'a accordé en acceptant d'être rapporteur de cette thèse. Sa relecture attentive et pointilleuse m'ont été d'une grande aide et je lui en suis très reconnaissante.

Je tiens également à remercier Mme Inbar Fijalkow, de l'École Nationale Supérieure de l'Électronique et de ses Applications, pour elle aussi avoir accepté de rapporter ma thèse. Ses commentaires m'ont beaucoup touchée et je l'en remercie vivement.

Merci également à M. Philippe Godlewski, de Télécom ParisTech, d'avoir accepté d'être examinateur à ma soutenance de thèse. Je le remercie sincèrement du temps qu'il m'a consacré avant la soutenance, pour discuter de mes travaux de recherche, ainsi que de m'avoir aidée pendant ma "recherche de thèse".

Ce travail de thèse n'aurait pu être ce qu'il est sans l'encadrement, la patience, les conseils, et la disponibilité de mes deux directeurs de thèse. Un très grand merci à M. Philippe Ciblat, mon directeur de thèse à Télécom ParisTech, et M. Christophe Le Martret, mon directeur de thèse industriel chez THALES Communications. Tous deux m'ont énormément appris, et se sont toujours montrés disponibles pour m'orienter dans mon travail.

Je souhaite également remercier MM. Cédric Demeure et Gilles Bourde de THALES Communications, de m'avoir accueillie au sein du service SPM. Un grand merci aussi à M. Dominique Merel de THALES Communications de m'avoir accueillie au sein de son laboratoire et de m'avoir accordé une liberté quant à l'organisation de ce travail de thèse entre Télécom ParisTech et THALES.

Je souhaite aussi remercier Mme Houda Labiod de Télécom ParisTech pour son aide, notamment au sujet des canaux de Gilbert-Elliot.

Je tiens à remercier M. Raphaël Massin de THALES Communications pour son aide et sa disponibilité dans mon travail de thèse.

Un grand merci également à mon collègue et surtout ami, M. Benjamin Gadat, qui n'a jamais hésité à prendre de son temps libre pour m'aider, répondre à mes questions et me faire part de ses connaissances. Son aide tout au long de ces trois ans m'a été très précieuse.

Merci également à M. Patrick Devriendt de l'ESME-Sudria, qui m'a orientée vers le doctorat et mise en relation avec Télécom ParisTech.

Merci à mes amis collègues de THALES Communications : Benjamin, Cécile, Jérémy, Alexandre, Isabelle, Catherine, Guillaume, Michaël, Nicolas, Célien, PAL, Bertrand, Martine ...Merci à eux pour les pauses café sur la plate-forme, ou encore pour les pots à l'annexe !

Merci à mes amis collègues thésards de Télécom ParisTech : Mélanie, Laura, Eric, Steevy, Maya, Ali, Fabrice, Alban, Chadi, Sami, Mireille, Zahir, Qing, Yang, Charlotte, Aziz, Haribo ... Merci à eux pour les moments de détente autour d'un gâteau du cours "cuisine" ou d'une boîte de bonbons !

Toutes ces personnes ont contribué à ce que ce travail de thèse se déroule dans de bonnes conditions. Merci à vous tous pour tous les bons moments passés ensemble !

Je ne pourrais achever cette page de remerciements sans citer ma famille : mon fiancé Sébastien, mes parents Isabelle et Jean-François, mes soeurs Ariane, Céline et Stéphanie, mes beaux-frères Stéphane, Jean-Maurice et Julien, mes futurs beaux-parents Hubert et Dominique, ainsi que mes grandes amies Irina, Aude, Nancy et Estelle. Tous, ils n'ont pas cessé de m'encourager pendant ces trois ans.

Enfin, un grand merci à tous mes proches et mes amis !

Table of Contents

Abstract

This thesis deals with the performance closed-form derivations and the analysis of Hybrid ARQ retransmission schemes in a cross-layer context. Hybrid ARQ mechanism enables us to take benefit from the properties of both ARQ and FEC according to the SNR value. As a consequence, the total amount of transmitted redundancy is automatically adapted to the channel quality, based on the acknowledgment of the previous transmission. Actually, since all the systems operate or are going to operate under IP, it is of interest to evaluate their performance at the IP layer. The metrics considered here are the Packet Error Rate, the efficiency, the delay and the jitter. These four metrics are useful since the different QoS requirements depend on a combination of them. However, in the literature, the performance of HARQ based systems are mainly analyzed at the MAC level only. Furthermore, the analyses are often carried out by means of simulations. The goal is then here to derive the four performance metrics for any HARQ mechanism, at both MAC and IP level, by taking into account some existing optimizations between the IP and MAC layers. Our theoretical derivations are proven to be useful for building algorithms dedicated to radio resource management as well as to unequal data protection.

Résumé

Cette thèse porte sur l'établissement des expressions analytiques des performances des schémas Hybrid ARQ dans un contexte d'optimisation inter-couches. Le mécanisme Hybrid ARQ permet de tirer profit des propriétés de l'ARQ et des propriétés d'un FEC, selon la valeur du SNR. Par conséquent, la quantité de redondance transmise est automatiquement adaptée à la qualité du canal, en se basant sur l'acquittement de la transmission précédente. Sachant qu'aujourd'hui, tous les systèmes opèrent (ou sont sur le point d'opérer) sous le protocole IP, il est intéressant d'étudier leurs performances au niveau de la couche IP. Les métriques considérées sont le taux d'erreur paquet, l'efficacité, le délai et la gigue. Ces métriques sont utiles puisque les besoins en QoS dépendent d'une combinaison de celles-ci. Cependant, dans la littérature, les performances des schémas HARQ sont surtout analysées au niveau MAC. De plus, elles sont souvent évaluées au moyen de simulations. L'objectif est donc ici d'établir les expressions analytiques des quatre métriques de performances pour tout type de mécanisme HARQ, aux niveaux MAC et IP, en prenant en compte des solutions existantes d'optimisation entre les couches MAC et IP. Il est ensuite prouvé que ces dérivations sont utiles pour construire des algorithmes dédiés à la gestion des ressources radio aussi bien qu'à la protection inégale des données.

Acronyms List

Acronym	English Language
AMC	Adaptive Modulation and Coding
ARQ	Automatic Repeat reQuest
ACK	Acknowledgment
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPSK	Binary Phase-Shift Keying
CRC	Cyclic Redundancy Checksum
CC	Chase-Combining
DL	Data-Link
EDL	Equal DPDU Length
FEC	Forward Error Correcting
FER	Frame Error Rate
HARQ	Hybrid ARQ
IC	Incremental redundancy with Chase combining
IP	Internet Protocol
IR	Incremental Redundancy
LDPC	Low-Density Parity-Check
MAC	Medium Access Control
NACK	Negative Acknowledgment
OSI	Open Systems Interconnection
PBS	SAR-PDU-based strategy
PDU	Protocol Data Unit
PER	Packet Error Rate
PHY	Physical
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
QPSK	Quadrature Phase-Shift Keying
RCPC	Rate Compatible Punctured Convolutional
RM	Retransmission Mechanism
RP	Receiver Processing
RTM	ReTransmission Manager
SAR	Segmentation And Reassembly
SBS	SAR-PDU-based strategy
SDU	Service Data Unit

SNR Signal to Noise Ratio
TCP Transport Control Protocol
TG Transmission Group
UDL Unequal DPDU Length
UDP User Datagram Protocol

Notations List

Notation	English Language
L_{DSDU}	Length (in bits) of the DSDU
L_F	Length (in bits) of the FRAG
L_{DPDU}	Length (in bits) of the DPDU
L_{PPDU}	Length (in bits) of the PPDU
R_{PHY}	Rate of the FEC code at the PHY layer
P	Number of retransmissions granted per DPDU
$P_{\max}$	Number of transmissions granted per FRAG
R_0	Rate of the mother code
N	Number of FRAGs belonging to a same IP packet
C	Number of transmissions granted to the N FRAGs belonging to a same IP packet
t_0	Number of DPDUs resulting of the puncturing code in the IR-HARQ scheme (noted $\{\text{DPDU}(i)\}_{i=1}^{t_0}$)
δ_i	Length (in bits) of $\{\text{DPDU}(i)\}$
Π_F	Packet Error Rate at the FRAG level
Π_S^P	Packet Error Rate at the NET level for the PBS
Π_S^S	Packet Error Rate at the NET level for the SBS
η_F	Efficiency at the FRAG level
η_S^P	Efficiency at the NET level for the PBS
η_S^S	Efficiency at the NET level for the SBS
ρ	Coefficient accounting for overhead and coding scheme
$\bar{n}_F$	Delay at the FRAG level
$\bar{n}_S^P$	Delay at the NET level for the PBS
$\bar{n}_S^S$	Delay at the NET level for the SBS
$\sigma^2_{n_F}$	Jitter at the FRAG level
$\sigma^2_{n_{SP}}$	Jitter at the FRAG level for the PBS
$\sigma^2_{n_{SS}}$	Jitter at the FRAG level for the SBS
$p_1(k)$	Probability to successfully receive one FRAG in exactly k DPDU transmissions
π_0	Probability that the first DPDU transmission associated with one FRAG fails
π_j	Probability that the $(j+1)^{\text{th}}$ DPDU transmission associated with one FRAG fails given that the j previous DPDU transmissions associated with the same FRAG failed

$\hat{n}_F$	Average number of transmitted bits when the information packet of length L_F is successfully received
$\check{n}_F$	Average number of transmitted bits when the information packet of length L_F is unsuccessfully received
$\hat{n}_S^P$	Average number of transmitted bits when the information packet of length L_{DSDU} is successfully received for the PBS
$\check{n}_S^P$	Average number of transmitted bits when the information packet of length L_{DSDU} is unsuccessfully received for the PBS
$\hat{n}_S^S$	Average number of transmitted bits when the information packet of length L_{DSDU} is successfully received for the SBS
$\check{n}_S^S$	Average number of transmitted bits when the information packet of length L_{DSDU} is unsuccessfully received for the SBS
w_k	Total number of bits transmitted at the k-th transmission (including transmissions from DPDU(1) up to DPDU(k))
$m_{\underline{i}}(j)$	Number of transmissions consumed by j MAC packets
$r_{\underline{i}}(j)$	Number of transmitted bits for j received MAC packets

Résumé long de la thèse en français

Introduction

Depuis quelques années, les moyens de communications sans fil sont devenus très populaires puisqu'ils permettent aux utilisateurs de communiquer les uns avec les autres indépendemment de l'endroit où ils se situent. De plus, la demande pour des hauts débits ne cesse d'augmenter. Afin d'atteindre la Qualité de Service (*Quality of Service* - QoS) requise, il est nécessaire de mettre en oeuvre des systèmes qui en satisfont les contraintes (débit, délai, fiabilité, mobilité), ce qui représente une lourde tâche. En général, dans les communications sans fil, le Rapport Signal à Bruit (*Signal to Noise Ratio* - SNR) du côté récepteur varie dans une large gamme de valeurs en raison de la mobilité, du fait que le canal varie dans le temps, ..., ce qui a pour conséquence de diminuer le débit atteignable. Pour pallier cette dégradation, il est fréquent d'adapter le schéma de modulation et de codage (*Modulation and Coding Scheme* - MCS) au niveau des couches accès radio (celles-ci correspondent aux deux premières couches du modèle OSI). Une autre solution est d'utiliser le mécanisme *Hybrid Automatic Repeat reQuest* (HARQ). Ce modèle combine les techniques ARQ, qui consistent en la retransmission des paquets de données sur détection d'une erreur, avec un codage canal ou *Forward Error Correcting codes* (FEC). Ce schéma puissant permet donc de tirer profit, en fonction du SNR, des propriétés de l'ARQ et des propriétés du FEC. Par conséquent, la quantité de redondance transmise est automatiquement adaptée à la qualité du canal, en se basant sur l'acquittement de la transmission précédente. Ces schémas HARQ font aujourd'hui partie des nouvelles normes sans fil telles que le Wimax mobile (IEEE 802.16e) [1] et la 3GPP-LTE [2]. Ainsi il est d'un grand intérêt d'étudier les systèmes basés sur les HARQ.

Sachant qu'aujourd'hui, tous les systèmes opèrent (ou sont sur le point d'opérer) sous le protocole IP, il est intéressant d'étudier leurs performances au niveau de la couche IP. De manière plus générale, il est pertinent d'analyser les performances des systèmes basés sur l'HARQ au niveau de la couche réseau, quel que soit le protocole réseau qui y est implémenté. Dans cet esprit, un schéma inter-couches a été récemment proposé, qui consiste à optimiser de manière conjointe les couches MAC et réseau, alors que celles-ci sont habituellement indépendamment considérées. Il a été prouvé que cette stratégie permet d'améliorer les performances globales du réseau. Mais qu'entendons-nous par "analyse des performances" ? Dans la plupart des travaux, les auteurs se concentrent seulement sur une ou un maximum de deux métriques telles que le taux d'erreur paquet, ou l'efficacité, ou le délai. Dans notre étude, nous voudrions étudier toutes les métriques nécessaires à la conception d'un système : le taux d'erreur paquet, l'efficacité, le délai,

et la gigue. Notez que ces métriques sont toutes utiles puisque: *i)* la QoS pour la voix exige un faible délai, *ii*) la QoS des flux vidéo exige une grande efficacité et une faible gigue, *iii*) la QoS pour le transfert de fichier exige un faible taux d'erreur paquet et une efficacité élevée.

Dans la littérature, les performances des schémas HARQ sont surtout analysées au niveau MAC. De plus, elles sont souvent évaluées au moyen de longues simulations et les expressions analytiques des quatre métriques liées à un mécanisme HARQ ne peuvent être trouvées que partiellement. En effet, les équations obtenues sont souvent prouvées seulement pour un type d'HARQ et au niveau d'une seule couche (habituellement la couche MAC). Enfin, en ce qui concerne le schéma d'optimisation inter-couches, il a été étudié au niveau de la couche réseau, mais seulement pour un schéma ARQ et seulement en termes de taux d'erreur paquet et de délai. En conséquence, il manque un grand nombre de formules théoriques concernant les quatre métriques pour les mécanismes d'HARQ et l'objectif de cette thèse est de combler le manque ainsi identifié. Nous proposons donc d'analyser les quatre métriques introduites précédemment pour les mécanismes de type HARQ au moyen de formules théoriques. Ces expressions analytiques nous permettront d'accélérer l'évaluation numérique de tout type de mécanisme HARQ, d'en fournir une analyse judicieuse et de gérer plus efficacement les algorithmes de ressources radio.

Positionnement de la problématique

Dans ce chapitre, nous introduisons tout d'abord le modèle du système que nous considérons tout au long de la thèse. Les principales caractéristiques du modèle sont indiquées ci-après. Au niveau de la source, un *Data-Link Service Data Unit* (DSDU) est envoyé de la couche réseau à la couche MAC, où il est fragmenté en un nombre N de paquets MAC (ou FRAGs). Chaque paquet MAC est codé (en fonction du schéma de retransmission utilisé), le paquet codé ainsi obtenu étant appelé *Data-Link Packet Data Unit* (DPDU). Le DPDU est alors envoyé sur le canal de propagation après passage par la couche physique. Au niveau de la destination, le DPDU reçu est décodé puis, si les N paquets MAC issus du même DSDU sont tous correctement reçus, le DSDU est réassemblé et envoyé aux couches supérieures.

Traditionnellement, chaque paquet MAC dispose du même crédit de transmissions $P_{\max}$. Cette stratégie conventionnelle est notée *PDU-Based Strategy (PBS)*. Cependant, sachant qu'un paquet IP est divisé en N FRAGs, si seulement un seul des N FRAGs n'est pas reçu correctement, même après avoir consommé ses $P_{\max}$ transmissions, alors non seulement le FRAG sera jeté à la réception, mais il en sera de même pour le DSDU auquel il appartient. Il est donc tout à fait absurde de continuer à transmettre des FRAGs appartenant à un DSDU corrompu. Une méthode d'optimisation inter-couches proposée par Choi *et al.* [3] pour pallier ce problème consiste à prendre en considération le fait que les N FRAGs sont issus du même DSDU. Ainsi, au lieu d'attribuer $P_{\max}$ transmissions par FRAG, on autorise un crédit de transmissions global, noté C, à l'ensemble des N paquets MAC appartenant au même DSDU. Il s'agit de la technique *SDU-Based Strategy (SBS)*, destinée à améliorer les performances globales du réseau.

Nous présentons ensuite les différents mécanismes de retransmissions HARQ existant dans l'état de l'art. Nous proposons alors une classification de ces schémas en distinguant le mécanisme de retransmission, qui opère au niveau de l'émetteur, du traitement appliqué au niveau du récepteur afin de reconstituer l'information émise. Cette classification n'est pas exhaustive mais les schémas les plus fréquemment adoptés y sont exposés. Les quatre schémas les plus importants parmi ceux présentés dans la thèse sont :

- les HARQ de type I :
  - À la source, le paquet MAC est tout d'abord codé en utilisant un FEC, puis le DPDU résultant est envoyé sur le canal de propagation.
  - À la destination, aucun traitement n'est appliqué sur le DPDU reçu avant le décodage de celui-ci.
- les HARQ à redondance incrémentale (*Incremental Redundancy* HARQ - IR-HARQ) :
  - À la source, le paquet MAC est codé puis le mode de code obtenu est divisé en un jeu de t_0 DPDUs, notés $\{\text{DPDU}(i)\}_{i=1}^{t_0}$, qui peuvent être de tailles quelconques (égales ou inégales). Les DPDUs sont envoyés successivement si nécessaire.
  - À la destination, les DPDUs issus du même mot de code sont concaténés puis décodés. Quand les t_0 DPDUs ont tous été envoyés mais décodés avec des erreurs, la mémoire est vidée et le processus recommence à partir de la transmission de DPDU(1).
- les HARQ avec *Chase combining* (CC-HARQ) :
  - À la source, le paquet MAC est codé et le DPDU ainsi obtenu est transmis autant de fois que nécessaire.
  - À la destination, la version reçue est combinée avec les versions précédentes du même DPDU selon l'algorithme de Chase.
- les HARQ à redondance incrémentale et avec *Chase combining* (IC-HARQ) :
  - À la source, le paquet MAC est codé puis le mode de code obtenu est divisé en un jeu de t_0 DPDUs, notés $\{\text{DPDU}(i)\}_{i=1}^{t_0}$, qui peuvent être de tailles quelconques (égales ou inégales). Les DPDUs sont envoyés successivement si nécessaire.
  - À la destination, les DPDUs issus du même mot de code sont concaténés puis décodés. Quand les t_0 DPDUs ont tous été envoyés mais décodés avec des erreurs, on applique l'algorithme de Chase sur les différentes versions d'un même DPDU.

Nous présentons alors une étude approfondie de l'état de l'art portant sur les performances théoriques de ces différents schémas HARQ au niveau des couches MAC et IP. Celles-ci sont principalement analysées de deux manières : par l'approche combinatoire

et par l'approche basée sur les chaînes de Markov. Cette étude met en évidence deux résultats importants. Tout d'abord, il existe une certaine confusion dans la définition des métriques telles que le délai et l'efficacité. Nous définissons donc de manière rigoureuse les quatre métriques sur lesquelles nous nous focalisons dans cette thèse.

Le **taux d'erreur paquet**, noté Π, est défini comme étant égal à un moins le rapport entre le nombre de paquets d'information reçus correctement et le nombre de paquets d'information transmis. Il est noté Π_F au niveau MAC et Π_S au niveau réseau. Comme les paquets sont respectivement des FRAGs au niveau MAC et des DSDUs au niveau réseau, nous obtenons :

$$\Pi_F := 1 - \frac{\text{Nombre moyen de FRAGs reçus correctement}}{\text{Nombre moyen de FRAGs transmis}},$$
$$\Pi_S := 1 - \frac{\text{Nombre moyen de DSDUs reçus correctement}}{\text{Nombre moyen de DSDUs transmis}},$$

où ":=" signifie *par definition*.

L'**efficacité**, notée η, est définie comme le rapport entre le nombre de bits d'information reçus correctement et le nombre de bits transmis. Elle est notée η_F pour la couche MAC et η_S pour la couche réseau. Par conséquent, nous avons :

$$\eta_{F \text{ or } S} := \frac{\text{Nombre moyen de bits d'information reçus correctement}}{\text{Nombre moyen de bits transmis}}.$$

Sous l'hypothèse de DPDUs de mêmes tailles, nous obtenons la reformulation suivante (avec une nouvelle notation) :

$$\dot{\eta}_F := \frac{\rho}{\text{Nombre moyen de DPDUs transmis pour recevoir correctement un FRAG}},$$
$$\dot{\eta}_S := \frac{N\rho}{\text{Nombre moyen de DPDUs transmis pour recevoir correctement un DSDU}},$$

où $\rho := R_{\text{PHY}} L_F / L_{\text{DPDU}}$ correspond à l'en-tête et au rendement du schéma de codage. L'efficacité est souvent associée au throughput, qui est défini comme le nombre de bits reçus correctement par unité de temps. L'efficacité n'a pas de dimension alors que le throughput est donné en bit/s. Puisque nous supposons que nous transmettons un DPDU par trame, le throughput est proportionnel à l'efficacité et ne sera pas considéré dans la suite.

Le **délai**, noté $\bar{n}$, est défini comme le nombre moyen de DPDUs transmis pour recevoir correctement un paquet d'information (nous supposons qua la transmission de l'ACK/NACK est sans erreur). Soit FRAG(i,j) le $j^{\text{ième}}$ FRAG associé au $i^{\text{ième}}$ DSDU.

- $n_F(i,j)$ est le nombre de DPDUs qui sont transmis pour recevoir correctement FRAG(i,j),
- $n_S(i)$ est le nombre de DPDUs qui sont transmis pour recevoir correctement DSDU(i).

Les termes $n_F(i,j)$ et $n_S(i)$ sont appelés délais instantanés. Le délai que nous considérons dans cette thèse peut donc être obtenu de la manière suivante :

$$\begin{aligned}\bar{n}_F &:= \text{nombre moyen de DPDUs transmis pour recevoir correctement un FRAG,}\\ &= \mathbb{E}[n_F(i,j)],\\ \bar{n}_S &:= \text{nombre moyen de DPDUs transmis pour recevoir correctement un DSDU,}\\ &= \mathbb{E}[n_S(i)].\end{aligned}$$

La **gigue**, notée σ_n, est définie comme la variation du délai et est égale à l'écart-type du délai instantané. Nous avons donc :

$$\begin{aligned}\sigma_{n_F} &:= \sqrt{\text{Moyenne}\left((\text{Délai inst. au niveau MAC} - \text{Délai moyen au niveau MAC})^2\right)},\\ &= \sqrt{\mathbb{E}\left[(n_F(i,j) - \bar{n}_F)^2\right]},\\ \sigma_{n_S} &:= \sqrt{\text{Moyenne}\left((\text{Délai inst. au niveau réseau} - \text{Délai moyen au niveau réseau})^2\right)},\\ &= \sqrt{\mathbb{E}\left[(n_S(i) - \bar{n}_S)^2\right]}.\end{aligned}$$

Ensuite, il apparaît qu'un grand nombre d'expressions théoriques n'ont pas été établies. En particulier, au niveau IP, seul l'ARQ a été considéré. L'objectif principal de cette thèse est donc de combler ce manque.

Cette partie est traitée dans le chapitre 1 de la thèse.

Établissement des expressions analytiques des performances

Dans ce chapitre, nous établissons les expressions analytiques des métriques introduites au chapitre 1 pour les schémas HARQ. Pour cela, nous avons besoin d'introduire quelques notations. Tout d'abord, nous notons $p_1(k)$, la probabilité de recevoir correctement un FRAG en exactement k transmissions. Ensuite, nous définissons π_0 comme étant la probabilité que la première transmission de DPDU associée à un FRAG soit un échec. Enfin, π_j est la probabilité que la $(j+1)^{\text{ième}}$ transmission de DPDU associée à un FRAG soit un échec, sachant que les j transmissions précédentes de DPDU associées au même FRAG ont également échouées. Nous pouvons alors écrire :

$$p_1(k) = \begin{cases} 1 - \pi_0 & \text{for } k = 1,\\ (1 - \pi_{k-1}) \prod_{j=0}^{k-2} \pi_j & \text{for } k > 1. \end{cases}$$

De la même manière, nous notons $p_N^S(k)$, la probabilité que les N paquets MAC appartenant au même paquet IP soient correctement reçus en k transmissions lorsque la stratégie utilisée est le SBS. Sachant que les N FRAGs sont indépendents, nous pouvons écrire :

$$p_N^S(k) = \sum_{\underline{i} \in \mathcal{Q}_{N,k}^S} \prod_{m=1}^{N} p_1(i_m), \tag{1}$$

où l'ensemble $\mathcal{Q}_{N,k}^S$ est défini de la manière suivante :

$$\mathcal{Q}_{N,k}^S = \{\underline{i} = [i_1, i_2, \cdots, i_N] \in \mathbb{N}_*^N \mid \sum_{m=1}^{N} i_m = k\}. \tag{2}$$

Au moyen de ces notations, nous établissons au chapitre 2 de manière analytique le taux d'erreur paquet, l'efficacité, le délai et la gigue. Les formules sont données au niveau réseau pour la méthode conventionnelle PBS et pour la méthode d'optimisation inter-couches SBS. Étant donné que la stratégie SBS a pour but d'améliorer les performances du système au niveau réseau, il est suffisant de les étudier à ce niveau là uniquement. C'est pourquoi seules les performances liées à la stratégie PBS sont aussi considérées au niveau MAC. Nous présentons ici les résultats en termes de taux d'erreur

paquet, de délai et de gigue, pour les deux stratégies au niveau de la couche réseau, les résultats au niveau MAC pour la stratégie PBS étant obtenus à partir des résultats au niveau réseau en posant $N = 1$.

Métrique	PBS	SBS
Π_S	$\Pi_S^P = 1 - (1 - \Pi_F)^N$	$\Pi_S^S = 1 - \sum_{k=N}^{C} p_N^S(k)$
$\bar{n}_S$	$\bar{n}_S^P = N \frac{\sum_{k=1}^{P\max} k p_1(k)}{\sum_{k=1}^{P\max} p_1(k)}$	$\bar{n}_S^S = \frac{\sum_{k=N}^{C} k p_N^S(k)}{1-\Pi_S^S}$
$\sigma^2_{n_S}$	$\sigma^2_{n_S^P} = N\big(\frac{\sum_{k=1}^{P\max} k^2 p_1(k)}{\sum_{k=1}^{P\max} p_1(k)} - (\frac{\sum_{k=1}^{P\max} k p_1(k)}{\sum_{k=1}^{P\max} p_1(k)})^2\big)$	$\sigma^2_{n_S^S} = \frac{\sum_{k=N}^{C} k^2 p_N^S(k)}{1-\Pi_S^S} - (\bar{n}_S^S)^2$

Table 1: Expressions analytiques du taux d'erreur paquet, du délai et de la gigue au niveau réseau en PBS et en SBS

L'établissement des expressions analytiques pour l'efficacité a été complexe et a nécessité l'introduction et le calcul d'un certain nombre de variables intermédiaires. C'est pourquoi nous traitons cette métrique à part. Étant donnés :

- L_F, la longueur (en bits) d'un FRAG,
- δ_k, la longueur en bits de DPDU(k),
- $w_k := \sum_{i=1}^{k} \delta_i$, le nombre total de bits transmis à la $k^{\text{ième}}$ transmission (ce qui inclut les transmission depuis DPDU(1) jusqu'à DPDU(k)),
- $\hat{n}_F$, le nombre moyen de bits transmis lorsque la transmission des L_F bits d'information est un succès : $\hat{n}_F = \frac{\sum_{k=1}^{P\max} w_k \cdot p_1(k)}{\sum_{k=1}^{P\max} p_1(k)}$
- et $\check{n}_F$, le nombre moyen de bits transmis lorsque la transmission des L_F bits d'information est un échec : $\check{n}_F = w_{P_{\max}}$,

nous obtenons en PBS :

$$\eta_S^P = = \frac{L_F(\sum_{k=1}^{P\max} p_1(k))^N}{(1 - \sum_{k=1}^{P\max} p_1(k))\check{n}_F + (\sum_{k=1}^{P\max} p_1(k))\hat{n}_F}. \tag{3}$$

De la même manière, en utilisant les variables précédentes et en notant :

- $r_{\underline{i}}(j) := \sum_{k=1}^{j} w_{i_k}$, le nombre de bits transmis pour j FRAGs reçus correctement,
- $m_{\underline{i}}(j) := \sum_{k=1}^{j} i_k$, l nombre de transmissions consommées par j FRAGs reçus correctement,
- $q(i)$, la probabilité suivante :

$$q(i) := \begin{cases} 1 & \text{for } i = 0, \\ \prod_{k=0}^{i-1} \pi_k & \text{for } i > 0, \end{cases}$$

- et l'ensemble $\mathcal{T}_j$, défini par $\mathcal{T}_j := \{\underline{i} = [i_1, i_2, \cdots, i_{j-1}] \in \mathbb{N}_*^{j-1} | \sum_{k=1}^{j-1} i_k < C\}$,

nous démontrons en SBS :

$$\begin{aligned}\check{n}_S^S = (\Pi_S^S)^{-1} \Bigg[& q(C-1) \cdot w_C \\ & + \sum_{j=2}^{N-1} \sum_{\underline{i} \in \mathcal{T}_j} \prod_{k=1}^{j-1} p_1(i_k) q(C - m_{\underline{i}}(j-1) - 1)(r_{\underline{i}}(j-1) + w_{C-m_{\underline{i}}(j-1)}) \\ & + \sum_{\underline{i} \in \mathcal{T}_N} \prod_{k=1}^{N-1} p_1(i_k) q(C - m_{\underline{i}}(j-1))(r_{\underline{i}}(N-1) + w_{C-m_{\underline{i}}(N-1)}) \Bigg]. \end{aligned} \tag{4}$$

Au regard des expressions des quatre métriques, nous constatons qu'elles sont fonction de la probabilité $p_1(k)$ en PBS et de la probabilité $p_N^S(k)$ en SBS (à l'exception de l'efficacité, directement exprimée en fonction de $p_1(k)$). Sachant que p_N^S est entièrement exprimable en fonction de p_1 et que p_1 ne dépend que des taux d'erreur conditionnés π_j, il est donc nécessaire de calculer les π_j correspondants pour évaluer numériquement les performances des schémas HARQ. Malheureusement, nous ne connaissons pas actuellement d'expressions analytiques pour ces taux d'erreur conditionnés, ce qui demande de les estimer de manière empirique.

Toutes ces expressions analytiques sont valables pour tout type d'HARQ, de modulation, de canal de propagation et sont données dans le cas général, c'est-à-dire pour des DPDUs de tailles différentes. Cette notion de tailles de DPDUs prend vraiment un sens lorsque le procédé de redondance incrémentale est utilisé. En effet, dans un tel cas de figure, les DPDUs issus du paquet MAC codé peuvent avoir des tailles inégales ou non. Ainsi, le cas général dans lequel nous nous sommes placés pour établir les formules théoriques concerne des tailles inégales de DPDUs. Cependant, il s'avère que les DPDUs peuvent être de mêmes tailles. Dans cette situation particulière, nous montrons que l'expression de l'efficacité, seule métrique affectée par la taille des DPDUs, se trouve fortement simplifiée :

$$\dot{\eta}_S^P = \frac{\rho(\sum_{k=1}^{P_{\max}} p_1(k))^N}{(1 - \sum_{k=1}^{P_{\max}} p_1(k))P_{\max} + \sum_{k=1}^{P_{\max}} k p_1(k)} \tag{5}$$

et

$$\dot{\eta}_S^S = \frac{N\rho \cdot \sum_{k=N}^{C} p_N^S(k)}{C(1 - \sum_{k=N}^{C} p_N^S(k)) + \sum_{k=N}^{C} k p_N^S(k)}. \tag{6}$$

Il existe d'autres cas pour lesquelles les expressions des métriques peuvent se simplifier. En particulier, lorsque les taux d'erreur conditionnés π_j sont indépendents de j, comme par exemple dans le cas des HARQ de type I où $\pi_j = \pi_0 \quad \forall j$. Dans un tel contexte, l'expression de $p_1(k)$ est réduite de manière significative : $p_1(k) = (1-\pi_0)\pi_0^{(k-1)}$. Par conséquent, nous établissons dans ce chapitre des expressions analytiques simplifiées pour les quatre métriques, aux niveaux MAC et IP, et pour les stratégies PBS et SBS.

Nous considérons deux autres cas particuliers. Le premier concerne l'efficacité au niveau MAC dans le cas des HARQ à redondance incrémentale pour des tailles de DPDUs quelconques. Ce mécanisme est caractérisé par le fait, comme dit précédemment, qu'un paquet MAC est poinçonné en un nombre t_0 de DPDUs. Si chaque DPDU dispose d'un nombre de retransmissions P (également appelé persistence), il est fréquent d'accorder au FRAG un crédit de transmissions $P_{\max}$ simplement égal à $t_0(P+1)$. Nous démontrons alors que dans un tel cas, l'efficacité au niveau MAC est indépendante de la persistence P.

Le dernier cas particulier sur lequel nous portons notre attention est la comparaison théorique des taux d'erreur paquet obtenus au niveau réseau en SBS et en PBS pour un crédit de transmissions global comparable. Sachant qu'un DSDU est fragmenté en N FRAGs et qu'un FRAG dispose de $P_{\max}$ transmissions, la comparaison est juste pour $C = NP_{\max}$. Nous prouvons dans ce chapitre que lorsque cette égalité est respectée ou encore lorsque $C > NP_{\max}$, le taux d'erreur paquet au niveau réseau est toujours plus faible avec la méthode d'optimisation inter-couches.

Enfin, nous terminons ce chapitre en indiquant les étapes à suivre rigoureusement pour évaluer numériquement, à partir des formules théoriques ainsi établies, les performances des différentes métriques. L'objectif établi au chapitre 1, qui était donc de combler le manque en termes d'expressions analytiques des performances associées aux schémas HARQ, est donc atteint.

Cette partie est traitée dans le chapitre 2 de la thèse.

Étude des performances

Le chapitre 3 a deux objectifs principaux: valider l'exactitude des expressions théoriques établies au chapitre 2 d'une part, et étudier l'influence des différents mécanismes HARQ, des stratégies (SBS, PBS), de la modulation et du canal, sur les performances du système d'autre part.

Dans un premier temps, nous validons par la simulation toutes les formules théoriques proposées précédemment et mettons en évidence le fait que ces formules sont applicables à tout type de schéma HARQ, mais aussi quelle que soit la modulation et quel que soit le canal de propagation. Dans un second temps, nous montrons que la conception d'un système est facilitée par ces nouvelles expressions. En effet, au lieu d'avoir à simuler les HARQ pour une modulation donnée, un canal donné, pour connaître les performances du modèle ainsi considéré, il suffit d'évaluer numériquement les formules dès lors que l'on a estimé les π_j correspondants. Ainsi, l'étude de l'influence des différentes composantes d'un modèle sur les performances de celui-ci devient plus simple. Nous exploitons donc cet avantage important dans ce chapitre pour comparer les différents schémas HARQ entre eux, les stratégies SBS et PBS et étudier l'influence de la modulation et du canal de propagation.

À l'issue de ces études, nous construisons des tables de références qui, pour une métrique donnée et pour un intervalle de SNR donné, permettent à un utilisateur de

sélectionner la combinaison "schéma HARQ + stratégie + modulation" qui donnera les meilleurs résultats. Par exemple, à fort SNR, nous obtenons la table de références suivante :

	Schéma	Stratégie	Modulation
Taux d'erreur paquet	IR-HARQ ou IC-HARQ	SBS	BPSK
Efficacité	IR-HARQ ou IC-HARQ	SBS	64QAM
Délai	IR-HARQ ou IC-HARQ	PBS, SBS	64QAM
Gigue	IR-HARQ ou IC-HARQ	PBS, SBS	BPSK

Table 2: Meilleure combinaison " schéma HARQ + stratégie + modulation " pour une métrique donnée et à fort SNR.

De plus, les métriques que nous considérons sont directement liées à la Qualité de Service (*Quality of Service* - QoS). En effet:

- les applications de type voix nécessitent un faible délai,
- les applications de type vidéo exigent une efficacité élevée et une faible gigue
- et les applications de type transfert de fichiers ont besoin d'un faible taux d'erreur paquet et d'une forte efficacité.

Ainsi, nous pouvons construire, à partir des tables telles que la table ci-dessus, de nouvelles tables de références qui permettent à un utilisateur de choisir la combinaison qui satisfera au mieux ses exigences en termes de QoS. À fort SNR, il vient donc :

	Schéma	Stratégie	Modulation
Voix	IR-HARQ ou IC-HARQ	PBS, SBS	64QAM
Vidéo	IR-HARQ ou IC-HARQ	PBS, SBS	QPSK ou 16QAM
Transfert de fichiers	IR-HARQ ou IC-HARQ	SBS	QPSK ou 16QAM

Table 3: Meilleure combinaison " schéma HARQ + stratégie + modulation " en fonction de l'application du système et à fort SNR.

Cette partie est traitée dans le chapitre 3 de la thèse.

Application aux systèmes de communications

Dans ce dernier chapitre, nous nous intéressons aux applications pour lesquelles notre cadre d'étude peut être utilisé : gestion des ressources radio, schéma de protection

inégale des données et réseau multi-bonds. En ce qui concerne la gestion des ressources radio, nous évaluons l'impact des paramètres principaux du système : le crédit de transmission ($P_{\max}$ pour le PBS, C pour le SBS), le nombre de FRAGs issus de la fragmentation du paquet IP N, le rendement du code R_0 et le codage canal. Pour ce dernier critère, nous considérons les codes *Rate Compatible Punctured Convolutional* (RCPC), performants pour des paquets de petite taille, et les codes *Low Density Parity Checksum* (LDPC), performants pour des paquets de grande taille. Nous fournissons alors quelques tendances concernant la politique de gestion des ressources radio que nous devons suivre pour trois types de données : la voix, la vidéo et le transfert de fichiers. En d'autres termes, de la même façon qu'au chapitre 3, nous construisons des tables de références de sorte qu'un utilisateur sache quel est le meilleur jeu de paramètres à attribuer à son système, pour répondre à ses besoins en QoS. Ainsi, à fort SNR, nous obtenons la table de références suivante :

	C	R_0	N	Codage canal
Voix	C faible	R_0 quelconque	N faible	LDPC
Vidéo	C faible	R_0 quelconque	N élevé	RCPC
Transfert de fichiers	C élevé	R_0 faible	N élevé	RCPC

Table 4: Meilleur paramétrage du système en fonction de l'application du système et à fort SNR.

Pour certaines applications, il est nécessaire de protéger les données différemment. Par exemple, dans un contexte de flux vidéo, les données associées à la première image sont plus importantes que celles associées à l'image de compensation du mouvement. De plus, dans un contexte IP, l'en-tête IP doit être lu correctement. En conséquence, certaines données doivent être plus protégées que d'autres. Pour d'autres applications, il est possible de perdre certains paquets MAC sans pour autant affecter de manière significative les performances du système. C'est le cas de la voix qui est très robuste à la perte de paquets. Il est donc judicieux d'étudier pour les deux stratégies (PBS et SBS), la probabilité de perte du $i^{\text{ième}}$ FRAG et la probabilité d'avoir i FRAGs erronés sur les N FRAGs issus du même paquet IP.

De cette étude, nous montrons que la stratégie SBS fournit en fait par construction une protection inégale des paquets au niveau FRAG due au fait que les FRAGs sont transmis de manière ordonnée. En effet, le $i^{\text{ième}}$ FRAG ne sera jamais transmis tant que le $(i-1)^{\text{ième}}$ FRAG n'aura pas été correctement reçu. Ainsi, le premier FRAG transmis est toujours plus protégé que les autres puisqu'il peut bénéficier d'un plus grand nombre de transmissions. Dans le cas PBS en revanche, puisque chaque paquet MAC se voit attribué un même crédit $P_{\max}$, les FRAGs sont protégés également. Nous proposons donc d'étendre la méthode PBS au schéma de protection inégale des paquets en attribuant aux FRAGs des crédits de transmissions différents. Dans ce cas, les FRAGs les plus protégés ne sont pas nécessairement les premiers, mais ceux ayant la valeur de $P_{\max}$ la plus grande. En notant $R_N^P(i)$ et $R_N^S(i)$, un moins la probabilité

de recevoir correctement FRAG(i) appartenant à un DSDU constitué de N FRAGs, respectivement dans les contextes PBS et SBS, nous obtenons :

$$R_N^S(i) = 1 - \sum_{k=i}^{C} p_i(k), \quad \forall 1 \leq i \leq N, \tag{7}$$

et

$$R_N^P(i) = R_N^P = 1 - \sum_{k=1}^{P_{\max}} p_1(k), \quad \forall 1 \leq i \leq N. \tag{8}$$

En notant $S_N^P(i)$ et $S_N^S(i)$, la probabilité d'avoir i FRAGs erronés parmi N FRAGs respectivement dans les contextes PBS et SBS, nous obtenons :

$$S_N^S(i) = \sum_{k=N-i}^{C} p_{N-i}^S(k)(1 - \sum_{k'=1}^{C-k} p_1(k')), \quad \forall 0 < i < N, \tag{9}$$

et

$$S_N^P(i) = \binom{N}{N-i} \left(\sum_{k=1}^{P_{\max}} p_1(k) \right)^{N-i} \left(1 - \sum_{k=1}^{P_{\max}} p_1(k) \right)^{i}, \quad \forall 0 \leq i \leq N. \tag{10}$$

Le coefficient binomial dans l'expression de $S_N^P(i)$, $\binom{N}{N-i}$, est dû au fait que les indices des i FRAGs erronés peuvent être choisis parmi les N FRAGs. Pour le SBS, quand $i = N$ ou $i = 0$, nous pouvons prouver de manière similaire :

$$S_N^S(N) = 1 - \textstyle\sum_{k=1}^{C} p_1(k) \quad \text{et} \quad S_N^S(0) = \textstyle\sum_{k=N}^{C} p_N^S(k).$$

Enfin, nous comparons les performances obtenues à partir des expressions analytiques établies au chapitre 2 avec celles obtenues à partir d'un simulateur réseau basé sur des événements discrets, utilisant le logiciel OMNeT++ et implémentant le protocole IR-HARQ. Nous montrons sur un seul lien que, en termes de taux d'erreur paquet au niveau IP, nous avons directement les mêmes résultats avec les deux approches. En revanche, le délai théorique étant exprimé en nombre de DPDUs et le délai simulé étant exprimé en millisecondes (ms), nous devons appliquer une formule de translation au délai théorique pour : d'une part, le convertir en ms, et d'autre part, prendre en compte les caractéristiques réelles de l'implémentation OMNeT++. Nous écrivons donc :

$$\bar{d}_s^x = \alpha \cdot \bar{n}_S^x \cdot (1 + \beta), \tag{11}$$

où $\bar{d}_S^x$ est le délai prédit (en ms) pour la stratégie xBS, α est la durée moyenne par transmission de DPDU et β est le nombre moyen de transmissions qui ne sont pas liées à la transmission du DPDU (mais ont aussi une durée moyenne égale à α). Au moyen de cette formule, nous obtenons les mêmes résultats avec les deux approches.

De plus, nous considérons le cas multi-bonds et proposons des équations pour prédire, à partir de nos formules, le taux d'erreur paquet et le délai de bout en bout (du noeud de départ au noeud d'arrivée) au niveau IP. En ce qui concerne le taux d'erreur paquet, nous démontrons :

$$\Pi_S^{x,\mathrm{N}_0 \to \mathrm{N}_L} = 1 - \prod_{i=0}^{L-1} (1 - \Pi_S^{x,\mathrm{N}_i \to \mathrm{N}_{i+1}}). \tag{12}$$

où $\Pi_S^{x,\mathrm{N}_i\to\mathrm{N}_j}$ est le taux d'erreur paquet au niveau IP entre les noeuds N_i et N_j (en supposant que les données traversent les $(j-i-1)$ noeuds de relais depuis le noeud N_i jusqu'au noeud N_j, soit qu'elles traversent $j-i+1$ noeuds).

De la même façon, nous obtenons pour le délai de bout en bout (en nombre de DPDUs transmis):

$$\bar{n}_S^{x,\mathrm{N}_0\to\mathrm{N}_L} = \sum_{i=0}^{L-1} \bar{n}_S^{x,\mathrm{N}_i\to\mathrm{N}_{i+1}}, \tag{13}$$

où $\bar{n}_S^{x,\mathrm{N}_i\to\mathrm{N}_j}$ est le délai moyen au niveau IP entre les noeuds N_i and N_j.

En simulant un lien à deux bonds (trois noeuds : un émetteur, un relais et un récepteur), nous montrons que la prédiction est précise.

Cette partie est traitée dans le chapitre 4 de la thèse.

Conclusion et perspectives

Conclusion

Le travail réalisé à travers cette thèse traite de l'analyse des différents schémas HARQ, qui sont l'une des techniques essentielles utilisées dans les nouvelles normes telles que le Wimax, la 3G-LTE et la 4G. Le principal objectif étant de trouver la meilleure manière de satisfaire les besoins des utilisateurs, nous nous sommes focalisés sur les performances des HARQ en termes de taux d'erreur paquet, d'efficacité, de délai et de gigue, en fonction du type de flux de données (données, voix, vidéo). En effet, ces métriques jouent un rôle important dans les besoins en QoS. En général, les performances des mécanismes de retransmission sont évaluées au niveau de la couche MAC, sachant que chaque paquet est considéré indépendemment des autres avec sa propre persistence. Cependant, le niveau IP est plus représentatif de la QoS perçue par l'utilisateur que le niveau MAC. C'est la raison pour laquelle nous avons considéré tout au long de la thèse un concept d'optimisation inter-couches, basé sur le fait que les paquets MAC sont issus d'un même paquet IP et qu'on leur alloue alors un crédit global de transmissions.

Afin de combler le manque important en termes d'expressions analytiques des performances souligné au chapitre 1, nous avons dérivé au chapitre 2 les quatre métriques au niveau MAC puis au niveau réseau pour les deux stratégies (conventionnelle et d'optimisation inter-couches). Les formules théoriques que nous avons établies sont valides pour tout type d'HARQ, de modulation et de canal de propagation. En outre, elles peuvent être appliquées quelles que soient les tailles des DPDUs, qui peuvent être différentes quand le procédé de redondance incrémentale est utilisé.

Au moyen des évaluations numériques des expressions des quatre métriques, nous avons étudié au chapitre 3 l'influence des schémas HARQ, des stratégies, de la modulation et du canal de propagation sur les performances du système. Nous avons alors construit des tables de références qui permettent à tout utilisateur de sélectionner la meilleure combinaison "schéma + stratégie + modulation, en fonction de l'application à laquelle le système est destiné et de l'intervalle de SNR dans lequel il opère.

Finalement, nous nous sommes focalisés sur certaines applications pour lesquelles notre cadre d'étude pouvait être utilisé : gestion des ressources radio, protection inégale des données, et réseau multi-bonds. Pour la gestion des ressources radio, de la même manière qu'au chapitre 3, le lecteur n'a qu'à se référer aux tables proposées dans le chapitre 4 pour trouver le jeu de paramètres (crédit de transmission, rendement du code, codage canal, nombre de paquets MAC issus du même paquet IP) le plus judicieux en fonction de ses besoins en QoS. Grâce au cadre d'étude développé au chapitre 2, nous avons pu mettre en évidence le fait que le schéma de protection inégale des données était inhérent à la stratégie SBS mais non à la stratégie PBS. Nous avons alors proposé une solution pour la méthode PBS afin que la protection inégale de données puisse être appliquée dans ce contexte. Enfin, nous avons montré que nos expressions analytiques sont valides sur un lien dans un réseau multi-bonds et nous permettent de prédire les performances globales du réseau, de la source à la destination, en termes de taux d'erreur paquet et de délai au niveau IP.

Perspectives

Les travaux réalisés dans cette thèse ont soulevé un certain nombre de problématiques qui mériteraient d'être traitées par la suite. Celles-ci sont listées ci-dessous.

En ce qui concerne les expressions analytiques établies, elles ont toutes été exprimées en fonction des taux d'erreur conditionnés π_j. Ces taux d'erreur dépendent du schéma de retransmission, de la taille des paquets MAC, de la modulation, du canal de propagation, *etc.*. Ils doivent donc être estimés au moyen de simulations, aussitôt que l'une des caractéristiques citées ci-dessus change. Bien que nous n'ayons pas à simuler le processus de segmentation et de réassemblage du paquet IP pour calculer les π_j, la programmation des schémas HARQ doit être faite. Il serait donc très utile de trouver les formules théoriques ou des bornes très proches pour les π_j.

Nous avons juste abordé la capacité des HARQ à fournir une protection inégale des paquets dans un contexte d'optimisation inter-couches, en donnant les équations basiques de performances. Des travaux supplémentaires devraient être conduits pour, dans le cas de problèmes spécifiques tels que la protection de l'en-tête IP, mettre au point des stratégies en relation avec les solutions existantes.

En ce qui concerne le contexte multi-bonds, il serait intéressant d'étendre les études menées pour le taux d'erreur paquet et le délai à l'efficacité et à la gigue. Ainsi, nous serions capables de déterminer entièrement la QoS perçue par l'utilisateur dans un réseau multi-bonds sans avoir recours à de longues simulations.

General Introduction

Problem statement

The work presented in this PhD thesis has been produced thanks to the collaboration of the department "Communications et Électronique" (COMELEC) of Télécom ParisTech and the department "Signal Processing and Multimedia" (SPM) of THALES Land and Joint Systems in the framework of a "Convention Industrielle de Formation par la REcherche" (CIFRE).

In the past few years, wireless communication devices have become increasingly popular since they allow users to communicate with each other independently from their locations. Moreover the demand for high data rate has grown. In order to achieve the required Quality of Service (QoS), the transmission needs to be reliable enough. Consequently designing systems satisfying these constraints (rate, delay, reliability, mobility) is a challenging task. In wireless communications, the Signal to Noise Ratio (SNR) at the receiver side usually varies within a large range of values due to mobility, time-varying channel, diversity scheme, ... Thus, it is typical to adapt the modulation and coding scheme at the radio access layers (corresponding to the first two layers of the OSI model) in order to get close to the achievable throughput. A complementary way to adapt the coding scheme is to use the Hybrid Automatic Retransmission reQuest (HARQ) mechanism. This model combines ARQ-like technique consisting in retransmitting the data packets upon error detection with channel coding or Forward Error Correcting (FEC) code. Rather than adapting the FEC rate thanks to a feedback carrying information on the SNR, the amount of transmitted redundancy is adapted based on the acknowledgment of the previous transmission. These HARQ schemes are now part of new wireless standards such as the mobile Wimax (IEEE 802.16e) [1] and 3GPP LTE [2]. Consequently it does add value to carry out **the HARQ based system analysis**.

Until a few years ago, communication systems were designed by considering independently the different layers following the spirit of the OSI model. Recently, a cross-layer design consisting in optimizing jointly the different layers has been considered in order to improve the whole network performance. Indeed, a global optimization of the system resource is preferable to a local optimization of each of the layers. Therefore it is worth focusing on **the optimization and the analysis of the HARQ based system at different layers**.

What do we mean by the analysis of the HARQ based system? In many works the

authors only implement practical HARQ mechanisms and analyze them through intensive simulation runs. Usually they only focus on one or a maximum of two performance metrics such as the packet error rate or the efficiency or the delay. In our study, we would like to investigate all the useful performance metrics: the packet Error Rate, the efficiency, the delay, and the jitter. Note that these metrics are necessary since *i*) the voice QoS requires small delay, *ii*) the video streaming QoS requires high efficiency and small jitter, *iii*) the file transfer QoS requires small packet error rate and high efficiency.

In addition, as simulations are very time-consuming, we suggest to analyze these four performance metrics of HARQ mechanisms through closed-form expressions. These closed-form expressions enable us to speed up the numerical evaluation of any HARQ mechanism, to provide some insights about the HARQ based systems, and to manage more efficiently the radio resource algorithms. In the literature, closed-form expressions for the four performance metrics associated with a HARQ mechanism can be found partially. The obtained equations are often proven only for a specific HARQ mechanism and at one layer (usually the MAC one). When two layers are taken jointly into account (usually MAC and IP layers), only simple ARQ mechanism has been studied. Consequently a lot of closed-form expressions regarding the four performance metrics of HARQ mechanisms is missing and the objective of this thesis is to fulfill the identified gap. Therefore it is of great interest **to derive in closed-form expressions the packet error rate, the efficiency, the delay, and the jitter for any HARQ mechanism at any layer, including cross-layer schemes**.

Outline and contributions

In this section, we give the thesis outline and we mention the most important results.

In Chapter 1, we initially introduce the layer structure of any system and the different types of existing HARQ mechanisms. We then present the state-of-the-art of the existing closed-form expressions for the four performance metrics. This chapter highlights the fact that a gap has to be fulfilled in terms of theoretical formulas, specifically at the IP layer.

In Chapter 2, we consider two strategies: the standard one in which the layers are treated separately, and the optimized one in which they are treated jointly. This chapter is firstly dedicated to the rigorous definition of the four performance metrics of interest: packet error rate, efficiency, delay, and jitter. We then develop closed-form expressions for each of them. For the standard approach we derive expressions at the MAC layer and extend them to the IP layer. Since the cross-layer approach aims at optimizing the performance at the IP layer, it is sufficient to establish their expressions at the IP layer only. The formulas are specific to the related strategy but for each of them, they are generic in the sense that they can be applicable regardless of the HARQ scheme, the size of the MAC packets, the modulation and the propagation channel.

The first part of Chapter 3 consists in validating the closed-form expressions previously established. We then focus on the comparison between the different HARQ

mechanisms and cross-layer strategies considered in the thesis by using the numerical evaluation of the proposed closed-form expressions. Moreover, we analyze the influence of the modulation and the propagation channel on the performance metrics. Thanks to those analyses, we end up by providing a selection of the best combinations "HARQ mechanism + cross-layer strategy + modulation" according to the desired QoS requirements.

Chapter 4 is split into three parts. Firstly, we study for the four performance metrics, the influence of the following parameters: the transmission credit per IP packet, the code rate, the channel coding, and the number of MAC packets per IP packet. Once again, we are able to select the best combinations of these design parameters according to the desired QoS requirements. Secondly, we address the unequal packet loss protection technique in the HARQ context. We start by showing that the cross-layer strategy naturally provides such an unequal protection. We theoretically analyze this property thanks to the framework developed in Chapter 2. Furthermore, we propose to adapt the standard strategy for unequal packet loss protection and derive its performance. Lastly, we show that we can predict the packet error rate and delay performance of a discrete event based realistic simulation implementing the HARQ scheme, using the previous derived closed-form expressions.

Publications

The work presented in this manuscript has lead to the following publications:

Journal paper

- A. Le Duc, C. J. Le Martret, and P. Ciblat, "Theoretical Performance of Cross-Layer Hybrid ARQ Schemes", *under preparation for IEEE Trans. on Communications.*

Conference papers

- A. Le Duc, P. Ciblat and C. J. Le Martret, "Delay and Jitter Closed-form Expressions for Cross-Layer Hybrid ARQ Schemes", *IEEE Vehicular Technology Conference VTC*, Anchorage (Alaska, USA), Sep. 2009.
- A. Le Duc, C. J. Le Martret and P. Ciblat, "Packet Error Rate and Efficiency Closed-form Expressions for Cross-Layer Hybrid ARQ Schemes", *IEEE Workshop on Signal Processing Advances in Wireless Communications*, Perugia (Italia), Jun. 2009.
- A. Le Duc, C. J. Le Martret and P. Ciblat, "Procédé de retransmission à redondance incrémentale adapté aux paquets IP fragmentés", *Colloque GRETSI*, Troyes (France), Sep. 2008.

International patent

- A. Le Duc and C. J. Le Martret, "Incremental Redundancy Retransmission Method for Fragmented Packets", *International Patent PCT/EP2007/064392*, 2009.

Chapter 1

Problem Statement

1.1 Introduction

Wireless communication with high data rate has recently received a lot of attention since new applications, such as data file transfer or video streaming, have become increasingly popular. These types of applications usually require powerful schemes to sustain or enhance link reliability. One way to obtain such protection is to resort to recent channel coding or Forward Error Correcting (FEC) code such as turbo-code or Low Density Parity Check codes (LDPC) which enable to get close to the Shannon capacity at the expense of high computational load. However, it is well-known that using FEC solutions has the two following drawbacks: first, for a given code rate, when the Signal to Noise Ratio (SNR) is high enough (such as the packet error rate is far below the target one), the real throughput is lower than the achievable one since FEC redundancy is useless for this operating point. Second, when Adaptive Modulation and Coding (AMC) is used to overcome the previous drawback, the feedback information needed to select the right Modulation and Coding Scheme (MCS) may be inaccurate or obsolete in a quick time-varying channel, and thus not adapted to the current channel condition. The Hybrid Automatic Retransmission reQuest (HARQ) which combines FEC and retransmission mechanisms based upon acknowledgments is a solution that allows to overcome the previous exposed problems. Regarding the first one, it systematically adapts the code rate to the channel state and thus always offers the best throughput. As for the second drawback, if the channel state information overestimates the link capability, the HARQ will automatically adapt to the new condition, avoiding the packet drop at the receiver side.

In order to better understand such error correcting mechanisms or to build a relevant radio resource management, it is always worth deriving performance of such mechanisms in closed-form. Since the global performance of a communication system is not only connected to the physical or Medium Access Control (MAC) layer but also to the upper layer such as Network (or IP) layer, it is of great interest to assess the theoretical performance of such mechanisms at different layers. Therefore the main contribution of this PhD thesis is the development of new theoretical expressions for packet error rate, efficiency, delay, and jitter, associated with HARQ mechanisms at different layers.

This chapter aims at the following:

- introducing the considered system model (layer model, several HARQ mechanisms, *etc.*),

- analyzing the state-of-the-art about theoretical performance derivations,
- describing and justifying the purpose of this PhD thesis.

Finally this chapter is organized as follows: in Section 1.2, we introduce the layer model that we consider throughout this thesis. In Section 1.3, we present the main existing HARQ retransmission mechanisms. Section 1.4 is dedicated to the state-of-the-art analysis of HARQ theoretical performance at different layers, which shows a few gaps in the literature. The purpose of the thesis, drawn in Section 1.5, is to address them.

Note that only a few papers in the literature deal with theoretical performance analysis. In addition, to the best of our knowledge, a thorough and overall state-of-the-art is not available in the papers we have found. Consequently, even if our state-of-the-art is not exhaustive, we hope that it offers a comprehensive overview that can be useful to the reader independently of the scientific contributions of this thesis.

1.2 Layer model description

In this work, we consider the first three layers of the seven layers Open Systems Interconnection (OSI) model: the Physical layer (PHY), the Data-Link layer (DL) or also called MAC layer, and the Network layer (NET). The PHY layer enables the bit transmission through a physical propagation medium. The MAC layer is the protocol layer which transfers data between adjacent wireless network nodes and carries out the radio resource management in a multi-user context. The NET layer takes care of establishing a route from the transmitter node to the receiver node (sink) via a multi-hop communication (if necessary) by using other existing nodes as relays.

Before going further, we introduce some notations [4] that will be used along the manuscript. Let us consider the layer X. The input packet (coming from the adjacent upper layer) is called a XSDU, with SDU referring to as Service Data Unit. The output packet (going to the adjacent lower layer) is called a XPDU, with PDU referring to as Protocol Data Unit. The XSDU is usually different from the XPDU since the XSDU may be fragmented into several XPDUs in order to satisfy the packet size constraint of the adjacent lower layer. A XPDU also becomes a YSDU where the layer Y is the adjacent lower layer associated with layer X. Definitions of SDU and PDU are illustrated on Fig. 1.1.

Let us now detail more precisely our layer model from the NET to the PHY and for both transmitter and receiver sides. The layer model we propose is meant to be general enough to be standard agnostic and thus resorts to generic naming. We use the conventional naming of SDU and PDU as previously explained. For the PHY layer, we use the acronyms PSDU and PPDU. For the MAC/DL layer, we use the acronyms DSDU and DPDU. From now on the notation DxDU will stand for both DSDU and DPDU. Note that when IP is used, the DSDU is equivalent to the IP datagram.

We consider that the MAC layer gets a DSDU of length L_{DSDU} from the NET layer. This DSDU is fragmented by the Segmentation And Reassembly (SAR) block into N fragments (shorten to FRAG in the sequel) of length $L_F = L_{\text{DSDU}}/N$ (we do not consider here the overhead added by the SAR). These packets are fed into the ReTransmission Manager (RTM) block which implements the retransmission scheme (coding,

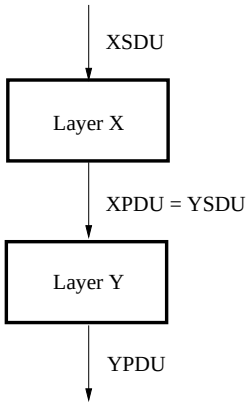

Figure 1.1: Illustration of the SDU and PDU notations.

HARQ mechanism, *etc.*) and generates DPDUs that are transmitted to the PHY layer. In general the DPDUs may have different lengths, for instance when an Incremental Redundancy (IR) HARQ mechanism is considered. When DPDUs are assumed to have the same length, the length is denoted by $L_{\mathrm{DPDU}} \geq L_F$. Obviously, the way the DPDUs are built depends upon the retransmission scheme and will be detailed later on. The DPDU is transformed into a Physical PDU (PPDU) of length $L_{\mathrm{PPDU}} = L_{\mathrm{DPDU}}/R_{\mathrm{PHY}}$ that may include a FEC code of rate R_{PHY} and sent through the propagation channel into a frame structure. After propagating through the channel, the packet is demodulated (and decoded if necessary) and the DPDU is sent to the MAC layer into the RTM. The RTM processes the DPDU and decides whether the transmission is successful or not. It then sends an ACKnowledgment (ACK) or a Negative ACKnowledgment (NACK) back to the transmitter accordingly. The RTM delivers fragments that are sent to the SAR that reconstitutes the DSDUs. All these operations are summarized on Fig. 1.2.

1.3 HARQ mechanisms description

The ARQ scheme allows to establish reliable wireless links by retransmitting corrupted packets upon error detection. Various versions have been proposed, such as Stop and Wait, Go-Back-N, Selective Repeat, ... (see [5] and references therein for a global survey). The HARQ can be seen as an evolution of the ARQ that associates retransmission mechanisms with channel coding.

In the literature, the HARQ are classified under different *types*, which, although not always very clear, can be summarized along the line of [5] as follows. The type I HARQ uses a fix rate FEC along with the ARQ scheme. The type II HARQ is based on the concept of IR that consists in sending redundancy piecewise upon error detection, instead of the whole redundancy the first time as in type I HARQ. This scheme thus automatically adapts the equivalent code rate according to the channel condition. Various channel coding schemes allow to implement the type II HARQ [5–9]. The type III HARQ was proposed by [10] using the concept of *self-decodable* codes. The HARQ schemes are now used in modern standards such as IEEE802.16 [11] and 3GPP LTE [2] in a parallel Stop and Wait mode. They provide a powerful link enhancement

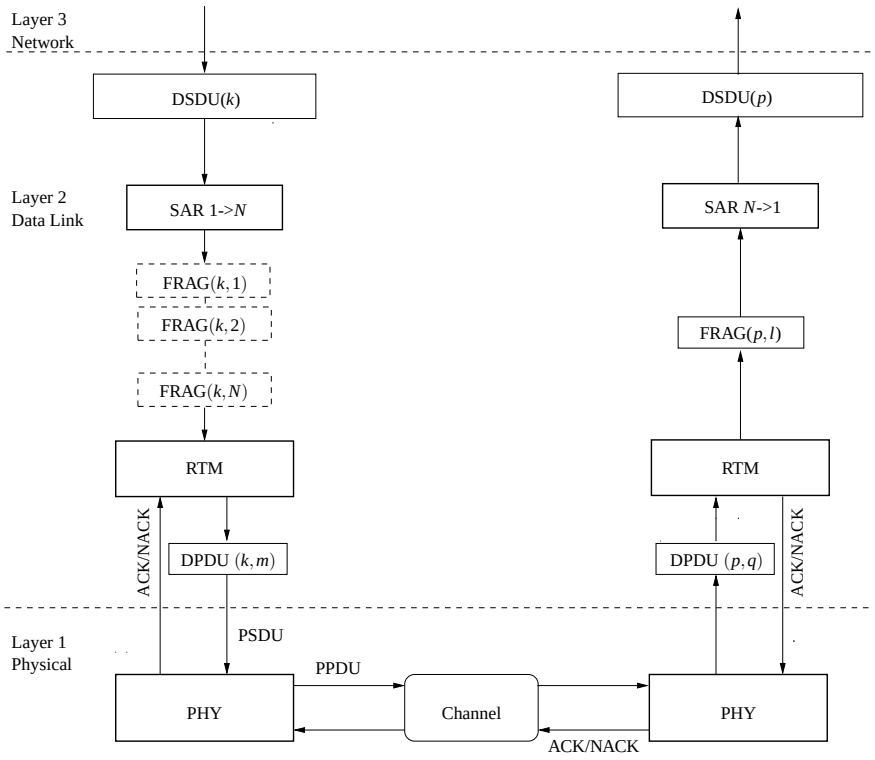

Figure 1.2: Illustration of NET, MAC, and PHY layers for a one-to-one transmission.

associated with AMC schemes.

It is clear from the previous classification that the HARQ *type* characterizes the way information and redundancy are managed at the transmitter side. The way packets are processed at the receiver side is not specified. For instance, for the type I HARQ, authors in [5] say literally that the packet is *rejected* upon error detection. However, in the so-called *Chase Combining HARQ* proposed in new standards and based on the same packet transmission scheme, it is assumed that the packets are accumulated and combined using a maximal ratio combining approach at the receiver side. So, does this new scheme belong to type I HARQ? If one decides to implement sub-optimal combination such as equal-gain combining or selective combining, how do we call the resulting scheme? We thus often face lack of precision in the way the schemes are implemented, and schemes belonging to the same type may have different performance, which can be found rather confusing. In the standards actually the protocol is only given at the transmitter side, the packet processing at the receiver being let to the manufacturer, leaving him the liberty to find the best implementation taking into account the trade-off between performance and constraints (memory, processing capability, ...). To be more specific, when we refer to *Incremental Redundancy HARQ* proposed in the standards, we have information about the different channel coding rates, the maximum number of retransmissions but nothing about the way packets are processed at the receiver side. We have found at least 3 ways of processing the packets at the receiver side. As another example, the conventional ARQ is always considered with a memoryless processing at the receiver based upon hard decision bits. But one can imagine combining the ARQ packets with soft bits and wonder what would be the gain doing so (how to enhance the

performance without changing the protocol). Our approach allows us to consider such schemes and is meant to be very general to cover the maximum of possible cases.

Thus, for the sake of clarity and precision in the definition of the retransmission schemes, we suggest to describe the HARQ mechanisms by distinguishing the operations conducted at the transmitter side (referred later on to as the Retransmission Mechanism (RM)) from the operations conducted at the receiver side (referred later on to as the Receiver Processing (RP)). A specific HARQ mechanism corresponds to a combination of one RM and one RP. Actually the RM describes the way the FRAGs are transformed into DPDUs and how the transmitter reacts to a NACK reception (to an ACK reception, all the RMs start again the process with the next information packet). In contrast, the RP describes the way the DPDUs are processed at the receiver side in order to decide if the "estimated" FRAG is corrupted or not.

In this work we consider truncated HARQ schemes with a finite number of transmissions per DPDU as introduced in [12]. It also allows to take into account conventional HARQ schemes as well by letting the transmission credit to go to infinity. We call P the *persistence*, that represents the maximum number of retransmissions granted per DPDU and plays an important role in the HARQ performance. This persistence induces another persistence at the FRAG level that we call *transmission credit per FRAG* and noted $P_{\max}$. In the simple case where one FRAG is represented by one DPDU, we have $P_{\max} = P + 1$.

In the rest of the manuscript, a packet is said to be "received" when received with no error.

This section is organized as follows: in Section 1.3.1, we describe the most well-spread Retransmission Mechanisms. In Section 1.3.2, we describe the most well-spread Receiver Processing. Note that this presentation of RMs and RPs is not exhaustive, but the most important schemes have normally been considered.

1.3.1 Retransmission Mechanisms

As a reminder, the RM explains the transmitter behavior when continuously receiving NACKs. For all the RMs, an header is systematically added to the incoming FRAG, followed by a Cyclic Redundancy Checksum (CRC) encoding in order to check the packet integrity at the receiver side.

1.3.1.1 Standard Retransmission Mechanisms

RM1:
The DPDU is simply constituted by the FRAG with the header and the CRC. The transmitter retransmits the same DPDU until the retransmission credit (*i.e.*, the persistence) is consumed. This RM was introduced in the early 50's [13] to ensure more reliable telegraph communications over wireless channels. Since then, various versions of this RM have been proposed, specifically to handle the issue when an incoming DPDU has to be sent while the ACK or the NACK of the previous DPDU has not been received. This leads to the Stop and Wait (SW), Go-Back-N (GB-N), and Selective Repeat (SR). For more details, the reader can refer to [5] and references therein. The main drawback of this RM occurs at low and medium SNR as no coding scheme is associated with. This type of RM is strongly linked to ARQ mechanism.

RM2:
The DPDU is constituted by the FRAG with the header and the CRC which is then encoded by a FEC of rate R_0. The transmitter retransmits the same DPDU until the retransmission credit (*i.e.*, the persistence) is consumed. Note that the FRAG and DPDU sizes are then linked to R_0 and N and should be chosen carefully. Different FEC codes can be used (such as block coding, convolutional codes, *etc.*). This additional FEC enables us to overcome the main drawback of RM1 since, at low or medium SNR, RM2 will offer better Packet Error Rate (PER) than RM1. Now, the main drawback of this RM is to systematically send redundancy bits even when not necessary (*i.e.*, at high SNR) and consequently to reduce the data rate. This scheme is illustrated on Fig. 1.3. This type of RM is strongly linked to the type I HARQ mechanism.

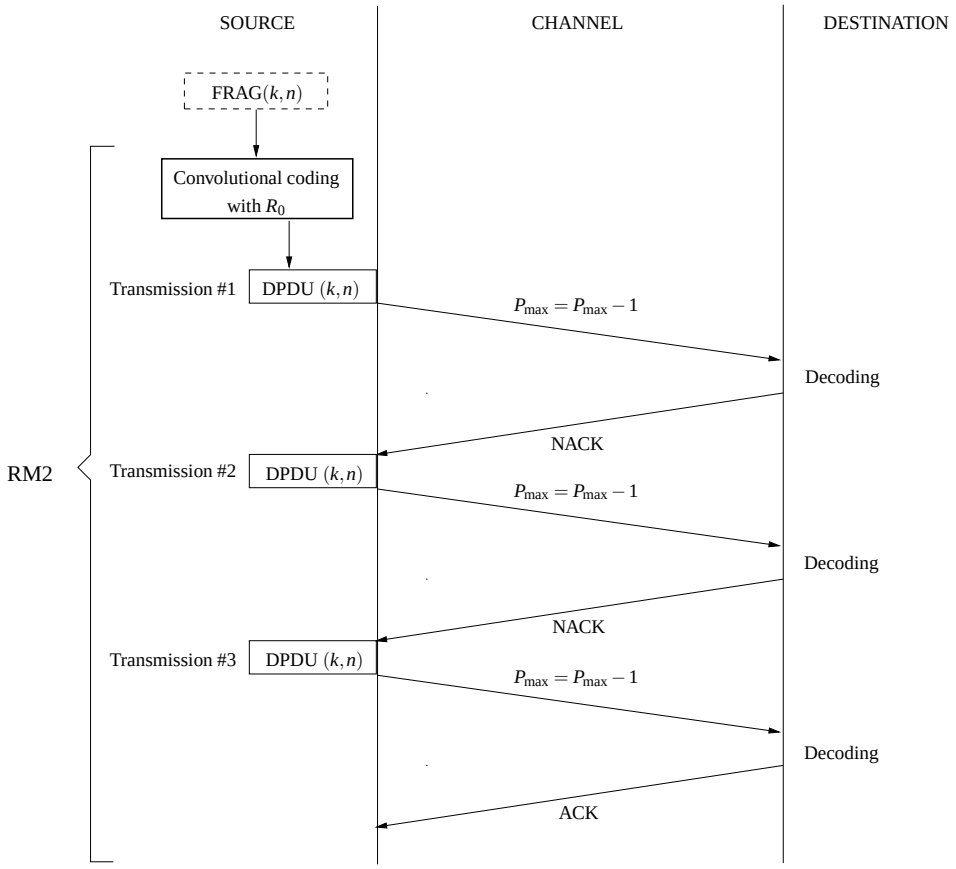

Figure 1.3: RM2 scheme (where $P_{\max} \geq 3$).

Because of the respective drawbacks of RM1 and RM2, more sophisticated mechanisms have been proposed in the literature. These mechanisms, introduced below, correspond to several kinds of the type II HARQ approach. In the sequel, we only describe the most commonly used mechanisms in existing standards.

RM3:
The FRAG in which the header and the CRC have been added, is encoded by a FEC of rate $R_0 = s_0/r_0$ (known as the *mother code*). The encoded FRAG is then split into t_0 subblocks, usually thanks to a puncturing technique of the mother code. Each subblock is then transformed into a DPDU. The DPDUs may have different sizes depending on the considered puncturing technique. The DPDUs associated with the same FRAG are thus numbered as $\{\text{DPDU}(i)\}_{i=1}^{t_0}$, and DPDU($i$) has the length δ_i. DPDU(1) corresponds either to the information bits (fragment + overhead) if the code is systematic, or to the coded bits of a rate 1 punctured version of the mother code. DPDU(2) is the block of redundancy bits which concatenated with DPDU(1) gives the redundancy bits of the punctured code of rate $\delta_1/(\delta_1 + \delta_2)$. This can be generalized as follows: for $i > 1$, DPDU(i) is the block that, when concatenated with DPDU(j) for $1 \leq j \leq i-1$, gives the redundancy bits of the punctured code of rate $\delta_1/\sum_{j=1}^{i} \delta_j$. The transmitter starts to transmit DPDU(1), then DPDU(2) (if a NACK is received), then DPDU(3) (if a second NACK is received), and so on up to DPDU(t_0). The transmission credit per FRAG $P_{\max}$ is linked to the persistence P granted per DPDU, so that there are at most $P_{\max} = t_0(P+1)$ successive DPDU transmissions. The transmitter then retransmits the same DPDUs sequence until the transmission credit per FRAG $P_{\max}$ is reached.

This kind of RM scheme corresponds to the HARQ with Incremental Redundancy (IR-HARQ). This scheme automatically takes benefits of both ARQ and FEC schemes according to the channel condition. The first idea of IR-HARQ was given in [6] with punctured Reed-Solomon codes. Some other punctured codes have then been proposed, such as the punctured convolutional codes, also called Rate Compatible Punctured Convolutional (RCPC) codes [7], the punctured turbo codes [8], and the punctured LDPC [14].

It is usual to introduce the RM3 scheme only with a mother code of rate $1/r_0$, *i.e.* $s_0 = 1$. In such a case, we only consider r_0 subblocks, thus r_0 DPDUs with the same length L_{DPDU} (corresponding to the fragment size plus overhead). Consequently, the transmission credit per FRAG is given by $P_{\max} = r_0(P+1)$. This simplified version of the RM3 scheme is illustrated on Fig. 1.4. From now on, when the transmission credit per FRAG $P_{\max}$ is equal to $r_0(P+1)$, that implicitly means that the DPDUs are of equal lengths whereas when it is given by $t_0(P+1)$, that implies that the redundancy increments are of different lengths.

1.3.1.2 Other Retransmission Mechanisms

Although RM1, RM2, and RM3 are the best known retransmission schemes, other less known but nevertheless interesting types of RM can be found.

RM4:
The FRAG with an header and a CRC is encoded using a high rate LDPC code (called mother LDPC code) to build the DPDU. When a NACK is received, the transmitter punctures some of the information bits contained in the FRAG, according to a given puncturing scheme. This puncturing scheme is known at the receiver as well. The punctured FRAG is then encoded using the same LPDC code to build the new DPDU

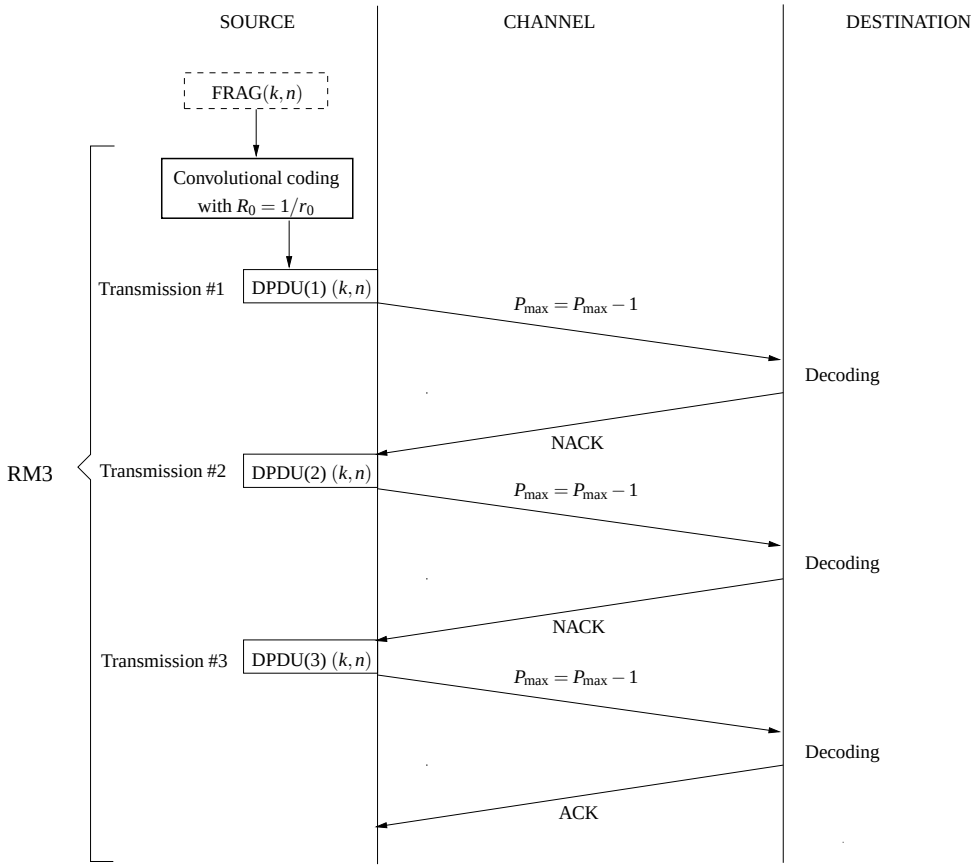

Figure 1.4: RM3 scheme (where $P_{\max} \geq 3$ and $R_0 = 1/r_0$ with $r_0 \geq 3$).

that will be sent. Note that this DPDU is associated with a lower rate LDPC than the mother code since the same amount of parity bits protects a reduced amount of information bits. Then, only the parity bits of the DPDU are transmitted, not the information ones. This process can be repeated $P_{\max} - 1$ times using different puncturing schemes. This slight modification of RM3 has been proposed in [15] and allows to improve the throughput.

RM5:
The FRAG in which the header and the CRC have been adjuncted, is encoded by a FEC of rate $R_0 = 1/r_0$ (known as the *mother code*). The encoded FRAG is then split into r_0 DPDUs of the same length, denoted by $\{\text{DPDU}(i)\}_{i=1}^{r_0}$. The transmitter firstly sends DPDU(1). When a NACK is received, instead of sending DPDU(2), the transmitter sends the concatenation of DPDU(1) and DPDU(2), denoted by DPDU(1,2). More generally, for $i > 1$, when a NACK is received for DPDU$(1, 2, \cdots, i-1)$, the next transmitted DPDU is the concatenation of DPDU(i) with DPDU$(1, 2, \cdots, i-1)$, denoted by DPDU$(1, 2, \cdots, i-1, i)$. As usual the maximum number of transmissions is given per FRAG and is equal to $P_{\max}$. Such a technique was proposed in [16] and allows to improve the performance in terms of PER at the expense of the efficiency.

RM6:
To implement this scheme, we need to have the ability to send two packets at the same time without interference. This constraint can be satisfied, for instance, in an Orthogonal Frequency Division Multiplexing (OFDM) system. The FRAG with the header and the CRC is encoded by a FEC of rate $R_0 = 1/r_0$ (known as *mother code*). The encoded FRAG is then split into r_0 DPDUs of the same length, denoted by $\{\text{DPDU}(i)\}_{i=1}^{r_0}$. The first two DPDUs are then transmitted on parallel subcarriers, then consuming only one transmission. If a NACK is received (because the decoding of the information based on the first two DPDUs has failed), an additional DPDU (actually DPDU(3)) is sent. The process continues until the correct reception of the FRAG occurs or the transmission credit per FRAG is achieved. This mechanism has been proposed in [17] for improving the PER. This scheme is illustrated on Fig. 1.5.

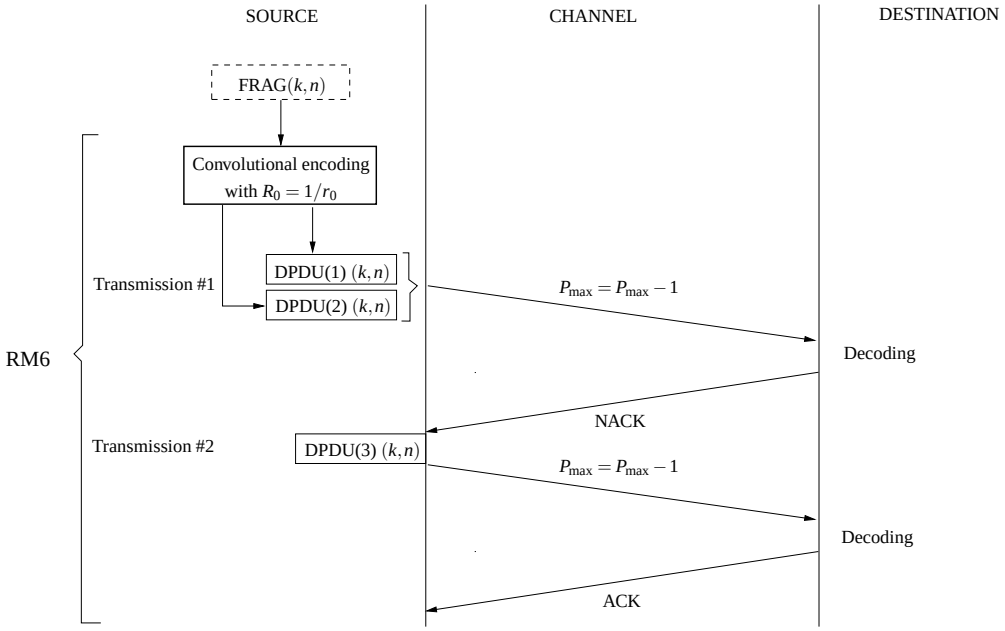

Figure 1.5: RM6 scheme (where $P_{\max} \geq 2$ and $R_0 = 1/r_0$ with $r_0 \geq 3$).

RM7:
The main idea of this mechanism consists in considering a set of K FRAGs, instead of considering each FRAG separately. This set is encoded by a FEC code of rate K/N with $N > K$. The FEC output is then split into N DPDUs of identical lengths. This set of DPDUs is referred to as a Transmission Group (TG). The N DPDUs are sent on the channel. If a NACK is received, the transmitter generates an extra redundant DPDU obtained by the coding of the K original FRAGs with a code different from the initial FEC. The procedure continues until the K FRAGs are recovered with no error. Let us assume that N' is the number of extra redundant DPDUs needed to successfully receive the K original DPDUs of the current TG. If the next TG is not successfully received after its first transmission, then N' extra redundant DPDUs are directly generated from the K original FRAGs. If a NACK is still received after the transmission of these N' extra redundant DPDUs, other extra redundant DPDUs are sent one by one. On the

contrary, if the TG is decoded without error after the transmission of these N' extra redundant DPDUs, N' is decremented by a number arbitrarily chosen. Actually N' is dynamically adapted to the channel conditions: when the channel is good, N' will decrease whereas when the channel is bad, N' will increase. This mechanism has been proposed in [18] and improves the delay and the throughput.

1.3.2 Receiver Processing

As a reminder, the Receiver Processing (RP) explains how the receiver treats the received DPDUs to recover the FRAG successfully.

1.3.2.1 Standard Receiver Processing

We firstly start with the presentation of the RPs used in most works and standards.

RP1:
The receiver decodes the DPDUs one by one and so does not use any system memory. This mechanism can be used with RM1 and RM2 but does not have sense for the other described RMs. For instance, the receiver only checks the CRC in the RM1 case and the receiver decodes the FEC and then checks the CRC in the RM2 case.

RP2:
The receiver decodes the incoming DPDUs jointly as follows: the receiver checks the CRC of DPDU(1) which is the first DPDU associated with one FRAG. If the FRAG is not successfully decoded, then the receiver decodes the concatenation of DPDU(1) and DPDU(2) and checks the CRC, and so on up to the reception of DPDU(t_0) which is decoded with mother code of rate R_0 followed by the CRC checking. Then, if the FRAG is received with errors and the persistence is not reached, the received packet memory is flushed (put to zero) and the process starts again. This scheme is applicable to RM3 and RM6. The RM3-RP2 combination is illustrated on Fig. 1.6.

RP3:
The receiver combines the received DPDUs using the Chase Combining (CC) algorithm [19], that is to say, by applying a maximal ratio combining to the different received DPDUs associated with the same FRAG. The RP3 scheme is applicable to any above-mentioned RM scheme as soon as $P_{\max} > 1$. For instance, in the case of the RM1 and RM2 schemes, the decision is made on the sum of the DPDUs associated with the same FRAG. In the case of the RM3 scheme, the RP3 scheme is slightly more complicated. For instance, after processing the t_0 first DPDUs as done in the RP2 scheme, the memory is not flushed. Consequently the second transmission of DPDU(1) (done if a NACK is still sent at the transmitter after the reception of the t_0 first DPDUs) is combined with the previous DPDU(1) associated with the same FRAG. Then the new entire codeword of rate R_0 is decoded, followed by CRC checking. If the decoded FRAG is still erroneous, a NACK is sent to the transmitter and the second transmission of DPDU(2) occurs. The received DPDU(2) is then combined with the previous DPDU(2) and the new entire codeword of rate R_0 is decoded followed by CRC checking, and so on. This scheme is illustrated on Fig. 1.7 when RM2 is used at the transmitter side.

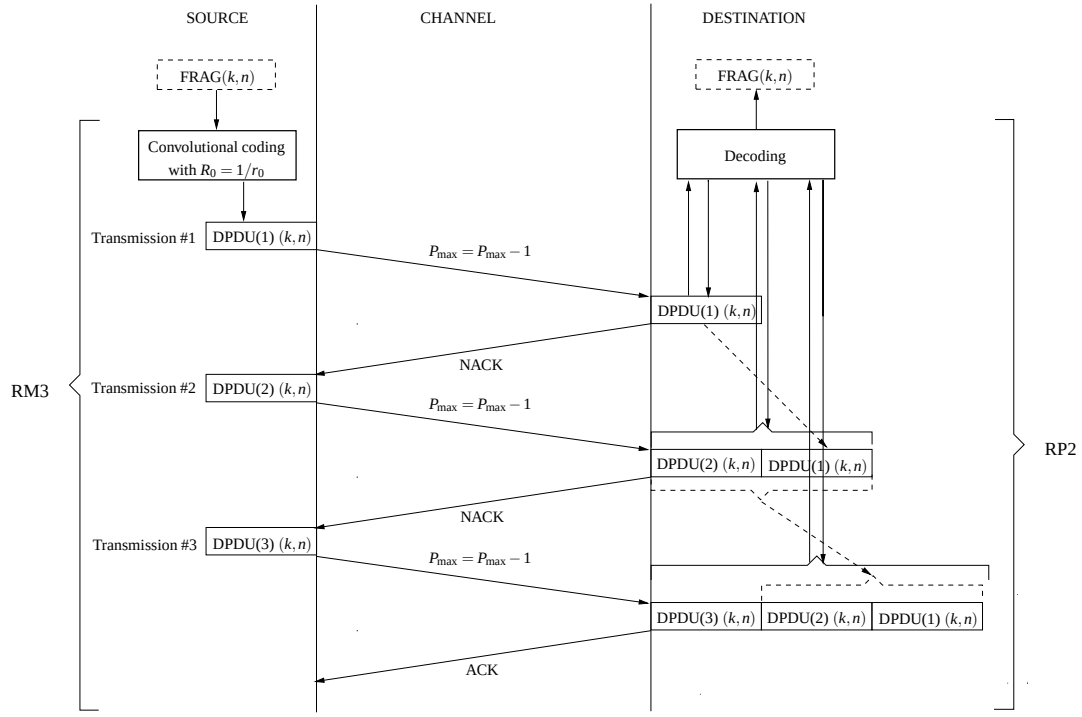

Figure 1.6: RP2 associated with RM3 (where $P_{\max} \geq 3$ and $R_0 = 1/r_0$ with $r_0 \geq 3$).

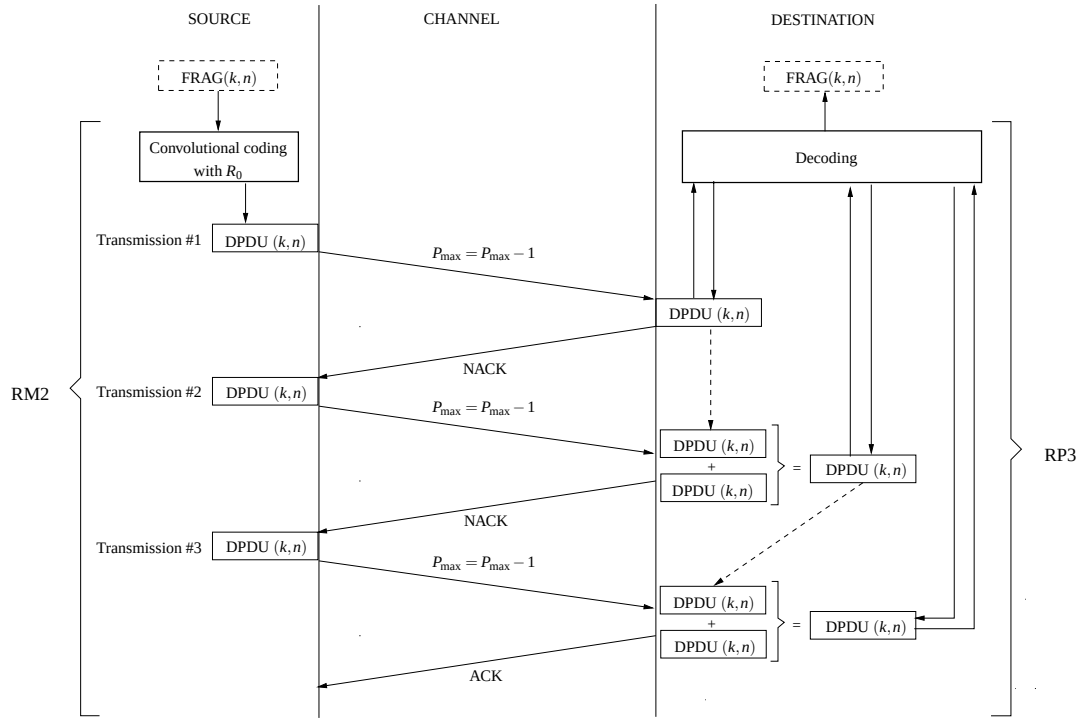

Figure 1.7: RP3 associated with RM2 (where $P_{\max} \geq 3$).

1.3.2.2 Other Receiver Processing

We here focus on other types of RPs that are less commonly used. However, the following RPs could be applied to some above-described RMs.

RP4:
This RP scheme can be only performed in association with RM4 scheme. The first received DPDU is decoded and the corresponding CRC is checked. If there is still an error, then a NACK is sent to the transmitter. As previously explained, the transmitter, when receiving the NACK, sends a new DPDU containing the parity bits coming from an encoded version of a smaller set of information bits. The receiver knows the puncturing scheme used for building this subset of information bits. Thanks to the reception of the second DPDU, the receiver tries to decode the subset of information bits, also called the not punctured bits. Those information bits decoding is obviously more likely due to the lower code rate. The information about the not punctured bits, obtained after the second decoding process, is merged with the information about the punctured bits. Then the original received parity bits are used to recover the entire fragment and the CRC of this new decoded block is checked.

RP5:
This RP scheme can be only performed in association with RM5. The received DPDU associated with the first transmitted DPDU is simply decoded. The second received DPDU is actually composed by the retransmitted bits (equivalent to the first DPDU) and additional redundancy bits. The retransmitted bits of the second DPDU are linearly combined with the bits of the first DPDU according to the Chase algorithm (as in RP3). Then the resulting codeword (sum of retransmitted bits plus additional redundancy bits) is decoded (as in RP2) [16].

RP6:
This RP scheme can be only performed in association with RM6 scheme. The first two DPDUs sent simultaneously at the transmitter are concatenated at the receiver in order to decode the FRAG. After checking the CRC, if the FRAG is not received, a NACK is sent at the transmitter and a new DPDU is then received. Once again the concatenation of both DPDUs (the already concatenated one and the new received one) is performed and a decoding is carried out, and so on [17].

RP7:
This RP scheme can be only performed in association with RM7 scheme. The TG can be successfully received if at least K DPDUs over the whole set of the N transmitted DPDUs are decoded without errors. If it is not the case, the decoding of the same set of DPDUs plus one redundant DPDU (or more) will take place, and so on until successful decoding or no more transmission credit is available [18].

RP8:
This scheme can be applied to each RM (except RM7) as soon as $P_{\max} > 1$. When a retransmission occurs (for example, the transmission of additional redundancy DPDU in the case of RM3 or the transmission of the same DPDU in the case of RM1/RM2), the receiver selectively combines some of the received DPDUs coming from the same FRAG in order to recover the information. In other terms, the receiver tries several possible combinations and chooses one combination (thanks to a certain criteria) instead

of systematically combining the received DPDU with the previous one [20,21].

1.3.3 Summary

In Tab. 1.1, we represent all the possible combinations of the different RMs and RPs previously defined. A cross ("×") indicates that the combination is feasible while a blank means that the combination does not make sense.

	RM1	RM2	RM3	RM4	RM5	RM6	RM7
RP1	×	×					
RP2			×			×	
RP3	×	×	×	×	×	×	×
RP4				×			
RP5					×		
RP6						×	
RP7							×
RP8	×	×	×	×	×	×	

Table 1.1: Table of all the RM-RP combinations.

For the remaining of the PhD thesis, although our results hold for any RM-RP combination, we focus on the following RM-RP combinations which are the most used in the literature and standards:

- RM1-RP1: ARQ
- RM2-RP1: ARQ with FEC
- RM1-RP3: CC-ARQ
- RM2-RP3: CC-HARQ
- RM3-RP2: IR-HARQ
- RM3-RP3: IC-HARQ (IC: Incremental redundancy with Chase combining).

Obviously the list of given RMs and RPs in Tab. 1.1 is not exhaustive and thus it exists a lot of other RMs and RPs. These other kinds of HARQ are very rarely studied and used. Therefore we do not consider them in this manuscript. Two other classes of HARQ mechanisms can still be mentioned. The first one works with extra information feedback, *i.e.* not only the ACK/NACK but other types of information feedback such as average SNR or detection fiability [22–29]. The second one is based on the principle of the *invertible* codes introduced in [30]. These invertible codes are able to decode the redundancy alone without necessarily combining it with the previously transmitted DPDUs. This overcomes the following situation: the first packet is received in very bad condition whereas the redundancy is received in very good conditions, *i.e.* without error. The invertible code enables us to retrieve the FRAG only with the redundant part which has been well received. This idea was generalized in [10] with the concept of *self-decodable* codes and referred to as type III HARQ scheme.

1.4 State-of-the-art for theoretical performance

This section is dedicated to the state-of-the-art of the existing closed-form performance expressions. We specifically observe that almost all the works focus on the MAC layer and not on the upper layers such as NET layer. In addition, at the MAC layer, some metrics (PER, efficiency) are rather well analyzed whereas some other important metrics (delay and jitter) are less studied.

There are two main interests for providing performance closed-form expressions: *i*) it is faster to numerically compute closed-form expressions than really implement a HARQ mechanism and *ii*) it enables us to have insights about the influence of design parameters in order to carry out a relevant resource allocation algorithm.

This section is organized as follows: in Section 1.4.1 we provide the definition of the performance metrics of interest. Section 1.4.2 is dedicated to the presentation of the existing results about analytical performance of HARQ mechanisms at the MAC layer. In Section 1.4.3, we focus on the existing results about analytical performance of HARQ mechanisms at the NET layer. Section 1.4.4 introduces existing approaches for system design based on HARQ mechanism.

1.4.1 Performance metrics definition

We define in this section the metrics that may play an important role to satisfy a required Quality of Service (QoS). Obviously the QoS is application-dependent. For instance, applications, such as file transfer or email, need small PER and high data rate (or equivalently high efficiency). Note that we will see some more links between the data feature and the QoS later, once all the metrics have been properly defined. In the rest of the manuscript, we focus on the following four metrics:

- PER,
- Efficiency,
- Delay,
- Jitter.

Some metric definitions are confusing in the literature. For instance, the delay is often mis-defined which leads to some mistakes on the delay derivations and indirectly on the efficiency derivations. As a consequence we spend time to define carefully and precisely the metrics before investigating them. Before going further, we introduce some important notations. The subscript 'F' stands for an analysis at the MAC layer (equivalently, FRAG/DPDU level) whereas the subscript 'S' stands for an analysis at the NET layer (equivalently, DSDU level).

The **PER**, denoted by Π, is defined as one minus the ratio between the number of received information packets and the number of transmitted information packets. As previously mentioned, the PER is denoted by either Π_F at the MAC layer (equiv. FRAG/DPDU level) or Π_S at the NET layer (equiv. DSDU level). As the information packets are the FRAGs at the MAC layer and the DSDUs at the NET layer respectively,

we get:

$$\Pi_F := 1 - \frac{\text{Average number of FRAGs successfully received}}{\text{Average number of transmitted FRAGs}},$$
$$\Pi_S := 1 - \frac{\text{Average number of DSDUs successfully received}}{\text{Average number of transmitted DSDUs}},$$

where ":=" means *by definition*.
The **efficiency**, denoted by η, is defined as the ratio between the number of successfully received information bits and the number of transmitted bits. The efficiency will be denoted by either η_F at the MAC layer (equiv. FRAG/DPDU level) or η_S at the NET layer (equiv. DSDU level). Consequently, we have:

$$\eta_{F \text{ or } S} := \frac{\text{Average number of successfully received information bits}}{\text{Average number of transmitted bits}}.$$

Under the assumption of identical DPDU lengths, we obtain the following reformulation (with a new notation):

$$\dot{\eta}_F := \frac{\rho}{\text{Average number of transmitted DPDUs to successfully receive one FRAG}},$$
$$\dot{\eta}_S := \frac{N\rho}{\text{Average number of transmitted DPDUs to successfully receive one DSDU}},$$

where $\rho := R_{\text{PHY}} L_F / L_{\text{DPDU}}$ accounting for the overhead and coding scheme rates.
The efficiency is often associated with the throughput that is defined as the number of successfully received bits per unit of time. The efficiency has no dimension whereas the throughput is given in bit/s. Since we assume that we transmit one DPDU per frame, the throughput is proportional to the efficiency and will not be considered in the later.

The **delay**, denoted by $\bar{n}$, is defined as the average number of transmitted DPDUs to successfully receive the information packet (we assume without loss of generality that the ACK/NACK transmissions are delay-free). Let FRAG(i,j) be the j^{th} FRAG associated with the i^{th} DSDU. Different useful types of delays can be defined in the following manner in order to analyze the retransmission schemes performance:

- $n_{F,r}(i,j)$ is the number of DPDUs that are transmitted between the received FRAG(i,j) and the next received fragment,
- $n_F(i,j)$ is the number of DPDUs that are transmitted when FRAG(i,j) is successfully received,
- $n_{S,r}(i)$ is the number of DPDUs that are transmitted between the received DSDU(i) and the next received DSDU,
- $n_S(i)$ is the number of DPDUs that are transmitted when DSDU(i) is successfully received.

In order to illustrate those different delays, Fig. 1.8 shows a time representation of some DPDU and DSDU transmissions from NET layer through MAC layer and PHY layer, and from the transmitter to the receiver in the context of ARQ scheme. OK means that the FRAG (respectively DSDU) after the receiver processing (respectively

after the SAR) is received without error (respectively without error on each FRAG constituting the considered DSDU). KO means that the FRAG (respectively DSDU) is received with error (respectively it exists at least one FRAG that has been received with error). For sake of simplicity, the acknowledgments are not represented on the figure. In this example, the DSDU is fragmented into $N = 3$ fragments, and the transmission credit per FRAG is equal to $P_{\max} = 3$.

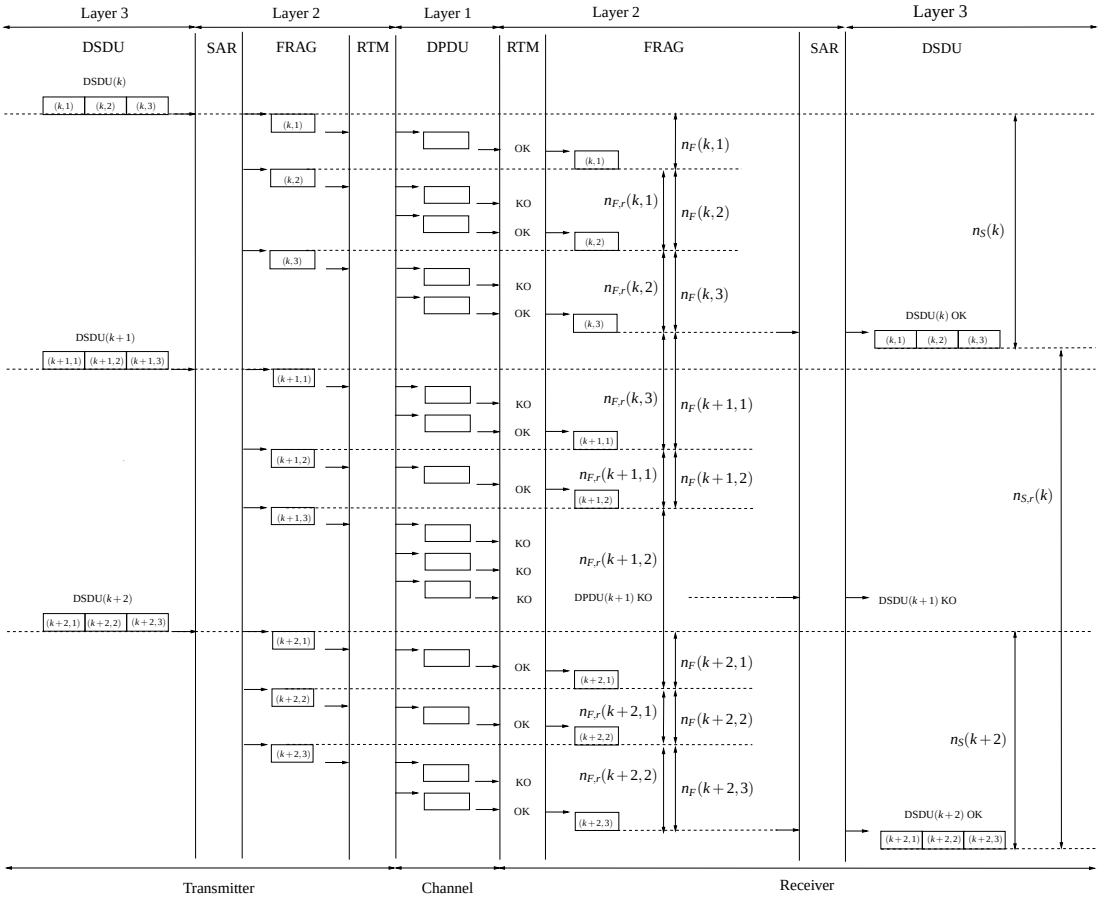

Figure 1.8: Different delay definitions (for ARQ scheme, $N = 3$ and $P_{\max} = 3$).

The terms $n_F(i,j)$ and $n_S(i)$ are called instantaneous delays. The delay can be thus obtained as the expectation of the instantaneous delay and so can be written as follows:

$$\begin{aligned}\bar{n}_F &:= \text{Average number of DPDUs sent to successfully receive one FRAG,}\\ &= \mathbb{E}[n_F(i,j)],\\ \bar{n}_S &:= \text{Average number of DPDUs sent to successfully receive one DSDU,}\\ &= \mathbb{E}[n_S(i)].\end{aligned}$$

We can also define the average delay between two successive received packets, denoted by $\bar{n}_r$, which can be written as follows:

$$\begin{aligned}\bar{n}_{F,r} &:= \text{Average number of transmitted DPDUs between two successive received FRAGs,}\\ &= \mathbb{E}[n_{F,r}(i,j)],\\ \bar{n}_{S,r} &:= \text{Average number of transmitted DPDUs between two successive received DSDUs,}\\ &= \mathbb{E}[n_{S,r}(i)].\end{aligned}$$

One can point out that $\bar{n}_r$ is conversely proportional to the efficiency when the DPDUs have the same length, whereas $\bar{n}$ (which is the true delay) usually gives the information about delay packet delivery (from the source to the destination). For instance $\bar{n}_S$ is the so-called *One-Way Delay* in the IP context [31]. We will see in the rest of this state-of-the-art that some authors mis-use $\bar{n}_r$ instead of $\bar{n}$ and vice-versa.

The **jitter**, denoted by σ_n, is defined as the delay variation and is equal to the standard deviation of the instantaneous delay. Therefore, we have:

$$\begin{aligned}
\sigma_{n_F} &:= \sqrt{\text{Average}\left((\text{Inst. delay at MAC layer} - \text{Average delay at MAC layer})^2\right)}, \\
&= \sqrt{\mathbb{E}\left[(n_F(i,j) - \bar{n}_F)^2\right]}, \\
\sigma_{n_S} &:= \sqrt{\text{Average}\left((\text{Inst. delay at NET layer} - \text{Average delay at NET layer})^2\right)}, \\
&= \sqrt{\mathbb{E}\left[(n_S(i) - \bar{n}_S)^2\right]}.
\end{aligned}$$

These four metrics are very useful to analyze the performance of a system. Nevertheless the importance of each metric depends on the required QoS. For streaming traffic (such as video), having a small jitter is of great interest. For voice, the delay has clearly to be minimized whereas, for the file transfer, it is more interesting to minimize the PER and to maximize the efficiency.

Before going further, we introduce some notations that will be used throughout the manuscript:

- Let $p_1(k)$ be the probability to successfully receive one FRAG in exactly k DPDU transmissions.
- Let π_0 be the probability that the first DPDU transmission associated with one FRAG fails.
- Let π_j be the probability that the $(j+1)^{\text{th}}$ DPDU transmission associated with one FRAG fails, given that the j previous DPDU transmissions associated with the same FRAG failed.

It is easy to find that $p_1(k)$ is also the probability that the k^{th} DPDU transmission succeeds whereas the $k-1$ previous DPDU transmissions have failed, *i.e.* $p_1(k) :=$ Pr{FRAG received in k DPDUs}, and so leads to the following expression:

$$p_1(k) = \begin{cases} 1 - \pi_0 & \text{for } k = 1, \\ (1 - \pi_{k-1}) \prod_{j=0}^{k-2} \pi_j & \text{for } k > 1. \end{cases}$$

1.4.2 HARQ performance analysis at the MAC level

In this section, we focus on the derivations of the four above-defined metrics for HARQ mechanisms at the MAC layer. To theoretically derive these metrics, there are two main approaches in the literature: the combinational approach and the Markovian approach.

This section is organized as follows: in Section 1.4.2.1, we introduce the existing results obtained by the combinational approach. In Section 1.4.2.2, we describe the existing results obtained by the Markov chain based approach. Section 1.4.2.3 is dedicated to the introduction of some works exhibiting only theoretical approximations, bounds or numerical simulations of the metrics.

In the literature the notations and the shapes of the equations are absolutely not standardized. Therefore writing all the existing equations coming from the quoted papers with respect to the function $p_1(k)$ (defined in Section 1.4.1) has corresponded to a large amount of work. But this work of re-writing the existing equations was necessary for at least two reasons: *i*) to compare the formulas obtained by different authors for the same metric (we have found a lot of mistakes or misunderstandings); *ii*) to compare the existing expressions with those that are proposed in Chapter 2.
Lastly, we consider an instantaneous and perfect feedback (this implies that there is no difference between SW and SR for instance) throughout the thesis and in this state-of-the-art as well. If a paper cited in the state-of-the-art works with non-instantaneous or/and imperfect feedback, we obviously consider it but we only introduce the result of this paper in our context of instantaneous and perfect feedback. As already mentioned, a confusion between $\bar{n}$ and $\bar{n}_r$ has been made in some papers, which implies that some closed-form expressions found in the literature are not correct. We nevertheless report them below. To distinguish the right ones from the ones we think are wrong, we use the notation $\overset{w}{=}$ for the wrong ones.

1.4.2.1 Theoretical analysis using a combinational approach

In this section, we start by introducing the closed-form expressions when the transmission credit per FRAG $P_{\max}$ (connected to the persistence P) is infinite.

Case 1: infinite $P_{\max}$
As the number of transmissions per FRAG is infinite, the PER vanishes regardless of the used HARQ mechanism, thus:

$$\lim_{P_{\max}\to\infty} \Pi_F = 0. \tag{1.1}$$

This previous expression has been found in numerous papers such as in [5].

The efficiency has the following expression:

$$\lim_{P_{\max}\to\infty} \eta_F = \frac{\rho}{\sum_{k=1}^{\infty} k p_1(k)} \tag{1.2}$$

provided in [32] (for ARQ, ARQ with FEC, CC-HARQ, IR-HARQ) and in [33] (for IC-HARQ). It is worth noting that (1.2), when ARQ occurs, boils down to:

$$\lim_{P_{\max}\to\infty} \eta_F = \rho(1-\pi_0) \quad \text{(for ARQ only)} \tag{1.3}$$

as reported in [5]. Actually (1.3) is valid as soon as the terms π_j are independent of j which occurs for ARQ and ARQ with FEC.

As for the delay, one can first notice that:

$$\lim_{P_{\max}\to\infty} \bar{n}_F = \lim_{P_{\max}\to\infty} \bar{n}_{F,r}$$

since the PER is zero. Consequently, when $P_{\max}$ is infinite, there is no confusion between $\bar{n}_F$ and $\bar{n}_{F,r}$ as they have the same limit, and all the expressions found in the literature are correct. The following expression:

$$\lim_{P_{\max}\to\infty} \bar{n}_F = \lim_{P_{\max}\to\infty} \bar{n}_{F,r} = \sum_{k=1}^{\infty} k p_1(k) \tag{1.4}$$

is found in [34] (for ARQ, ARQ with FEC), in [32] (for ARQ, ARQ with FEC, CC-HARQ, IR-HARQ), and in [33] (for IC-HARQ). In the case of ARQ and ARQ with FEC, we have:

$$\lim_{P_{\max}\to\infty} \bar{n}_F = \lim_{P_{\max}\to\infty} \bar{n}_{F,r} = \frac{1}{1-\pi_0} \quad \text{(for ARQ only)}. \tag{1.5}$$

Some authors (specifically [9,32]) noted that the efficiency can be connected to the delay as follows:

$$\begin{aligned} \lim_{P_{\max}\to\infty} \eta_F &= \frac{\rho}{\lim_{P_{\max}\to\infty} \bar{n}_F}, \\ &= \frac{\rho}{\lim_{P_{\max}\to\infty} \bar{n}_{F,r}}. \end{aligned}$$

We will see in next section (associated with finite $P_{\max}$) that both delays $\bar{n}_F$ and $\bar{n}_{F,r}$ are different. We will also see that the efficiency (when the DPDUs have the same length) is proportional to the inverse of $\bar{n}_{F,r}$ and not of $\bar{n}_F$. Consequently the second equality is always correct whereas the first one only holds for infinite transmission credit per FRAG. Unfortunately, in a lot of papers (as can be seen below), the authors incorrectly believed that the first equation still holds.

As a conclusion, we notice that the analysis for the jitter is always omitted by the authors. In addition the closed-form expressions of the metrics, written with respect to $p_1(k)$, are insensitive to the nature of the considered HARQ mechanism (although the authors established them for specific HARQ mechanism, and were not aware of the generic nature of their approach). Actually the expression of $p_1(k)$, through the expression of π_j, depends on the nature of the considered HARQ mechanisms.

Now we move on the second case which is in line with real implementation of new standards: finite $P_{\max}$. In the literature, this case corresponds to the so-called *truncated* HARQ [12,35] introduced in order to limit the maximum delay per packet.

Case 2: finite $P_{\max}$

The PER can be described by the following expression:

$$\Pi_F = 1 - \sum_{k=1}^{P_{\max}} p_1(k) \tag{1.6}$$

given in [9]. In the case of ARQ, the previous expression reduces to the following one [24]:

$$\Pi_F = \pi_0^{P_{\max}} \quad \text{(for ARQ only)}. \tag{1.7}$$

Both previous terms go to zero when the transmission credit per FRAG goes to infinity which confirms (1.1).

Different expressions for the efficiency have been found in the literature. Fortunately most of them are equivalent and correct. As the transmission credit $P_{\max}$ is finite, note that the obtained closed-form expressions are much more complicated than in the infinite case. The following expression for the efficiency:

$$\eta_F = \rho \frac{\sum_{k=1}^{P_{\max}} p_1(k)}{1 + \sum_{i=1}^{P_{\max}-1} (1 - \sum_{j=1}^{i} p_1(j))} \tag{1.8}$$

has been provided in [9] and in [36] for IR-HARQ. In [37], an other expression for the efficiency has been established:

$$\eta_F = \rho \frac{1-\Pi_F}{P_{\max}\Pi_F + \sum_{k=1}^{P_{\max}} k p_1(k)}, \tag{1.9}$$

$$= \rho \frac{\sum_{k=1}^{P_{\max}} p_1(k)}{P_{\max}\left(1-\sum_{k=1}^{P_{\max}} p_1(k)\right) + \sum_{k=1}^{P_{\max}} k p_1(k)}, \tag{1.10}$$

where Π_F has been given by (1.6). (1.9) and (1.10) have been proven for CC-HARQ and IR-HARQ. One can note that (1.9) fortunately reduces to (1.2) when assuming infinite $P_{\max}$, since $P_{\max}\Pi_F$ goes to zero when $P_{\max}$ goes to infinity. One can also prove that (1.8) and (1.9) are actually identical.

In [17], the following expression for the efficiency has been put:

$$\eta_F \overset{w.}{=} \frac{\rho}{\sum_{k=1}^{P_{\max}} k p_1(k)}. \tag{1.11}$$

This expression is not correct. In their proof, the authors made two mistakes: *i*) confusion between $\bar{n}_F$ and $\bar{n}_{F,r}$, and *ii*) mistake in the derivations on $\bar{n}_F$ as it can be seen later in (1.12).

Note that, for the case when ARQ or ARQ with FEC is used, the efficiency reduces to:

$$\eta_F = \rho(1-\pi_0) \quad \text{(for ARQ only)}$$

which is for example provided by [24]. The efficiency is independent of the persistence in ARQ scheme since the expression for finite persistence is the same as for infinite persistence (cf. (1.3)).

Let us now inspect the existing results about delay derivations. First of all, we focus on the derivations of the *true* delay $\bar{n}_F$. In [17], it has been established that:

$$\bar{n}_F \overset{w.}{=} \sum_{k=1}^{P_{\max}} k p_1(k) \tag{1.12}$$

which is not correct as we will see with the true expressions given in the next section and in Chapter 2. In [24], the authors found for the ARQ scheme:

$$\bar{n}_F \overset{w.}{=} \sum_{k=0}^{P_{\max}-1} \pi_0^k = \frac{1-\pi_0^{P_{\max}}}{1-\pi_0} \tag{1.13}$$

which is also not correct as we will see with the true expression for ARQ scheme provided in Chapter 2. Note that, by putting $P_{\max} = +\infty$ in (1.12) we find the correct expression given by (1.4) for the delay with infinite transmission credit. By putting $P_{\max} = +\infty$ in (1.13), we find the correct expression reported in (1.5).

Due to the non-null PER when $P_{\max}$ is finite, it is easy to prove that the equality between the two kinds of delays do not hold anymore. Consequently, we get:

$$\bar{n}_F \neq \bar{n}_{F,r}.$$

In contrast, we know that, if DPDUs have the same length, we have:

$$\eta_F = \rho \frac{1}{\bar{n}_{F,r}}.$$

In [9], the delay has been obtained as the inverse of the efficiency. Consequently, the authors derived the term $\bar{n}_{F,r}$ and not $\bar{n}_F$. Nevertheless, as they did not define the considered delay, we do not know if they were aware of these two kinds of delays. So, in [9], we have:

$$\bar{n}_{F,r} = \frac{1 + \sum_{i=1}^{P_{\max}-1}(1 - \sum_{j=1}^{i} p_1(j))}{\sum_{k=1}^{P_{\max}} p_1(k)}. \tag{1.14}$$

We remind that $\bar{n}_{F,r}$ is not the useful delay for system design.

Let us now focus on the jitter. This metric is very seldom taken into account in the literature although it is of great interest for upper layers. The following expression:

$$\sigma^2_{n_F} \overset{w.}{=} \sum_{k=1}^{P_{\max}} k^2 p_1(k) - (\sum_{k=1}^{P_{\max}} k p_1(k))^2. \tag{1.15}$$

has been given in [38] (for ARQ, ARQ with FEC, CC HARQ, IR-HARQ). Unfortunately, this expression does not correspond to the true jitter. Actually it is the "jitter" associated with the false expression of the delay given in (1.12). As (1.12) becomes correct for infinite transmission credit per FRAG, (1.15) becomes correct only for infinite transmission credit per FRAG.

Usually $p_1(k)$ corresponds to the probability that a FRAG is successfully received in exactly k transmissions. But other definitions for $p_1(k)$ are possible. For instance, in [9, 32, 39], $p_1(k)$ has been expressed as in (1.1) but where π_{j-1} is the outage probability and is equal to $\Pr\{c_j < r\}$ with c_j the channel capacity at the j^{th} transmission and r the target rate.

In the literature, we thus found correct closed-form expressions for PER and efficiency. Even if the authors proved their expressions for specific HARQ schemes, we will see in Chapter 2 that these expressions hold for any HARQ scheme. We also noticed that the delay and the jitter have not been expressed in closed-form via the combinational approach. Moreover a lot of papers confuse $\bar{n}_F$ with $\bar{n}_{F,r}$.

1.4.2.2 Theoretical analysis using a Markovian approach

In the literature, an other approach to establish the closed-form expressions of the metrics exists: this approach relies on the Markov chain. There are two ways to introduce the Markov chain for our problem as described below.

- The propagation channel is modeled by a Markov chain with N_b states. Each state corresponds to one level of the channel quality. The most commonly employed channel modeled by a Markov chain is the Gilbert-Elliot channel. Such a channel has only two states: the state "GOOD" associated with a high SNR (*i.e.* a low bit error rate) and the state "BAD" associated with a low SNR (*i.e.* a high bit error rate). Note that it is possible to merge these two states into a single one if the average SNR is identical for both states. For instance, the Gaussian channel and the Rayleigh channel can be viewed as a Gilbert-Elliot channel with only one single state.

- The HARQ scheme (regardless of the type of propagation channel) is modeled by a Markov chain.

As the first approach does not really use the Markov chain for modeling the underlying HARQ scheme, we only focus on the second approach which is the most interesting because strongly different from the combinational approach.

Case 1: infinite $P_{\max}$
Most works using the markovian approach have focused on the case when the transmission credit is infinite.

In [40], the efficiency has been derived for the Go Back-N ARQ scheme with unreliable feedback and timeout mechanism. The ARQ scheme has then been modeled by a four states Markov chain as follows:

- the first state corresponds to a transmission of an erroneous FRAG and a correct feedback,
- the second state corresponds to a transmission of a correct FRAG and a correct feedback,
- the third state corresponds to a transmission of an erroneous FRAG and an erroneous feedback,
- the fourth state corresponds to a transmission of a correct FRAG and an erroneous feedback.

In [40], it has been proven that:

$$\eta_F = \frac{\bar{R}}{\bar{D}} \tag{1.16}$$

where $\bar{R}$ is the average number of acknowledged transmissions during a cycle (which is the time between two consecutive passages to the reference state), and $\bar{D}$ is the average time between two consecutive entrances into the reference state. (1.16) has also been found in [41] for Selective Repeat ARQ scheme. Obviously the Markov chain developed in [40] still holds when the feedback is perfect (as assumed in this manuscript) by removing the third and fourth states. In [40], when only two states occur, it has been proven that $\bar{R} = p_0(P_{01}R_{01} + P_{00}R_{00}) + p_1(P_{10}R_{10} + P_{11}R_{11})$ and $\bar{D} = p_0(P_{01}D_{01} + P_{00}D_{00}) + p_1(P_{10}D_{10} + P_{11}D_{11})$ where p_i is the steady-state probability of state i, where P_{ij} is the probability of transition from state i to state j, and where R_{ij} (resp. D_{ij}) is the number of acknowledged transmissions associated with transition from state i to state j (resp. the delay associated with transition from state i to state j). In ARQ scheme, we have $p_0 = 1 - \pi_0$ and $p_1 = \pi_0$, $P_{01} = P_{11} = \pi_0$, $P_{10} = P_{00} = 1 - \pi_0$. In addition we can see that $R_{00} = R_{10} = 1$, $R_{01} = R_{11} = 0$, and $D_{00} = D_{01} = D_{10} = D_{11} = 1$. Then we find fortunately (1.3) with $\rho = 1$.

The delay has been studied by [42] as follows: the analysis holds when the retransmission mechanism is RM1 or RM2, and when the receiver side is RP1 or RP3. Consequently, the analysis is valid for ARQ, ARQ with FEC, CC-ARQ, CC-HARQ. Let $N_{\max}$ be the maximum number of the received versions of the same FRAG to be Chase-combined. Such a HARQ scheme can be modeled by a Markov chain with $(N_{\max} + 2)$ states. One of these states is called "FACK" and corresponds to an ACK error or loss with probability f. An other of these states is called "SUC" and corresponds to successful decoding. Note that the transmission credit per FRAG is infinite but the maximum number of FRAG copies (or DPDUs coming from the same FRAG) that can

be combined together is finite. This means that for the $(N_{\max}+1)^{\text{th}}$ transmission of the same FRAG, the Chase combination is not done and the decision is only made on the incoming FRAG. The closed-form expression for the delay has then been given by:

$$\bar{n}_F = \mathbf{v}(\mathbf{I}_{N_{\max}+1} - \mathbf{G})^{-1}\mathbf{w}, \tag{1.17}$$

where

- $\mathbf{v}$ is the $1 \times (N_{\max}+1)$ row vector with the component corresponding to the initial state equal to 1 and 0 otherwise,
- $\mathbf{I}_{N_{\text{x}}+1}$ is the $(N_{\text{x}}+1) \times (N_{\text{x}}+1)$ identity matrix,
- $\mathbf{G}$ is a $(N_{\max}+1) \times (N_{\max}+1)$ matrix, that elements are equal to the transition probabilities but in which the row and the column of the "SUC" state have been deleted
- and $\mathbf{w}$ is the column vector with every component equal to 1.

If we consider only ARQ or ARQ with FEC, we have $N_{\max} = 1$. After some algebraic manipulations, one can write:

$$\bar{n}_F = \frac{(1-f)+f(1-\pi_0)}{(1-f)(1-\pi_0)} \quad \text{(for ARQ only)}. \tag{1.18}$$

Knowing that $f = 0$ in our case, we obtain (1.5).

Case 2: finite $P_{\max}$

Clearly the most interesting work based on the markovian approach is [43] in which finite transmission credit $P_{\max}$ and almost every HARQ scheme (CC-HARQ, IR-HARQ, and even IC-HARQ) have been considered. In addition no perfect feedback and round-trip time have been taken into account. For instance, the round-trip time is here equal to m and each state of the chain is characterized by a state vector $\mathbf{v}$ of m couples of the form (l_i, r_i) $(1 \leq i \leq m)$ where r_i represents the number of retransmissions that have occurred for the i^{th} FRAG ($r_i = 0$ corresponds to the first transmission) and where l_i represents the total number of bit errors that have been undergone by the transmission of FRAG i after r_i retransmissions of the same FRAG i. If l_i is lower than a predetermined error correction threshold θ_{r_i}, the transmission is successful whereas if it is greater, the transmission fails. In order to compute the evolution of the error level l_i, the authors introduced an extra state, denoted by S, that corresponds to the bit errors ξ_S added by the channel to each DPDU transmission. Consequently $l_{i+1} = l_i + \xi_S$. This channel information is adjuncted to the couples (l_i, r_i) to build a state as follows:

$$\mathbf{v} = ((l_1, r_1), (l_2, r_2), \cdots, (l_m, r_m), S). \tag{1.19}$$

In the sequel, we denote the steady-state probability of state $\mathbf{v}$ by $\sigma(\mathbf{v})$. In [43], closed-form expressions for PER, efficiency, and delay have been provided. The authors then had:

$$\Pi_F = \sum_{\mathbf{v} \in \mathcal{B}} \sigma(\mathbf{v}) \tag{1.20}$$

where $\mathcal{B} = \{\mathbf{v}|\mathbf{v} = ((x, P_{\max} - 1), (l_2, r_2), \cdots, (l_m, r_m), S)$ with $\theta_{P_{\max}-1} < x\}$ is the set of vectors such that the first FRAG is still received with errors after $P_{\max}$ transmissions. For the efficiency, they get:

$$\eta_F = \sum_{\mathbf{v}\in\mathcal{A}} \sigma(\mathbf{v}) \tag{1.21}$$

where $\mathcal{A} = \{\mathbf{v}|\mathbf{v} = ((0, r_1), (l_2, r_2), \cdots, (l_m, r_m), S)\}$ is the set of vectors such that the first FRAG is successfully received after r_1 retransmissions. As for the delay, it has been given by:

$$\bar{n}_F = \frac{\sum_{\mathbf{v}\in\mathcal{A}} r_1\sigma(\mathbf{v})}{\sum_{\mathbf{v}\in\mathcal{A}} \sigma(\mathbf{v})}. \tag{1.22}$$

In order to translate these expressions in our framework, we consider now that the feedback is instantaneously sent which implies that $m = 1$. Furthermore, in Gaussian channel, l_1 only takes two values: 0 if the transmission succeeds (ACK) and 1 if it fails (NACK). The term π_j represents the probability to have l_1 equal to 0 or 1 given the number j of retransmissions. The link between π_j and π_{j+1} does not need an extra state. Consequently, we can remove it. Then the state vector is only described by the couple (l_1, r_1). Finally, $\mathcal{A}$ and $\mathcal{B}$ are simply given by $\mathcal{A} = \{\mathbf{v}|\mathbf{v} = (0, r_1)\}$ and $\mathcal{B} = \{\mathbf{v}|\mathbf{v} = (1, P_{\max} - 1)\}$. After these simplifications, we can write (1.20), (1.21), and (1.22) as follows:

$$\Pi_F = 1 - \sum_{k=1}^{P_{\max}} p_1(k), \tag{1.23}$$

$$\eta_F \overset{w.}{=} \sum_{k=1}^{P_{\max}} p_1(k) \tag{1.24}$$

and

$$\bar{n}_F = \frac{\sum_{k=1}^{P_{\max}} k p_1(k)}{\sum_{k=1}^{P_{\max}} p_1(k)}. \tag{1.25}$$

Some comments are given below: (1.25) is the first closed-form expression for the delay valid when the persistence is finite. Therefore this paper definitely does add value on that point. In contrast, one can notice that combining (1.23) and (1.24) leads to $\eta_F = 1 - \Pi_F$ which is not correct according to our definition. Consequently, (1.24) is not true. Given (1.24), the efficiency should be the probability to successfully transmit one FRAG in maximum $P_{\max}$ transmissions. Such a definition for the efficiency is not in agreement with the one previously done. Fortunately (1.23) concerning the PER is the same as (1.6).

1.4.2.3 Approximation and simulation based analysis

Instead of trying to derive the metrics in closed-form, some papers only concentrated on calculating some approximations or bounds for the considered metrics. Actually most works dealt with delay evaluation.

In [44], the authors provided the following lower and upper bounds for the delay in case of CC-HARQ:

$$1 + \sum_{i=1}^{P_{\max-1}} \prod_{j=1}^{i} \left(1 - \sum_{k=1}^{j} p_1(k)\right) \leq \bar{n}_F \leq 1 + \sum_{i=1}^{P_{\max-1}} \left(1 - \sum_{j=1}^{i} p_1(j)\right) \tag{1.26}$$

We can note that the upper bound of (1.26) is similar to (1.14) but without the denominator. As a result the upper bound of $\bar{n}_F$ is really a lower bound of $\bar{n}_{F,r}$ since the denominator is less than 1. Therefore we are not sure that previous lower and upper bounds are correct for the delay.

In [45], the authors approximated the delay in case of IR-HARQ as follows:

$$\bar{n}_F \approx \frac{1}{1-\Pi_F} \left(= \frac{1}{\sum_{k=1}^{P_{\max}} p_1(k)} \right). \tag{1.27}$$

This approximation is rough since the true numerator equal to $\sum_{k=1}^{P_{\max}} k p_1(k)$ (as seen in (1.25)) is omitted.

To find bounds on the efficiency, [46] used the fact that the efficiency is inversely proportional to the delay given by (1.26) in the case of infinite transmission credit per FRAG.

As the derivations for the HARQ are rather difficult, a lot of authors often prefered to assess the HARQ performance by simulating the whole system. A high quantity of works based on this approach can be found. As an example, the authors of the most articles mentioned in Section 1.3 studied their proposed scheme through extensive simulations, even if these simulations are time consuming and do not provide useful insights about the influence of the design parameters.

1.4.3 HARQ performance analysis at the NET level

As a system is not only composed by the MAC layer, it is useful to evaluate the performance of HARQ mechanism on upper layer, such as the adjacent NET layer. This enables us to be more realistic on the true system performance and so more representative of the real QoS observed by the users. This section is organized as follows: in Section 1.4.3.1, we introduce a new manner (valid for each HARQ mechanism) of distributing the transmission credit using the fact that DPDUs at the MAC layer belong to the same DSDU at the NET layer. This cross-layer optimization strategy has been proposed in [3] for ARQ mechanism. Section 1.4.3.2 is devoted to the state-of-the-art about theoretical performance analysis at the NET layer. Finally Section 1.4.3.3 deals with simulation based performance analysis for the NET layer and even the uppest layers.

1.4.3.1 Cross-layer optimization strategy

Before studying the performance evaluation at the NET layer, we first introduce a cross-layer optimization strategy proposed in [3] between both MAC and NET layers. Usually, a fixed number of retransmissions is granted per FRAG. Consequently if the last possible retransmission fails, the FRAG is dropped and the transmission is started again with the next FRAG. Note that if one FRAG is dropped at the receiver side, the corresponding DSDU at the NET layer is also dropped. Then it is absurd to continue to send FRAGs associated with the corrupted DSDU. Therefore the authors in [3] suggested to enhance the ARQ scheme by taking into account the fact that a set of FRAGs belongs to the same DSDU. To do that, they proposed to grant a global transmission credit, noted C, to the set of FRAGs belonging to the same DSDU. Thus, rather than allowing each of the N fragments to be transmitted $P_{\max}$ times, the new

strategy authorizes C transmissions to the set of N FRAGs. Results in [3] show that this cross-layer strategy outperforms the conventional one in terms of PER. In the sequel, we will refer the conventional one to as PDU-Based Strategy (PBS) and the cross-layer one to as SDU-Based Strategy (SBS). Furthermore, depending on the retransmission strategy (PBS or SBS), we will differentiate the variables by adding an upper script, respectively with a 'P' or a 'S'. As an example, η_S^P will denote the efficiency at the NET layer when PBS is employed. In Fig. 1.9 (resp. Fig. 1.10), we draw the scheme of the PBS (resp. SBS).

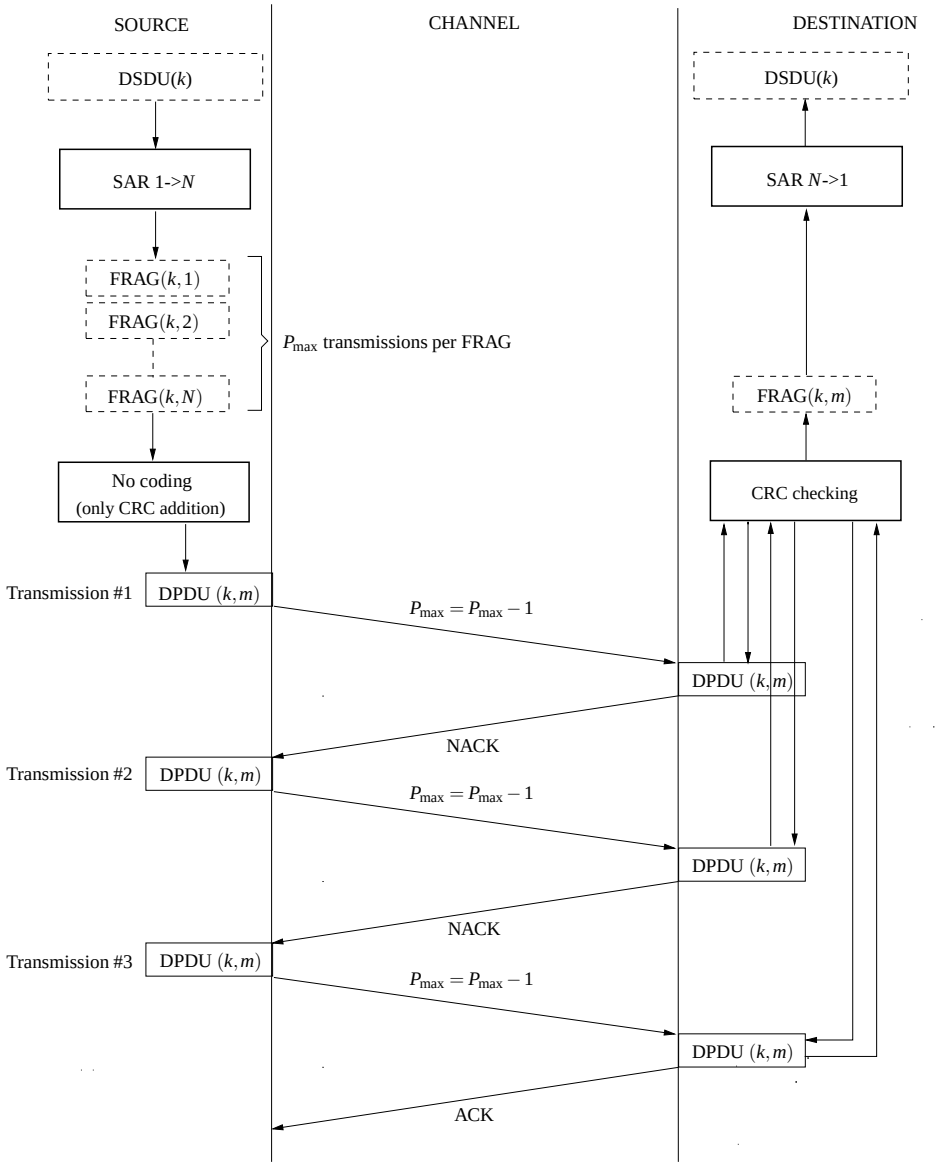

Figure 1.9: PBS scheme (for ARQ with $P_{\max} \geq 3$).

1.4.3.2 Theoretical analysis

Only a few papers have focused on the theoretical performance of HARQ mechanism at the NET layer.

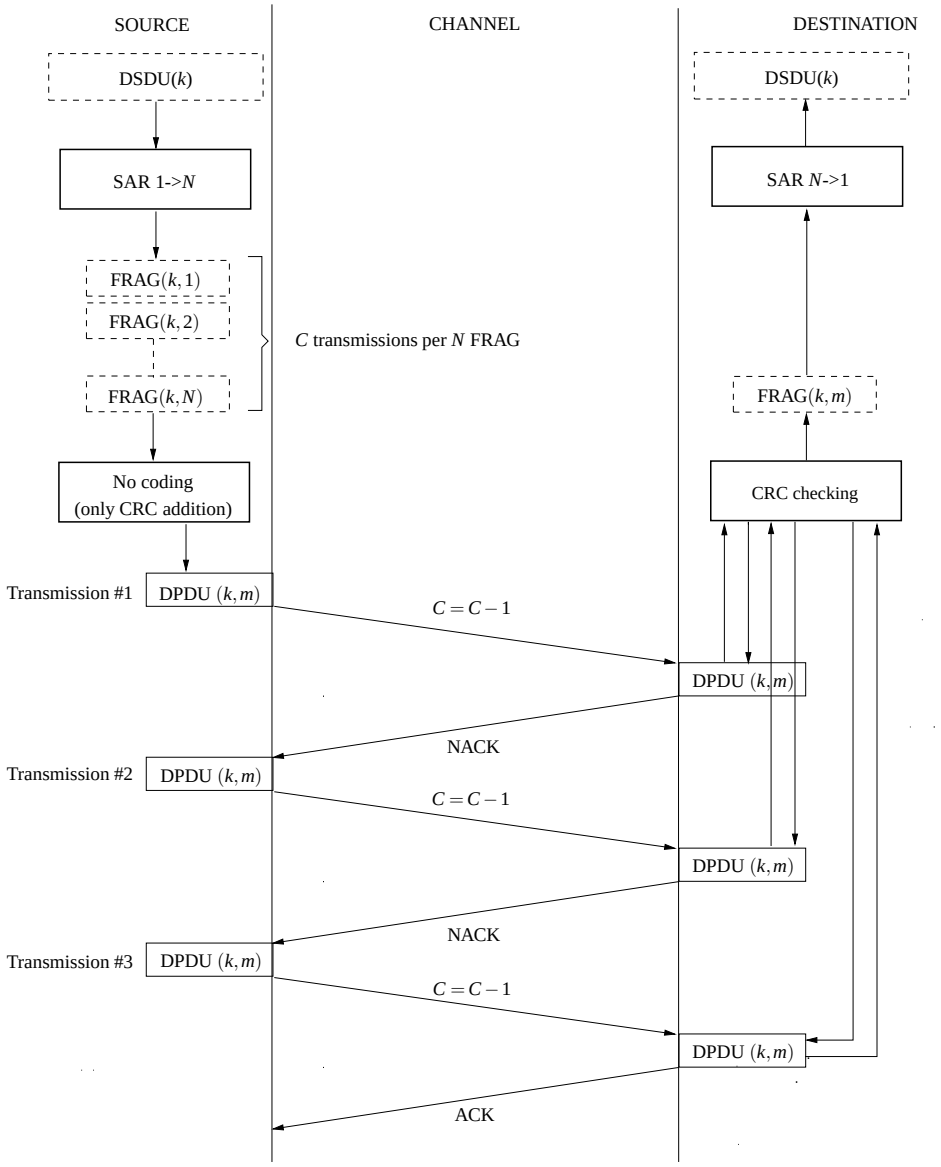

Figure 1.10: SBS scheme (for ARQ with $C \geq 3$).

Let us start with [3] which deals with ARQ mechanism and both PBS and SBS strategies. The authors proposed PER and delay closed-form expressions. According to the notations introduced in Section 1.4.1, they obtained:

$$\Pi_S^P = 1 - (1 - {\pi_0}^{P_{\max}})^N \tag{1.28}$$

and

$$\Pi_S^S = 1 - \sum_{k=N}^{NP_{\max}} p_N^S(k), \tag{1.29}$$

with $p_N^S(k)$ the probability to successfully receive N FRAGs in exactly k transmissions in the context of SBS strategy. Note that (1.28) is available in [47] as well. In the case of ARQ scheme, the term $p_N^S(k)$ can be written as follows:

$$p_N^S(k) = \binom{k-1}{N-1}(1-\pi_0)^N {\pi_0}^{(k-N)}. \tag{1.30}$$

The authors obtained the following expressions for the delay:

$$\bar{n}_S^P = \frac{\sum_{k=N}^{NP_{\max}} k p_N^P(k)}{(1-\pi_0{}^{P_{\max}})^N} \tag{1.31}$$

and

$$\bar{n}_S^S = \frac{\sum_{k=N}^{NP_{\max}} k p_N^S(k)}{\sum_{k=N}^{NP_{\max}} p_N^S(k)}, \tag{1.32}$$

where $p_N^P(k)$ is the probability to successfully receive N FRAGs in exactly k transmissions in the context of the PBS strategy. By using (1.30), the delay for the SBS strategy takes the following form:

$$\bar{n}_S^S = N \frac{\sum_{k=N}^{NP_{\max}} \binom{k}{N} \pi_0^k}{\sum_{k=N}^{NP_{\max}} \binom{k-1}{N-1} \pi_0^k}.$$

Moreover the closed-form expression $p_N^P(k)$ is available in the paper as follows:

$$p_N^P(k) = \sum_{(n_1,\cdots,n_N)\in Q_k^P} \prod_{j=1}^{N} (1-\pi_0)\pi_0^{n_j-1}$$

where $Q_k^P = \{(n_1,\cdots,n_N) \in \mathbb{N} | \sum_{j=1}^N n_j = k \text{ and } 1 \leq n_j \leq P_{\max}\}$. In [47], it has also been proven that

$$\bar{n}_S^P = N \bar{n}_F^P. \tag{1.33}$$

An other work has proposed closed-form expression for the PER in the case of ARQ mechanism and PBS strategy at the NET layer [48]. The paper is based on a Gilbert-Elliot channel model. In order to simplify the presentation, we will consider a degenerate Gilbert-Elliot model with only one state. As already mentioned, Gaussian or Rayleigh channels can be treated as a Gilbert-Elliot model with one single state. The authors obtained that:

$$\Pi_S^P = 1 - \sum_{k=N}^{NP_{\max}} p_N^P(k). \tag{1.34}$$

It is interesting to notice that (1.34) concerning the PBS strategy has the same form as (1.29) concerning the SBS strategy. Actually (1.34) is identical to (1.28) since one can prove that $\sum_{k=N}^{NP_{\max}} p_N^P(k) = (1-\pi_0{}^{P_{\max}})^N$ (see Chapter 2).

Finally, in [49], closed-form expression for the delay has been proposed in the case of ARQ mechanism and PBS strategy at the NET layer. To derive the delay, a Gilbert-Elliot channel model has been considered. Once again, we introduce the degenerate case. The authors established that:

$$\bar{n}_S^P \overset{w.}{=} \mathbf{v}(\mathbf{I} - \mathbf{G}^{P_{\max}})(\mathbf{I} - \mathbf{G})^{-1}\mathbf{w} \tag{1.35}$$

where

- $\mathbf{v}$ is the N components row vector $\begin{bmatrix} 0, & 0, & \cdots & 1 \end{bmatrix}$,
- $\mathbf{I}$ is the identity matrix with the same size as $\mathbf{G}$,
- $\mathbf{G}$ is the following matrix $\begin{bmatrix} \pi_0, & 0, & \cdots & 0 \\ \pi_0(1-\pi_0), & \pi_0^2, & \cdots & 0 \\ \vdots, & \vdots, & \ddots & \vdots \\ \pi_0(1-\pi_0)^{N-1}, & \pi_0^2(1-\pi_0)^{N-2}, & \cdots, & \pi_0^N \end{bmatrix}$

- and $\mathbf{w}$ is the column vector with every component equal to 1.

In order to check this previous expression, we consider $N = 1$ (we thus boil down to MAC layer). Then (1.35) becomes $(1 - \pi_0^{P_{\max}})/(1 - \pi_0)$ and is equal to (1.13) which is not correct. Consequently (1.35) is not correct.

We finally notice that the literature about theoretical performance of HARQ mechanism at the NET layer is poor since HARQ is never considered (only ARQ) and efficiency and jitter are never derived (only PER and delay).

1.4.3.3 Simulation based analysis

Some papers have considered simulation based performance analysis for ARQ/HARQ mechanism at the NET layer and even other upper layers.

For example, in [49], the authors studied the interactions between ARQ mechanism and Transport level (TCP for TCP/IP protocol) over wireless links. They showed, by simulations, that limiting the number of retransmissions granted per FRAG degrades the TCP goodput (defined as the total useful TCP payload data received at the end of each simulation run and expressed in FRAGs per second) and leads to a loss in wireless efficiency (bandwidth and energy consumption). Then, the authors concluded that the best and simplest choice at the TCP level is to adopt a fully reliable link layer ARQ protocol. In [50], they extended their previous work to the HARQ. They obtained the same conclusion as in [49]. In [38], the authors confirmed experimentally the conclusion done in [49] and moreover showed that, even at the TCP level, IR-HARQ and CC-HARQ enable to achieve better performance than the ARQ and ARQ with FEC.

1.4.4 System performance optimization

Furthermore, the system performance depend on the design parameters of the system. Indeed, design parameters (like persistence, number of fragments per DSDU, *etc.*) play an important role in the performance and can be adjusted in order to satisfy the different QoS requirements. Hence, some works have been made by simulations to analyze the impact of these parameters on the system performance at the upper layers.

We can note that the persistence has an important influence on the upper layers as pointed in [51] where the authors assessed the TCP goodput for IR-HARQ by simulations. But other parameters have to be taken into account in order to satisfy the required performance. Indeed, the same authors also demonstrated that the persistence and the Maximum Segment Size (MSS) are interdependent: if the persistence is low, a medium value of the MSS is optimal, whereas if the persistence is medium, a high value of the MSS is necessary. Furthermore, based on ARQ mechanism, [48] shows that, when the transmitter is far from the receiver, the DSDU has to be fragmented in a high number of FRAGs for improving the PER.

Little effort has then been made in order to determine the impact of the HARQ mechanisms of the cross-layer strategies on the whole system. Clearly the related works are not sufficient enough and a rigorous study has still to be made in order to define a relevant selection of the HARQ mechanisms, cross-layer strategies, and design parameters according to one target QoS.

1.5 Thesis objective

In Tab. 1.2, Tab. 1.3, and Tab. 1.4, we sum up all the closed-form expressions available in the literature with respect to HARQ mechanisms, considered layers, and cross-layer strategies. In the tables, only the correct expressions of the metrics (according to our previous analysis) have been reported. The cross "×" means that the corresponding expression does not exist in the literature.

SBS (NET)		
	ARQ	HARQ
PER	$1-\sum_{k=N}^{C} p_N^S(k)$	×
Efficiency, UDL	Not applicable	×
Efficiency, EDL	×	×
Delay	$\frac{\sum_{k=N}^{C} k p_N^S(k)}{\sum_{k=N}^{C} p_N^S(k)}$	×
Jitter	×	×

Table 1.2: Available theoretical expressions from the state-of-the-art for the SBS at the NET level. (Unequal DPDU Length: UDL, Equal DPDU Length: EDL.)

PBS (NET)		
	ARQ	HARQ
PER	$1-\sum_{k=N}^{NP_{\max}} p_N^P(k)$	×
Efficiency, UDL	Not applicable	×
Efficiency, EDL	×	×
Delay	$\frac{\sum_{k=N}^{NP_{\max}} k p_N^P(k)}{\sum_{k=N}^{NP_{\max}} p_N^P(k)}$	×
Jitter	×	×

Table 1.3: Available theoretical expressions from the state-of-the-art for the PBS at the NET level. (Unequal DPDU Length: UDL, Equal DPDU Length: EDL.)

We notice that all the metrics depend on the term $p_N(k)$ defined as the probability

PBS (MAC)		
	ARQ	HARQ
PER	$1-\sum_{k=1}^{P_{\max}} p_1(k)$	$1-\sum_{k=1}^{P_{\max}} p_1(k)$
Efficiency, UDL	Not applicable	$\times$
Efficiency, EDL	$\frac{\rho \sum_{k=1}^{P_{\max}} p_1(k)}{P_{\max}\left(1-\sum_{k=1}^{P_{\max}} p_1(k)\right)+\sum_{k=1}^{P_{\max}} k p_1(k)}$	$\frac{\rho \sum_{k=1}^{P_{\max}} p_1(k)}{P_{\max}\left(1-\sum_{k=1}^{P_{\max}} p_1(k)\right)+\sum_{k=1}^{P_{\max}} k p_1(k)}$
Delay	$\frac{\sum_{k=1}^{P_{\max}} k p_1(k)}{\sum_{k=1}^{P_{\max}} p_1(k)}$	$\frac{\sum_{k=1}^{P_{\max}} k p_1(k)}{\sum_{k=1}^{P_{\max}} p_1(k)}$
Jitter	$\times$	$\times$

Table 1.4: Available theoretical expressions from the state-of-the-art at the MAC level. (Unequal DPDU Length: UDL, Equal DPDU Length: EDL.)

of receiving successfully N FRAGs in exactly k transmissions. Closed-form expressions for $p_N(k)$ are listed on Tab. 1.5.

	ARQ	HARQ
$p_N^S(k)$	$\binom{k-1}{N-1}(1-\pi_0)^N \pi_0^{(k-N)}$	$\times$
$p_N^P(k)$	$(1-\pi_0)^N \sum_{(n_1,\cdots,n_N)\in S_k} \prod_{j=1}^{N} \pi_0^{n_j-1}$	$\times$
$p_1(k)$	$(1-\pi_0)\pi_0^{k-1}$	$(1-\pi_{k-1})\prod_{j=0}^{k-2}\pi_j$

Table 1.5: Available theoretical expressions from the state-of-the-art for $p_N(k)$.

The main purpose of the PhD thesis is to fulfill both previous tables. Consequently the thesis is organized as follows:

- In Chapter 2, the missing expressions are derived. They are valid for any HARQ mechanism regardless of the considered cross-layer strategy and layer. We specifically provide theoretical expression for efficiency when DPDUs associated with the same FRAG have different lengths.
- In Chapter 3, we firstly check that there is a perfect agreement between the empirical values of performance metrics and their closed-form expressions developed in Chapter 2. Then we also compare the different HARQ mechanisms and the different cross-layer strategies for several modulations and propagation channels.
- In Chapter 4, we use the theoretical expressions obtained in Chapter 2 to carry out

relevant resource allocation. We firstly analyze the influence of design parameters (N, C, $P_{\max}$, FEC scheme, and channel coding rate) on the global system performance versus the SNR. From that study we determine the best set of parameters to fulfill the various QoS.

Secondly, we show that the cross-layer strategy introduced in [3] induces an unequal packet loss protection and is thus relevant for practical scheme such as multimedia application. This concept is extended to the conventional strategy. Finally, we show that we can predict the packet error rate and delay performance of a discrete event based realistic simulation implementing the HARQ scheme, using the previous derived closed-form expressions.

Chapter 2

Performance Closed-Form Expressions

2.1 Introduction

In Chapter 2, we derive the closed-form expressions of the metric performance introduced in Chapter 1 as a function of the elementary packet error rate π_j and the parameters N, $P_{\max}$, and C.

In Section 2.2, we first derive expressions in the most general case using compact notations that do not always explicitly show the π_j values. We then derive the expressions for particular cases in Section 2.3, for equal length DPDUs and type I HARQ schemes. We discuss about the independence of the efficiency *vs.* the persistence and also prove that the cross-layer approach SBS performs better in terms of PER at the NET level than the PBS one. We finally end by giving a summary of the different steps needed to compute the performance from the π_j in Section 2.4, followed by a conclusion in Section 2.5.

2.2 General expressions

We derive the calculation according to the following order. We first start by the PER which is quite easy to obtain in a compact manner, all the complexity being hidden by the intermediate probability notations that we use. It is followed by the delay and the jitter expressions with the same comment as for the PER. We then finally end the section by deriving the efficiency expression with unequal DPDU lengths. This is the most challenging part of this work since the combinatorial is quite complex and is not easy to express in a compact way.

The assumptions made to derive the general expressions are the following conventional ones:

- the ACK/NACK are assumed error-free and instantaneous,
- the CRC is ideal with overhead zero,
- the DPDU packet length can be different from one transmission to another (for the IR scheme),
- the delay and jitter are counted in number of transmissions,

- the noisy DPDU packets are independent.

Note that the last assumption is obvious in an Additive White Gaussian Noise (AWGN) channel, and corresponds to a slow fading channel (typically Rayleigh) where the attenuation is constant within one DPDU transmission and is independent from one DPDU to another.

2.2.1 Packet error rate

At the MAC level, the PER is easily obtained in the PBS case by writing $\Pi_F := 1 - \Pr\{\text{FRAG received}\}$. The probability that a fragment is successfully received is equal to the probability that the fragment is received in one transmission, or in two transmissions, ... up to the maximum number of transmissions. Thus, from (1.1), we obtain:

$$\Pi_F = 1 - \sum_{k=1}^{P_{\max}} p_1(k), \tag{2.1}$$

where $P_{\max}$ takes different expressions depending on the RM (see Section 1.3.1).

At the NET level, inasmuch as the fragment transmissions are independent, it is straightforward to get:

$$\Pi_S^P = 1 - (1 - \Pi_F)^N, \tag{2.2}$$

which drops down to (2.1) when $N = 1$.

In the SBS, we consider only Π_S^S since Π_F is of no use. Using the same reasoning as for Π_F, the probability that a DSDU is successfully received is equal to the probability that the DSDU is received after N transmitted DPDUs, or received after $(N + 1)$ DPDUs, ... up to the maximum number of transmissions C. We thus obtain:

$$\Pi_S^S = 1 - \sum_{k=N}^{C} p_N^S(k). \tag{2.3}$$

where $p_N^S(k)$ is the probability that N FRAGs are transmitted in exactly k transmissions in the SBS. Let us introduce the following event definition: $F_j(\ell) := \{\text{FRAG } \#j \text{ is received in } \ell \text{ DPDUs}\}$, thus $p_N^S(k)$ corresponds to the event $\{F_1(i_1)$ and $F_2(i_2)$ and $\dots$ and $F_N(i_N)\ \}$ where the indices are constrained to verify $\sum_{m=1}^{N} i_m = k$ to account that the SDU is transmitted in k transmissions. From (1.1) and the $F_j(k)$ definition we find $\Pr\{F_j(k)\} = p_1(k)$. Since the FRAG transmissions are independent, we can thus deduce:

$$p_n^S(k) = \sum_{\underline{i} \in \mathcal{Q}_{n,k}^S} \prod_{m=1}^{n} p_1(i_m), \tag{2.4}$$

where the set $\mathcal{Q}_{n,k}^S$ is defined as

$$\mathcal{Q}_{n,k}^S = \{\underline{i} = [i_1, i_2, \cdots, i_n] \in \mathbb{N}_*^n \mid \sum_{m=1}^{n} i_m = k\}. \tag{2.5}$$

2.2.2 Delay

At the MAC level, the average number of transmitted DPDUs per successful FRAG transmission for the PBS is given by:

$$\bar{n}_F = \sum_{k=1}^{P_{\max}} k \Pr\{\text{FRAG received in } k \text{ DPDUs} \mid \text{FRAG received}\}. \tag{2.6}$$

The Bayesian rule allows to remove the conditioning in the probability in (2.6) and using (1.1) we get:

$$\begin{aligned} \bar{n}_F &= \frac{\sum_{k=1}^{P_{\max}} k \Pr\{\text{FRAG received in } k \text{ DPDUs}\}}{\Pr\{\text{FRAG received}\}}, \\ &= \frac{\sum_{k=1}^{P_{\max}} k p_1(k)}{1 - \Pi_F}, \\ &= \frac{\sum_{k=1}^{P_{\max}} k p_1(k)}{\sum_{k=1}^{P_{\max}} p_1(k)}. \end{aligned} \tag{2.7}$$

At the NET level for the PBS, the instantaneous delay for a DSDU is the summation of the N FRAG transmission delays. Noting $n_F(k)$, the instantaneous delay of the k-th FRAG, we have:

$$\bar{n}_S^P = \sum_{k=1}^{N} \mathbb{E}\left[n_F(k)\right]. \tag{2.8}$$

As the N FRAG transmission delays are independent and identically distributed, we have:

$$\bar{n}_S^P = N \bar{n}_F. \tag{2.9}$$

For the SBS, following the same reasoning as for $\bar{n}_F$ at the NET level, we get:

$$\bar{n}_S^S = \frac{\sum_{k=N}^{C} k p_N^S(k)}{1 - \Pi_S^S}. \tag{2.10}$$

This expression drops down to the expression given in [3] Eq. (7) for the particular ARQ case (see (1.32)).

Expressions (2.7), (2.9) and (2.10) are apparently new since, as for the PER, they can here be applied to all the different RM-RP combinations whereas they were previously given only for specific schemes.

2.2.3 Jitter

Using the same approach as above for the average delay, we obtain at the FRAG level:

$$\sigma_{n_F}^2 = \mathbb{E}[n_F]^2 - (\bar{n}_F)^2, \tag{2.11}$$

which leads to:

$$\sigma_{n_F}^2 = \frac{\sum_{k=1}^{P_{\max}} k^2 p_1(k)}{1 - \Pi_F} - (\bar{n}_F)^2. \tag{2.12}$$

At the NET level, the derivation for the PBS is straightforward since the FRAGs are *i.i.d.*, and we get:

$$\sigma_{n_S^P}^2 = N \sigma_{n_F}^2. \tag{2.13}$$

For the SBS case, following the same reasoning as for $\sigma^2_{n_F}$ at the NET level leads to:

$$\begin{aligned}\sigma^2_{n_S^S} &= \mathbb{E}[\left(n_S^S\right)^2] - (\bar{n}_S^S)^2, \\ &= \frac{\sum_{k=N}^{C} k^2 p_N^S(k)}{1-\Pi_S^S} - (\bar{n}_S^S)^2. \end{aligned} \tag{2.14}$$

2.2.4 Efficiency

From the efficiency definition (ratio between the average received bits and average transmitted bits), we can write:

$$\eta_y^x = \frac{n_1}{n_2} \tag{2.15}$$

where n_1 is the average number of received bits (with no error) and n_2 is the average number of transmitted bits. As illustrated in Fig. 2.1 the number of received bits with no error is equal to the transmitted information bits length I multiplied by the probability of success ($n_1 = I \cdot (1 - \Pi_y^x)$) where I is the number of information bits. The average number of transmitted bits is equal to the average number of transmitted bits when the packet of length I is successfully received (noted $\hat{n}$) multiplied by the probability of success, plus, the average number of transmitted bits when the transmission fails (noted $\check{n}$) multiplied by the probability of error ($n_2 = \check{n} \cdot \Pi_y^x + \hat{n} \cdot (1 - \Pi_y^x)$). It thus leads to:

$$\eta_y^x = \frac{I \cdot (1 - \Pi_y^x)}{\check{n}_y^x \cdot \Pi_y^x + \hat{n}_y^x \cdot (1 - \Pi_y^x)}. \tag{2.16}$$

This expression is very general and applies for any kind of retransmission scheme at

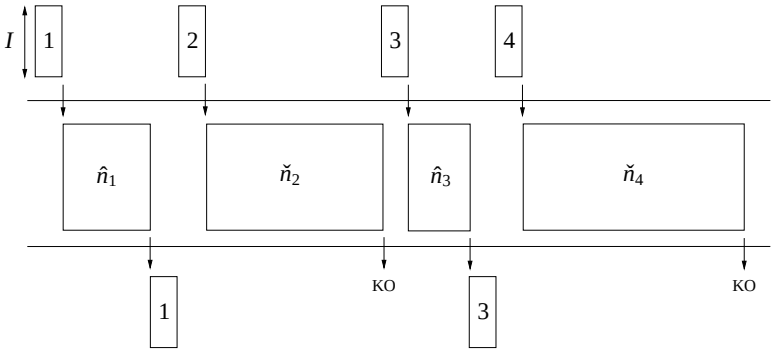

Figure 2.1: Illustration of $\check{n}$ and $\hat{n}$ definitions.

any layer. For instance, at the MAC level, we will have $I = L_F$, whereas at the NET level, we will have $I = L_{\text{DSDU}}$, the quantities $\hat{n}_y^x$ and $\check{n}_y^x$ being a function of I as well. We are now deriving the expressions for $\hat{n}_y^x$ and $\check{n}_y^x$ for the different considered cases (PBS at FRAG and NET levels, and SBS at NET level).

Before, let us introduce δ_k the number of transmitted bits for DPDU $\#k$ and $w_k := \sum_{i=1}^{k} \delta_i$ which gives the total number of bits transmitted at the k-th transmission (including transmissions from DPDU $\#1$ up to DPDU $\#k$) as illustrated in Fig. 2.2).

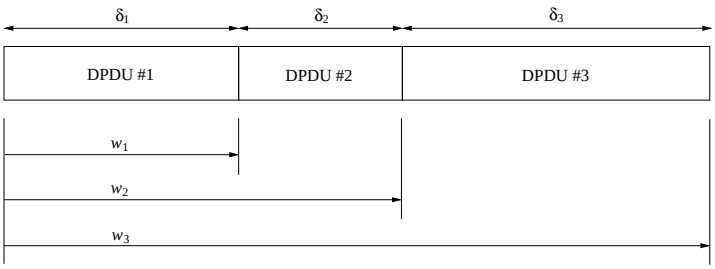

Figure 2.2: Illustration of w_k and δ_k definitions.

2.2.4.1 Calculation of $\hat{n}_F$

Thus, the average number of transmitted bits per successful transmitted FRAG is given by:

$$\begin{aligned}\hat{n}_F &= \sum_{k=1}^{P_{\max}} w_k \cdot \Pr\{F(k)|\text{ FRAG OK}\} \\ &= \frac{1}{1-\Pi_F} \sum_{k=1}^{P_{\max}} w_k \cdot p_1(k).\end{aligned} \tag{2.17}$$

2.2.4.2 Calculation of $\check{n}_F$

The average number of transmitted bits when the FRAG fails to be received is equal to:

$$\check{n}_F = \sum_{k=1}^{P_{\max}} \delta_k = w_{P_{\max}}. \tag{2.18}$$

2.2.4.3 Calculation of $\hat{n}_S^P$

For the PBS, since the fragments are independent, the average number of transmitted bits is equal to N times the average number of transmitted bits per FRAG:

$$\hat{n}_S^P = N \cdot \hat{n}_F. \tag{2.19}$$

2.2.4.4 Calculation of $\hat{n}_S^S$

For the SBS, we have to list all the events corresponding to a successful DSDU packet transmission. Each event is composed of N successful transmissions provided the total number of transmissions does not overshoot C. Let us define $\hat{E}_S^S(\underline{i}) := \{F_1(i_1)$ and $F_2(i_2)$ and $\cdots$ $F_N(i_N)\}$ this event, with $\underline{i} = [i_1, i_2, \cdots, i_N]$, and with the constraint $\sum_{k=1}^{N} i_k \leq C$. Then the average number of transmitted bits in this case is given by:

$$\hat{n}_S^S = \sum_{\underline{i} \in \mathcal{R}} e(\underline{i}) \cdot \Pr\{\hat{E}_S^S(\underline{i})|\text{ IP packet OK}\}, \tag{2.20}$$

where $e(\underline{i})$ is the number of bits corresponding with the event $\hat{E}_S^S(\underline{i})$, and $\mathcal{R}$ is the set of integer vectors $\underline{i}$ belonging to the set

$$\mathcal{R} = \{\underline{i} = [i_1, i_2, \cdots, i_N] \in \mathbb{N}_*^N | \sum_{k=1}^{N} i_k \leq C\}. \tag{2.21}$$

One can easily deduce that $e(\underline{i})$ is equal to the total number of bits transmitted for the N fragments, and is equal to $e(\underline{i}) = \sum_{k=1}^{N} w_{i_k}$. Since events $F_j(k)$ are independent, we have $\Pr\{\hat{E}(\underline{i})|\text{ IP packet KO}\} = (1-\Pi_S^S)^{-1} \cdot \prod_{k=1}^{N} \Pr\{F_k(i_k)\}$. Since $\Pr\{F_i(k)\} = p_1(k)$, (2.20) takes the following form:

$$\hat{n}_S^S = (1-\Pi_S^S)^{-1} \cdot \sum_{\underline{i}\in\mathcal{R}} \Big[\prod_{k=1}^{N} p_1(i_k) \cdot \sum_{k=1}^{N} w_{i_k} \Big]. \tag{2.22}$$

2.2.4.5 Calculation of $\check{n}_S^P$

The events associated with $\check{n}_S^P$ are events where at least one FRAG is not received. Let us note the event $G_{\underline{i}}(j) := \{\bar{F}_1 \text{ and } \bar{F}_2 \text{ and } \ldots \text{ and } \bar{F}_j \text{ and } F_{j+1}(i_1) \text{ and } \ldots \text{ and } F_N(i_{N-j})\}$, where $\bar{F}_k$ is the event FRAG $\#k$ is not received. The event $G_{\underline{i}}(j)$, $1 \le j \le N$, corresponds to the case where the first j fragments are not received and the $(N-j)$ remaining ones are successfully transmitted. For a given j, the indices $\{i_1, i_2, \ldots, i_{N-j}\}$ belong to the set

$$\mathcal{S}_j := \{\underline{i} = [i_1, i_2, \cdots, i_{N-j}] \in \mathbb{N}_*^{N-j} | \forall k, i_k \le P_{\max}\}. \tag{2.23}$$

Noticing that the number of possible events that have j FRAGs KO is $\binom{N}{j}$, we can write:

$$\begin{aligned} \check{n}_S^P &= \sum_{j=1}^{N} \binom{N}{j} \sum_{\underline{i}\in\mathcal{S}_j} g_{\underline{i}}(j) \Pr\{G_{\underline{i}}(j)|\text{ IP packet KO}\} \\ &= (\Pi_S^P)^{-1} \sum_{j=1}^{N} \binom{N}{j} \sum_{\underline{i}\in\mathcal{S}_j} g_{\underline{i}}(j) \Pr\{G_{\underline{i}}(j)\}, \end{aligned} \tag{2.24}$$

where $g_{\underline{i}}(j)$ is the number of transmitted bits associated with the event $G_{\underline{i}}(j)$.

When a FRAG is not received, the corresponding number of transmitted bits is equal to $w_{P_{\max}}$ bits. Thus we can write

$$g_{\underline{i}}(j) = j \cdot w_{P_{\max}} + r_{\underline{i}}(N-j) \tag{2.25}$$

where $r_{\underline{i}}(N-j) := \sum_{k=1}^{N-j} w_{i_k}$ is the number of transmitted bits for the $(N-j)$ remaining received FRAGs. For $j = N$, all the FRAGs are in error and thus we have $\Pr\{G_{\underline{i}}(j)\} = (\Pi_F)^N$ with $g_{\underline{i}}(j) = N \cdot w_{P_{\max}}$. For $j < N$, the probability that j FRAGs are not received is equal to $(\Pi_F)^j$ and the probability that the $(N-j)$ remaining FRAGs are received is equal to $\prod_{k=1}^{N-j} p_1(i_k)$, which gives:

$$\Pr\{G_{\underline{i}}(j)\} = \begin{cases} (\Pi_F)^j \cdot \prod_{k=1}^{N-j} p_1(i_k) & \text{for} \quad 1 \le j < N, \\ (\Pi_F)^N & \text{for} \quad j = N. \end{cases} \tag{2.26}$$

Putting (2.25) and (2.26) into (2.24) leads to:

$$\begin{aligned} \check{n}_S^P = \frac{1}{\Pi_S^P} \Bigg[&\sum_{j=1}^{N-1} \binom{N}{j} (\Pi_F)^j \sum_{\underline{i}\in\mathcal{S}_j} \prod_{k=1}^{N-j} p_1(i_k)(j \cdot w_{P_{\max}} + r_{\underline{i}}(N-j)) \\ &+ N \cdot w_{P_{\max}} \cdot (\Pi_F)^N \Bigg]. \end{aligned} \tag{2.27}$$

We now show that (2.27) can be simplified into a different form that is more compact and less complex to compute. First, let us rewrite (2.27) as the sum of 3 terms $\check{n}_S^P = A_1 + A_2 + A_3$ with:

$$A_1 = N w_{P_{\max}} \frac{(\Pi_F)^N}{\Pi_S^P}, \tag{2.28}$$

$$A_2 = \frac{w_{P_{\max}}}{\Pi_S^P} \sum_{j=1}^{N-1} \binom{N}{j} j(\Pi_F)^j \sum_{\underline{i} \in \mathcal{S}_j} \prod_{k=1}^{N-j} p_1(i_k), \tag{2.29}$$

$$A_3 = \frac{1}{\Pi_S^P} \sum_{j=1}^{N-1} \binom{N}{j} (\Pi_F)^j \sum_{\underline{i} \in \mathcal{S}_j} \left(\prod_{k=1}^{N-j} p_1(i_k) \cdot \sum_{k=1}^{N-j} w_{i_k} \right), \tag{2.30}$$

$$\tag{2.31}$$

where A_2 and A_3 can be simplified. In order to do so, we need first to establish the 2 following properties:

Property 1 *The following equality holds:*

$$\sum_{\underline{i} \in \mathcal{S}_j} \prod_{k=1}^{N-j} p_1(i_k) = (1 - \Pi_F)^{(N-j)}.$$

Proof : We can write:

$$\begin{aligned} \left(\sum_{i=1}^{P_{\max}} p_1(i) \right)^{(N-j)} &= \sum_{i_1=1}^{P_{\max}} p_1(i_1) \times \sum_{i_2=1}^{P_{\max}} p_1(i_2) \times \ldots \times \sum_{i_{N-j}=1}^{P_{\max}} p_1(i_{N-j}), \\ &= \sum_{i_1, i_2, \ldots, i_{N-j} \in [1, P_{\max}]} p_1(i_1) p_1(i_2) \ldots p_1(i_{N-j}), \end{aligned} \tag{2.32}$$

where the set of indices of the multiple summation in (2.32) can be identified as the set defined in (2.23), which allows to first establish that:

$$\left(\sum_{i=1}^{P_{\max}} p_1(i) \right)^{(N-j)} = \sum_{\underline{i} \in \mathcal{S}_j} \prod_{k=1}^{N-j} p_1(i_k). \tag{2.33}$$

Then, from (2.1) we deduce that $\sum_{i=1}^{P_{\max}} p_1(i) = 1 - \Pi_F$, which inserted into (2.33) completes the proof. ■

Property 2 *The following equality holds:*

$$\sum_{\underline{i} \in \mathcal{S}_j} \left(\prod_{k=1}^{N-j} p_1(i_k) \cdot \sum_{k=1}^{N-j} w_{i_k} \right) = (N-j)(1 - \Pi_F)^{(N-j-1)} \sum_{k=1}^{P_{max}} w_k p_1(k).$$

Proof : The Property 2 can be proved by recurrence. First let us introduce the following notation

$$X_{N-j} := \sum_{\underline{i} \in \mathcal{S}_j} \left(\prod_{k=1}^{N-j} p_1(i_k) \cdot \sum_{k=1}^{N-j} w_{i_k} \right), \tag{2.34}$$

with N fixed.

For $j = N-1$, we have $\mathcal{S}_{N-1} = \{i = 1, \ldots, P_{\max}\}$ and we simply deduce that:

$$X_1 = \sum_{k=1}^{P_{\max}} w_k p_1(k). \tag{2.35}$$

For $j = N-2$, we have $\mathcal{S}_{N-2} = \{\underline{i} = [i_1, i_2] \in \mathbb{N}_*^2 | \forall k, i_k \leq P_{\max}\}$. Let us express X_2 as the summation of $P_{\max}$ terms, $X_2 = y_1 + y_2 + \cdots + y_{P_{\max}}$ where y_m collects all the terms with $i_1 = m$. Thus $p_1(m)$ can be put in factor and we can write $y_m = p_1(m) z_m$ with $z_m = \sum_{k=1}^{P_{\max}} (w_m + w_k) p_1(k)$. We can split the summation into 2 sums, $z_m = w_m \sum_{k=1}^{P_{\max}} p_1(k) + \sum_{k=1}^{P_{\max}} w_k p_1(k)$ where the first term can be identified as $w_m(1 - \Pi_F)$ using (2.1) and the last term as X_1. We thus deduce $y_m = p_1(m)(w_m(1 - \Pi_F) + X_1)$. Replacing y_m into X_2 and computing the summation gives $X_2 = \sum_{k=1}^{P_{\max}} p_1(k)(w_k(1 - \Pi_F) + X_1)$ or equivalently:

$$X_2 = 2(1 - \Pi_F) X_1. \tag{2.36}$$

For $j = N-3$, we have $\mathcal{S}_{N-3} = \{\underline{i} = [i_1, i_2, i_3] \in \mathbb{N}_*^3 | \forall k, i_k \leq P_{\max}\}$. Let us express X_3 as the summation of $P_{\max}$ terms, $X_3 = y_1 + y_2 + \cdots + y_{P_{\max}}$ where y_m collects all the terms with $i_1 = m$. Thus $p_1(m)$ can be put in factor and we can write $y_m = p_1(m) z_m$, with $z_m = \sum_{\underline{i} \in \mathcal{S}_{N-2}} (w_m + w_{i_1} + w_{i_2}) p_1(i_1) p_1(i_2)$. We can split the summation into 2 sums, $z_m = w_m \sum_{\underline{i} \in \mathcal{S}_{N-2}} p_1(i_1) p_1(i_2) + \sum_{\underline{i} \in \mathcal{S}_{N-2}} (w_{i_1} + w_{i_2}) p_1(i_1) p_1(i_2)$ where the first term can be identified as $w_m(1 - \Pi_F)^2$ using Property 1 for $N - j = 2$, and the last term as X_2. We thus deduce $y_m = p_1(m)(w_m(1 - \Pi_F)^2 + X_2)$. Replacing y_m into X_3 gives $X_3 = \sum_{k=1}^{P_{\max}} p_1(k)(w_k(1 - \Pi_F)^2 + X_2)$ or equivalently:

$$X_3 = (1 - \Pi_F)^2 X_1 + (1 - \Pi_F) X_2. \tag{2.37}$$

The previous reasoning can be applied recursively by grouping terms with $i_1 = m$, factorizing those terms by $p_1(m)$, and identifying X_{m-1} and X_1 when computing X_m. This leads to the general term expression

$$X_m = (1 - \Pi_F)^{m-1} X_1 + (1 - \Pi_F) X_{m-1}. \tag{2.38}$$

Applying (2.38) recursively, we finally get:

$$X_m = m(1 - \Pi_F)^{m-1} X_1, \tag{2.39}$$

which completes the proof using (2.35) and for $m = N - j$. ■

From Property 1, we can write $A_2 = \frac{w_{P_{\max}}}{\Pi_S^P} \sum_{j=1}^{N-1} \binom{N}{j} j (\Pi_F)^j (1 - \Pi_F)^{(N-j)}$ which drops down to:

$$A_2 = N w_{P_{\max}} \frac{\Pi_F}{\Pi_S^P} (1 - (\Pi_F)^{(N-1)}). \tag{2.40}$$

From Property 2 we have:

$$A_3 = \frac{1}{\Pi_S^P} \sum_{j=1}^{N-1} \binom{N}{j} (\Pi_F)^j (N - j)(1 - \Pi_F)^{N-j-1} X_1,$$

which after some calculations leads to:

$$A_3 = N(1 - \frac{\Pi_F}{\Pi_S^P}) \frac{X_1}{(1 - \Pi_F)}. \tag{2.41}$$

From (2.35) we can rewrite (2.17) as $\hat{n}_F = X_1/(1-\Pi_F)$, and (2.41) then takes the following form:

$$A_3 = N(1 - \frac{\Pi_F}{\Pi_S^P})\hat{n}_F. \tag{2.42}$$

From (2.28) and (2.40) we have $A_1 + A_2 = N w_{P_{\max}} \frac{(\Pi_F)}{\Pi_S^P}$. Thus from (2.18) and (2.42), we finally end up with the final expression:

$$\check{n}_S^P = N\frac{\Pi_F}{\Pi_S^P}(\check{n}_F - \hat{n}_F) + N\hat{n}_F. \tag{2.43}$$

This later expression is much more compact than (2.27) and it is also less complex to program since we have succeeded to remove the summation over the sets $\mathcal{S}_j$ in (2.27) to simple summations. Indeed, for a given j, the summation over $\mathcal{S}_j$ in (2.27) requires $P_{\max}^{N-j}$ additions which leads to a total complexity of $\frac{P_{\max}^N - P_{\max}}{P_{\max}-1}$ additions when j goes from 1 to $(N-1)$, whereas the summations in (2.43) have a complexity of $P_{\max}$.

2.2.4.6 Calculation of $\check{n}_S^S$

For the SBS, we have to take into account that the N FRAGs are sharing the same global transmission credit and cannot be considered independently. Thus, the event the IP does not go through corresponds to the event $\check{E}_S^S := \bigcup_{j=1}^N H(j)$ with:

- $H(1)$:={FRAG #1 consumes all the credit},
- $H(j)$:={{FRAG #1 OK, and FRAG #2 OK, and ... and FRAG #j consumes the remaining credit}, for $j < 1 < N$},
- $H(N)$:={FRAG #1 OK, and FRAG #2 OK, and ... and FRAG #$(N-1)$ OK and FRAG #N KO with the remaining credit},

the rationale for considering 3 different cases will appear later since the computation of these terms are different. Going further in characterizing the previous events, we find

- $H(1) = \{\bar{F}_1\}$,
- $H(j) = \bigcup_{\underline{i} \in \mathcal{T}_j} H_{\underline{i}}(j)$, for $1 < j < N$,
- $H(N) = \bigcup_{\underline{i} \in \mathcal{T}_N} H_{\underline{i}}(N)$,

where $H_{\underline{i}}(j)$, for $1 < j < N$, is the event defined as $H_{\underline{i}}(j)$:={$F_1(i_1)$ and $F_2(i_2)$ and ... and $F_{j-1}(i_{j-1})$ and $\tilde{F}_j(\underline{i})$} where $\tilde{F}_j(\underline{i})$ is the event that FRAG #j consumes the remaining credit whenever it is successfully received or not, $H_{\underline{i}}(N)$:= {$F_1(i_1)$ and $F_2(i_2)$ and ... and $F_{N-1}(i_{N-1})$ and $\bar{F}_N$}, and the set $\mathcal{T}_j$ is defined as:

$$\mathcal{T}_j := \{\underline{i} = [i_1, i_2, \cdots, i_{j-1}] \in \mathbb{N}_*^{j-1} | \sum_{k=1}^{j-1} i_k < C\}. \tag{2.44}$$

Fig. 2.3 illustrates for $N = 4$ and $C = 5$ all the cases contributing to $\check{n}_S^S$ where cross means KO and circle OK. The superposition of the cross and circle represents cases where both KO and OK have to be considered.

We can know give the expression for the mean number of transmitted bits when an IP packet does not go through:

$$\begin{aligned}\check{n}_S^S &= h(1)\Pr\{H(1)|\text{ IP packet KO}\} + \sum_{j=2}^{N}\sum_{\underline{i}\in\mathcal{T}_j} h_{\underline{i}}(j)\Pr\{H_{\underline{i}}(j)|\text{ IP packet KO}\} \\ &= (\Pi_S^S)^{-1}\left(h(1)\Pr\{H(1)\} + \sum_{j=2}^{N}\sum_{\underline{i}\in\mathcal{T}_j} h_{\underline{i}}(j)\Pr\{H_{\underline{i}}(j)\}\right). \qquad (2.45)\end{aligned}$$

where $h_{\underline{i}}(j)$ (resp. $h(1)$) is the number of transmitted bits associated with the event $H_{\underline{i}}(j)$ (resp. $H(1)$).

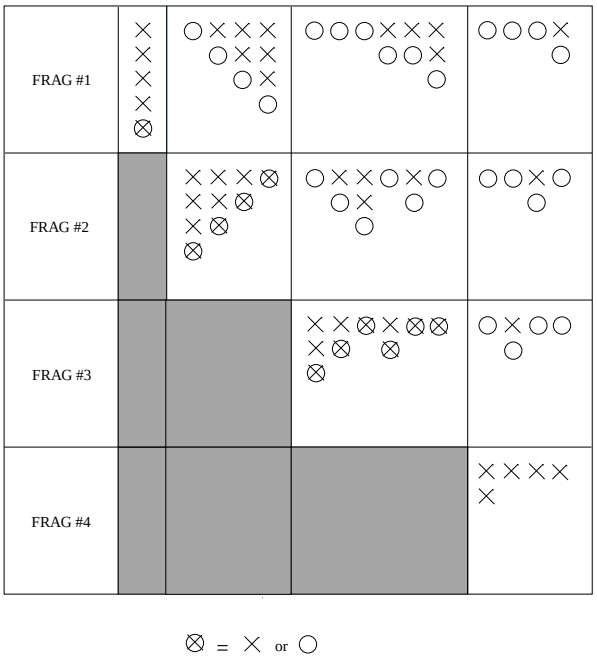

Figure 2.3: Crosses and circles for $\check{n}_S^S$ case, $N = 4$, $C = 5$.

We now present the computation of $h(1)$, $\Pr\{H(1)\}$, $h_{\underline{i}}(j)$ and $\Pr\{H_{\underline{i}}(j)\}$ for the different cases:

- $j = 1$
  The probability $\Pr\{H(1)\} = \Pr\{\bar{F}_1\}$ is the probability that the first FRAG has consumed the C credits whenever it is successfully received or not (if it is transmitted in C transmissions, there is no longer credits to transmit the remaining packets and thus the IP packet transmission fails). Thus we have to consider both cases the FRAG is OK and KO which correspond respectively to the probability $(1-\pi_{C-1})\prod_{k=0}^{C-2}\pi_k$ and $\prod_{k=0}^{C-1}\pi_k$. Introducing the following notation:

$$q(i) := \begin{cases} 1 & \text{for } i = 0, \\ \prod_{k=0}^{i-1}\pi_k & \text{for } i > 0. \end{cases}$$

  we deduce that $\Pr\{H(1)\} = (1-\pi_{C-1})q(C-1) + q(C)$ which drops down to

$$\Pr\{H(1)\} = q(C-1), \qquad (2.46)$$

  and that the corresponding number of transmitted bits $h(1) = w_C$ for both cases.

- $1 < j < N$
  In this case, the event $H_{\underline{i}}(j)$ corresponds to the successful transmission of the first $(j-1)$ FRAGs, followed by the transmission of FRAG $\#j$ which consumes the remaining credits whenever it is successfully received or not. The remaining credit left after the $(j-1)$ FRAGs transmission is equal to $C - m_{\underline{i}}(j-1)$ where $m_{\underline{i}}(j) := \sum_{k=1}^{j} i_k$. Thus, following the same reasoning as for $j = 1$, we have $\Pr\{\tilde{F}_j(\underline{i})\}\} = q(C - m_{\underline{i}}(j-1) - 1)$. Since FRAG transmissions are independent and remembering that $\Pr\{F_i(k)\} = p_1(k)$, we thus easily deduce that:

$$\Pr\{H_{\underline{i}}(j)\} = \sum_{\underline{i} \in \mathcal{T}_j} \prod_{k=1}^{j-1} p_1(i_k) q(C - m_{\underline{i}}(j-1) - 1), \tag{2.47}$$

  and that the corresponding number of transmitted bits $h_{\underline{i}}(j) = r_{\underline{i}}(j-1) + w_{C-m_{\underline{i}}(j-1)}$.

- $j = N$
  In that case, FRAG $\#N$ is necessarily KO at the C-th transmission, which gives $\Pr\{\bar{F}_N\} = q(C - m_{\underline{i}}(N-1))$. We then deduce that:

$$\Pr\{H(N)\} = \sum_{\underline{i} \in \mathcal{T}_N} \prod_{k=1}^{N-1} p_1(i_k) q(C - m_{\underline{i}}(N-1)), \tag{2.48}$$

  and that the corresponding number of transmitted bits is equal to $h_{\underline{i}}(N) = r_{\underline{i}}(N-1) + w_{C-m_{\underline{i}}(N-1)}$.

Thus, from (2.46), (2.47), (2.48), and the corresponding number of transmitted bits, (2.45) takes the following expression:

$$\begin{aligned} \check{n}_S^S = (\Pi_S^S)^{-1} \Bigg[& q(C-1) \cdot w_C \\ & + \sum_{j=2}^{N-1} \sum_{\underline{i} \in \mathcal{T}_j} \prod_{k=1}^{j-1} p_1(i_k) q(C - m_{\underline{i}}(j-1) - 1)(r_{\underline{i}}(j-1) + w_{C-m_{\underline{i}}(j-1)}) \\ & + \sum_{\underline{i} \in \mathcal{T}_N} \prod_{k=1}^{N-1} p_1(i_k) q(C - m_{\underline{i}}(N-1))(r_{\underline{i}}(N-1) + w_{C-m_{\underline{i}}(N-1)}) \Bigg]. \end{aligned} \tag{2.49}$$

Conversely to the $\check{n}_S^P$ expression, we have found no way of simplifying the expression (2.49) in order to put it in a more compact form with a reduced computation complexity. This could be part of a future research track.

2.2.4.7 Expressions summary

We give here a summary of the efficiency expressions from the expressions established above for η_F and η_S^P since they can take a reasonable compact form. However this is not possible (or useless) for η_S^S since $\check{n}_S^S$ is huge and has no compact form.

When putting (2.17) and (2.18) into (2.16) and using (2.1), we get:

$$\eta_F = \frac{L_F(1 - \Pi_F)}{\Pi_F \check{n}_F + (1 - \Pi_F)\hat{n}_F}, \tag{2.50}$$

$$= \frac{L_F \sum_{k=1}^{P_{\max}} p_1(k)}{(1 - \sum_{k=1}^{P_{\max}} p_1(k)) w_{P_{\max}} + \sum_{k=1}^{P_{\max}} w_k p_1(k)}. \tag{2.51}$$

When putting (2.19) and (2.43) into (2.16) and using (2.2), we get:

$$\begin{aligned}\eta_S^P &= \frac{NL_F(1-\Pi_F)^N}{N\Pi_F(\check{n}_F-\hat{n}_F)+N\Pi_S^P\hat{n}_F+(1-\Pi_S^P)N\hat{n}_F},\\ &= \frac{L_F(1-\Pi_F)^N}{\Pi_F\check{n}_F+(1-\Pi_F)\hat{n}_F}.\end{aligned} \tag{2.52}$$

We can identify η_F (2.50) in (2.52) which allows to write:

$$\eta_S^P = \eta_F\cdot(1-\Pi_F)^{(N-1)}, \tag{2.53}$$

$$= \eta_F\cdot\left(\sum_{k=1}^{P_{\max}} p_1(k)\right)^{(N-1)}, \tag{2.54}$$

and shows that η_S^P can be simply deduced from η_F by a multiplicative coefficient which depends on the PER at the MAC level. We verify that when setting $N=1$ into the previous equations we obtain, as expected, the efficiency at the MAC level. We can also deduce that since $(1-\Pi_F)\leq 1$, we have $\eta_S^P \leq \eta_F$ which sounds natural and traduces the price to pay to transmit N FRAGs.

2.3 Particular cases performance derivations

In this section we give some particular cases performance results deduced from the general expressions derived in the previous section. We first compute the expressions in the particular case when the DPDUs have the same length. We then proceed by the ARQ or type I HARQ case for which all the π_j are equal. We then discuss about the independence of the efficiency *vs.* the persistence. We finally end by proving that the PER in the SBS is always smaller than the PBS one when $C \geq NP_{\max}$.

2.3.1 Constant length DPDUs

The performance metric expressions given in the previous Section 2.2 were derived with no assumption on the length of the transmitted packets. More specifically, those expressions allow to take account retransmission schemes for which the length of the packets may vary from one (re)transmission to another as it is generally the case for the Incremental Redundancy scheme. If we look at the expressions in the previous sections of Section 2.2, we conclude that the length of the transmitted packets matters only for the efficiency, the PER and delay-jitter being independent of this parameter.

In this section we derive specific expressions for the particular case of constant length packets that are more simple than for the general case and can be applied to HARQ-CC (RM2-RP3) or to HARQ-IR (RM3-RP2) when the redundancy increments have the same length. When the DPDUs have all the same length we have $\forall k, \delta_k = L_{\text{DPDU}}$, which implies $w_k = k\cdot L_{\text{DPDU}}$ and thus $r_{\underline{i}}(n) = \sum_{k=1}^{n} w_k = L_{\text{DPDU}}\sum_{k=1}^{n} i_k$, or equivalently $r_{\underline{i}}(n) = L_{\text{DPDU}}\cdot m_{\underline{i}}(n)$.

We first start by giving the new expressions for $\check{n}$ and $\hat{n}$ and then we will include those terms in the equations derived in the previous section. Since η_S^P can be easily deduced from η_F using (2.53), we just need to work on $\hat{n}_F$ and $\check{n}_F$ for both. For η_S^S we need to work on $\hat{n}_S^S$ and $\check{n}_S^S$. In order to distinguish the variable expressions derived for

this special assumption from the others, we will add up a dot on top of them (*e.g.* $\hat{\dot{n}}_S^P$ denotes the equivalence of $\hat{n}_S^P$ for constant length packets.)

It is straightforward to get:

$$\hat{\dot{n}}_F = \frac{L_{\text{DPDU}}}{1-\Pi_F} \sum_{k=1}^{P_{\max}} k \cdot p_1(k), \tag{2.55}$$

$$\check{\dot{n}}_F = L_{\text{DPDU}} \cdot P_{\max}, \tag{2.56}$$

One can notice that, from (2.7), we can identify the mean delay at MAC level in (2.55) which can be rewritten as:

$$\hat{\dot{n}}_F = L_{\text{DPDU}} \cdot \bar{n}_F. \tag{2.57}$$

which confirms that when the DPDUs have the same length, the average amount of transmitted bits when the transmission is successful is proportional to the average number of transmitted packets. Thus, putting (2.55), (2.56), (2.57) into (2.50) gives the 2 equivalent expressions:

$$\dot{\eta}_F = \frac{L_F(1-\Pi_F)}{L_{\text{DPDU}}(\Pi_F \cdot P_{\max} + (1-\Pi_F)\bar{n}_F)}, \tag{2.58}$$

$$\dot{\eta}_F = \frac{L_F \sum_{k=1}^{P_{\max}} p_1(k)}{L_{\text{DPDU}} \left((1 - \sum_{k=1}^{P_{\max}} p_1(k)) P_{\max} + \sum_{k=1}^{P_{\max}} k p_1(k) \right)}, \tag{2.59}$$

that boil down to:

$$\dot{\eta}_F = \frac{\rho(1-\Pi_F)}{\Pi_F \cdot P_{\max} + (1-\Pi_F)\bar{n}_F}, \tag{2.60}$$

$$\dot{\eta}_F = \frac{\rho \sum_{k=1}^{P_{\max}} p_1(k)}{(1 - \sum_{k=1}^{P_{\max}} p_1(k)) P_{\max} + \sum_{k=1}^{P_{\max}} k p_1(k)} \tag{2.61}$$

since we remind $\rho = L_F / L_{\text{DPDU}}$.

The SBS case is a little bit more complex. Let us start with $\hat{n}_S^S$. Since $\sum_{k=1}^{N} w_{i_k} = L_{\text{DPDU}} \sum_{k=1}^{N} i_k$, (2.22) becomes:

$$\hat{\dot{n}}_S^S = \frac{L_{\text{DPDU}}}{(1-\Pi_S^S)} \cdot \sum_{\underline{i} \in \mathcal{R}} \Big[\prod_{k=1}^{N} p_1(i_k) \cdot \sum_{k=1}^{N} i_k \Big]. \tag{2.62}$$

Then, the idea is to split the summation over the set $\mathcal{R}$ in subsets for which $\sum_{k=1}^{N} i_k$ is constant. This is actually the property of the set $\mathcal{Q}_{n,k}^S$ defined in (2.5) which verifies $\mathcal{R} = \bigcup_{j=N}^{C} \mathcal{Q}_{N,j}^S$. We can rewrite (2.22) as:

$$\hat{\dot{n}}_S^S = \frac{L_{\text{DPDU}}}{(1-\Pi_S^S)} \cdot \sum_{j=N}^{C} j \sum_{\underline{i} \in \mathcal{Q}_{N,j}^S} \prod_{k=1}^{N} p_1(i_k), \tag{2.63}$$

in which the last term can be identified using (2.4) as $p_N^S(j)$, and thus can be rewritten as:

$$\hat{\dot{n}}_S^S = \frac{L_{\text{DPDU}}}{(1-\Pi_S^S)} \cdot \sum_{j=N}^{C} j \cdot p_N^S(j). \tag{2.64}$$

From the definition of the average delay given by (2.10), we can identify $\bar{n}_S^S$ in (2.64) which takes the following form:

$$\hat{n}_S^S = L_{\text{DPDU}} \cdot \bar{n}_S^S, \tag{2.65}$$

which confirms that when the DPDUs have the same length, the average amount of transmitted bits when the transmission is successful is proportional to the average number of transmitted packets.

For $\check{n}_S^S$, we use the same idea as for $\hat{n}_S^S$ by splitting the summation over the set $\mathcal{T}_j$ on subsets for which $\sum_{k=1}^{n} i_k$ is constant using $\mathcal{Q}_{k,n}^S$ set. Noticing that $\mathcal{T}_j = \bigcup_{\ell=j-1}^{C-1} \mathcal{Q}_{j-1,\ell}^S$, we can rewrite (2.49) as:

$$\begin{aligned}\check{n}_S^S = (\Pi_S^S)^{-1} \Bigg[& \sum_{\ell=N-1}^{C-1} \sum_{\underline{i} \in \mathcal{Q}_{N-1,\ell}^S} \prod_{k=1}^{N-1} p_1(i_k) q(C - m_{\underline{i}}(j-1)) (r_{\underline{i}}(N-1) + w_{C-m_{\underline{i}}(N-1)}) \\ & + \sum_{j=2}^{N-1} \sum_{\ell=j-1}^{C-1} \sum_{\underline{i} \in \mathcal{Q}_{j-1,\ell}^S} \prod_{k=1}^{j-1} p_1(i_k) q(C - m_{\underline{i}}(j-1) - 1) (r_{\underline{i}}(j-1) + w_{C-m_{\underline{i}}(j-1)}) \\ & + q(C-1) \cdot w_C \Bigg]. \end{aligned} \tag{2.66}$$

Applying the constant packet length assumption leads to $m_{\underline{i}}(j-1) = \sum_{k=1}^{j-1} i_k = \ell$, $r_{\underline{i}}(j-1) = \ell L_{\text{DPDU}}$, and thus $r_{\underline{i}}(j-1) + w_{C-m_{\underline{i}}(j-1)} = C \cdot L_{\text{DPDU}}$, which enable us to simplify (2.66) into:

$$\begin{aligned}\check{n}_S^S = \frac{C \cdot L_{\text{DPDU}}}{\Pi_S^S} \Bigg[& q(C-1) + \sum_{j=2}^{N-1} \sum_{\ell=j-1}^{C-1} \sum_{\underline{i} \in \mathcal{Q}_{j-1,\ell}^S} \prod_{k=1}^{j-1} p_1(i_k) q(C - \ell - 1) \\ & + \sum_{\ell=N-1}^{C-1} \sum_{\underline{i} \in \mathcal{Q}_{N-1,\ell}^S} \prod_{k=1}^{N-1} p_1(i_k) q(C - \ell) \Bigg]. \end{aligned} \tag{2.67}$$

To go further, we need to note that the 3 terms inside the square brackets can be identified as (2.46), (2.47) and (2.48) when the split over the set $\mathcal{Q}_{k,n}^S$ is done. Thus, we can rewrite (2.67) as:

$$\check{n}_S^S = \frac{C \cdot L_{\text{DPDU}}}{\Pi_S^S} \left[\Pr\{H_{\underline{i}}(1)\} + \sum_{j=2}^{N-1} \Pr\{H_{\underline{i}}(j)\} + \Pr\{H_{\underline{i}}(N)\} \right], \tag{2.68}$$

where the terms into the square bracket can be identified, from the $H_{\underline{i}}(j)$ definition, as the SDU PER for the SBS, and thus $\sum_{j=1}^{N} \Pr\{H_{\underline{i}}(j)\} = \Pi_S^S$, which finally gives:

$$\check{n}_S^S = C \cdot L_{\text{DPDU}}. \tag{2.69}$$

This result was predictable since, when a SDU is not transmitted, the corresponding number of transmissions used is equal to the maximum credit C and thus in the case of constant length packets, we naturally find (2.69).

Putting (2.65) and (2.69) into (2.16) gives the 2 equivalent expressions:

$$\dot{\eta}_S^S = \frac{N\rho(1-\Pi_S^S)}{C\cdot\Pi_S^S + (1-\Pi_S^S)\bar{n}_S^S}, \tag{2.70}$$

$$\dot{\eta}_S^S = \frac{N\rho\cdot\sum_{k=N}^{C} p_N^S(k)}{C(1-\sum_{k=N}^{C} p_N^S(k)) + \sum_{k=N}^{C} k p_N^S(k)}. \tag{2.71}$$

2.3.2 Type I HARQ performance expressions

In the previous sections we have shown that we can derive any retransmission scheme performance as a function of $p_1(k)$ in an explicit manner or through $p_n^S(k)$. Those expressions cannot be simplified further, except for the type I HARQ scheme with memoryless receiver processing (RP1) for which $\forall j, \pi_j = \pi_0$ and $P_{\max} = P+1$. This includes also the ARQ scheme which can be seen as a special case of type I HARQ with no channel coding at the transmitter side. We present here the closed form expressions for the different performance metric for both PBS and SBS as a function of π_0 which do not include the summation operator. Note that those expressions can be used for any kind of propagation channel and channel coding since it is captured in π_0. Moreover, except for the PER Π_F and Π_S^P and the efficiency η_F^P at the PBS level, all the expressions presented in this section have apparently never been published.

For type I HARQ schemes, (1.1) takes the simple form:

$$p_1(k) = (1-\pi_0)\pi_0^{(k-1)}. \tag{2.72}$$

When putting (2.72) into (2.4), it comes:

$$p_n^S(k) = \binom{k-1}{n-1}(1-\pi_0)^n\pi_0^{(k-n)}, \tag{2.73}$$

that was given in [3] Eq. (2).

From (2.72) we compute:

$$\sum_{k=1}^{P+1} p_1(k) = 1-\pi_0^{(P+1)}, \tag{2.74}$$

$$\sum_{k=1}^{P+1} k p_1(k) = \frac{1-(P+2)\pi_0^{(P+1)}+(P+1)\pi_0^{(P+2)}}{1-\pi_0}, \tag{2.75}$$

that will be useful in the following sections.

2.3.2.1 Packet error rate

From (2.1) and (2.74) we find:

$$\Pi_F = \pi_0^{(P+1)}, \tag{2.76}$$

which was given in [24] Eq. (15). Replacing Π_F by (2.76) in (2.2) gives:

$$\Pi_S^P = 1-(1-\pi_0^{(P+1)})^N, \tag{2.77}$$

that was given in [3] Eq. (1) and in [47] Eq. (2).

In the SBS, putting (2.73) into (2.3) brings (using *Mathematica*):

$$\Pi_S^S = \frac{\pi_0^{(C-N+1)}\Gamma(C+1)(1-\pi_0)^N F_{C+1}^{C-N+2}(1,\pi_0)}{\Gamma(N)}, \tag{2.78}$$

with $F_x^y(w,z) := {}_2F_1(w,x,y,z)/\Gamma(y)$, where $\Gamma(x)$ and ${}_2F_1(w,x,y,z)$ are respectively the conventional gamma and hypergeometric functions [52]. Note that (2.78) is an explicit version of Eq. (3) in [3].

2.3.2.2 Delay

Putting (2.72) into (2.7) and using (2.9) we get:

$$\bar{n}_S^P = N\left(P_{\max} + \frac{1}{1-\pi_0} + \frac{P_{\max}}{\pi_0^{P_{\max}}-1}\right), \tag{2.79}$$

from which we can easily deduce $\bar{n}_F$ by setting $N=1$.

For $\bar{n}_S^S$, replacing Π_S^S and $p_N^S(k)$ respectively by (2.3) in (2.73) into (2.10) leads to:

$$\bar{n}_S^S = \frac{\sum_{k=N}^{C}\binom{k-1}{N-1}k\pi_0^k}{\sum_{k=N}^{C}\binom{k-1}{N-1}\pi_0^k}. \tag{2.80}$$

The summations in (2.80) can be computed using *Mathematica* which gives the following equivalent expression:

$$\bar{n}_S^S = \frac{(1-\pi_0)^{(N+1)}\pi_0^{(C+1)}\Gamma(C+2)(F_{C+1}^{C-N+2}(1,\pi_0)+\pi_0 F_{C+2}^{C-N+3}(2,\pi_0)) - N\Gamma(N)\pi_0^N}{(1-\pi_0)((1-\pi 0)^N\pi_0^{(C+1)}\Gamma(C+1)F_{C+1}^{C-N+2}(1,\pi_0) - \pi_0^N\Gamma(N))}, \tag{2.81}$$

that depends only on C, N, and π_0.

Those equations allow to compute the asymptotic values of the delay, *e.g.* at high and low SNR. At high SNR we have $\lim_{\mathrm{SNR}\to+\infty}\pi_0 = 0$, thus replacing $\pi_0 = 0$ in (2.79) leads immediately to $\bar{n}_S^P = N$, which sounds coherent since when the channel is error free, the N IP packets are transmitted in N DPDUs.

For the low SNR we have $\lim_{\mathrm{SNR}\to 0}\pi_0 = 1$. Putting $\pi_0 = 1$ in (2.79) is not possible since it leads to a division by zero. Rearranging (2.79) as a fraction still leads to a 0 over 0 indetermination. We thus can resort to the l'Hospital's rule by deriving two times the numerator and denominator (since there is still an indetermination after the first derivation), and we get:

$$\lim_{\pi_0\to 1}\bar{n}_S^P = N\left.\frac{\frac{\partial^2}{\partial\pi_0^2}\left(\pi_0^{P_{\max}}(1-\pi_0)P_{\max}+\pi_0^{P_{\max}}-1\right)}{\frac{\partial^2}{\partial^2\pi_0^2}\left((1-\pi_0)(\pi_0^{P_{\max}}-1)\right)}\right|_{\pi_0=1} = \frac{N}{2}(P_{\max}+1). \tag{2.82}$$

For the SBS at high SNR, the Taylor expansion of (2.80) around $\pi_0 = 0$ gives $\bar{n}_S^S = N + O(\pi_0)$ that leads to $\lim_{\pi_0\to 0}\bar{n}_S^S = N$, which is again an expected result. At high SNR, replacing $\pi_0 = 1$ into (2.80) gives $\bar{n}_S^S|_{\pi_0=1} = (\sum_{k=N}^{C}\binom{k-1}{N-1}k)/(\sum_{k=N}^{C}\binom{k-1}{N-1})$ for which we have found no way to simplify. Thanks to *Mathematica* we find:

$$\lim_{\pi_0\to 1}\bar{n}_S^S = \frac{N(C+1)}{N+1}. \tag{2.83}$$

We have reported in Tab. 2.1 the different values for the instantaneous and average delays at the NET level for PBS and SBS. We can see that whereas the minimum value for the instantaneous and average delays are equal, it is not true for the maximum values. The maximum of the average delays are smaller than those of the instantaneous ones. This means that at very low SNR, there are still some IP packets that are transmitted without consuming all the allowed credits. Those figures also allow to compare the

	$\min[n_S^x(i)]$	$\max[n_S^x(i)]$	$\bar{n}_S^x$	$\min \bar{n}_S^x$	$\max \bar{n}_S^x$
PBS	N	$NP_{\max}$	(2.79)	N	$\frac{N(P_{\max}+1)}{2}$
SBS	N	C	(2.80) or (2.81)	N	$\frac{N(C+1)}{N+1}$

Table 2.1: Min and max values for instantaneous and average delay at the NET level for PBS and SBS.

maximum average delay between both strategies. We find that the SBS average delay is larger when $C > \frac{N(P_{\max}+1)+P_{\max}-1}{2}$. In particular, it shows that when the number of maximum retransmission per DSDU is equal in both strategies, (*e.g.* $C = NP_{\max}$), the SBS has always a larger delay than the PBS one (which will be illustrated by the simulations in Chapter 3).

2.3.2.3 Jitter

Putting (2.76), (2.79) into (2.12) and using (2.13) we get:

$$\sigma^2_{n_S^P} = N\left(\frac{\pi_0 + \pi_0^{(2P+3)} - \pi_0^{(P+1)}((P+1)^2 + \pi_0^2(\pi_0+1)^2 - 2\pi_0 P(P+2))}{(\pi_0-1)^2(\pi_0^{(P+1)}-1)^2}\right), \quad (2.84)$$

from which we can easily deduce $\sigma^2_{n_F}$ by setting $N = 1$.

Putting (2.78), (2.81) into (2.14) we obtain:

$$\sigma^2_{n_S^S} = \frac{N(N+\pi_0)\Gamma(N)}{A(1-\pi_0)^2} - \frac{(1-\pi_0)^N\pi_0^{(C-N+1)}(B_1+B_2)}{A} - \frac{D}{E}, \quad (2.85)$$

with

$$\begin{aligned}
A &= \Gamma(N) - (1-\pi_0)^N\pi_0^{(C-N+1)}\Gamma(C+1)F_{C+1}^{C-N+2}(1,\pi_0),\\
B_1 &= C^3\Gamma(C)F_{C+1}^{C-N+2}(1,\pi_0) + (2C+1)\Gamma(C+1)F_{C+1}^{C-N+2}(1,\pi_0),\\
B_2 &= \pi_0((2C+3)\Gamma(C+2)F_{C+2}^{C-N+3}(2,\pi_0) + 2\pi_0\Gamma(C+3)F_{C+3}^{C-N+4}(3,\pi_0)),\\
D &= ((1-\pi_0)^N(\pi_0-1)\Gamma(C+2)\pi_0^{(C+1)}(F_{C+1}^{C-N+2}(1,\pi_0) + \pi_0 F_{C+2}^{C-N+3}(2,\pi_0))\\
&\quad + N\Gamma(N)\pi_0^N)^2,\\
E &= (1-\pi_0)^2((1-\pi_0)^N\pi_0^{(C+1)}\Gamma(C+1)F_{C+1}^{C-N+2}(1,\pi_0) - \pi_0^N\Gamma(N))^2. \quad (2.86)
\end{aligned}$$

2.3.2.4 Efficiency

Putting (2.74) and (2.75) into (2.61) gives the conventional result:

$$\eta_F = \rho(1-\pi_0) \quad (2.87)$$

which shows that the efficiency is independent of the persistence. We can then easily deduce η_S^P by putting (2.87) and (2.74) into (2.54):

$$\eta_S^P = \rho(1-\pi_0)(1-\pi_0^{(P+1)})^{(N-1)}. \tag{2.88}$$

For η_S^S, putting (2.78), (2.81), and (2.69) into (2.16), we get:

$$\eta_S^S = \rho\frac{A}{B+D}, \tag{2.89}$$

with

$$\begin{aligned}
A &= N(\pi_0-1)((1-\pi_0)^N\pi_0^{(C+1)}\Gamma(C+1)F_{C+1}^{C-N+2}(1,\pi_0)-\pi_0^N\Gamma(N)),\\
B &= (1-\pi_0)^N(\pi_0-1)\pi_0^{(C+1)}((\Gamma(C+2)-C\Gamma(C+1))F_{C+1}^{C-N+2}(1,\pi_0),\\
D &= (1-\pi_0)^N(\pi_0-1)((\pi_0^{(C+1)}\pi_0\Gamma(C+2)F_{C+2}^{C-N+3}(2,\pi_0))+N\Gamma(N)\pi_0^N).
\end{aligned}$$

2.3.3 On the independence of the efficiency *vs.* persistence

From (2.87) we have deduced that the efficiency at the MAC level is independent of the persistence in the case of ARQ or type I HARQ (thus equal length DPDUs). It is worth noting that this result was presented in [24] in a different way by saying "the throughput only depends on N", which with our notations is equivalent to say "the efficiency only depends on π_0", which seems to us less instructive than stressing on the independence with regard to the persistence.

One can wonder if this property still holds at the NET level or for other retransmission schemes. Using the previous derivations, we prove the following results:

- In the case of IR-HARQ with flush memory at the receiver side (RM3-RP2), when the persistence is a multiple integer of t_0 (we remind that t_0 corresponds to the number of DPDUs per FRAG), the efficiency at the MAC level is independent of the persistence, even for unequal DPDU lengths.
- The efficiency at the NET level is dependent of the persistence for the PBS.

For the first result, let us pose $P_{\max} = a \cdot t_0$ with $a \in \mathbb{N}_*$. We can rewrite $z = \sum_{k=1}^{P_{\max}} w_k p_1(k)$ equivalently as $z = \sum_{i=0}^{a-1}\sum_{k=1}^{t_0} w_{it_0+k}p_1(it_0+k)$. Since the RP uses the flush memory, we have $w_{it_0+k} = iw_{t_0}+w_k$ and $p_1(it_0+k) = p_1(k)(q(t_0))^i$. Inserting the two last relations into z gives $z = z_1 + z_2$ with $z_1 = \sum_{i=0}^{a-1} iw_{t_0}(q(t_0))^i\sum_{k=1}^{t_0}p_1(k)$ and $z_2 = \sum_{i=0}^{a-1}(q(t_0))^i\sum_{k=1}^{t_0}w_k p_1(k)$. From (2.46) and (1.1) it is useful to notice that $q(k-1)-q(k) = p_1(k)$ and thus $\sum_{k=1}^{j}p_1(k) = 1-q(j)$. This later expression allows to write $z_1 = w_{t_0}(1-q(t_0))\sum_{i=0}^{a-1} i(q(t_0))^i$ which transforms into $z_1 = \frac{w_{t_0}q(t_0)}{1-q(t_0)}[1 - a(q(t_0))^{(a-1)} + (a-1)(q(t_0))^a]$. We also get $z_2 = \frac{1-(q(t_0))^a}{1-q(t_0)}\sum_{k=1}^{t_0}w_k p_1(k)$. Inserting z expression into (2.51) leads after simplifications to:

$$\eta_F = \frac{L_F(1-q(t_0))}{w_{t_0}q(t_0)+\sum_{k=1}^{t_0}w_k p_1(k)}, \tag{2.90}$$

which is not a function of $P_{\max}$ and thus is independent of the persistence.

For the second result, it can be simply deduced from (2.54) which shows that although η_F is independent of the persistence, η_S^P is not since $P_{\max}$ appears in the expression. Although we do not have such straightforward proof, the dependency of η_S^S with regard to the persistence at the NET level can be observed on the simulation results in Chapter 3.

2.3.4 PBS and SBS packet error rate comparison

In this section we prove that, when $C \geq NP_{\max}$ the SBS strategy always outperforms the PBS one in terms of PER, *i.e.* $\Pi_S^S < \Pi_S^P$. The proof uses the $p_n^P(k)$ and $p_n^S(k)$ definition based on the $p_1(k)$ that we remind here for the sake of clarity:

$$p_n^x(k) = \sum_{\underline{i} \in \mathcal{Q}_{n,k}^x} \prod_{m=1}^{n} p_1(i_m) \tag{2.91}$$

where the sets $\mathcal{Q}_{n,k}^x$ are defined as:

$$\mathcal{Q}_{n,k}^P = \{\underline{i} = [i_1, i_2, \cdots, i_n] \in \mathbb{N}_*^n \mid \sum_{m=1}^{n} i_m = k, i_k \leq P_{\max}\}, \tag{2.92}$$

$$\mathcal{Q}_{n,k}^S = \{\underline{i} = [i_1, i_2, \cdots, i_n] \in \mathbb{N}_*^n \mid \sum_{m=1}^{n} i_m = k\}. \tag{2.93}$$

In order to prove this result, we first establish the three following propositions.

Proposition 1 $n \leq k < P_{max} + n \Longrightarrow \mathcal{Q}_{n,k}^P = \mathcal{Q}_{n,k}^S$.

Proof : As long as the maximum number of transmitted DPDUs for the PBS, $P_{\max}$, is not reached, both PBS and SBS have the same set of vectors. When k increases, the maximum number of transmitted DPDUs needed per fragment increases too. The highest value for which both strategies give the same set of vectors arises when one fragment consumes the maximum of transmitted DPDUs and the $n-1$ remaining fragments are received in one DPDU. The $n-1$ transmissions consuming $n-1$ DPDUs, there are $k-n+1$ remaining DPDUs to transmit the last fragment. Thus, both strategies will have the same set until $P_{\max} = k - n + 1$ which concludes the proof. ■

Proposition 2 $P_{max} + n \leq k \leq nP_{max} \Longrightarrow \mathcal{Q}_{n,k}^P \subset \mathcal{Q}_{n,k}^S$.

Proof : We put $k = P_{\max} + n + m$ with $0 \leq m \leq (n-1)P_{\max} - n$. Then, let us take the case when the $n-1$ fragments are received in one DPDU and the last fragment is received in $k-(n-1) = P_{\max} + m + 1$ DPDUs. This event occurs for SBS due to the global credit management but never occurs for the PBS since the maximum number of transmitted DPDUs exceeds $P_{\max}$. Thus $\mathcal{Q}_{n,k}^P$ is strictly included in $\mathcal{Q}_{n,k}^S$. ■

We can illustrate the Proposition 2 by taking the previous example $n = 3$, $k = 7$, with $P_{\max} = 3$. In order to compute the set $\mathcal{Q}_{7,3}^P$, it suffices to take vectors from $\mathcal{Q}_{7,3}^S$ and to remove all the vectors with components greater than $P_{\max}$, which gives $\mathcal{Q}_{7,3}^P$ ={(3,3,1),(3,1,3),(1,3,3),(2,2,3),(2,3,2),(3,2,2)}.

Proposition 3 *For the PBS,* $k > nP_{max}, \Longrightarrow p_n^P(k) = 0$.

Proof : For the PBS, the maximum number of transmitted DPDUs per fragment is $P_{\max}$, and thus the maximum of transmitted DPDUs for n fragments is equal to $nP_{\max}$. As a consequence, when $k > nP_{\max}$, there is no way to deliver the n fragments in exactly k transmissions, which is equivalent to $p_n^P(k) = 0$, for $k > nP_{\max}$ and thus $\mathcal{Q}_{n,k}^P = \emptyset$. ■

Based upon the previous propositions, we can then state the following theorem:

Theorem 1 *For any retransmission scheme (RM and RP) and for $C \geq NP_{max}$, the packet error rate of the cross-layer strategy SBS at the NET level is strictly smaller than the PBS one (i.e. $\Pi_S^S < \Pi_S^P$).*

Proof : Let us note $\Delta := \Pi_S^P - \Pi_S^S$. For $C \geq NP_{\max}$, from (2.2) and (2.3), we get:

$$\Delta = \sum_{k=N}^{C} (p_N^S(k) - p_N^P(k)). \tag{2.94}$$

According to Proposition 1 and Proposition 3, (2.94) can be rewritten:

$$\Delta = \sum_{k=P_{\max}+N}^{NP_{\max}} (p_N^S(k) - p_N^P(k)) + \sum_{k=NP_{\max}+1}^{C} p_N^S(k). \tag{2.95}$$

Noting $\bar{\mathcal{Q}}_{k,n} := \mathcal{Q}_{n,k}^S \setminus \{\mathcal{Q}_{n,k}^S \cap \mathcal{Q}_{n,k}^P\}$, then according to Proposition 2, (2.95) becomes:

$$\Delta = \sum_{k=P_{\max}+N}^{NP_{\max}} \sum_{\underline{\mathbf{q}} \in \bar{\mathcal{Q}}_{k,N}} p_1(q_1)p_1(q_2)\cdots p_1(q_n) + \sum_{k=NP_{\max}+1}^{C} p_N^S(k). \tag{2.96}$$

Thus, since $\bar{\mathcal{Q}}_{k,n} \neq \emptyset$ according to Proposition 2 and since $p_1(k)$ and $p_N^S(k)$ are strictly positive quantities, we deduce that $\Delta > 0$ which concludes the proof. ■

Theorem 1 tells that, when $C \geq NP_{\max}$, it is always beneficial from a packet error point of view at the NET level to use the cross-layer scheme SBS rather than the PBS one. It means in particular that, for an equal number of maximal retransmission credit ($C = NP_{\max}$), the cross-layer strategy always outperforms the conventional one. When $C < NP_{\max}$, calculation shows that we cannot derive any theoretical results. We will see in Chapter 3 that we can play on the credit C for the SBS in order to adjust performance and balance the trade off throughput/delay, even when $C < NP_{\max}$.

2.4 Performance computation procedure

In this section we summarize the different steps that are needed to compute the different performance given by the equations derived in Section 2.2 and Section 2.3. The performance expressions show that they may involve the following quantities: π_j, $p_1(k)$, $q(k)$, and $p_n^S(k)$, and that they all can be computed from the π_j. The $p_1(k)$ and $q(k)$ can easily be computed from the π_j using (1.1) and (2.46) whereas an efficient computation of $p_n^S(k)$ requires some tricks.

We first discuss the estimation of the π_j and then how to compute $p_n^S(k)$. Then we list the different steps that enable to compute the performance from the π_j.

2.4.1 Estimation of π_j

In order to compute the performance, we assume that the π_j are available. They can be obtained from either simulations (estimated PER) or bounds (computed PER) as it is done for example in [24] for convolutional codes using tangential sphere bound technique, in the very simple case of type I HARQ with no memory at the receiver side. In our work we will base the performance study on estimated PER, the PER

computation being specific to the channel coding scheme (convolutional, turbo, LDPC, ...) and a whole research topic by itself.

The way to obtain the estimated π_j is then to perform a simulation in order to generate the events related to the corresponding RM-RP scheme at the MAC level only. Thus, although it may look close to do the programming of the HARQ protocol, it is less complex. Indeed, the simulation is done only at the MAC level and there is no necessity to consider the protocol at the NET level (thus avoiding the segmentation and the reassembly process) to compute the π_j, even if they allow to compute the performance of the PBS and the SBS at the NET level.

We can state that a given set of π_j associated with L_F, $P_{\max}$ and C, allows to compute all the performance (MAC and NET levels for PBS and SBS) for any N, transmission credit per FRAG $\leq P_{\max}$ and global credit $\leq C$. Furthermore, when the RP does not involve accumulating processing (such as Chase Combining type of processing) at the receiver side, the sufficient set of π_j (see step 2 in Section 2.4.3) allows to compute all the metric performance for any N, transmission credit per FRAG and global credit. In the contrary, new π_j need to be computed when $P_{\max}$ or C increases. When the FRAG length L_F varies, the π_j have also to be re-evaluated in any case since the channel coding performance usually varies with the packet length. The need of evaluating new π_j can be seen as drawback and a limitation of our method. Further work should be dedicated to bound derivation for π_j estimation.

We now illustrate the complexity of computing closed form expression for the π_j which is mainly due to the conditioning with respect to the previous transmissions. Let us consider here the ARQ case for which there is no channel coding scheme. In the very simple case of a binary shift keying modulation in an additive complex Gaussian circular noise channel with variance $N_0/2$ per real dimension, and $L_F = 1$ (*i.e.* a packet of one bit) with combining at the receiver side (RM1-RP3), we get:

$$\pi_1 = \frac{1}{2\pi\sigma^2\pi_0} \int_{-\infty}^{-\sqrt{E_b}} \int_{-\infty}^{-v-2\sqrt{E_b}} \exp -\frac{u^2+v^2}{N_0}\, dudv, \tag{2.97}$$

where E_b is the energy per bit and $\pi_0 = \frac{1}{2}\,\mathrm{erfc}\left\{\sqrt{\frac{E_b}{N_0}}\right\}$ is the conventional error probability in a Gaussian channel. Thus, one can understand that for more complex parameters this approach is not tractable.

2.4.2 Estimation of $p_n^S(k)$

The computation of the $p_n^S(k)$ can be done using the definition (2.91) but experience shows that it requires a huge complexity and is thus not tractable when k and n become large since we need to browse the whole set $\mathbb{N}_*^n$ to determine the elements of $\mathcal{Q}_{n,k}^S$.

We can show that we can actually compute recursively $p_n^S(k)$ in n. Indeed, $p_n^S(k)$ can be expressed using $p_{n-1}^S(k)$, noticing that the probability that n DPDUs are received in k transmissions is equal to the probability that $(n-1)$ DPDUs are received in $(n-1)$ transmissions along with 1 DPDU received in $(k-n+1)$ transmissions (*i.e.* $p_1(k-n+1)p_{n-1}^S(n-1)$) or, that $(n-1)$ DPDUs are received in n transmissions along with 1 DPDU received in $(k-n)$ transmissions (*i.e.* $p_1(k-n)p_{n-1}^S(n)$) or, ... or, that $(n-1)$ DPDUs are received in $(k-1)$ transmissions along with one DPDU received in

1 transmission (*i.e.* $p_1(1)p_{n-1}^S(k-1)$), which can be summarized by:

$$p_n^S(k) = \sum_{i=1}^{k-n+1} p_1(i)p_{n-1}^S(k-i), \tag{2.98}$$

where $p_1(i)$ is given by (1.1) and $p_1^S(i)$ is interpreted as $p_1(i)$. This algorithm has a polynomial complexity in *multiply and add* operations equal to $(n-1)(k-n+1)(k-n+2)/2$ that can be upper bounded by $n(k-n)^2$. Obviously, (2.98) is dedicated to the SBS strategy, since all the performance closed-form expressions in the PBS one can be given as function of $p_1(k)$. But we can note that an equivalent form of (2.98) was proposed in [3] for the PBS (although not needed) referring [53] were similar problem was addressed in the context of queuing networks.

2.4.3 Performance computation in 4 steps

We now give the way to compute the closed-form expressions following 4 successive different steps.

Step 1

The first step consists in evaluating the set $\{\pi_j\}_{j=0}^M$ by simulation, where M is defined as the maximum index for which the elementary PERs are needed. M depends on the RM-RP and cross-layer schemes whereas the π_j values depend on the channel model too. The different values taken by M for the different RM-RP and cross-layer schemes (in ascending RP order for convenience) are given by:

- RP1, $\forall$ RM: since the receiver processing is memoryless, we have $\pi_j = \pi_0$, thus $M = 0$ for both PBS and SBS.
- RM3-RP2: since the receiver flushes periodically the memory after receiving the maximal number of incremental redundancy blocks, we have $\pi_j = \pi_{(j \mod t_0)}$, thus $M = t_0 - 1$, for both PBS and SBS.
- RM1-RP3/RM2-RP3: since the CC processing accumulates the packets up to the maximum retransmission credit, we have $M = P$ for PBS and $M = C - 1$ for SBS.
- RM3-RP3: for the same reason as above, we get $M = (P+1)t_0 - 1$ for the PBS and $M = C - 1$ for the SBS.

Step 2

The second step consists in computing the $p_1(k)$ from the π_j estimated in step 1, using (1.1). The parameter to be determined here is the range of the index k.

- RM1, RM3 and RM4: for the PBS, it is easy to deduce from the different equations (*e.g.* (2.1)) that we have $1 \leq k \leq P_{\max}$, which gives $1 \leq k \leq P+1$ for RM1 and RM2, and $1 \leq k \leq (P+1)t_0$ for RM3 ; for the SBS, following the same reasoning on (2.3), and identifying the $p_1(k)$ needed in (2.4), we obtain $1 \leq k \leq C - N + 1$ for all RMs.

Step 3

The third step concerns only the SBS. It consists in computing $\{p_N^S(k)\}_{k=N}^C$ applying (2.98) with the $p_1(k)$ computed in step 2. This can be easily obtained from a double recursion in n and k as follows:

```
for n = 2 to N
  for k = n to C-N+n
    compute p_n^S(k) using (2.98)
  end for
end for
```

where all the intermediate value of $p_n^S(k)$ should be kept in memory.

Step 4

The last step deals with the performance computation. For the PBS, use the $p_1(k)$ computed in step 2 into (2.1) and (2.2) for packet error rates, into (2.7), and (2.9) for delays, into (2.12) and (2.13) for jitter, and into (2.51) and (2.54) for efficiency. For the SBS, use the $p_N^S(k)$ computed in step 3 into (2.3) for packet error rate, into (2.10) for delay, into (2.14) for jitter and into (2.49) for efficiency.

2.5 Conclusion

In this chapter, we have presented a general framework to derive theoretical performance (PER, delay, jitter, efficiency) of any retransmission scheme, ARQ, Hybrid ARQ, in a cross-layer mode or not, at the fragment or NET level, including finite transmission credit per FRAG (connected to the persistence P). Specifically, we have given closed form expressions for the efficiency in the case of unequal DPDU lengths that are apparently new. We have then used those expressions to derive particular cases when the length of the DPDUs is constant, and for type I HARQ. We have also discussed the independence of the efficiency with regard to the persistence parameter, and we then have established a theorem which says that the cross-layer strategy taking into account the IP nature of the fragments is always better in terms of PER than the conventional approach regardless of the retransmission scheme.

Tab. 2.2, Tab. 2.3, Tab. 2.4 and Tab. 2.5 summarize the closed-form expressions that have been derived in this chapter. Two kinds of expressions can be found in those tables compared to those of Chapter 1 (Tab. 1.2 to Tab. 1.5). First, the new expressions that replace the crosses. Second, new expressions that replace the existing ones when found to be simpler. Note that the expressions are referred by their number due to their big size and the lack of space. Furthermore, note that we have actually covered 100% of the missing formulas in the initial tables which makes 17 new expressions, and found 7 new simplified ones.

SBS (NET)		
	ARQ	HARQ
PER	(2.78)	(2.3)
Efficiency, UDL	Not applicable	(2.16) with (2.22) and (2.49)
Efficiency, EDL	(2.89)	(2.71)
Delay	(2.81)	(2.10)
Jitter	(2.85)	(2.14)

Table 2.2: Proposed theoretical expressions for the SBS at the NET level. (Unequal DPDU Length: UDL, Equal DPDU Length: EDL.)

PBS (NET)		
	ARQ	HARQ
PER	(2.77)	(2.2)
Efficiency, UDL	Not applicable	(2.54)
Efficiency, EDL	(2.88)	(2.61)
Delay	(2.79)	(2.9)
Jitter	(2.84)	(2.13)

Table 2.3: Proposed theoretical expressions for the PBS at NET level. (Unequal DPDU Length: UDL, Equal DPDU Length: EDL.)

PBS (MAC)		
	ARQ	HARQ
PER	(2.76)	Same as in Tab. 1.4
Efficiency, UDL	Not applicable	(2.51)
Efficiency, EDL	(2.87)	Same as in Tab. 1.4
Delay	(2.79)	Same as in Tab. 1.4
Jitter	(2.84)	(2.12)

Table 2.4: Proposed theoretical expressions at MAC level. (Unequal DPDU Length: UDL, Equal DPDU Length: EDL.)

	ARQ	HARQ
$p_N^S(k)$	Same as in Tab. 1.5	(2.91)
$p_N^P(k)$	Same as in Tab. 1.5	(2.91)
$p_1(k)$	Same as in Tab. 1.5	Same as in Tab. 1.5

Table 2.5: Proposed theoretical expressions for $p_N(k)$

Chapter 3

Performance Analysis

3.1 Introduction

The goal of this chapter is twofold: on the one hand, we check the accuracy of the theoretical expressions established in Chapter 2. On the other hand, based on the numerical evaluations of the theoretical expressions, we focus on the influence of the different HARQ mechanisms and cross-layer strategies as well as of the modulation and the propagation channel on all the performance metrics. This analysis enables us to provide some insights and guidelines about system design according to QoS requirements.

We remind that all the simulations are done under the same assumptions as stated in Section 2.2 for the closed-form derivations. In addition, we have selected five HARQ mechanisms defined in Chapter 1 to perform the simulations: ARQ, CC-ARQ, CC-HARQ, IR-HARQ, and IC-HARQ. The IR-HARQ and IC-HARQ are implemented with the RCPC codes introduced in [7]. Unless it is otherwise stated, the DPDUs have the same length, the modulation is QPSK, and the propagation channel is the Gaussian channel. We recall that the input/output relationship in a Gaussian channel is $y = s+b$ where s represents the transmitted symbols, b is the complex-valued, circular, Gaussian noise with zero mean and variance N_0 per real dimension, and y is the discrete-time received signal. In the sequel, we often analyze the performance with respect to the SNR defined as

$$\text{SNR} = \frac{\text{Average received power per symbol (around carrier frequency)}}{\text{Noise power (around carrier frequency)}}. \tag{3.1}$$

After some simple algebraic manipulations, the SNR in the case of Gaussian channel takes the following form

$$\text{SNR}_{\text{Gaussian}} = \frac{E_s}{2N_0} \tag{3.2}$$

where E_s is the average transmitted power per symbol in baseband.

As a second propagation channel, we also focus on the Rayleigh for which the input/output relationship is $y = hs + b$ where h is a complex-valued, circular, Gaussian process with zero mean and variance $\sigma_h^2/2$ per real dimension, and where s, b, and y are defined as in Gaussian channel. The channel is assumed slow-fading, that means that h is varying at each DPDU transmission and not at each transmitted symbol. The average SNR is given by

$$\text{SNR}_{\text{Rayleigh}} = \frac{\sigma_h^2 E_s}{2N_0}. \tag{3.3}$$

This chapter is organized as follows: in Section 3.2, the theoretical expressions are compared to the simulation results. Thanks to numerical evaluation of the theoretical expressions, we compare the HARQ mechanisms in Section 3.3 and the cross-layer strategies in Section 3.4 respectively. In Section 3.5 (resp. Section 3.6), we inspect the influence of the modulation (resp. an other propagation channel such as Rayleigh channel) on the HARQ mechanisms and cross-layer strategies. In Section 3.7 concluding remarks about the choice on relevant HARQ mechanisms and cross-layer strategies are drawn.

3.2 Empirical and theoretical performance comparison

This section is dedicated to the comparison between the theoretical performance and the empirical one for different configurations, that means for different types of HARQ, different cross-layer strategies, and different layers. All the metrics (PER, efficiency, delay, jitter) are considered and plotted versus the SNR. In all the figures, the lines represent the theory while the symbols represent the simulation.

In Fig. 3.1 and Fig. 3.2, we display the PER, efficiency, delay, and jitter with respect to the SNR for IR-HARQ and CC-HARQ respectively for both cross-layer strategies at the NET layer. We observe a perfect agreement between the theory and the simulation.

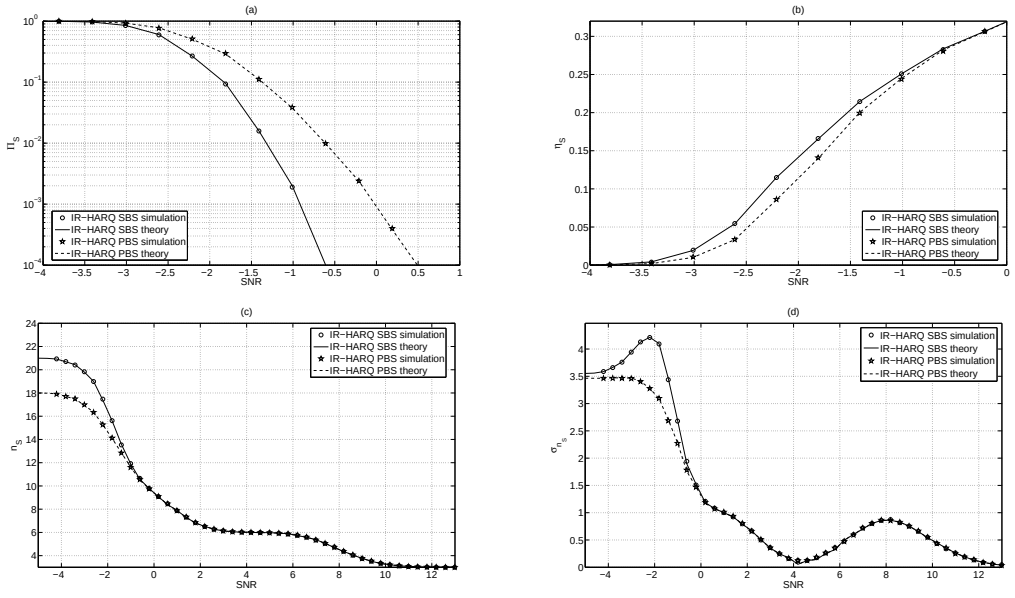

Figure 3.1: Theoretical and empirical performance for IR-HARQ at NET layer with SBS and PBS strategies: (a) PER, (b) efficiency, (c) delay, and (d) jitter. ($N = 3$, $L_F = 120$ bits, $R_0 = 1/4$, $C = 24$ (SBS), and $P_{\max} = 8$ (PBS).)

We also have to inspect the validity of our analysis at the MAC layer. Therefore, in Fig. 3.3, we plot the four metrics with respect to the SNR for ARQ and IC-HARQ (which were not considered in Fig. 3.1 and Fig. 3.2) at the MAC layer. As expected, there is a perfect matching between the numerical evaluation of the theoretical expressions and the performance of simulated system.

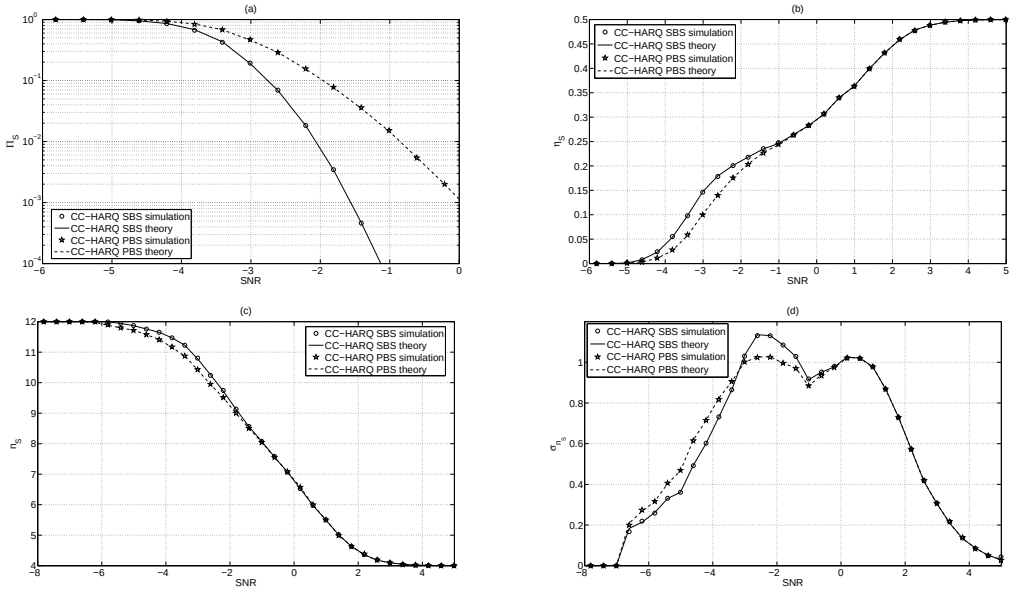

Figure 3.2: Theoretical and empirical performance for CC-HARQ at NET layer with SBS and PBS strategies: (a) PER, (b) efficiency, (c) delay, and (d) jitter. ($N = 4$, $L_F = 60$ bits, $R_0 = 1/2$, $C = 12$ (SBS), and $P_{\max} = 3$ (PBS).)

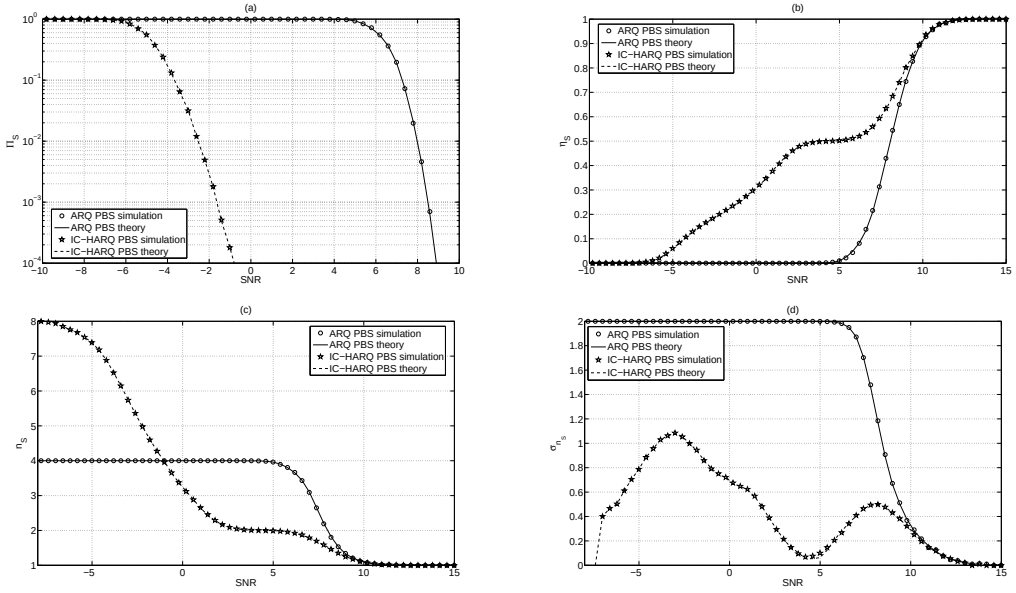

Figure 3.3: Theoretical and empirical performance for ARQ and IC-HARQ at MAC layer: (a) PER, (b) efficiency, (c) delay, and (d) jitter. ($L_F = 120$ bits, $P_{\max} = 7$ for the ARQ, and $R_0 = 1/2$ and $P_{\max} = 8$ for the IC-HARQ.)

The figures presented above in the case of HARQ mechanisms using incremental redundancy have been obtained with DPDUs of constant length. However, in Chapter 2, we have highlighted the fact that, if the PER, delay, and jitter theoretical formulas are not affected by the size of the DPDUs, the efficiency analytical expression takes an other form. We then represent on Fig. 3.4 the empirical and theoretical efficiencies in

the case of IR-HARQ with DPDUs of different lengths, for several sets of parameters (different C for SBS and $P_{\max}$ for PBS, and different N for both). This figure enables us to check the validity of the general efficiency closed-form expressions summarized in Tab. 2.2 and Tab. 2.3.

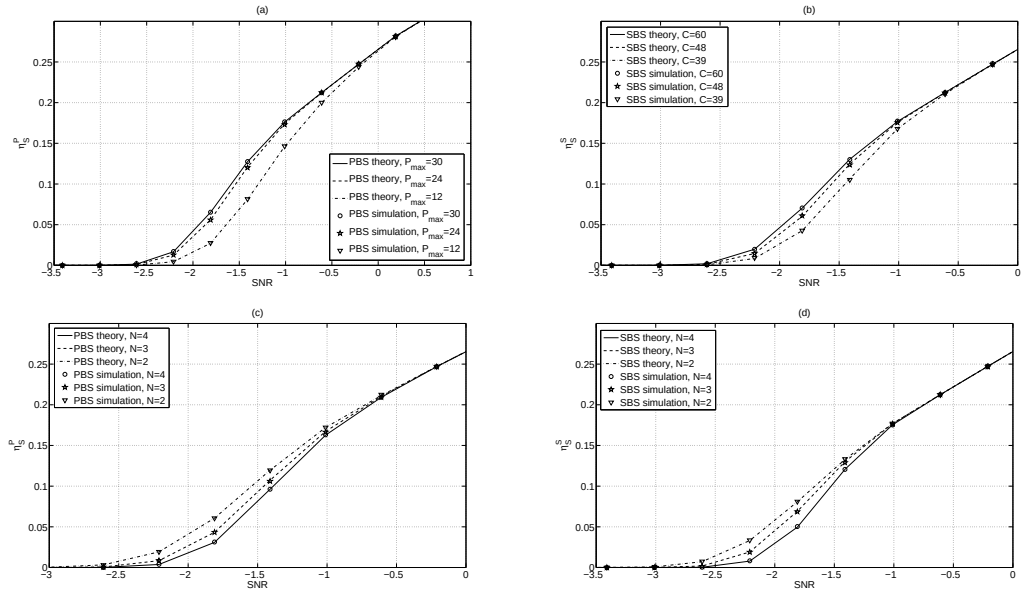

Figure 3.4: Theoretical and empirical efficiency for IR-HARQ at the IP layer: (a) PBS efficiency with $P_{\max}$ varying and $N = 3$, (b) SBS efficiency with C varying and $N = 3$, (c) PBS efficiency with N varying and $P_{\max} = 18$, and (d) SBS efficiency with N varying and $C = 64$. ($L_F = 320$ bits, $R_0 = 1/4$, $t_0 = 6$, with the different puncturing rates: 1, 4/5, 2/3, 1/2, 4/11, and 1/4.)

Until now, the comparison has been done with the same modulation (QPSK) and the same propagation channel (the Gaussian one). However, it is necessary to check also the accuracy of our expressions in the context of modulation with higher efficiency and more realistic wireless channel such as the Rayleigh one. Consequently, the four metrics are represented with respect to the SNR in Fig. 3.5 when 16QAM is employed for the IR-HARQ at the NET layer with both cross-layer strategies. Once again, we observe that our theoretical derivations are accurate even when the modulation is modified.

In Fig. 3.6, we plot the four metrics with respect to the SNR in the context of a slow-fading Rayleigh channel for IR-HARQ at the NET layer with both cross-layer strategies. We check that our theoretical expressions remain valid although we have changed significantly the nature of the propagation channel.

Other configurations (BPSK modulation, ARQ mechanism, *etc.*) have been tested and we always have observed a perfect matching between theory and practice. As the number of configurations that can be considered is infinite and as we have presented one case for each major kind (different HARQ schemes, SBS and PBS, different layers, modulations, and channels), we decide to now concentrate on the comparison of the main HARQ mechanisms and/or cross-layer strategies studied throughout this thesis. From now on, we only focus on the numerical evaluation of the theoretical expressions to do the proposed comparisons in next sections.

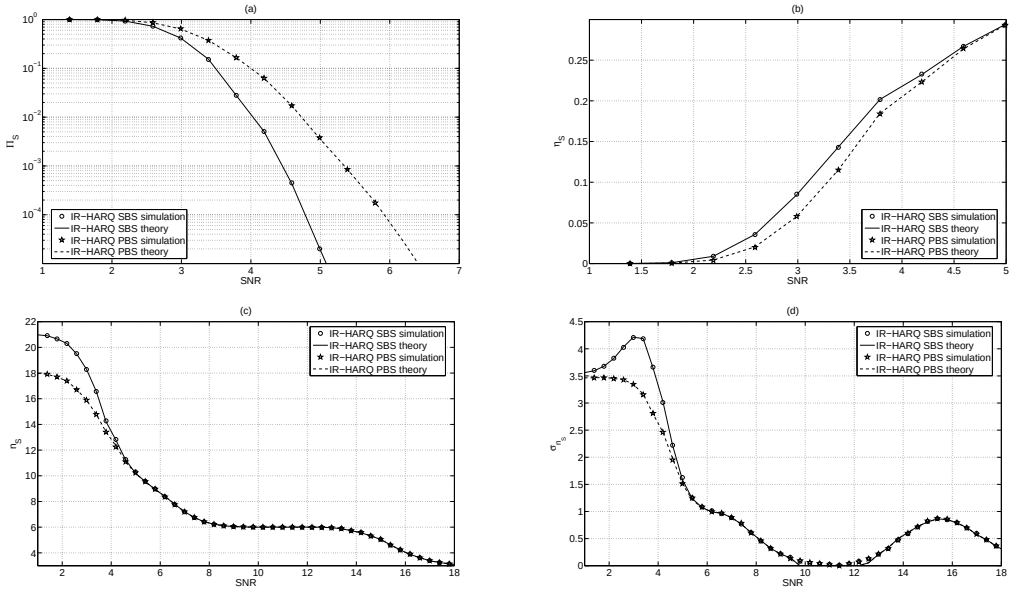

Figure 3.5: Theoretical and empirical performance in 16QAM context for IR-HARQ at the NET layer with SBS and PBS strategies: (a) PER, (b) efficiency, (c) delay, and (d) jitter. ($N = 3$, $L_F = 240$ bits, $R_0 = 1/4$, $C = 24$ (SBS), and $P_{\max} = 8$ (PBS).)

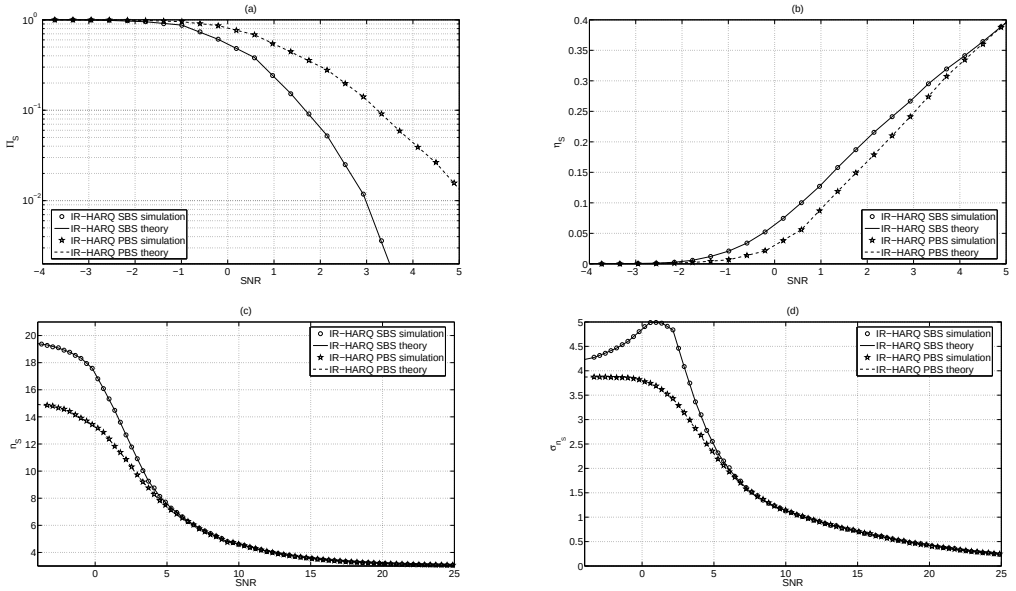

Figure 3.6: Theoretical and empirical performance in slow-fading Rayleigh channel for IR-HARQ at the NET layer with SBS and PBS strategies: (a) PER, (b) efficiency, (c) delay, and (d) jitter. ($N = 3$, $L_F = 120$ bits, $R_0 = 1/2$, $C = 24$ (SBS), and $P_{\max} = 8$ (PBS).)

3.3 HARQ mechanisms performance comparison

Our ability to evaluate the performance metrics enables us to compare the different HARQ mechanisms easily. Thanks to Chapter 2, we are able to propose a complete comparison of the HARQ mechanisms whereas, in the literature, most papers often

focus on the comparison of two HARQ mechanisms for one performance metric only (*e.g.* [54,55]).

This section is organized as follows: in Section 3.3.1, we introduce our simulation set-up chosen so that the comparison between different HARQ mechanisms will be fair. In Section 3.3.2, we really compare the considered HARQ mechanisms for the four performance metrics with respect to the SNR and the QoS requirements. Only PBS strategy is considered in this section, but similar results and remarks could be obtained by considering SBS strategy. Moreover we inspect the influence of two important parameters: the code rate (in Section 3.3.3) and the transmission credit (in Section 3.3.4).

3.3.1 Design parameters setting

In order to proceed into a fair comparison between different HARQ mechanisms, we consider that:

- the number of information bits per DSDU is constant whatever the HARQ mechanism and is denoted by L_{DSDU}
- and the number of bits (regardless of the nature of the bits, that is to say that the bit may correspond either to an information bit or to a redundancy bit) per DPDU is constant and is denoted by L_{DPDU}. Consequently the number of symbols is equal to $L_{\text{DPDU}}/2$ per DPDU since the QPSK modulation is used. In other terms, sending a DPDU is equivalent to $L_{\text{DPDU}}/2$ "channel uses" regardless of the HARQ mechanism.

We then have to determine the number of bits per FRAG L_F and the number of FRAGs per DSDU N, that both depend on L_{DSDU} and L_{DPDU}. Let us consider firstly the case when IR-HARQ (with mother code rate R_0) is studied. With this scheme, we get:

- $L_F = L_{\text{DPDU}}$
- and $N = L_{\text{DSDU}}/L_F = L_{\text{DSDU}}/L_{\text{DPDU}}$.

It comes easily that the parameters L_F and N defined for IR-HARQ are numerically identical for the ARQ, CC-ARQ, and IC-HARQ (with the same mother code rate).

In contrast, the CC-HARQ (with the same code rate as the IR-HARQ) has to be designed very carefully. Indeed, in such a case, the information bits contained in one DSDU are passed through a coding with rate R_0. As a consequence, we get:

- $L_F = R_0 \cdot L_{\text{DPDU}}$
- and $N = L_{\text{DSDU}}/L_F = L_{\text{DSDU}}/(R_0 \cdot L_{\text{DPDU}})$.

As the size of a DPDU is the same for each HARQ mechanism, it is fair to consider a constant transmission credit which corresponds to the maximum number of DPDUs that can be transmitted to receive one DSDU. This transmission credit is assumed to be fixed to C, for any retransmission scheme. As each DSDU is fragmented in N FRAGs and as the strategy used is the PBS, we then get $P_{\max} = C/N$ for each of the five schemes of interest. In Tab. 3.1, we summarize the relationship between the main parameters when various HARQ mechanisms are carried out. It thus resorts that in all cases, $C = Nr_0(P+1)$.

	ARQ + CC-ARQ	IR-HARQ + IC-HARQ	CC-HARQ
Coding rate R_0	No coding	$R_0 = \frac{1}{r_0}$	$R_0 = \frac{1}{r_0}$
Number of bits per FRAG L_F	L_{DPDU}	L_{DPDU}	$R_0 \cdot L_{\text{DPDU}}$
Number of FRAGs per DSDU N	$\frac{L_{\text{DSDU}}}{L_{\text{DPDU}}}$	$\frac{L_{\text{DSDU}}}{L_{\text{DPDU}}}$	$\frac{L_{\text{DSDU}}}{(R_0 \cdot L_{\text{DPDU}})}$
Transmission credit per FRAG $P_{\max}$ (PBS only)	$\frac{C}{N}$	$\frac{C}{N}$	$\frac{C}{N}$

Table 3.1: Parameters setting for different HARQ mechanisms, with fixed DSDU and DPDU lengths and with C constant.

3.3.2 Performance comparison

We now focus on the performance comparison of different HARQ mechanisms (ARQ, CC-ARQ, CC-HARQ, IR-HARQ, IC-HARQ) for the following metrics: PER, efficiency, delay, and jitter.

At first, we concentrate on the PER and the efficiency which have been plotted with respect to the SNR in Fig. 3.7. Concerning the PER, we immediately notice that the ARQ is always the worst mechanism whatever the SNR. The best mechanisms are actually the CC-HARQ and the IC-HARQ for all SNR intervals. Between the best ones and the worst one, it lies the IR-HARQ and the CC-ARQ. Clearly at medium and high SNRs, the IR-HARQ outperforms the CC-ARQ. Consequently, at medium and high SNR, the FEC included in IR-HARQ mechanism and absent from the CC-ARQ mechanism provides a significant error correcting capacity. At low SNR, the CC-ARQ is slightly better than the IR-HARQ (around SNR of 0 dB on this specific figure) but with a PER around 30% for which any system does not work. As a reminder, the CC-HARQ is not only a Chase combining technique but a FEC is included in each DPDU, what explains that this scheme is excellent in terms of PER.

We now move on the efficiency analysis. The CC-HARQ looses its advantage (obtained with the PER analysis) since having a FEC included in each DPDU has a significant cost in terms of efficiency. At high SNR, we effectively observe that the CC-HARQ has the worst efficiency while all other HARQ mechanisms (even the simple ARQ one) offer the same efficiency close to 1. Actually, at high SNR, it is well-known that the efficiency of the CC-HARQ is equal to R_0 (on the figure, we get $R_0 = 1/2$) since the redundancy is always sent even if it is not needed. At medium SNR, the ARQ and the CC-ARQ efficiencies quickly drop down. As a consequence, at medium SNR, the IR-HARQ, IC-HARQ, and CC-HARQ offer the best performance in terms of efficiency. At low SNR, the efficiency of the IR-HARQ vanishes whereas the efficiencies associated with IC-HARQ and CC-HARQ continue to provide reasonable performance. Finally, the best efficiency is always reached with the IC-HARQ regardless of the SNR.

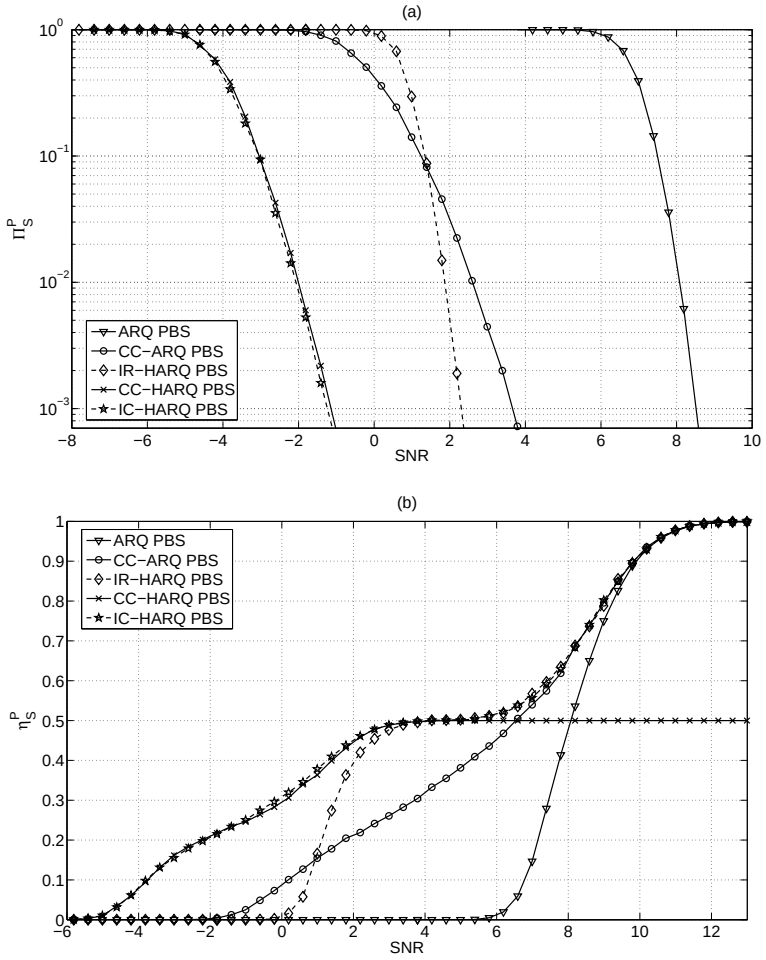

Figure 3.7: Theoretical PER (a) and efficiency (b) for various HARQ mechanisms with respect to the SNR at the NET layer with PBS strategy. ($L_{\text{DSDU}} = 360$ bits, $L_{\text{DPDU}} = 120$ bits, and $C = 24$; for ARQ and CC-ARQ, $N = 3$, $L_F = 120$ bits, and $P_{\max} = 8$; for IR-HARQ and IC-HARQ, $N = 3$, $L_F = 120$ bits, $P_{\max} = 8$, and $R_0 = 1/2$; for CC-HARQ, $N = 6$, $L_F = 60$ bits, $P_{\max} = 4$, and $R_0 = 1/2$.)

We then analyze the HARQ mechanisms performance in terms of delay and jitter. The both metrics are plotted on Fig. 3.8 with respect to the SNR with the same design parameters as in Fig. 3.7.

At high SNR, since almost all the FRAGs are received without re-sending them, the delay goes to the number of FRAGS per DSDU. Therefore the CC-HARQ is the worst mechanism since the number of FRAGs per DSDU is greater than those of other HARQ mechanisms (cf. Section 3.3.1). We observe that at high and medium SNR, the IC-HARQ and the IR-HARQ provide the smallest delay. At medium SNR, the CC-ARQ is clearly the worst one. Finally, at low SNR, the simple ARQ and the IR-HARQ offer the best delays. Actually using (2.82), one can prove that the delay for ARQ goes to $N(P_{\max}+1)/2$ when SNR becomes weak (cf. Section 2.3.2.2) whereas the delay at low SNR for mechanisms based on Chase combining (CC-ARQ, CC-HARQ, and IC-HARQ) is $NP_{\max}$. Obviously previous remark about delay at low SNR cannot be used to design a system since any system does not operate at such a SNR because of corresponding

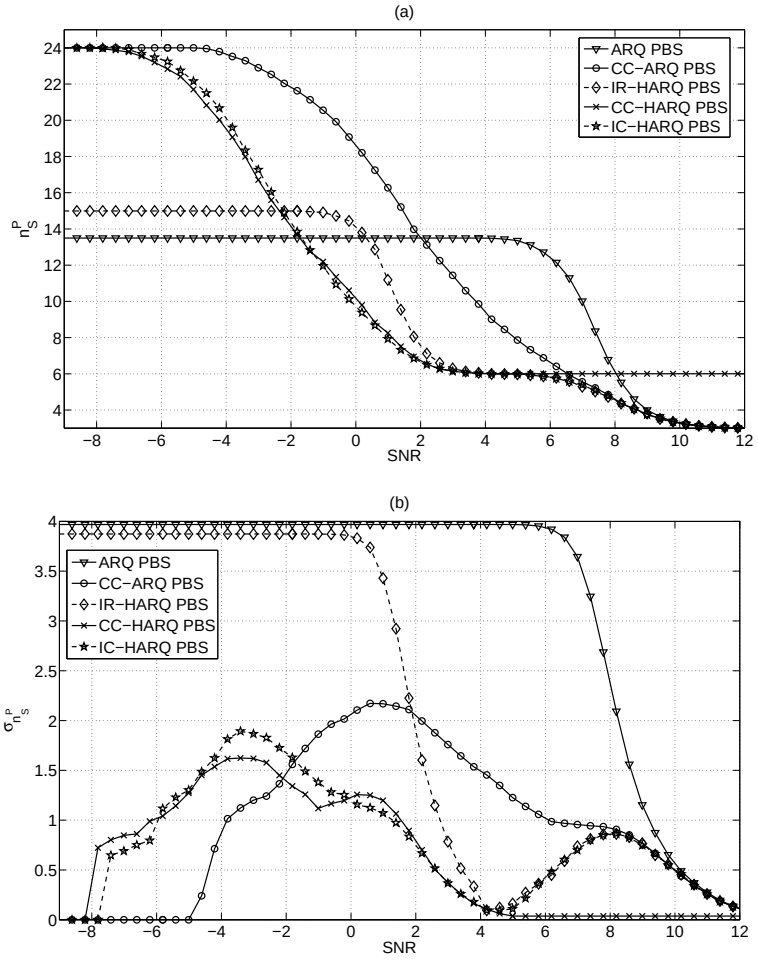

Figure 3.8: Theoretical delay (a) and jitter (b) for various HARQ mechanisms with respect to the SNR at the NET layer with PBS strategy. ($L_{\text{DSDU}} = 360$ bits, $L_{\text{DPDU}} = 120$ bits, and $C = 24$; for ARQ and CC-ARQ, $N = 3$, $L_F = 120$ bits, and $P_{\max} = 8$; for IR-HARQ and IC-HARQ, $N = 3$, $L_F = 120$ bits, $P_{\max} = 8$, and $R_0 = 1/2$; for CC-HARQ, $N = 6$, $L_F = 60$ bits, $P_{\max} = 4$, and $R_0 = 1/2$.)

poor efficiency and high PER.

Except at low SNR, we observe a matching between the shape of the jitter's curve and the shape of the delay's one. Indeed high jitter occurs when the slope of the delay's curve is sharp. When this slope is sharp, the system oscillates between different values of delay which leads to a high delay variation and so to a high jitter. Consequently the jitter is strongly (but not mathematically) connected to the derivative function of the delay. At high SNR, all HARQ mechanisms lead to small jitter. At medium and low SNR, the jitter is still well-bounded except for ARQ and IR-HARQ mechanisms. This can be explained as follows: at low SNR, the ARQ and IR-HARQ provide a lower average delay than the maximum instantaneous delay at the expense of a great variation of the instantaneous delay.

In Tab. 3.2, we select the best HARQ mechanisms with respect to the four performance metrics and the SNR. For sake of simplicity, we do not to classify the ARQ-like mechanisms (ARQ and CC-ARQ) which are very rarely the best ones. It turns out that

the IC-HARQ is a very relevant mechanism except for the delay at low SNR. The main drawback of the IC-HARQ is the complexity since it combines Incremental Redundancy (which leads to complex FEC codes) and the Chase combining (which leads to large size of buffer). We now compare the other HARQ mechanisms (without considering the IC-HARQ). In most realistic situations, a system often operates in high SNR context. In such a case, the CC-HARQ seems to be very relevant for the PER whereas the IR-HARQ is better for the efficiency and the delay. The CC-HARQ and the IR-HARQ offer the same jitter at high SNR. In stronger propagation environment, the CC-HARQ seems to offer a good trade-off between the four performance metrics.

	Low SNR	Medium SNR	High SNR
PER	IC, CC	IC, CC	IC, CC
Efficiency	IC, CC	IC, IR, CC	IC, IR
Delay	IR	IC, IR, CC	IC, IR
Jitter	IC, CC	IC, CC	IC, IR, CC

Table 3.2: Best HARQ mechanisms with respect to performance metrics and SNR (without ARQ like mechanisms).

Instead of considering the four performance metrics, let us consider the nature of the application. We remind that *i*) voice needs small delay, *ii*) video streaming needs high efficiency and small jitter, and *iii*) file transfer needs small PER and high efficiency. In Tab. 3.3, we select the best HARQ mechanism with respect to the application and the SNR.

	Low SNR	Medium SNR	High SNR
Voice	IR	IC, IR, CC	IC, IR
Video	IC, CC	IC	IC, IR
File transfer	IC, CC	IC, CC	IC, IR

Table 3.3: Best HARQ mechanisms with respect to application and SNR.

We now have to check that the same trends for the HARQ mechanisms marking can be observed when either other code rates or other transmission credits are used. Consequently we propose to study the influence of the code rate in Section 3.3.3 and of the transmission credit in Section 3.3.4. Notice that the goal of both next sections is not to assess the influence of code rate and transmission credit on the performance of HARQ mechanisms (actually done in Section 4.2), but only on the relative performance between HARQ mechanisms.

3.3.3 Influence of code rate

In this section, we modify the previously used value of R_0. Performance in previous sections have been always computed with $R_0 = 1/2$. Here, we consider that $R_0 = 1/4$. In contrast, the global transmission credit per DSDU is invariant. As seen in Tab. 3.1, this leads to change the FRAG size and the transmission credit per FRAG. In the rest of this section, we consider that the FRAG size is 60 bits for CC-HARQ (as done in previous section) and the FRAG size becomes 240 bits for other HARQ mechanisms instead of 120 as done before. As the transmission credit per DSDU remains the same, the transmission credit per FRAG in the CC-HARQ case is divided by 2 since the code rate is also divided by 2.

In Fig. 3.9, we plot the PER and the efficiency for the five HARQ mechanisms with respect to the SNR in PBS context and a value of R_0 equal to 1/4. Clearly,

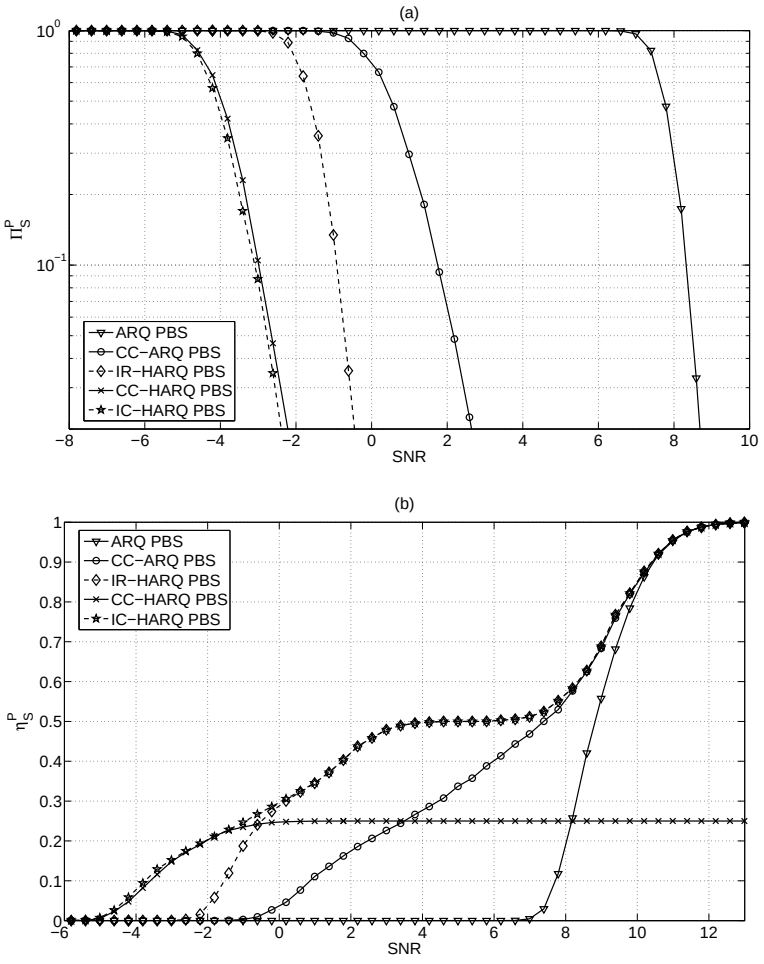

Figure 3.9: Theoretical PER (a) and efficiency (b) for various HARQ mechanisms with respect to the SNR at NET layer with PBS strategy and $R_0 = 1/4$. ($L_{\text{DSDU}} = 720$ bits, $L_{\text{DPDU}} = 240$ bits, and $C = 24$; for ARQ and CC-ARQ, $N = 3$, $L_F = 240$ bits, and $P_{\max} = 8$; for IR-HARQ and IC-HARQ, $N = 3$, $L_F = 240$ bits, and $P_{\max} = 8$; for CC-HARQ, $N = 12$, $L_F = 60$ bits, and $P_{\max} = 2$.)

the performance in PER of the IR-HARQ improves and becomes nearer to those of IC-

HARQ and CC-HARQ. In addition, the gap in PER between the CC-HARQ and the IC-HARQ is slightly larger. Decreasing the code rate enables us to offer higher performance for the IR-like mechanisms in terms of PER. The advantage of the CC-HARQ reduces accordingly. In terms of efficiency, the CC-HARQ performance dramatically degrades at high SNR, since the efficiency of the CC-HARQ goes then to R_0. Conversely, by decreasing the code rate, the efficiency of the IR-HARQ is improved at low SNR.

Let us inspect the delay and the jitter displayed in Fig. 3.10. The trend on the HARQ mechanisms comparison remains the same as the one described in Section 3.3.2. We notice a significant degradation at high SNR for the delay of the CC-HARQ. Therefore once again the CC-HARQ is affected by the decrease of the code rate.

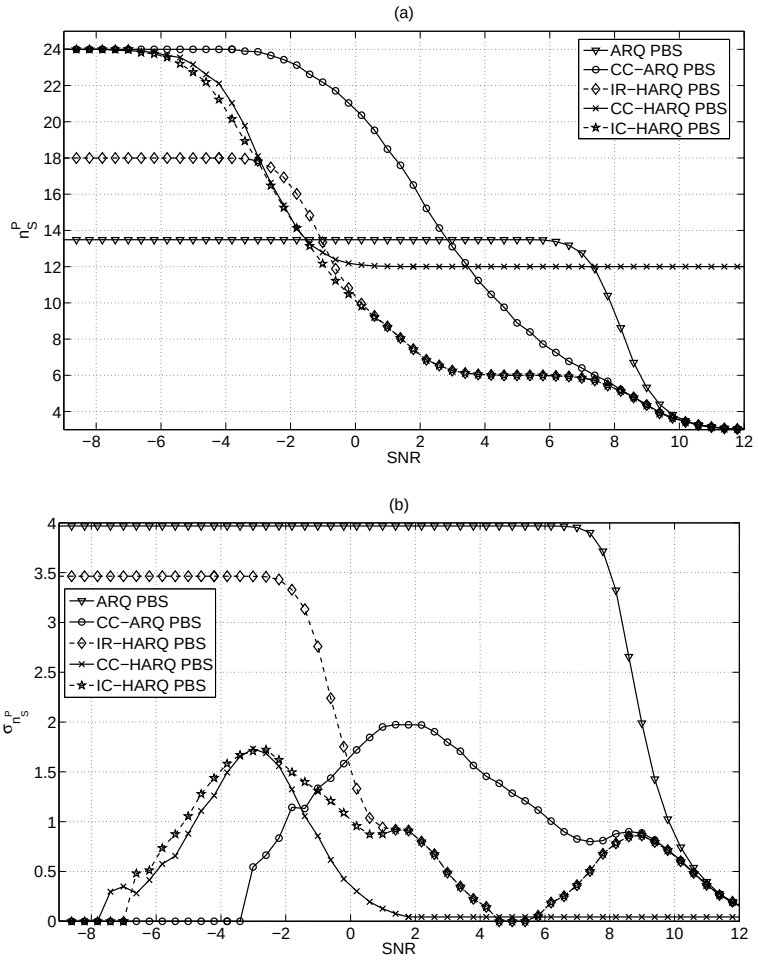

Figure 3.10: Theoretical delay (a) and jitter (b) for various HARQ mechanisms with respect to the SNR at NET layer with PBS strategy and $R_0 = 1/4$. ($L_{\text{DSDU}} = 720$ bits, $L_{\text{DPDU}} = 240$ bits, and $C = 24$; for ARQ and CC-ARQ, $N = 3$, $L_F = 240$ bits ,and $P_{\max} = 8$; for IR-HARQ and IC-HARQ, $N = 3$, $L_F = 240$ bits, and $P_{\max} = 8$; for CC-HARQ, $N = 12$, $L_F = 60$ bits, and $P_{\max} = 2$.)

Finally, decreasing the code rate allows to reduce the gap in performance of the IR-HARQ compared to the CC-HARQ when the IR-HARQ offered not as good performance as the CC-HARQ. Consequently decreasing the code rate is particularly in favor of the IR-like mechanisms. Nevertheless the comparison done in Tab. 3.2 and Tab. 3.3 is not

significantly modified and thus remains still valid.

3.3.4 Influence of the transmission credit

Until now, we have computed all the figures with the same transmission credit $C = 24$. We now assess if using another value for C modifies or not the conclusion drawn in Tab. 3.2 and Tab. 3.3. As previously mentioned, C is equal to $NP_{\max}$ where $P_{\max}$ is the transmission credit per FRAG. Furthermore $P_{\max}$ can be decomposed into $r_0(P+1)$ for the IR-like mechanisms (because of the assumption of constant DPDU length) and into $(P+1)$ for the Chase combining based mechanisms, where P is the persistence per DPDU. In this section, for changing C, we keep r_0 and N fixed, and we just modify the value of P. Therefore we will plot the four performance metrics versus the persistence P. Actually, for IR-HARQ, $r_0 = 2$ and $N = 3$. In contrast, the persistence P varies from 1 to 5. Consequently, C varies from 12 to 36.

In Fig. 3.11, we plot the PER and the efficiency for only two HARQ mechanisms with respect to P in PBS context and for different values of SNR. Actually, we present here three SNR values: a low one, a medium one, and a high one. Notice that the considered SNR values are not exactly the same from a performance metric to an other one. Actually, we select the SNRs that enable us to exhibit relevant insights. In order to not overwhelm the figure, we only plot the performance metrics for two kinds of HARQ which are the most representative: the CC-HARQ and the IR-HARQ.

In terms of PER, it comes easily that increasing the persistence leads to an important gain in performance for both mechanisms. The gap between both mechanisms is not significantly reduced with the increase of P. As a result, the classification of both mechanisms according to the PER is unchanged regardless of the SNR. As for the efficiency, the persistence allows to increase the performance for both mechanisms specifically at low and medium SNRs. However, as soon as P exceeds a certain threshold, the efficiency is not improved anymore for the IR-HARQ scheme as for the CC-HARQ one. This threshold value decreases as the SNR increases and for high SNR, becomes null. Once again, modifying P does not mainly change the conclusion put in Tab. 3.2 about efficiency.

In Fig. 3.12, we plot the delay and the jitter for only two HARQ mechanisms with respect to P in PBS context and for different values of SNR.

Except at low SNR, the persistence has a very limited impact on the delay and the jitter. Let us focus now on low SNR context. One can then observe that the IR-HARQ which was the best mechanism in previous simulations concerning the delay becomes the worst one for large P. Consequently the conclusion given in Tab. 3.2 for the row "delay" and the column "low SNR" is not general at all. Nevertheless, at low SNR, realistic system does not operate, thus Tab. 3.3 may remain the same. Regarding the jitter at low SNR, only the IR-HARQ performance are degraded by the persistence increase.

As a conclusion, we have noticed that the persistence has a great influence on the PER and a slighter influence on the other performance metrics. The greater the persistence, the better the PER for each HARQ mechanism. As for the efficiency, as soon as P reaches a certain threshold (which is not large in practice), the efficiency becomes numerically almost independent of P. Note that this observation is only mathematically true for ARQ mechanism at MAC layer. The persistence modifies only very slightly the

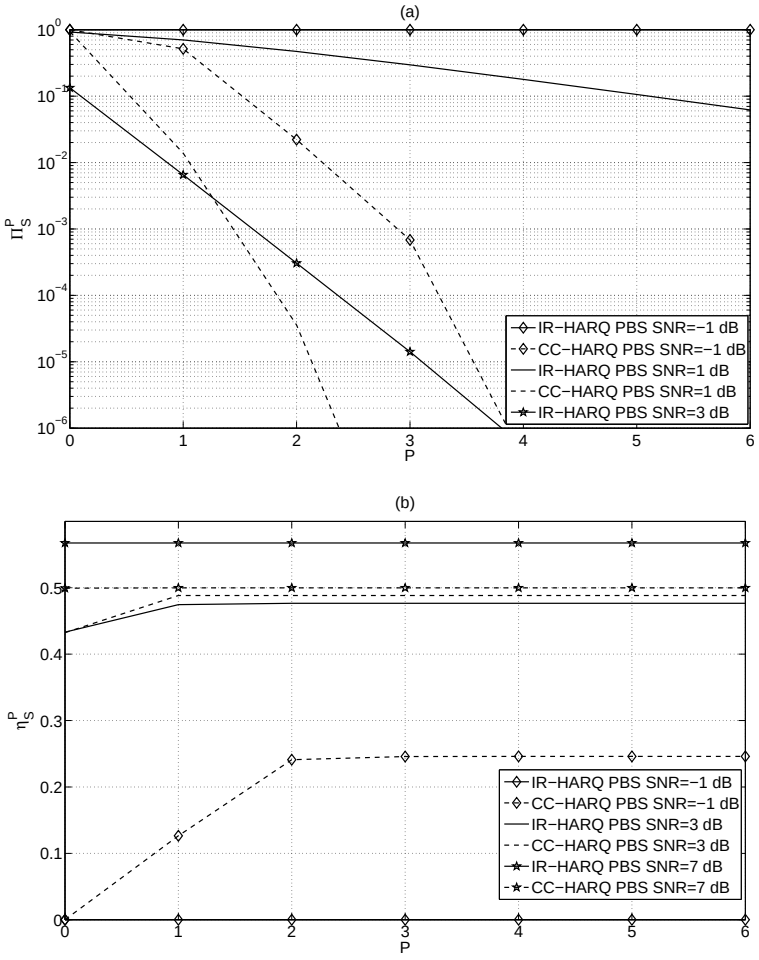

Figure 3.11: Theoretical PER (a) and efficiency (b) for various HARQ mechanisms with respect to P at NET layer with PBS strategy and different SNRs. ($L_{\text{DSDU}} = 360$ bits, $L_{\text{DPDU}} = 120$ bits, and $R_0 = 1/2$; for IR-HARQ, $N = 3$ and $L_F = 120$ bits ; for CC-HARQ, $N = 6$ and $L_F = 60$ bits.)

values of the delay and the jitter except at low SNR.

As a consequence, playing on the amount of FEC redundancy (via the code rate) or the amount of simple retransmission (via the persistence) does not really impact the relative performance metrics between considered HARQ mechanisms, specifically around the operating points (at medium and high SNRs). As a result the conclusion previously drawn in Tab. 3.2 and Tab. 3.3 still holds. Finally the IC-HARQ almost always offers the best performance. If we consider only the IR-HARQ and the CC-HARQ, the IR-HARQ is appropriate for communications needing high efficiency and low delay whereas the CC-HARQ is more relevant for communications needing very low PER.

3.4 Cross-layer strategies performance comparison

In the previous section, we have provided insights about the influence of the HARQ mechanisms on the four performance metrics and about the choice of the HARQ mech-

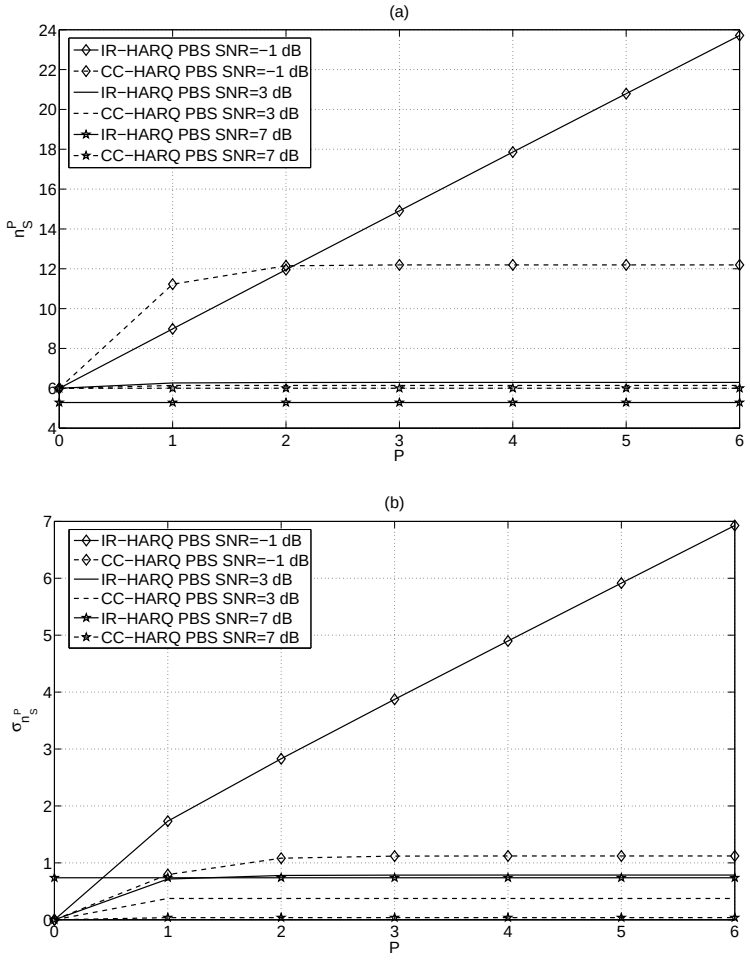

Figure 3.12: Theoretical delay (a) and jitter (b) for various HARQ mechanisms with respect to P at NET layer with PBS strategy and different SNRs. ($L_{\text{DSDU}} = 360$ bits, $L_{\text{DPDU}} = 120$ bits, and $R_0 = 1/2$; for IR-HARQ, $N = 3$ and $L_F = 120$ bits ; for CC-HARQ, $N = 6$ and $L_F = 60$ bits.)

anisms according to the QoS requirements through Tab. 3.2 and Tab. 3.3. We would like hereafter to proceed in a similar way but concerning the cross-layer strategies (PBS or SBS). Therefore this section deals with the evaluation of the four performance metrics according to the considered cross-layer strategies (PBS or SBS) at the NET layer. For the same reasons as the ones evoked in Section 3.3.4, we only consider the two most representative HARQ mechanisms: the IR-HARQ (which enables us to illustrate the influence of the incremental redundancy based error correcting code), and the CC-HARQ (which enables us to illustrate the influence of the linear combining). In order to make a fair comparison, the transmission credit per DSDU and the number of FRAGs per DSDU for a given retransmission scheme are the same regardless of the strategy used.

In Fig. 3.13 and Fig.3.14, we display the four performance metrics with respect to the SNR for the both cross-layer approaches (PBS and SBS) with the IR-HARQ and the CC-HARQ mechanisms respectively. We then can observe that the PER is significantly improved when the SBS strategy is used. In addition the SNR penalty for the PBS strategy compared to the SBS strategy increases with the SNR. This remark holds

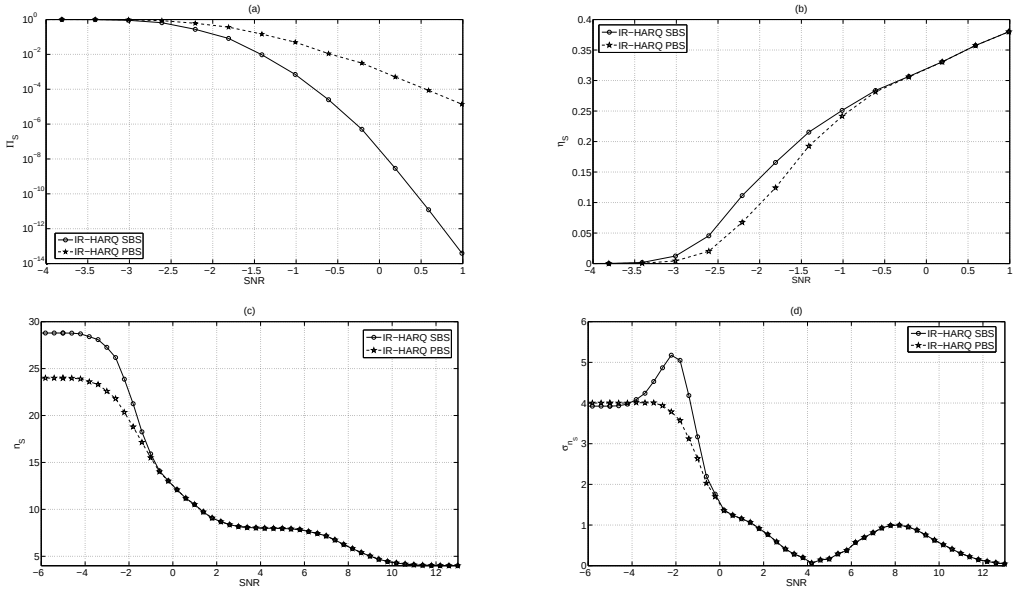

Figure 3.13: Theoretical PER (a), efficiency (b), delay (c), and jitter (d) of the SBS and the PBS strategies for IR-HARQ mechanism. ($C = 32$, $N = 4$, $L_F = 120$, and $R_0 = 1/4$.)

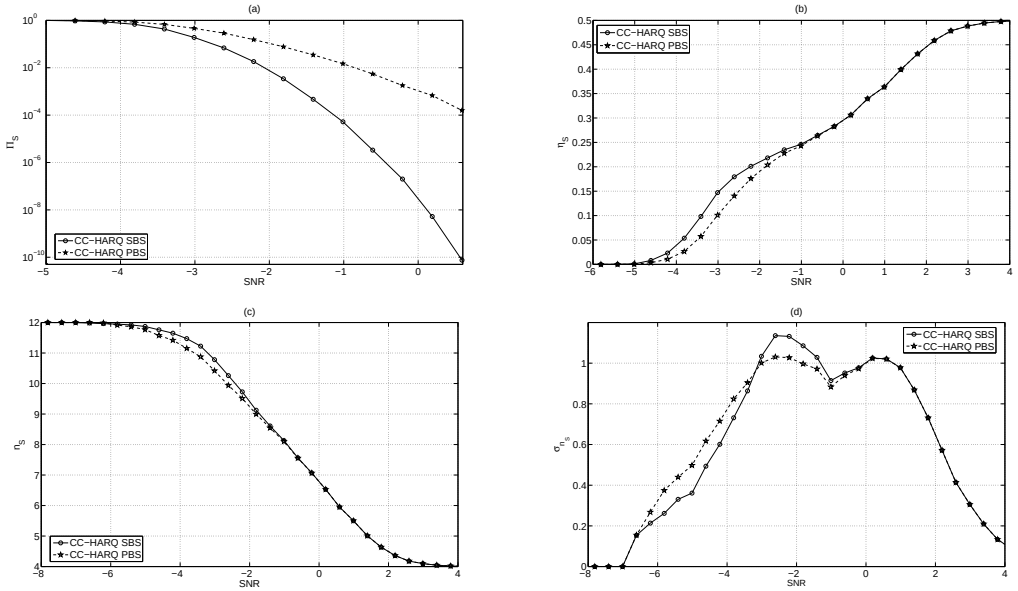

Figure 3.14: Theoretical PER (a), efficiency (b), delay (c), and jitter (d) of the SBS and the PBS strategies for CC-HARQ mechanism. ($C = 12$, $N = 4$, $L_F = 60$, and $R_0 = 1/2$.)

for both HARQ mechanisms (IR-HARQ and CC-HARQ) and confirms the Theorem 1 proven in Chapter 1. The same remark about the PER comparison of both cross-layer strategies has been already done in [3] but only for ARQ mechanism. In contrast, the efficiency is quite insensitive to the cross-layer strategy. To be more precise, the SBS strategy enables us to improve very slightly the efficiency at medium SNR. As for the delay and the jitter, they are also close to each other for both strategies except at low SNR for IR-HARQ. At low SNR, the PBS strategy (*i.e* that does not take into account

the link between both layers) is better than the SBS strategy in terms of delay and jitter. We specifically observe that the jitter exhibits a peak at low SNR for the SBS strategy and not for the PBS strategy. It is due to a sharper slope of the delay for the SBS strategy than for the PBS strategy as seen on part (c) of the figure. We nevertheless remind that at such low SNR, any system does not operate due to too high PER and too low efficiency.

To highlight more precisely the relationship between PER and delay, we plot the PER with respect to the delay for both strategies (PBS and SBS) on Fig. 3.15. The curves displayed on this figure have been obtained with different SNRs but only with the IR-HARQ mechanism for the sake of clarity. At low SNR, the delay is worse in the SBS strategy case for eventually a little benefit in terms of PER. However, at high SNR, the SBS approach allows to really achieve a lower PER than that of the PBS approach for almost the same delay. Therefore at high SNR, it is relevant to select the SBS strategy which relies on the optimization between the MAC layer and the NET layer. Similar analysis and conclusion were presented in [3] for the ARQ mechanism.

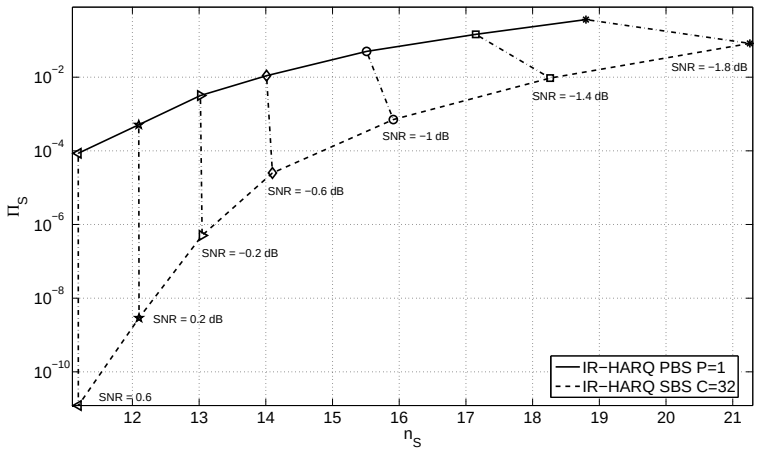

Figure 3.15: PER versus delay of the SBS and the PBS strategies for IR-HARQ mechanism. ($N = 3$, $L_F = 120$ bits, and $R_0 = 1/4$.)

In Fig. 3.16, we plot the efficiency with respect to the SNR for both strategies (PBS and SBS). The curves displayed on this figure have been obtained with different SNRs but only with the CC-HARQ mechanism for the sake of clarity. Given a PER, we observe that the SBS strategy offers a better efficiency than the PBS strategy. Given a reasonable value of SNR (*i.e.* at medium or high SNR), the SBS allows to improve significantly the performance in terms of PER and slightly the performance in terms of efficiency.

We are now able to classify the merit of the both strategies according to the SNR and the considered performance metric. This is summarized in Tab. 3.4. The SBS strategy enables us to boost the PER. At high SNR, the SBS strategy is the best one for each performance metric. In Tab. 3.5, we provide the strategy that we have to choose given a certain application and a certain SNR as done in Tab. 3.3.

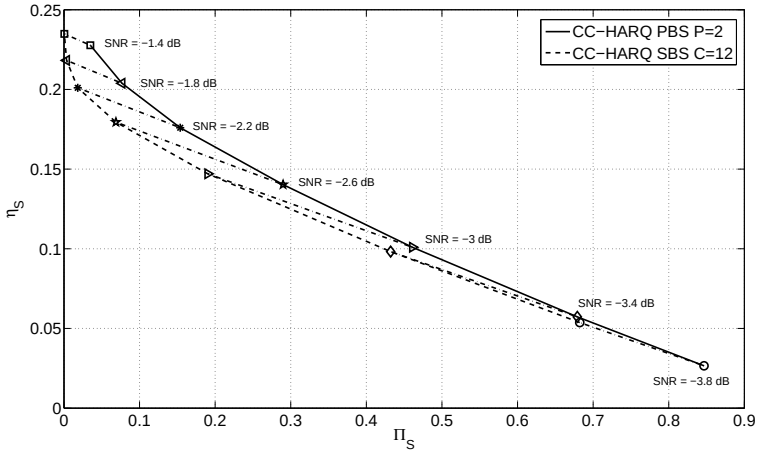

Figure 3.16: Efficiency versus PER of the SBS and the PBS strategies for CC-HARQ mechanism. ($N = 4$, $L_F = 60$ bits, and $R_0 = 1/2$.)

	Low SNR	Medium SNR	High SNR
PER	SBS	SBS	SBS
Efficiency	SBS	SBS	SBS
Delay	PBS	PBS	PBS, SBS
Jitter	PBS	PBS	PBS, SBS

Table 3.4: Best cross-layer strategies with respect to performance metrics and SNR

	Low SNR	Medium SNR	High SNR
Voice	PBS	PBS	PBS, SBS
Video	PBS	PBS, SBS	PBS, SBS
File transfer	SBS	SBS	SBS

Table 3.5: Best cross-layer strategies with respect to application and SNR

3.5 Influence of the modulation

We here analyze the influence of the modulation on the four performance metrics for both SBS and PBS strategies. For sake of simplicity, we only focus on the IR-HARQ mechanism.

In order to compare transmission based on different data rate, we consider that the

number of symbols per DPDU, denoted by L_s, is constant. In contrast, the number of bits (of information or redundancy) may differ from a DPDU to another one according to the chosen modulation scheme. Moreover the size in bits of the DSDU is also unchanged. Each DSDU then contains L_{DSDU} information bits.

Let us consider m_s the number of bits per symbol. The number of bits in a DPDU, or equivalently in a FRAG (since the scheme is here the IR-HARQ), is thus equal to $m_s L_s$, and the number of FRAGs per DSDU is $N = L_{\text{DSDU}}/(m_s L_s)$. The chosen parameters given in the captions of the figures satisfy these constraints.

Before going further, we have to mention that the efficiency has been replaced with the capacity, denoted hereafter by κ. Indeed, the efficiency only inspects the ratio between the well received bits and the transmitted bits and so does not take into account that several bits can be transmitted per channel use. We define the capacity as the product between the efficiency and the number of transmitted bits per channel use. Consequently $\kappa = m_s \eta$.

In Fig. 3.17, we plot the PER and the data rate with respect to the SNR for both cross-layer strategies at the NET layer and for different modulations (BPSK, QPSK, 16QAM, 64QAM). The lowest PER and capacity have been obtained by the BPSK. The gap between the PBS PER and the SBS one is slightly modified by the modulation. The highest capacity is obtained for constellations with large size as soon as SNR is high enough (specifically for the 16-QAM and 64-QAM on the figure).

In Fig. 3.18, we plot the delay and the jitter with respect to the SNR for both cross-layer strategies at the NET layer and for different modulations (BPSK, QPSK, 16QAM, 64QAM). At low SNR, the best modulation in terms of delay is the 64QAM, and the worst one is the BPSK. When the SNR increases and becomes medium, the classification is modified. Between low and medium SNR, the BPSK becomes the best one. Then at medium SNR, the QPSK is highlighted. Finally between medium and high SNR, the 16QAM offers the best delay. At high SNR, all modulations are equivalent for the delay performance metric. We specifically note that the 64QAM becomes once again interesting at high SNR. Indeed, at high SNR, the delay goes to N and N decreases when the constellation size increases. In terms of jitter, the higher the modulation, the higher the jitter. These conclusions are identical for the SBS and the PBS strategies.

In Tab. 3.6, we classify the different modulations with respect to the four performance metrics and the SNR. For the voice, as the delay has to be minimized, the BPSK modulation (even if it is not always the best one in terms of delay, it is not far away from the best one) seems to be a good candidate. For the video streaming, maximizing the capacity and minimizing the jitter leads to two different modulations, therefore a trade-off has to be done. For the file transfer, at high SNR, the 64QAM can be chosen since the PER obtained for 64QAM at high SNR is small enough for the required QoS.

3.6 Influence of the propagation channel

As a reminder, all the previous analyses have been carried out when a Gaussian channel is implemented. In wireless context, the Rayleigh channel is more realistic. Therefore, we analyze in this section the four performance metrics in the context of Rayleigh channel compared to the context of Gaussian channel. We recall that the considered Rayleigh channel is slow flat fading, that is to say that the channel realization is con-

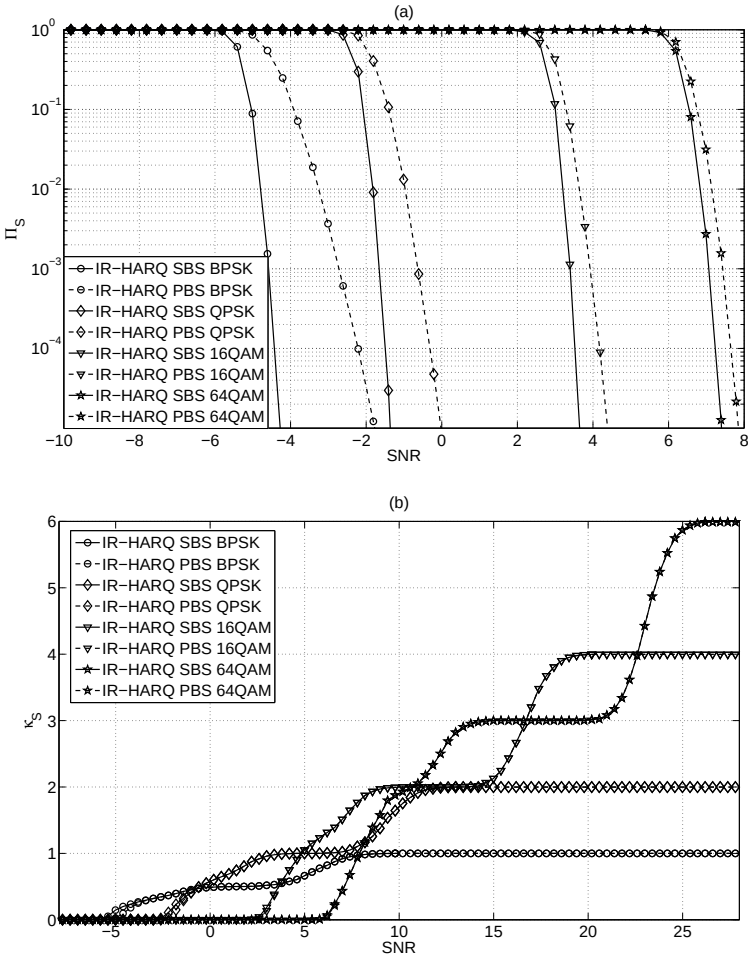

Figure 3.17: Theoretical PER (a) and capacity (b) versus SNR for IR-HARQ mechanism in the SBS and PBS contexts. ($C = 96$ and $R_0 = 1/4$; for BPSK, $L_F = 120$ bits and $N = 12$; for QPSK, $L_F = 240$ bits and $N = 6$; for 16QAM, $L_F = 480$ bits and $N = 3$; for 64QAM, $L_F = 720$ bits and $N = 2$.)

	Low SNR	Medium SNR			High SNR
		Low-Medium	Medium-Medium	High-Medium	
PER	BPSK	BPSK	BPSK	BPSK	BPSK
Efficiency	ALL	BPSK	QPSK	16QAM	64QAM
Delay	64QAM	BPSK	QPSK	16QAM	64QAM
Jitter	BPSK	BPSK	BPSK	BPSK	BPSK

Table 3.6: Best modulation with respect to performance metrics and SNR.

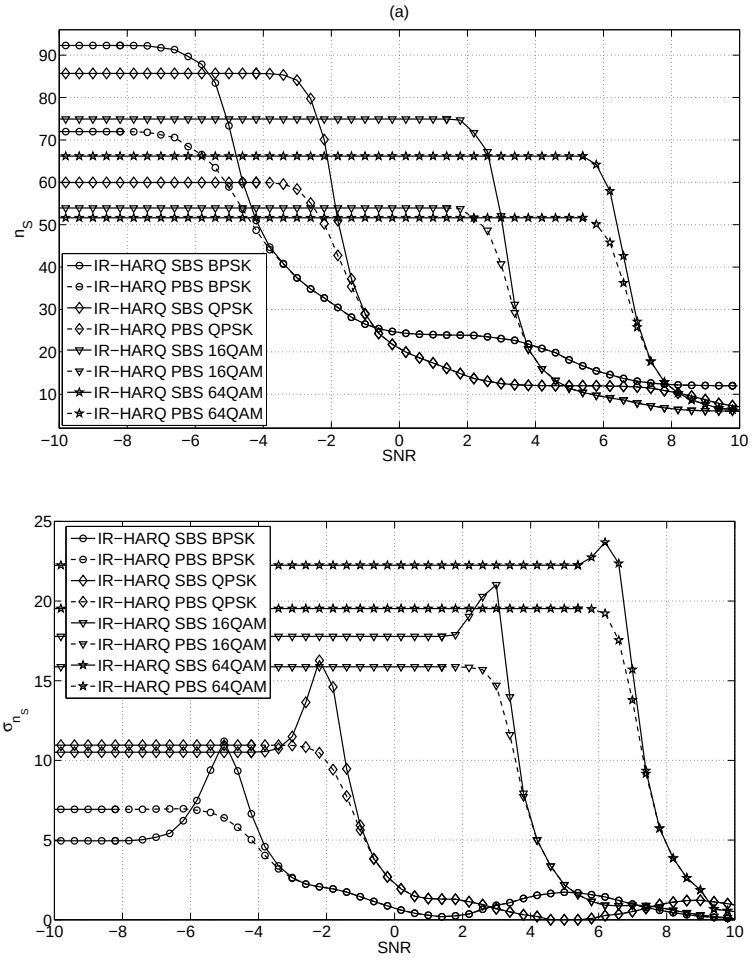

Figure 3.18: Theoretical delay (a) and jitter (b) versus SNR for IR-HARQ mechanism in the SBS and PBS contexts. ($C = 96$ and $R_0 = 1/4$; for BPSK, $L_F = 120$ bits and $N = 12$; for QPSK, $L_F = 240$ bits and $N = 6$; for 16QAM, $L_F = 480$ bits and $N = 3$; for 64QAM, $L_F = 720$ bits and $N = 2$.)

stant over one DPDU and varies independently from one DPDU to another one. The mathematical model of the Rayleigh channel is introduced in Section 3.1. For sake of simplicity, we only focus on the IR-HARQ mechanism but we consider both SBS and PBS schemes.

In Fig. 3.19, we plot the PER and the efficiency for the IR-HARQ mechanism in the context of the SBS and PBS schemes for a Gaussian channel and a Rayleigh channel. Note that for the PBS strategy, similar curves can be found in [56]. As proven in Theorem 1, the SBS is always better than the PBS for the PER regardless of the propagation channel. Fig. 3.19 confirms this statement. Nevertheless the gap between SBS and PBS is not the same in a Gaussian channel and a Rayleigh channel. In a Gaussian channel, we observe that the PBS based curve is only shifted but the shape is the same as the SBS one. The difference is thus only a gain in SNR. As for a Rayleigh channel, the difference between SBS and PBS is more important. We observe a gain in SNR but also a gain in diversity since the slope of the SBS based curve is larger than the slope of the PBS based curve. Consequently the SBS handles more relevantly the

channel diversity. Concerning the efficiency, the difference between SBS and PBS is

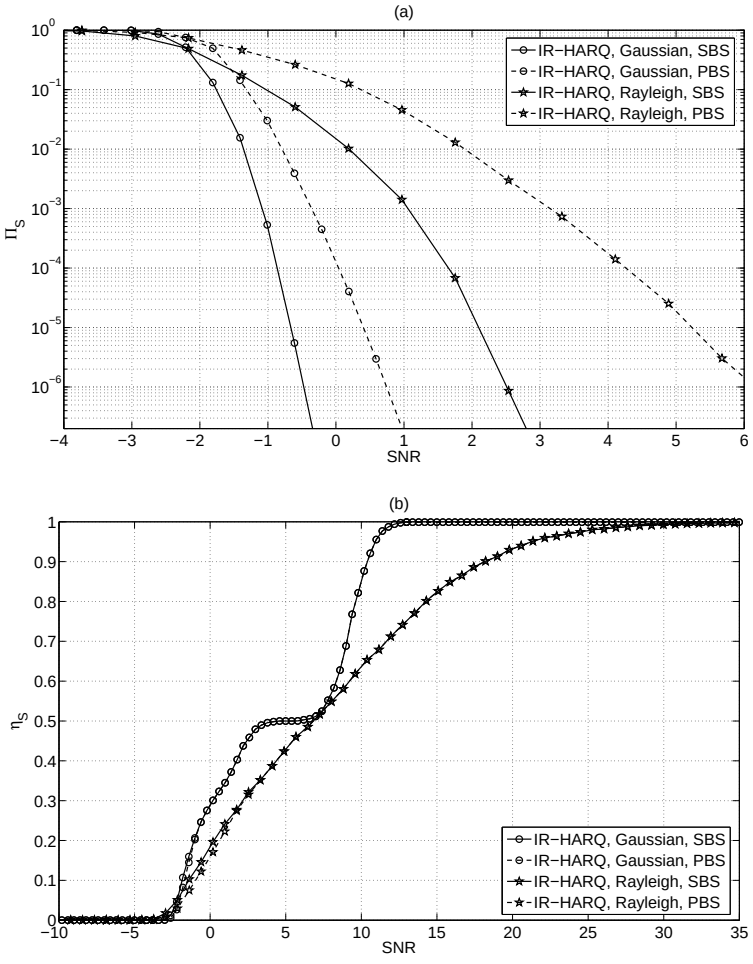

Figure 3.19: Theoretical PER (a) and efficiency (b) versus SNR for IR-HARQ mechanism in the SBS and PBS contexts and with a Gaussian channel and a Rayleigh channel. ($C = 36$, $N = 3$, $L_F = 240$ bits, and $R_0 = 1/4$.)

very small for the both propagation channels. In contrast, the Gaussian channel offers better performance than the Rayleigh channel in terms of efficiency specifically at useful SNRs (such as those around 10dB).

In Fig. 3.20, we display the delay and the jitter for the IR-HARQ mechanism in the context of the SBS and PBS schemes for a Gaussian channel and a Rayleigh channel. Not surprisingly, the delay and the jitter are slightly worse for the Rayleigh channel than for the Gaussian channel (except for a few values of SNRs). Nevertheless, the trends between SBS and PBS are unchanged.

To conclude, the same trade-off between the cross-layer and the conventional strategies has to be made regardless of the propagation channel.

3.7 Conclusion

In this chapter, we have analyzed, by means of the numerical evaluations of the four performance metrics expressions established in Chapter 2, the influence of the HARQ

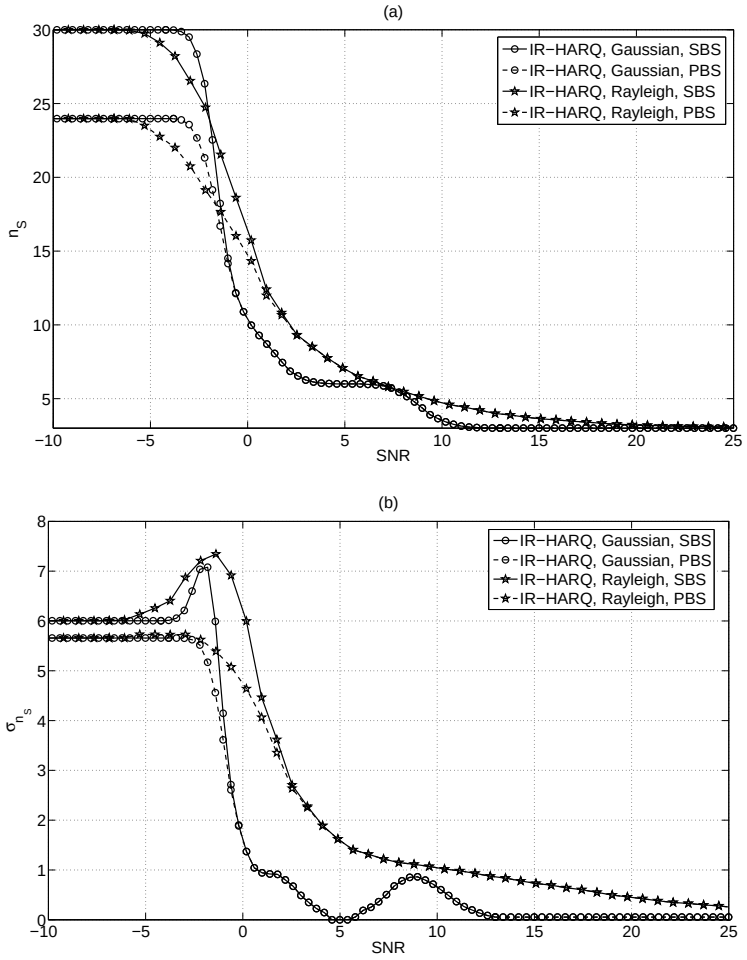

Figure 3.20: Theoretical delay (a) and jitter (b) versus SNR for IR-HARQ mechanism in the SBS and PBS contexts and with a Gaussian channel and a Rayleigh channel. ($C = 36$, $N = 3$, $L_F = 240$ bits, and $R_0 = 1/4$.)

mechanism, the cross-layer strategies, the modulation, and the propagation channel on the system performance. First of all, we have numerically validated the accuracy of the theoretical formulas for several system configurations. Then, depending on the system QoS, we have noticed that the CC-HARQ or the IR-HARQ can be of interest. In addition, we also have noted that the SBS scheme is well adapted to systems needing small PER and high efficiency whereas the PBS scheme is rather more relevant for real-time systems needing small jitter. The influence of the modulation and the propagation channel has been studied as well. Finally the reader may refer to Tab. 3.2, Tab. 3.4, and Tab. 3.6 to know how to satisfy the desired QoS.

Chapter 4

Applications to Communication Systems

4.1 Introduction

In Chapter 3, we have analyzed the influence of the different HARQ mechanisms (ARQ, CC-ARQ, IR-HARQ, CC-HARQ, and IC-HARQ), the different cross-layer strategies (PBS and SBS), the different modulations (BPSK, QPSK, 16QAM, and 64QAM), and the different propagation channels (Gaussian and Rayleigh ones) on the four performance metrics (PER, efficiency, delay, and jitter). Thanks to this analysis, we now know how to select the most relevant configuration with respect to the required QoS and the SNR. In the previous chapter, we did not evaluate the impact of the design parameters such as the transmission credit ($P_{\max}$ for the PBS and C for the SBS), the number of FRAGs per DSDU N, the code rate R_0. Therefore, the main goal of Chapter 4 is to find the best selection of the design parameters in order to satisfy a required QoS for a given channel quality. In this chapter, we show that the SBS induces by construction an unequal packet loss protection. We theoretically analyze this property and extend it to the PBS. Lastly, we show that we can predict the packet error rate and delay performance of a discrete event based realistic simulation implementing the HARQ scheme, using the previous derived closed-form expressions. In the remaining of this chapter, we only consider the IR-HARQ mechanism for the sake of clarity. Unless it is otherwise stated, the evaluation is done under the assumptions of constant DPDU length, Gaussian channel and QPSK modulation.

The chapter is organized as follows: in Section 4.2, we study the influence of the main design parameters (C, N, R_0, the error correcting codes) on the PER, the efficiency, the delay, and the jitter. We then provide the best set of parameters according to the considered application. Section 4.3 is dedicated to the unequal packet loss protection. The application of the theoretical formulas in a multi-hop context is done is Section 4.4.

4.2 Radio resource management

We here investigate the influence of the main parameters on the four performance metrics. The goal of the section is to show that the choice of these design parameters consists of a trade-off since minimizing PER, delay, and jitter, and maximizing efficiency leads to contradictory choices of parameters. This analysis is facilitated by our ability to

compute easily the performance metrics thanks to the closed-form expressions given in Chapter 2 (once the π_j are obtained, following the approach described in Section 2.4). This section is organized as follows:

- How to choose the global transmission credit C? see section 4.2.1.
- How to choose the code rate R_0 and the persistence P? see section 4.2.2.
- How to choose the error-correcting code? see section 4.2.3.
- How to choose the number of FRAGs per DSDU N? see section 4.2.4.

4.2.1 Influence of the transmission credit

In this section, we study the influence of the transmission credit ($P_{\max}$ for the PBS and C for the SBS) on the performance. In order to do so, we fix the following parameters: the number of FRAGs per DSDU $N = 3$, the code rate $R_0 = 1/4$, the FRAG size $L_F = 120$ bits. In order to make a fair comparison between both SBS and PBS strategies, we grant the same maximum number of transmissions per DSDU [3], which leads to $C = NP_{\max}$. Since we consider here the IR-HARQ scheme with constant DPDU length, $P_{\max}$ is equal to $r_0(P+1)$. Thus, in the rest of the section, P will be adjusted in order to fit $C = Nr_0(P+1)$, and we will then only refer to C.

In Fig. 4.1, we plot the PER and the efficiency with respect to C in both PBS and SBS contexts at different SNRs. As expected, the higher C, the lower the PER. In addition, at given PER, the gap between the both strategies (SBS and PBS) increases as the SNR increases. In the same way as for the PER, increasing C leads to an improvement in terms of efficiency. Nevertheless the gain in efficiency is only slight and not significant for a system design. At low and high SNRs, the gain is even so tiny that it is difficult to see it on the figure. In Fig. 4.2, we represent the ratio between the efficiency obtained in the SBS and the efficiency obtained in the PBS with respect to the SNR for different values of C. Firstly, as this ratio is always greater than or equal to one, the efficiency for the SBS is greater than or equal to the efficiency for PBS whatever the SNR. Secondly, we can notice that the gap in efficiency between both cross-layer approaches increases with C specifically at low SNR.

In Chapter 2, we have proven that the efficiency at the MAC layer for HARQ schemes based on IR is independent of the persistence when $P_{\max}$ is a multiple of t_0. This result is valid for unequal DPDU sizes ($w_i \neq w_1$ for $i \neq 1$). Fig. 4.3 highlights this result for two values of t_0. The first one corresponds to $t_0 = 6$ with the following code rates: 1, 4/5, 2/3, 1/2, 4/11, and 1/4. Along with a number of information bits equal to 320, we add up 24 tailing bits, so that the number of bits fed into the convolutional encoder is equal to 344. Thus the sizes of the t_0 DPDUs are equal to: $w_1 = 344$, $w_2 = w_3 = 86$, $w_4 = 172$, $w_5 = 258$, and $w_6 = 430$. The second example corresponds to $t_0 = 4$ with the following code rates: 1, 1/2, 1/3, and 1/4. In this case, the DPDU sizes are equal to $w_i = 344$, $\forall i$. We can check on Fig. 4.3 that the efficiency hits the same value at $P_{\max} = \{6, 12, 18\}$ for $t_0 = 6$ and at $P_{\max} = \{8, 12, 16\}$ for $t_0 = 4$, which confirms the result of Section 2.3.3. It is also interesting to note that the efficiency is maximal when $P_{\max}$ is a multiple integer of t_0, *i.e.* $P_{\max} = a \cdot t_0$, with $a \in \mathbb{N}^*$, and that in the interval $[a \cdot t_0, (a+1) \cdot t_0]$, the efficiency reaches its minimum for $P_{\max} = (a+1) \cdot t_0 - 1$. However, those results have not been proven in Section 2.3.3.

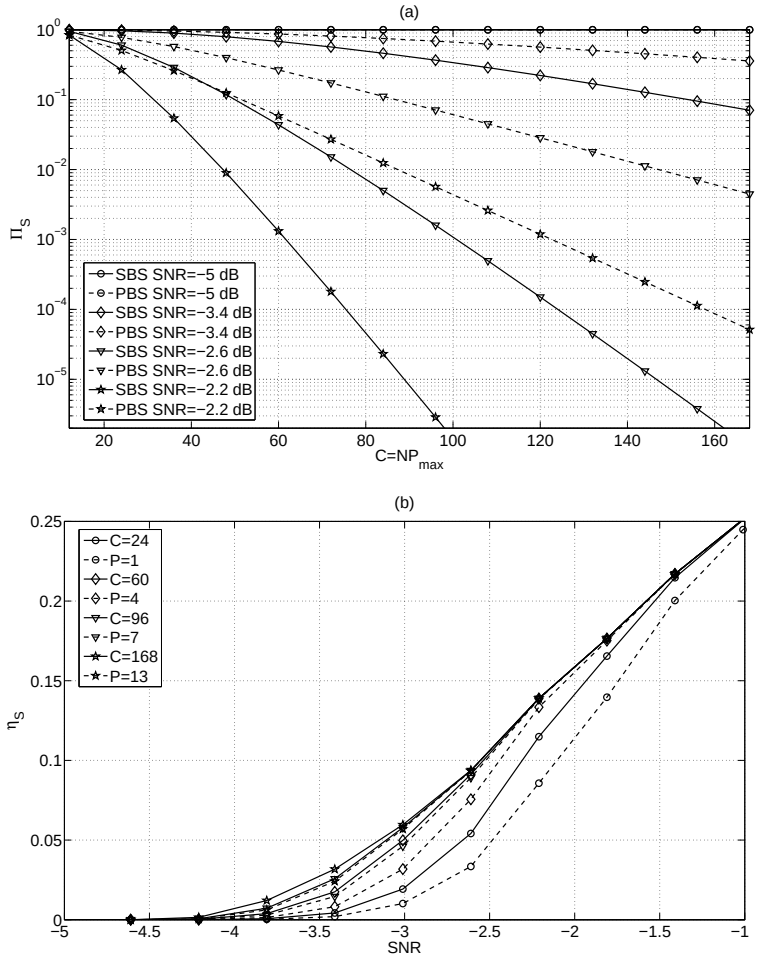

Figure 4.1: Theoretical PER (a) and efficiency (b) with respect to C for IR-HARQ mechanism in PBS and SBS contexts. ($R_0 = 1/4$, $L_F = 120$ bits, and $N = 3$.)

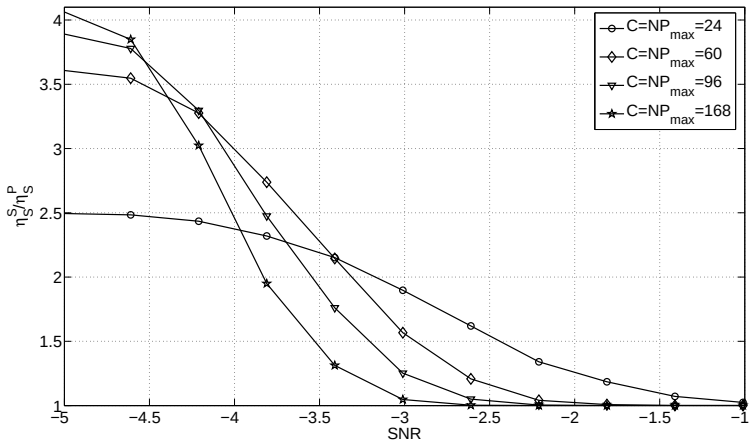

Figure 4.2: Ratio between the SBS efficiency and the PBS efficiency with respect to the SNR for IR-HARQ and different values of C. ($R_0 = 1/4$, $L_F = 120$ bits, and $N = 3$.)

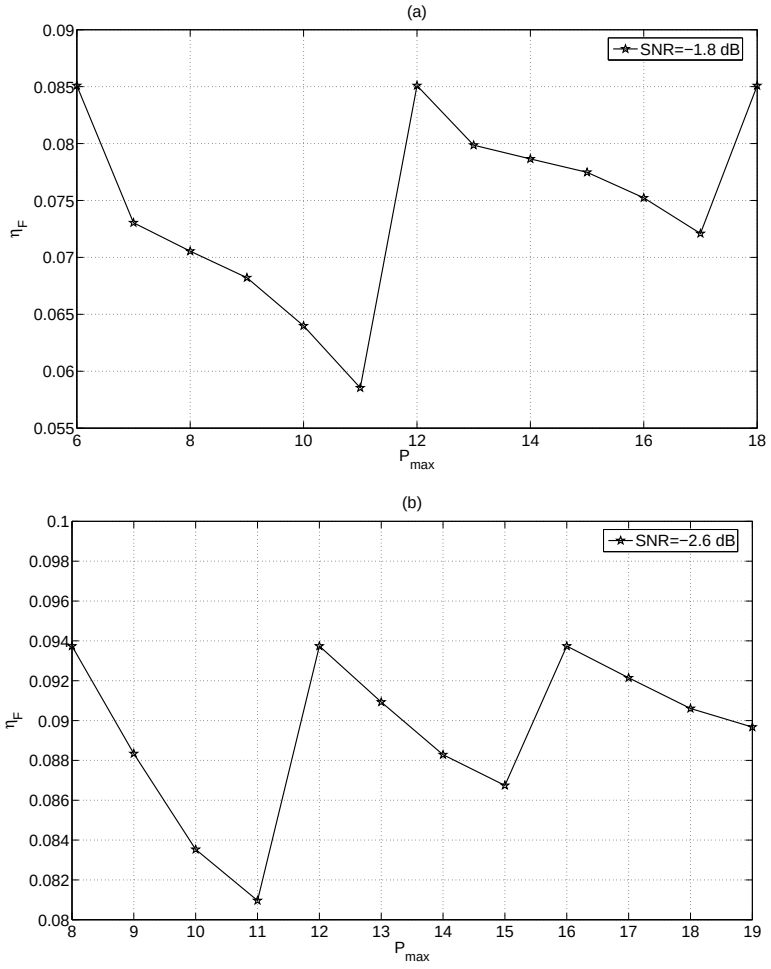

Figure 4.3: Theoretical efficiency for IR-HARQ at MAC layer: (a) $t_0 = 6$ and different DPDU lengths and (b) $t_0 = 4$ and constant DPDU lengths. ($L_F = 320$ bits and $R_0 = 1/4$; for (a), the different rates are 1, 4/5, 2/3, 1/2, 4/11, and 1/4; for (b), the different rates are 1, 1/2, 1/3, and 1/4.)

Let us now focus on the delay and the jitter. In Fig. 4.4, we display the delay and the jitter with respect to C in both PBS and SBS contexts at different SNRs. As expected, if more retransmissions per FRAG are allowed, the delay and the jitter increase. At low SNR, the delay and the jitter seem to increase linearly with respect to C. At medium and high SNR, a floor occurs as soon as C is large enough since the FRAGs are well received without consuming the whole global transmission credit.

To be more precise, in Fig. 4.5, we display the PER versus the delay and the efficiency versus the delay for both cross-layer strategies at different SNRs. We compare the PBS strategy with one given C (actually, $C = 24$) to several SBS strategies designed with different C. In Fig. 4.5.a we show that, for a given SNR, decreasing C gracefully degrades the performance in terms of PER to the benefit of a smaller delay, and thus allows to adjust the desired trade-off between PER and delay at the NET level. We also observe that equivalent performance for PBS and SBS can be achieved with a smaller transmission credit when SBS is prefered to PBS.

In Fig. 4.5.b, as noticed in Fig. 4.1, the efficiency can be slightly improved with the

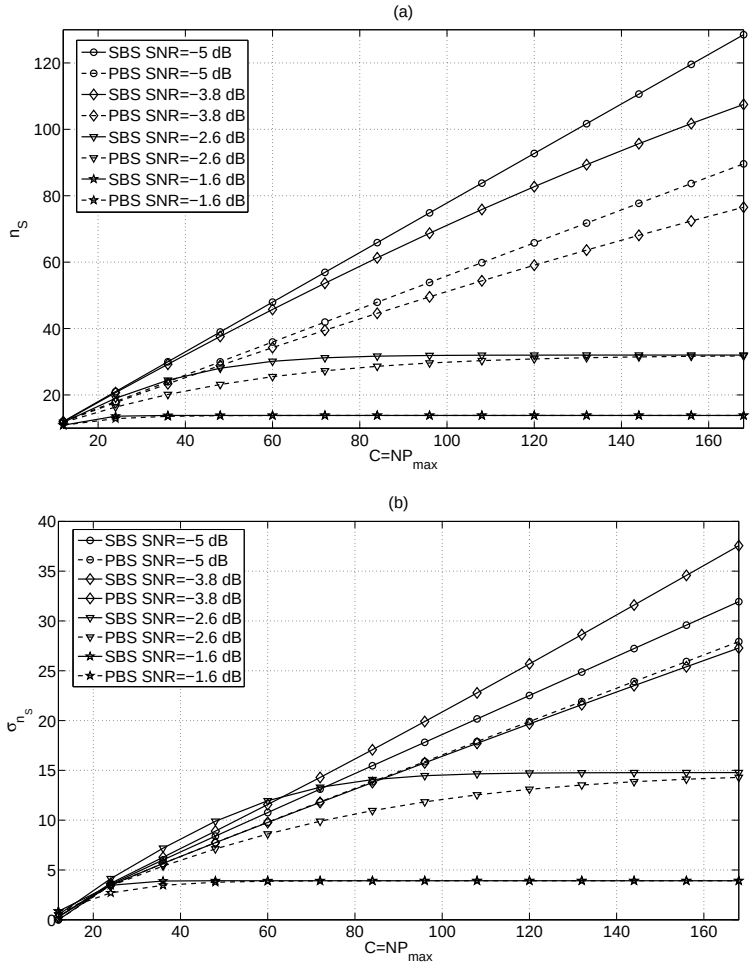

Figure 4.4: Theoretical delay (a) and jitter (b) with respect to C for IR-HARQ and different SNRs. ($R_0 = 1/4$, $L_F = 120$ bits, and $N = 3$.)

SBS strategy at the expense of the delay, or equivalently the PBS and the SBS can offer the same efficiency and almost the same delay but not with the same value of C.

In Tab. 4.1, we summarize the best choices to do in terms of global transmission credit with respect to the four performance metrics and the SNR. We do not comment the results at low SNR since any system does not operate at such a SNR. At medium and high SNR, one can select a medium C since it will ensure a small PER, a high efficiency and reasonable delay and jitter.

4.2.2 Influence of the code rate

In this section, we analyze the influence of R_0 when the global transmission credit per DSDU is fixed and when the number of FRAGs per DSDU is also fixed. Consequently, we consider the following configurations. In PBS context, $C = Nr_0(P + 1)$ is fixed, N is fixed, but r_0 and P may vary. We then analyze the trade-off between the code rate and the persistence. In SBS context, C is fixed, N is fixed, but r_0 may vary. In other words, we seek to know if we have to either transmit more times few different DPDUs or transmit few times more different DPDUs.

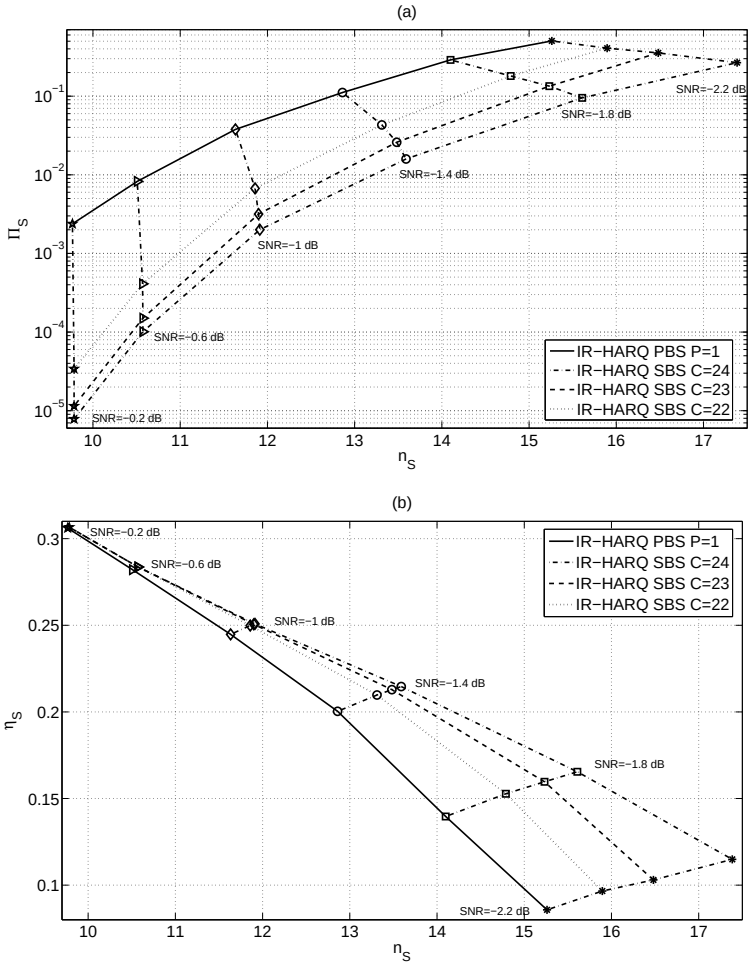

Figure 4.5: PER versus delay (a), efficiency versus delay (b) of the SBS and PBS strategies for IR-HARQ mechanism, different SNRs, and different C. ($N = 3$, $L_F = 120$ bits, and $R_0 = 1/4$.)

	low SNR	medium SNR	high SNR
PER	high C	high C	high C
Efficiency	all C	high C	all C
Delay	low C	low C, medium C	low C, medium C
Jitter	low C	low C, medium C	low C, medium C

Table 4.1: Best transmission credit per DSDU with respect to performance metrics and SNR.

In Fig. 4.6, we represent the PER and the efficiency in SBS and PBS contexts and for different code rates. We observe that it is worth selecting the lowest mother code

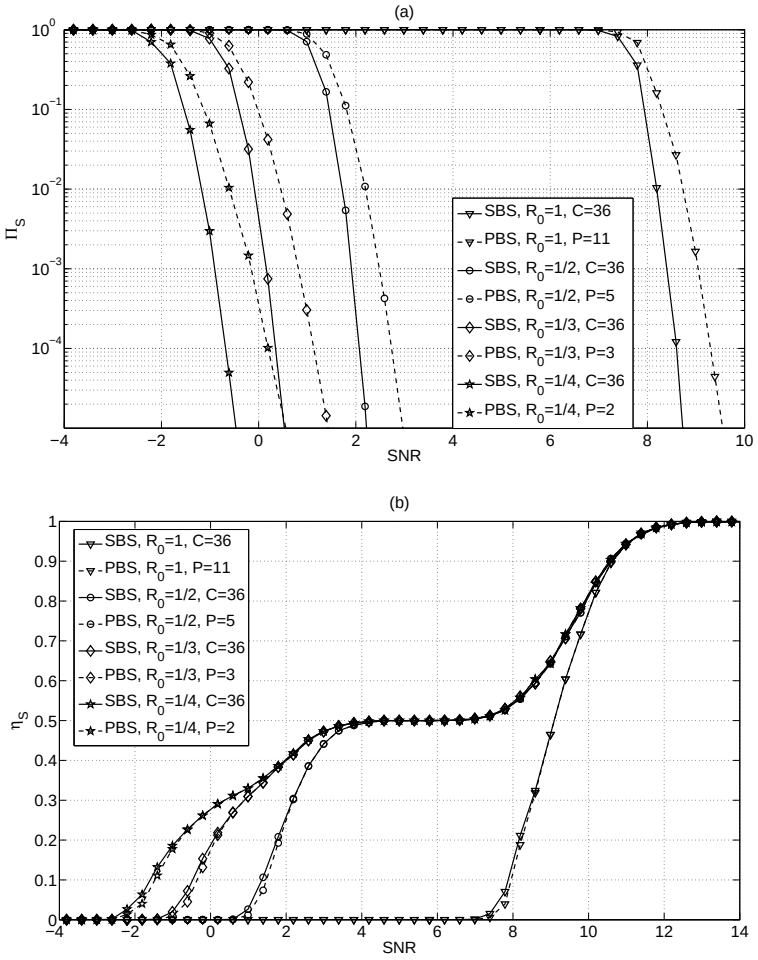

Figure 4.6: Theoretical PER (a) and efficiency (b) versus SNR for IR-HARQ, SBS and PBS contexts, and different R_0. ($N = 3$, $L_F = 320$ bits, and $C = 36$.)

rate in order to improve the PER whatever the cross-layer strategy used. That means that, for a same transmission credit C per DSDU, the redundancy is more relevant than the "simple" repetition for the PER metric. We have the same result concerning the efficiency. It is then interesting to put a large amount of redundancy rather than to grant a large number of "simple" repetitions. Note that the case $R_0 = 1$ can be assimilated to ARQ (even if the DPDU may not only contain information bits) while the other cases ($R_0 < 1$) are really IR-HARQ. It is already known that the IR-HARQ offers a better granularity in terms of efficiency than the simple ARQ. Indeed, in ARQ mechanism, the efficiency is either 0 or 1, and the other values occur but only over a very small SNR interval.

In Fig. 4.7, we respectively plot the delay and the jitter in SBS and PBS contexts and for different code rates. At (very) high SNR, the delay is insensitive to the code rate. Actually, as mentioned in the previous chapter, the delay goes to N when the SNR gets high. Since the number of FRAGs N is the same whatever the mother code rate, so is the delay in this range of SNR. At high and medium SNRs, it is interesting to consider small code rate rather than high transmission credit per FRAG. At (very) low SNR, that is the converse. We have similar phenomenon with the jitter, except at

very low SNR since the lowest jitter is still given by the lowest mother code rate for the both cross-layer approaches.

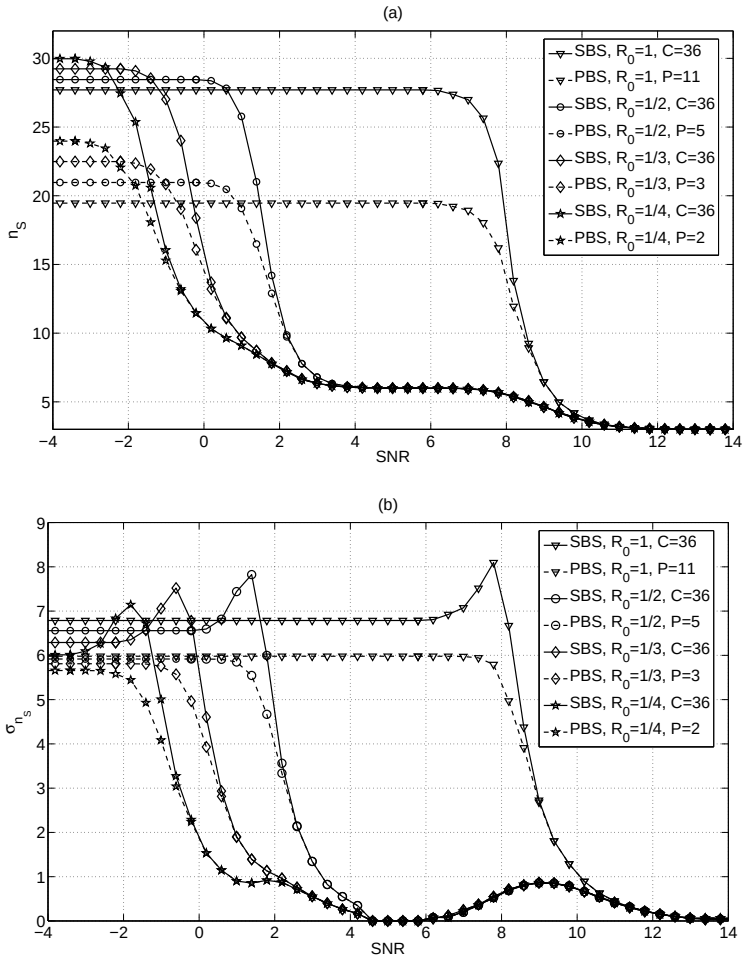

Figure 4.7: Theoretical delay (a) and jitter (b) versus SNR for IR-HARQ, SBS and PBS contexts, and different R_0. ($N = 3$, $L_F = 320$ bits, and $C = 36$.)

As a conclusion, the values of the mother code rate have to be chosen accordingly to Tab. 4.2. Except to very low SNR, we advocate to choose low code rate, that is to say that, the IR-HARQ mechanism really needs to take benefit of its associated error-correcting code.

4.2.3 Influence of the channel coding scheme

Until now, the channel coding schemes used for implementing IR-HARQ mechanism (and even the other HARQ mechanisms studied along the thesis) are based on the RCPC codes introduced in [7]. The error-correcting codes obviously may have an influence on the whole HARQ mechanism performance. Therefore, we propose hereafter to compare the RCPC codes with other interesting error correcting codes such as the LDPC codes. Indeed, it has been shown that the LDPC codes are adapted to IR [57] and offer solutions to improve the performance of HARQ mechanisms based on this concept [58]. However, this section is not dedicated to a thorough analysis of the LDPC codes, but rather to the

	low SNR	medium SNR	high SNR
PER	low R_0	low R_0	low R_0
Efficiency	all R_0	low R_0, medium R_0	all R_0
Delay	high R_0	low R_0	all R_0
Jitter	low R_0	low R_0	all R_0

Table 4.2: Best code rate with respect to performance metrics and SNR.

highlighting of the fact that the closed-form expressions established in Chapter 2 can be applied whatever the considered codes and enable us to easily assess the resulting performance. Even if there are more relevant ways to puncture the LDPC codeword [59], we have considered here random puncturing for simplicity.

It is well-known that these LDPC codes offer very small PER when the codeword length is large enough, *i.e.* when the FRAG is large enough.

We then here compare RCPC and LDPC codes for different values of FRAG length in terms of capacity as defined in Section 3.5: $\kappa = m_s \eta$. For sake of simplicity, but without loss of generality, we only focus on MAC layer, Gaussian channel, and QPSK modulation. As a benchmark, we also plot the Shannon capacity for QPSK input which is given by:

$$\kappa_{\text{QPSK}} \approx 2(1 - 2^{-\gamma}), \tag{4.1}$$

where γ is the SNR [60].

In Fig. 4.8, we plot the capacity versus SNR for both RCPC and LDPC codes when the FRAG (*i.e* the codeword) only contains 64 bits. At this FRAG length, the RCPC

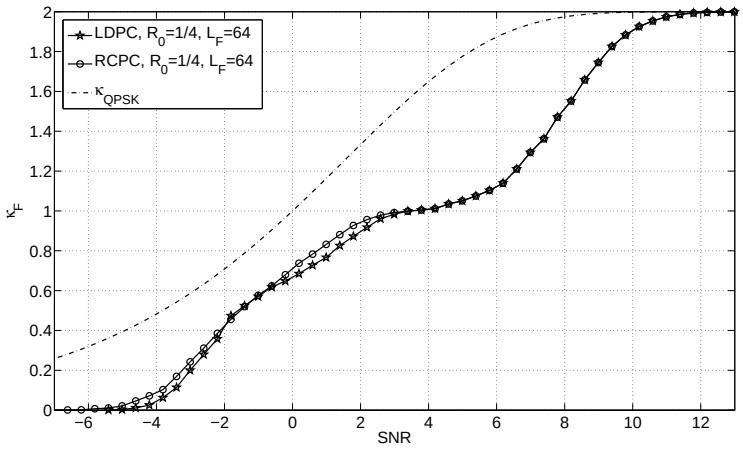

Figure 4.8: Capacity versus SNR for RCPC and LDPC when the FRAG length is equal to 64 bits at the MAC layer. ($P = 11$ and $R_0 = 1/4$.)

codes offer slightly better performance than the LDPC codes whatever the SNR. It is

not a surprise since LDPC codes become to be very interesting when the codeword is large enough.

In Fig. 4.9, we display the same criteria as in Fig. 4.8, but the number of bits in a FRAG is increased and equal to 240 bits. The LDPC performance gets slightly better than the RCPC one as soon as the SNR gets too bad.

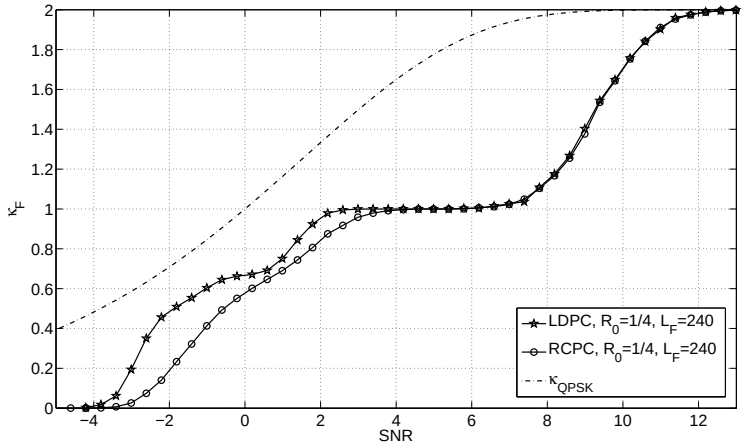

Figure 4.9: Capacity versus SNR for RCPC and LDPC when the FRAG length is equal to 240 bits at the MAC layer. ($P = 11$ and $R_0 = 1/4$.)

In Fig. 4.10, we plot the same metrics as in both previous figures but with the number of bits per FRAG equal to 1920 bits. The gap between RCPC and LDPC still increases specifically at low and medium SNR. The LDPC capacity becomes closer to the Shannon capacity as expected. That confirms that LDPC codes are relevant channel coding scheme candidates for HARQ as soon as the FRAG is large enough.

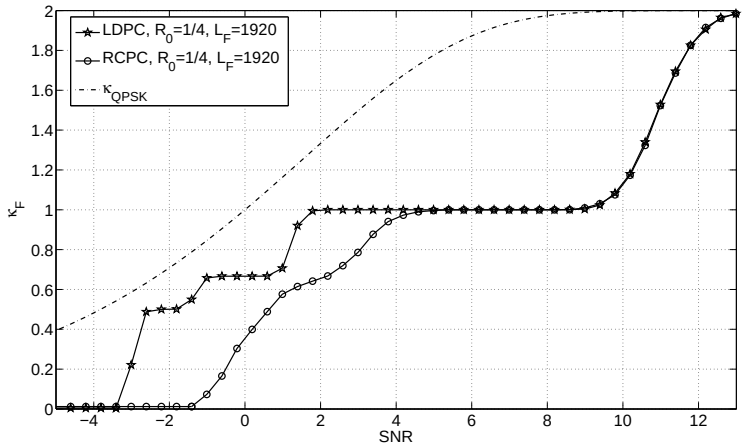

Figure 4.10: Capacity versus SNR for RCPC and LDPC when the FRAG length is equal to 1920 bits at the MAC layer. ($P = 11$ and $R_0 = 1/4$.)

4.2.4 Influence of the number of FRAGs per DSDU

In this section, we evaluate the influence of the fragmentation on the four performance metrics for PBS and SBS contexts. Thus we aim to determine if it is better to split one DSDU into a high number of FRAGs (each with small size) or into a small number of FRAGs (each with large size). Note that we do not into account the fragment overhead.

We consider that the number of bits per DSDU, denoted by L_{DSDU}, is fixed and that the global transmission credit is comparable for both strategies. We then get in PBS context $C = Nr_0(P+1)$, where the inverse of the code rate r_0 and the persistence P are fixed. Consequently, the global transmission credit per DSDU C is proportional to the number of FRAGs per DSDU N. In other words, we assume that:

$$\frac{C}{N} = \text{constant} \tag{4.2}$$

where the constant is equal to 6 since we put $R_0 = 1/2$ and $P = 2$. In SBS context, we also assume that (4.2) means that the "average" number of transmissions per FRAG is fixed. As L_{DSDU} is equal to NL_F and is fixed, increasing the number of FRAGs per DSDU leads to decreasing the number of bits per FRAG.

In Fig. 4.11, we plot the PER and the efficiency versus SNR for different values of N in PBS and SBS contexts. We observe that the PER decreases when the number of FRAGs increases, *i.e* when the FRAG is smaller. Therefore fragmenting the DSDU is interesting in terms of PER. Moreover the gap between the PBS and the SBS increases when N increases. Indeed, the SBS handles better the case when the number of FRAGs is high thanks to the credit sharing among all the FRAGs. In terms of efficiency, as already noticed by [61] at the MAC layer, the number of FRAGS has an impact on this performance metric. We observe on part (b) of the figure that it is a little advantageous to split one DSDU into a high number of FRAGs for several ranges of SNR.

In Fig. 4.12, we display the delay and the jitter versus SNR for different values of N in PBS and SBS contexts. As the delay corresponds to the number of DPDU transmissions needed to successfully transmit one DSDU, it is at least equal to N. Consequently, at high SNR, the higher N, the higher the delay, as observed on the figure. At low and medium SNRs, the delay is once again minimized when N is chosen as small as possible. Note that we compare FRAG delay with different sizes of FRAGs. If we can send the FRAGs successively (without guard time between FRAGs), the transmit duration (in second) is the same if the delay increasing is proportional to N. In PBS context, the delay is proportional to N which implies that the transmission duration is insensitive to N. In contrast, in SBS context, the delay is not proportional to N (except at high SNR), and a careful observation of this figure shows that the transmission duration increases with respect to N. As for the jitter, the lower N, the lower the jitter. Nevertheless at high SNRs, the gap for different values of N is very tiny. In contrast, at medium SNRs, we observe huge gap between small N and high N, specifically in the SBS context. At medium SNRs, we have to choose a sufficiently small N to ensure a reasonable jitter in SBS context.

In Tab. 4.3, we sum up the relevant choices for N with respect to the four performance metrics and the SNR.

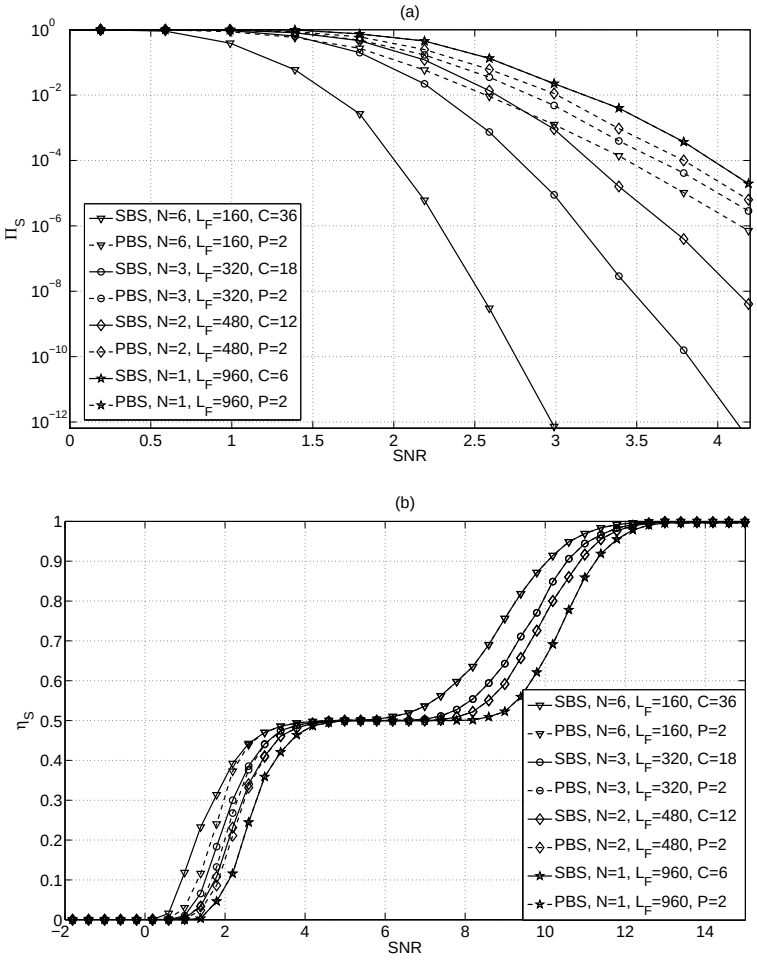

Figure 4.11: Theoretical PER (a) and efficiency (b) versus SNR for IR-HARQ mechanism, PBS and SBS contexts, and different N. ($L_{\text{DSDU}} = 960$ bits, $R_0 = 1/2$, and $C/N = 6$.)

	low SNR	medium SNR	high SNR
PER	high N	high N	high N
Efficiency	all N	high N	high N
Delay	low N	low N	low N
Jitter	low N	low N	all N

Table 4.3: Best number of FRAGs with respect to performance metrics and SNR.

4.2.5 Application to different QoS requirements

In this section, we provide some trends about the policy of radio resource management that we have to follow for three types of data: the voice, the video, and the file transfer. The policy is summed up in Tab. 4.4 and has been fulfilled thanks to Tab. 4.1,

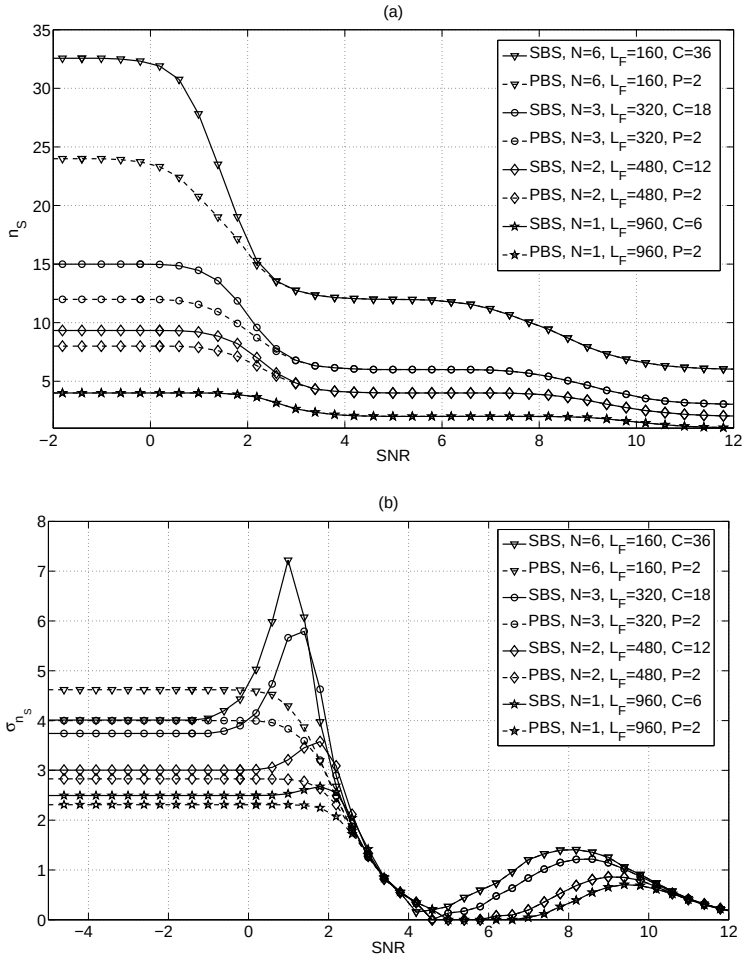

Figure 4.12: Theoretical delay (a) and jitter (b) versus SNR for IR-HARQ mechanism, PBS and SBS contexts, and different N. ($L_{\mathrm{DSDU}} = 960$ bits, $R_0 = 1/2$, and $C/N = 6$.)

Tab. 4.2, and Tab. 4.3 established in previous sections of this chapter. Therefore we do not mention the cross-layer strategies (PBS/SBS) and the error-correcting codes (RCPC/LDPC).

	Low SNR	Medium SNR	High SNR
Voice	low C, high R_0, low N	low C, low R_0, low N	low C, all R_0, low N
Video	low C, low R_0, low N	medium C, low R_0, low N	low C, all R_0, high N
File transfer	high C, low R_0, high N	high C, low R_0, high N	high C, low R_0, high N

Table 4.4: Best radio resource management with respect to application and SNR.

4.3 Unequal data protection

For some applications, it is worth protecting the data differently. For instance, in the video streaming context, the data associated with the first image or the data associated with the low-frequency image decomposition are more important than the data corresponding to the image of the motion compensation or the data corresponding to the high-frequency image decomposition. In addition, in IP context, the IP header has to be read correctly. As a consequence, some data have to be more protected than others [62–64].

For some other applications, we can loose some FRAGs without affecting significantly the system performance. This is the case for the voice which is very robust to packet loss. However, in this case the unequal protection is not necessarily an asset.

In any case, it is therefore of interest to inspect for both strategies:

- the packet loss of FRAG $\#i$,
- the probability to get i erroneous FRAGs out of N FRAGs of the same DSDU.

From this study, we can show that the SBS strategy actually provides by construction an unequal packet loss protection at the FRAG level due to the fact that the FRAGs are transmitted in an ordered manner. As a consequence, the first FRAG to be transmitted is always better protected than the others.

In the PBS, since each FRAG is granted with the same $P_{\max}$ value, the FRAGs are inherently equally likely protected. We then propose to extend the PBS to an unequal packet loss protection scheme by granting different maximum transmission credit per FRAG.

This section is organized as follows : in Section 4.3.1, we derive the probability to successfully receive the FRAG $\#i$ in PBS and SBS context. We discuss the capability of PBS and SBS to enable unequal packet protection. In Section 4.3.2, we derive the probability to successfully receive a given number of FRAGs among N FRAGs.

4.3.1 Probability to receive the FRAG $\#i$

Before deriving the probability to successfully receive the FRAG $\#i$, we roughly justify that SBS scheme is appropriate for doing unequal data protection. The first FRAG can be transmitted as many times as needed, as long as the global transmission credit C is not reached. Once the first FRAG is received without error, the second FRAG is then sent and can be only retransmitted as long as the global transmission credit C minus the credit used by the first FRAG is reached. Therefore, by noting k_1 the number of transmissions used by the first FRAG, there are $C - k_1$ possible transmissions for the second one. By assuming that the second FRAG is successfully received after k_2 transmissions (thus $k_1 + k_2$ transmissions have been already used by the two first FRAGs), the third FRAG can be sent $C - (k_1 + k_2)$ times at maximum. And so on until either the N FRAGs are successfully received or the global transmission credit C is reached. It appears then easily that the first FRAG has the highest probability to be received without errors, since it is more likely to successfully receive one FRAG in C transmissions at maximum than in less than C transmissions at maximum. More generally, it comes that the probability to successfully receive the FRAG $\#i$ is more

important than the one to successfully receive the FRAG $\#(i+1)$. Hence, the cross-layer approach allows an unequal protection of the data and thus, is a judicious choice for the applications presented at the beginning of this section.

Let $R_N^P(i)$ and $R_N^S(i)$ be one minus the probability to receive successfully the FRAG $\#i$ belonging to a DSDU consisting of N FRAGs in PBS and SBS contexts respectively. Let us start with the SBS scheme. The probability to successfully receive the FRAG $\#i$, knowing that the global transmission credit is C, is equivalent to the probability to successfully receive the i first FRAGs of the same DSDU in C transmissions at maximum. Indeed, if the $(i-1)$ previous FRAGs are not received without error, the FRAG $\#i$ would not be transmitted and would be considered as erroneous. We then obtain

$$R_N^S(i) = 1 - \sum_{k=i}^{C} p_i(k), \quad \forall 1 \leq i \leq N. \tag{4.3}$$

In Fig. 4.13, we represent the PER per FRAG index versus the SNR in the SBS scheme. The lines represent the theory and the symbols the simulation. We observe that theoretical and empirical results perfectly match. Furthermore, as previously mentioned, we observe that $R_N^S(i)$ is smaller than $R_N^S(i+1)$ for all SNR intervals. As a conclusion, the more protected FRAGs are the first FRAGs of the DSDU since they could benefit from more transmission credit. Application of this unequal protection property could be used for IP header protection which are more sensitive to errors than the payload for some traffic (video streaming, VoIP). In this case, the IP header should belong to the first FRAG.

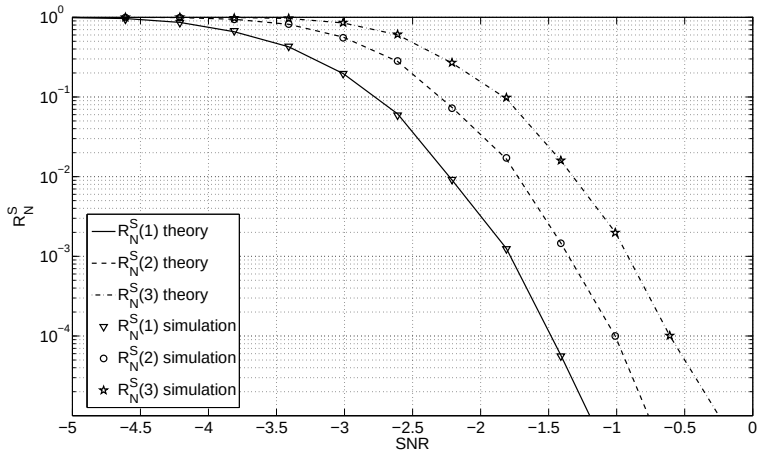

Figure 4.13: Theoretical and empirical PER versus SNR for different FRAG indices in the SBS scheme. ($N = 3$, $L_F = 120$ bits, $R_0 = 1/4$, and $C = 24$.)

In the PBS scheme, each FRAG has the same protection since the same transmission credit per FRAG is guaranteed. We then can obviously write

$$R_N^P(i) = R_N^P = 1 - \sum_{k=1}^{P_{\max}} p_1(k), \quad \forall 1 \leq i \leq N. \tag{4.4}$$

In Fig. 4.14, we represent $R_N^P(i)$ versus the SNR for different i. Theory and practice are in perfect agreement.

We propose here to do unequal data protection in PBS by allowing different transmission credit per FRAG. In such a case, the more protected FRAGs are not necessarily the first ones but those having the highest number of retransmissions. In Fig. 4.15, we illustrate this approach since we plot the theoretical PER per FRAG versus SNR when different values of transmission credit per FRAG are provided. Note that this new scheme, although derived from the PBS, can be cast in the cross-layer category since it considers the fact that the FRAGs are coming from the same IP packet in an ordered manner.

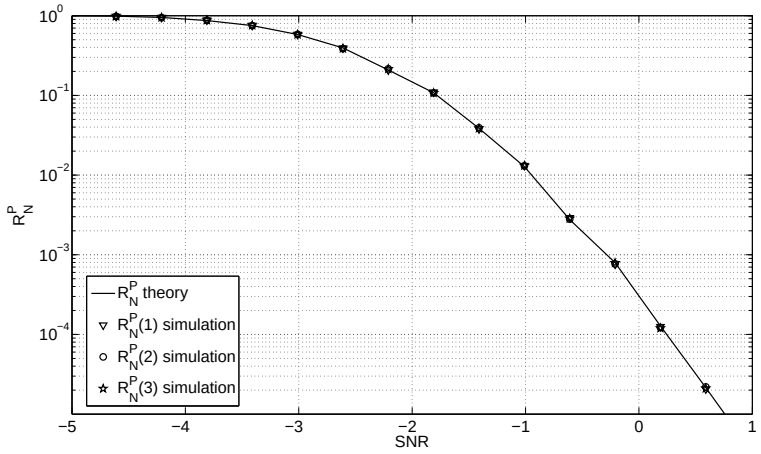

Figure 4.14: Theoretical and empirical PER versus SNR for FRAG having the same transmission credit in the PBS scheme. ($N = 3$, $L_F = 120$ bits, $R_0 = 1/4$, and $P = 1$.)

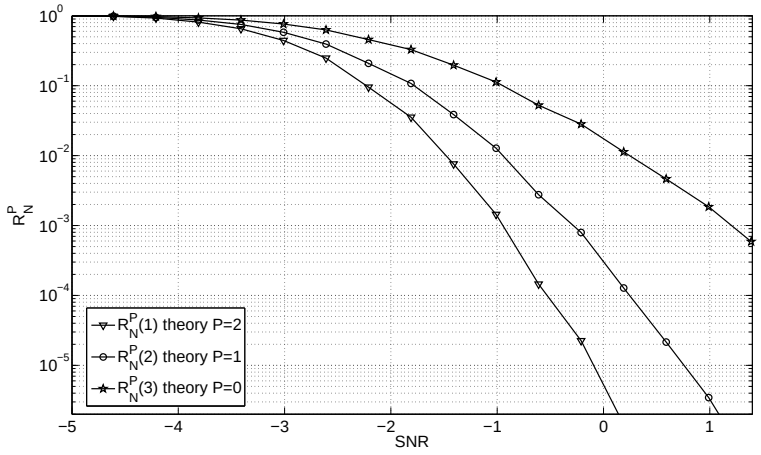

Figure 4.15: Theoretical and empirical PER versus SNR for FRAG having different transmission credit in the PBS scheme. ($N = 3$, $L_F = 120$, $R_0 = 1/4$, and $C = 24$.)

In Fig. 4.16, we merge the theoretical results obtained in Fig. 4.13 and Fig. 4.15. The global transmission credit per DSDU is the same for the SBS and PBS schemes, therefore it is then fair to compare their performance. By noting P_k the number of maximum retransmissions for the FRAG $\#k$ in the PBS case, we have $C = r_0(N +$

$\sum_{k=1}^{N} P_k$). We observe that, with the PBS strategy, it is not possible to achieve the SBS performance in terms of PER per FRAG. Indeed, the SBS adapts the persistence per FRAG according to the channel quality, whereas the persistence is *a priori* fixed in the PBS case. Then, if the PBS persistence for the FRAG $\#i$ is not sufficiently large to allow successful transmission, the FRAG would be erroneous whereas, at the expense of some additional transmissions, it could finally be received without errors in the SBS case. In the same way, if the FRAG $\#i$ is successfully received without having consumed its whole persistence, the reminder of the transmission credit would be wasted in the PBS context, instead of being used for the other FRAGs, if needed, in the SBS context. As a consequence, we get a higher PER for each of the N FRAGs in the PBS strategy.

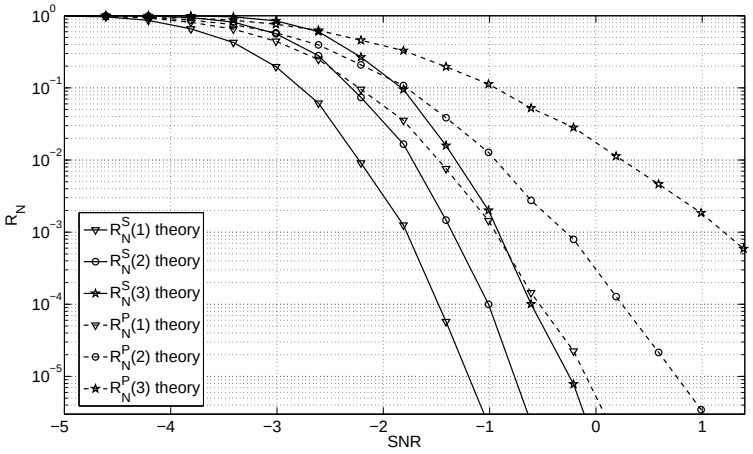

Figure 4.16: Theoretical PER versus the SNR in unequal data protection context. ($N = 3$, $L_F = 120$ bits, $R_0 = 1/4$; in SBS scheme, $C = 24$; in PBS scheme, $P_1 = 2$ for FRAG 1, $P_2 = 1$ for FRAG 2 and $P_3 = 0$ for FRAG 3.)

Note that in order to operate this unequal packet loss protection scheme, we need to let the MAC layer send to the IP level IP datagrams with corrupted FRAGs. We have to modify the SAR accordingly and let him know the maximum number of corrupted FRAGs that it can allow within one IP packet.

4.3.2 Probability to receive i erroneous FRAGs among N

In this section, we study the probability to have a fixed number, noted i, of erroneous FRAGs among the N FRAGs belonging to the same DSDU. By "erroneous FRAG", we mean that the FRAG is received with error or is never transmitted.

Let $S_N^P(i)$ and $S_N^S(i)$ be the probability to have i erroneous FRAGs among N FRAGs in PBS and SBS contexts respectively. Like in the previous section, we start with the SBS scheme. With this cross-layer strategy, the i erroneous FRAGs are necessarily the last ones. Indeed, while a FRAG is not successfully received, the following FRAGs are not sent. Then, the probability to have i erroneous FRAGs is equivalent to the probability to receive only $N-i$ FRAGs in exactly C transmissions or to the probability to receive only $N-i$ FRAGs in less than C transmissions when the FRAG $\#(N-i+1)$

is transmitted but always received with error. In other words, we get

$$S_N^S(i) = \sum_{k=N-i}^{C} p_{N-i}^S(k)(1 - \sum_{k'=1}^{C-k} p_1(k')), \quad \forall 0 < i < N. \tag{4.5}$$

When $i = N$ or $i = 0$, we also can prove in a similar way that

$$S_N^S(N) = 1 - \sum_{k=1}^{C} p_1(k), \tag{4.6}$$

and

$$S_N^S(0) = \sum_{k=N}^{C} p_N^S(k), \tag{4.7}$$

this last one being obviously equal to $1 - \Pi_S^S$ since the probability to have no erroneous FRAG among N is equivalent to the probability to have successfully received all the N FRAGs.

Let us move on the PBS scheme. In PBS, having i erroneous FRAGs among N is equivalent to assert that $N - i$ FRAGs are received without error and that i FRAGs are received with error even after having used all their transmission credits. Receiving a FRAG without error occurs with probability $\sum_{k=1}^{P_{\max}} p_1(k)$. Receiving a FRAG with error occurs with probability $1 - \sum_{k=1}^{P_{\max}} p_1(k)$. Consequently the final expression is

$$S_N^P(i) = \binom{N}{N-i} \left(\sum_{k=1}^{P_{\max}} p_1(k) \right)^{N-i} \left(1 - \sum_{k=1}^{P_{\max}} p_1(k) \right)^{i}, \quad \forall 0 \leq i \leq N. \tag{4.8}$$

The binomial coefficient, $\binom{N}{N-i}$, is due to the fact that the indices of the i erroneous FRAGs can be taken among all the N FRAGs.

In Fig. 4.17, we respectively plot $S_N^S(i)$ and $S_N^P(i)$ versus SNR. Theoretical results (in lines) are identical to empirical ones (in symbols), what shows that the closed-form expressions given by (4.5) and (4.8) are valid.

4.4 Comparison to OMNeT++ simulations

In this section, we compare the performance obtained by our closed-form derivations and the ones computed from a discrete event based simulator using OMNeT++ software as in [65] implementing the same HARQ protocol. This OMNeT++ implementation is very close to real system, including all the signaling and framing. We show on a one-hop link (2 nodes: one transmitter, one receiver) that for the PER we have a direct matching between both approaches. For the delay, we can also achieve a good matching by applying a translation formula (from number of packets to ms) which takes into account the real characteristics of the OMNeT++ implementation. Furthermore, we also simulate a two-hop link (3 nodes: one transmitter, one relay, one receiver) and propose equations to predict, from our closed-form derivations, the PER and the delay for the two-hop model. The prediction is actually quite accurate.

The simulation parameters are as introduced at the beginning of this chapter except for the modulation that is a BPSK.

The OMNeT++ simulation includes the following features:

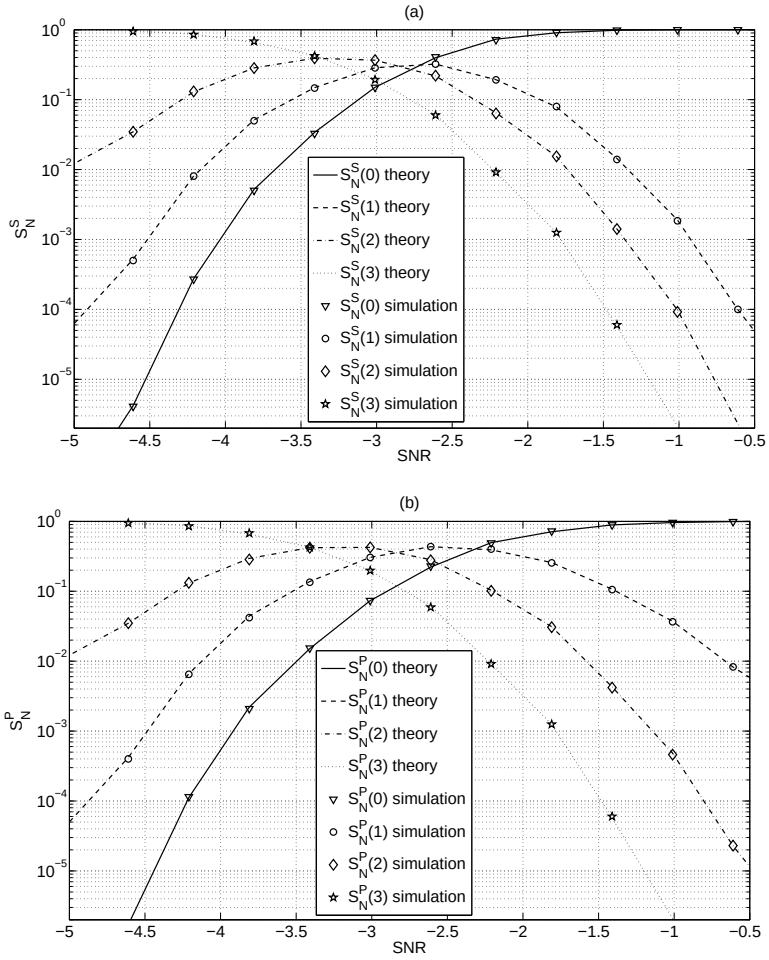

Figure 4.17: Theoretical and empirical probability to have i erroneous FRAGs among N for the SBS (a) and the PBS (b). ($N = 3$, $L_F = 120$ bits, $R_0 = 1/4$; in SBS scheme, $C = 24$; in PBS scheme, $P = 1$.)

- The protocol at the Transport layer is User Datagram Protocol (UDP). That means that no retransmission of the transport packet occurs, even if it has been received with errors at the destination transport layer.
- The sliding window is set to 1 in order to comply with our assumptions.
- The network protocol is the IP protocol, so that the DSDU is an IP packet.
- The IP packet has the following structure:
  - 4 bytes of signalling (SIG),
  - 64 bytes of payload (PL),
  - 8 bytes of UDP header (UDP),
  - 20 bytes of IP header (IP).

The number of payload bytes has been chosen so that the IP packet has a size equivalent to the one of the IP packets in applications like VoIP, but it obviously can

be fixed to an other value. The IP packet is then fragmented in N FRAGs, with N put arbitrarily to 4 in this section. The way the fragmentation is carried out is illustrated on Fig. 4.18. First of all, the IP packet is divided in N blocks of same lengths (4 blocks of 24 bytes on the scheme). To each block are added 9 bytes: 6 bytes of header and 3 bytes necessary for the Convolutional Coding (CCo) (leading here to a total of 33 bytes). The concatenation of the block bytes and of its header bytes is the payload of one FRAG. Thus, each FRAG is constituted by its payload (here, of 30 bytes) and by 3 bytes for the coding.

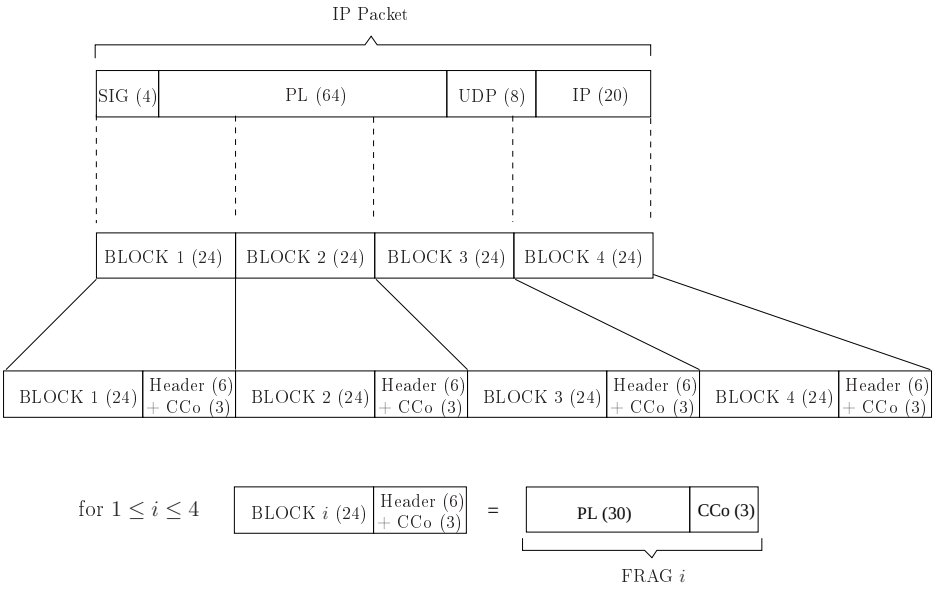

Figure 4.18: Fragmentation of the IP packet in 4 FRAGs.

Once the fragmentation is done, the FRAGs are sequentially transmitted, by applying the SBS or the PBS strategy conveniently, from the first node (N_0) to the second one (N_1). If the N FRAGs are successfully received at the node N_1, before the global transmission credit (C in the SBS case and $NP_{\max}$ in the conventional one) is entirely consumed, they are reassembled in one IP packet. Then, the received network packet is one more time fragmented and the resulting FRAGs are successively sent to the following node (N_2). The same process continues until the last node is reached or until there is no more credit. In the first case, the IP packet is reassembled and sent to the upper layer. In the other one, the IP packet is dropped and the next packet is treated in the same way as above described. In our simulation, there are three nodes, meaning that there is one source, one destination and one relay between them. Lastly, the SNR is assumed to be identical on both links, so that the performance on the first link ($N_0 \rightarrow N_1$) are the same as the performance on the second one ($N_1 \rightarrow N_2$). Fig. 4.19 illustrates this multi-hop context.

We show on Fig. 4.20 that (2.3) and (2.2) match the OMNeT++ simulation results for the PER on one-hop link, and (2.10) and (2.9) for the delay as well. The solid lines correspond to the theoretical results and the dashed lines to the simulation ones.

In the OMNeT++ simulation the delay is expressed in milliseconds (ms), whereas our theoretical delay is given in number of transmitted DPDUs. Moreover we have to take into account the ACK/NACK and the HELLO messages that can use the frame

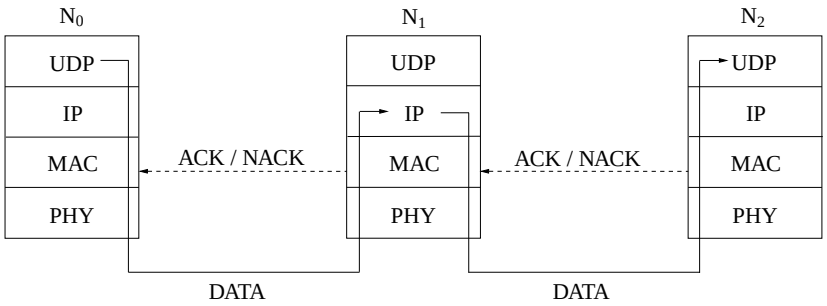

Figure 4.19: Framework.

slot dedicated to the DPDU transmission. Thus we have to find a translation formula that converts the theoretical delay into a delay expressed in ms. This expression is given by:

$$\bar{d}_s^x = \alpha \cdot \bar{n}_S^x \cdot (1+\beta), \tag{4.9}$$

where $\bar{d}_S^x$ is the predicted delay (in ms) for the xBS strategy, α is the average duration per DPDU transmission, and β is the average number of transmissions that are not related to DPDU transmission. One DPDU is transmitted in 3 frames (2 for the DPDU, 1 for the ACK/NACK). Since the frame duration is equal to 10ms, we get $\alpha = 30$ms. The HELLO messages are sent every 11 frames. However, approximately 2/3 of the HELLO messages are sent in the same frame as the ACK/NACK and thus should not be counted. Thus, on the average, $\beta = (1/3) * (1/11)$. One can note that α and β are independent of the cross-layer strategy.

We now study the two-hop case and try to propose equations to predict the PER and delay performance from our closed-form expressions. For the PER metric, assuming independent links, basic reasoning leads to the following expression:

$$\Pi_S^{x,\mathrm{N}_0\to\mathrm{N}_L} = 1 - \prod_{i=0}^{L-1}(1 - \Pi_S^{x,\mathrm{N}_i\to\mathrm{N}_{i+1}}). \tag{4.10}$$

where $\Pi_S^{x,\mathrm{N}_i\to\mathrm{N}_j}$ is the PER at the IP level between the nodes N_i and N_j assuming that the data go all through the $(j-i-1)$ relay nodes from node N_i to node N_j (thus $j-i+1$ nodes). If we assume equal SNR for all links, (4.10) boils down to:

$$\Pi_S^{x,\mathrm{N}_0\to\mathrm{N}_L} = 1 - (1 - \Pi_S^x)^L. \tag{4.11}$$

In the same way, we can guess that the delays accumulate at each link. Thus, we propose to compute the end-to-end delay (in number of transmitted DPDUs) as:

$$\bar{n}_S^{x,\mathrm{N}_0\to\mathrm{N}_L} = \sum_{i=0}^{L-1} \bar{n}_S^{x,\mathrm{N}_i\to\mathrm{N}_{i+1}}, \tag{4.12}$$

where $\bar{n}_S^{x,\mathrm{N}_i\to\mathrm{N}_j}$ is the mean delay at the IP level between the nodes N_i and N_j, which can be simplified for equal SNR on each link into:

$$\bar{n}_S^{x,\mathrm{N}_0\to\mathrm{N}_L} = L\bar{n}_S^x. \tag{4.13}$$

Note that if we want the delays to be expressed in ms to fit the OMNeT++ simulation, one have to resort to the translation formula in (4.9).

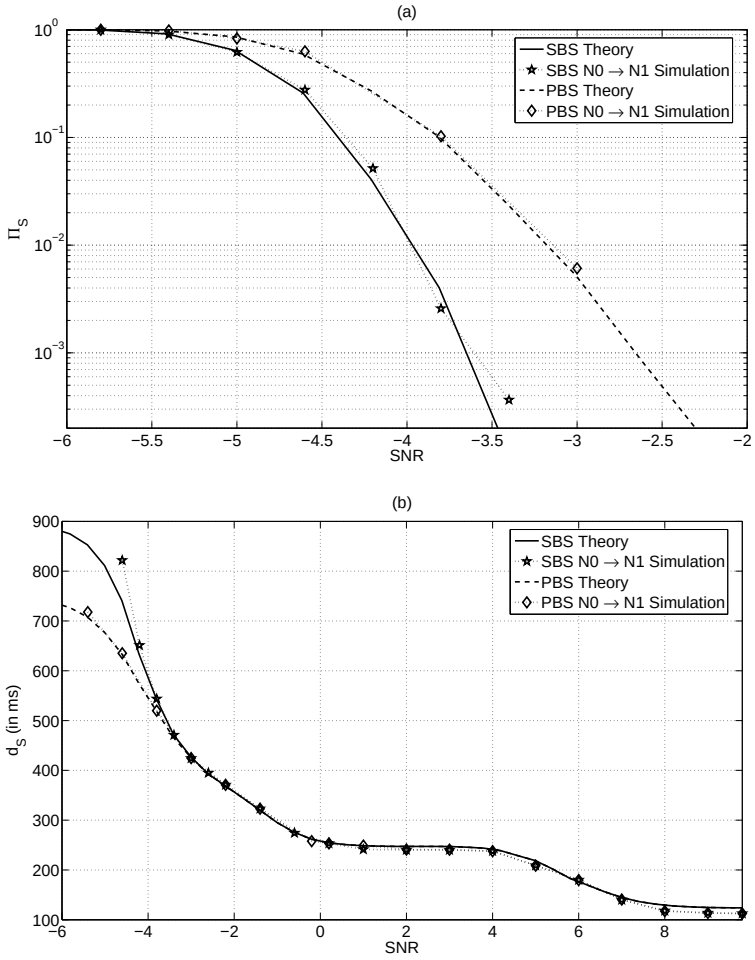

Figure 4.20: Theoretical and simulated PER (a) and delay (b) for one-hop link versus SNR, at the IP level and for both SBS and PBS strategies, with BPSK. ($N = 4$, $L_F = 240$ bits, $R_0 = 1/4$, $C = 32$ (SBS), and $P = 1$ (PBS).)

Fig. 4.21 shows the results of the 2-hop simulation for the PER and delay. We can clearly see that there is a good matching between both the proposed closed-form expressions in (4.11) and (4.13) and the OMNeT++ results.

These results are encouraging in the sense that it seems very probable to predict real system performance from our closed-form equations. One step further would be needed to take into account more realistic assumptions and specifically a sliding window greater than 1 (although it should not impact the PER performance).

4.5 Conclusion

In this chapter, we have focused on the applications of our performance metric derivations to practical issues (radio resource management, unequal packet loss protection and multi-hop context). For the radio resource management, we have deeply analyzed the influence of the following design parameters on the system:

- The global transmission credit ($P_{\max}$ for the PBS and C for the SBS). Judicious

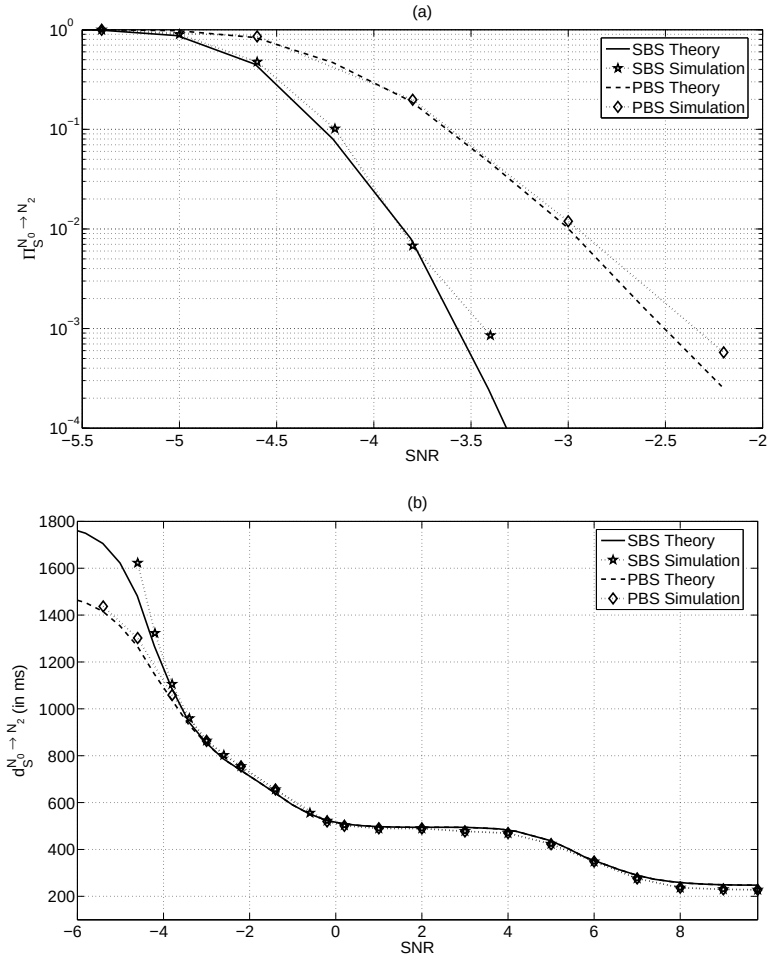

Figure 4.21: Theoretical and simulated PER (a) and delay (b) for two-hop link versus SNR, at the IP level and for both SBS and PBS strategies, with BPSK. ($N = 4$, $L_F = 240$ bits, $R_0 = 1/4$, $C = 32$ (SBS), and $P = 1$ (PBS).)

choices for this parameter are drawn in Tab. 4.1.

- The code rate R_0. Judicious choices for R_0 are drawn in Tab. 4.2.
- The channel coding. For short FRAG, the RCPC offers the same performance as the LDPC but it is less complex. For long FRAG, the LPDC becomes better at the expense of a higher computational load.
- The number of FRAGs per DSDU N. Judicious choices for N are drawn in Tab. 4.3.

We thus have been able to provide a design solution for a given QoS associated with the voice, or the video streaming, or the file transfer for different ranges of SNR. For the unequal packet loss protection, we have shown that it is induced in the cross-layer strategy introduced in [3], that is thus relevant for practical scheme such as multimedia application. We then have extended this concept of unequal packet loss protection to the conventional strategy. Finally, we have shown that we can predict the packet error

rate and delay performance of a discrete event based realistic simulation implementing the HARQ scheme, using the previous derived closed-form expressions.

General Conclusion and Perspectives

The work carried out in the thesis deals with the analysis of the different HARQ schemes, one of the key technique used in the new standards like Wimax, 3G LTE, and 4G. The main objective being to find the best way to satisfy the user needs, we have focused on the HARQ performance in terms of PER, efficiency, delay, and jitter depending on the type of data flow (data, voice, video). Indeed these metrics play an important role in the QoS requirements. Generally, the performance of the retransmission mechanisms are evaluated at the MAC layer, knowing that each MAC packet is considered independently of the others with its own persistence. However, the IP level is more representative of the QoS perceived by the user than the MAC level. That is the reason why we have considered along the thesis a cross-layer concept, based on the fact that the MAC packets are coming from a same IP packet and then are allocated a global transmission credit.

Taking into account the fact that the most relevant way to determine the system behavior is by means of closed-form expressions, those ones giving quickly accurate results compared to approximations or simulations, Chapter 1 has been dedicated to a thorough analysis of the existing theoretical formulas presented in the literature. We then have highlighted two major results. Firstly, some confusions have been made regarding the metrics definition, thus leading to some invalid expressions for the relative metrics. Secondly, the majority of the analytical expressions have been established at the MAC level. Given the complexity of the HARQ computations, the performance at the IP level have been mostly studied by either approximations and simulations or by considering the ARQ protocol only. Furthermore, only the conventional concept has been treated in the case of HARQ schemes.

In order to fulfill the important gap in terms of performance closed-form expressions at the IP level, we have derived in Chapter 2 the four metrics for both conventional and cross-layer strategies. Since the cross-layer approach aims at optimizing the performance at the IP layer, it was sufficient to establish their expressions at the IP layer only. For the classical approach, the performance have been given at the MAC and IP levels. The theoretical formulas we have established are valid for any type of HARQ, modulation and propagation channel. Moreover, they can be applied regardless of the length of the MAC packets, that can be different when using the incremental redundancy concept. The main part of the closed-form expressions presented in this chapter are totally new because of the difficult derivation of the HARQ schemes. In addition, the analytical expressions have been derived into two particular cases: MAC packets with constant

size and type I HARQ scheme. We also have discussed about the independence of the efficiency at the MAC level for incremental redundancy based schemes with regard to the persistence parameter. We then have theoretically proven that the independence is obtained when the transmission credit per FRAG is a multiple of the number of DPDUs resulting from the FRAG encoding. Finally we have demonstrated that the PER at the NET level is always lower in the cross-layer context than in the conventional one for comparable parameters.

By means of the numerical evaluations of the closed-form expressions of the four performance metrics, we have studied in Chapter 3 the influence of the HARQ mechanism, the cross-layer strategies, the modulation, and the propagation channel on the system performance. First of all, we have illustrated the perfect matching between theoretical and empirical results for several system configurations. Then, depending on the application the system is used for, we have noticed that, even if the best model is the IC-HARQ (and this regardless of the user needs), the CC-HARQ or the IR-HARQ can also achieve good performance. Furthermore, the analysis of the both cross-layer strategies has enabled us to conclude that systems requiring small PER and high efficiency are more satisfied by using the SBS whereas for real-time systems needing small jitter, it is more relevant to adopt the PBS scheme. We also have investigated the influence of the modulation and of the propagation channel. Thus, the reader may simply refer to Tab. 3.2, Tab. 3.4, and Tab. 3.6 to select the best configuration (retransmission scheme, cross-layer strategy, modulation) for its system according to the desired QoS.

Finally we have focused on some applications our derivations framework can be used for: radio resource management, unequal data protection, and multi-hop network. For the radio resource management, we have thoroughly analyzed the influence of the following design parameters on the system: the global transmission credit ($P_{\max}$ for the PBS and C for the SBS), the code rate R_0, the channel coding, and the number of FRAGs per DSDU N. The reader is invited to refer to Tab. 4.1, Tab. 4.2 and Tab. 4.3 where he will find the most relevant parameters choice according to its QoS requirements. Thanks to the derivations framework developed in Chapter 2, we also have highlighted the fact that the unequal protection of the data is inherent to the SBS scheme but not to the conventional one. We then have proposed a solution design for the PBS scheme so that such a technique can be applied with it. Lastly, we have shown that our closed-form expressions are valid for one link in a multi-hop network and enable us to easily predict its global performance from the source to the destination.

The works completed in the thesis have raised a number of problematics which should deserve to be treated in the future. These problematics are listed below.

- Regarding the established analytical expressions, they all have been expressed as function of the PERs π_j (as a reminder, π_j is the probability that the $(j+1)^{\text{th}}$ DPDU transmission associated with one FRAG fails, given that the j previous DPDU transmissions associated with the same FRAG failed). Those PERs depend on the retransmission scheme, the size of the MAC packets, the modulation, the propagation channel and so on. They then have to be estimated by means

of simulations, as soon as one of the characteristics quoted just above changes. Although we do not have to simulate the segmentation and reassembly process to compute the π_j, the programming of the HARQ mechanism has to be done. It would then be very helpful to find the theoretical formulas or tight bounds for the π_j. Such analytical expressions would avoid a high number of time-consuming simulations.

- We have just touched upon the capability of the HARQ to provide an unequal packet protection in a cross-layer strategy, giving the basic performance equations. Further work needs to be carried out to set up strategies in the case of specific problems such as IP header protection and video streaming in relationship with existing solutions driven by the application layer.

- Regarding the multi-hop context, it would be interesting to extend the studies carried out for the PER and the delay to the efficiency and the jitter. Thus, we would be able to fully determine what is the QoS perceived by the user in a multi-hop context without resorting to time-consuming simulations. In other terms, we would have a perfect knowledge of the system behavior from the Radio Access layers to the Transport layer, what would reinforce the cross-layer optimization.

Bibliography

[1] T. Kwon, S. Cho, H. Lee, S. Choi, J. Kim, S. Yun, W. H. Park, D. H. Cho, and K. H. Kim, "Design and implementation of simulator based on cross-layer protocol between MAC and PHY layers in WiBro compatible 802.16e OFDMA systems," *IEEE Communications Magazine*, vol. 43, pp. 136–146, Dec. 2005.

[2] K. C. Beh, A. Doufexi, and S. Armour, "Performance evaluation of hybrid ARQ schemes of 3GPP LTE OFDMA system," in *IEEE Personal, Indoor and Mobile Radio Communications (PIMRC)*, (Athens, Greece), Sept. 2007.

[3] Y. Choi, S. Choi, and S. Yoon, "MSDU-based ARQ scheme for IP-level performance maximization," in *IEEE Global Telecommunications Conference (GLOBECOM)*, (Saint-Louis, Missouri, USA), Oct. 2005.

[4] A. S. Tanenbaum, *Computer networks*. New Jersey, USA: Prentice Hall, 4th ed., 2002.

[5] D. Costello and S. Lin, *Error Control Coding*. New Jersey, USA: Pearson Higher Education, 2nd international ed., 2004.

[6] D. Mandelbaum, "An adaptive-feedback coding scheme using incremental redundancy," *IEEE Transactions on Information Theory*, vol. 20, pp. 388–389, May 1974.

[7] J. Hagenauer, "Rate-Compatible Punctured Convolutional codes (RCPC) and their applications," *IEEE Transactions on Communications*, vol. 36, pp. 389–400, Apr. 1988.

[8] A. S. Barbulescu, "Rate compatible turbo-codes," *IEEE Electronics Letters*, vol. 31, pp. 535–536, Mar. 1995.

[9] S. Sesia, G. Caire, and G. Vivier, "Incremental redundancy hybrid ARQ schemes based on Low-Density Parity-Check codes," *IEEE Transactions on Information Theory*, vol. 52, pp. 1311–1321, Aug. 2004.

[10] S. Kallel, "Complementary Punctured Convolutional (CPC) codes and their applications," *IEEE Transactions on Communications*, vol. 43, pp. 2005–2009, June 1995.

[11] M. Moh, T.-S. Moh, and Y. Shih, "On enhancing WiMAX hybrid ARQ: a multiple-copy approach," in *IEEE Consumer Communications and Networking Conference (CCNC)*, (Las Vegas, Nevada, USA), Jan. 2008.

[12] F. Adachi, S. Ito, and K. Ohno, "Performance analysis of a time diversity ARQ in land mobile radio," *IEEE Transactions on Communications*, vol. 37, pp. 177–187, Feb. 1989.

[13] H. van Duuren, "Error probability and transmission speed on circuits using error detection and automatic repetition of signals," *IRE Transactions on Communication Systems*, vol. 9, pp. 38–50, Mar. 1961.

[14] J. Ha, J. Kim, and S. T. McLaughlin, "Rate-compatible puncturing of Low-Density Parity-Check codes," *IEEE Transactions on Information Theory*, vol. 50, pp. 2824–2836, Nov. 2004.

[15] U. Dammer, E. Naroska, S. Schmermbeck, and U. Schwiegelshohn, "A data puncturing IR-scheme for type II hybrid ARQ protocols using LDPC codes," in *IEEE Global Telecommunications Conference (GLOBECOM)*, (Dallas, Texas, USA), Nov. 2004.

[16] Q. Chenand and P. Fan, "On the performance of type III hybrid ARQ with RCPC codes," in *IEEE Personal, Indoor and Mobile Radio Communications (PIMRC)*, (Beijing, China), Sept. 2003.

[17] Z. Jinwen, Y. Linlin, W. Wenbao, and L. Yuan'an, "Parallel hybrid ARQ for OFDM based broadband wireless packet access," in *IEEE International Conference on Communications, Circuits and Systems (ICCCAS)*, (Chengdu, China), June 2004.

[18] G. D. Papadopoulos, G. K. Koltsidas, and F.-N. Pavlidou, "Two hybrid ARQ algorithms for reliable multicast communications in UMTS networks," *IEEE Communications Letters*, vol. 10, pp. 260–262, Apr. 2006.

[19] D. Chase, "Code Combining - A maximum likelihood decoding approach for combining an arbitrary number of noisy packets," *IEEE Transactions on Communications*, vol. 33, pp. 385–393, May 1985.

[20] Q. Zhang and S. A. Kassam, "Hybrid ARQ with selective combining for fading channels," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 867–880, May 1999.

[21] M. Park, B. Keum, M. Lee, and H. S. Lee, "A selective HARQ scheme operating based on channel conditions for high speed packet data transmission systems," in *IEEE Personal, Indoor and Mobile Radio Communications (PIMRC)*, (Athens, Greece), Sept. 2007.

[22] L. Cai, Y. Wan, P. Song, and L. Gui, "Improved HARQ scheme using channel quality feedback for OFDM systems," in *IEEE Vehicular Technology Conference (VTC)*, (Milan, Italy), May 2004.

[23] A. Roongta and J. M. Shea, "Reliability-based hybrid ARQ and Rate-Compatible Punctured Convolutional (RCPC) codes," in *IEEE Wireless Communications and Networking Conference (WCNC)*, (Atlanta, Georgia, USA), Mar. 2004.

[24] Y. Zhou and J. Wang, "Optimum subpacket transmission for hybrid ARQ systems," *IEEE Transactions on Communications*, vol. 54, pp. 934–942, May 2006.

[25] J.-F. Cheng, Y.-P. E. Wang, and S. Parkvall, "Adaptive incremental redundancy," in *IEEE Vehicular Technology Conference (VTC)*, (Orlando, Florida, USA), Oct. 2003.

[26] R. H. Deng and M. L. Lin, "A type I hybrid ARQ system with adaptive code rates," *IEEE Transactions on Communications*, vol. 43, pp. 733–737, Apr. 1995.

[27] A. Avudainayagam, J. M. Shea, and A. Roongta, "Improving the efficiency of reliability-based hybrid ARQ with convolutional codes," in *IEEE Military Communications Conference (MILCOM)*, (Atlantic City, New Jersey, USA), Oct. 2005.

[28] V. Tripathi, E. Visotsky, R. Peterson, and M. Honig, "Reliability-based type II hybrid ARQ schemes," in *IEEE International Conference on Communications (ICC)*, (Anchorage, Alaska, USA), May 2003.

[29] D. Lee, K. Ko, C. H. Nam, and H. S. Lee, "A hybrid ARQ scheme using packet error prediction on OFDM based HSDPA systems," in *IST Mobile and Wireless Communications Summit*, (Budapest, Hungary), July 2007.

[30] S. Lin and P. S. Yu, "A hybrid ARQ scheme with parity retransmission for error control of satellite channels," *IEEE Transactions on Communications*, vol. 30, pp. 1701–1719, July 1982.

[31] G. Almes, S. Kalidindi, and M. Zekauskas, *A one-way delay metric for IPPM*. USA: RFC Editor, 1999.

[32] W. Yafeng, Z. Lei, and Y. Dacheng, "Performance analysis of type III HARQ with turbo codes," in *IEEE Vehicular Technology Conference (VTC)*, (Jeju, Korea), Apr. 2003.

[33] F. Chiti and R. Fantacci, "A soft combining hybrid ARQ technique applied to throughput maximization within 3G satellite IP networks," *IEEE Transactions on Vehicular Technology*, vol. 56, pp. 594–604, Mar. 2007.

[34] M. W. E. Bahri, H. Boujernaa, and M. Siala, "Performance comparison of type I, II and III hybrid ARQ schemes over AWGN channels," in *IEEE International Conference on Industrial Technology (ICIT)*, (Hammamet, Tunisia), Dec. 2004.

[35] Q. Yang and V. K. Bhargava, "Delay and Coding gain analysis of a truncated type II HARQ protocol," *IEEE Transactions on Vehicular Technology*, vol. 42, pp. 22–32, Feb. 1993.

[36] C. F. leanderson and G. Caire, "The performance of incremental redundancy schemes based on convolutional codes in the block fading gaussian collision channel," *IEEE Transactions on Wireless Communications*, vol. 3, pp. 843–854, May 2004.

[37] J.-W. Cho, Y. Chang, Y. Kim, and J. Chon, "Analytic Optimization of hybrid ARQ performance in wireless Packet Data Systems," in *IEEE Global Telecommunications Conference (GLOBECOM)*, (San Francisco, California, USA), Nov. 2006.

[38] Y. Yu, J. Hong, and G. Yue, "The effect of hybrid ARQ schemes on the TCP connection over wireless link," in *IEEE Wireless Communications and Networking Conference (WCNC)*, (Hong-Kong, China), Mar. 2007.

[39] D. Tuninetti and G. Caire, "The throughput of hybrid ARQ protocols for the gaussian collision channel," *IEEE Transactions on Informations Theory*, vol. 47, pp. 1971–1988, July 2001.

[40] M. Zorzi and R. Rao, "On the use of Renewal Theory in the analysis of ARQ protocols," *IEEE Transactions on Commununications*, vol. 44, pp. 1077–1081, Sept. 1996.

[41] M. Zorzi and R. Rao, "Throughput of Selective-Repeat ARQ with time diversity in Markov channels with unreliable feedback," *Wireless Networks*, vol. 2, pp. 63–75, Jan. 1996.

[42] S. Kallel, R. Link, and S. Bakhtiyari, "Throughput performance of memory ARQ scheme," *IEEE Transactions on Vehicular Technology*, vol. 48, pp. 891–899, May 1999.

[43] L. Badia, M. Levorato, and M. Zorzi, "Markov analysis of Selective Repeat type I HARQ using block codes," *IEEE Transactions on Communications*, vol. 56, pp. 1434–1441, Sept. 2008.

[44] Q. Chenand and P. Fan, "Extended SNR and its application to the performance analysis of hybrid ARQ with code combining system," in *Parallel and Distributed Computing, Applications and Technologies (PDCAT)*, (Beijing, China), Sept. 2003.

[45] B. B. Jémenez and A. G. Amat, "Performance analysis of a type II hybrid ARQ protocol based on RCPC codes for the IEEE 802.11a random access MAC protocol," in *IEEE Vehicular Technology Conference (VTC)*, (Stockholm, Sweden), June 2005.

[46] S. Kallel, "Analysis of a type II hybrid ARQ scheme with code combining," *IEEE Transactions on Communications*, vol. 38, pp. 1133–1137, Aug. 1990.

[47] F. Filali, "Link-layer fragmentation and retransmission impact on TCP Performance in 802.11-based networks," in *IFIP Mobile and Wireless Communications Networks Conference (MWCN)*, (Marrakech, Marrocco), Sept. 2005.

[48] J.-B. Seo, N.-H. Park, H.-W. Lee, and C.-H. Cho, "Impact of an ARQ scheme in the MAC/LLC layer on upper-layer packet transmissions over a markovian channel," in *IEEE Vehicular Technology Conference (VTC)*, (Melbourne, Australia), May 2006.

[49] F. Vacirca, A. D. Vendictis, A. Todini, and A. Baiocchi, "On the effects of ARQ mechanisms on TCP performance in wireless environments," in *IEEE Global Telecommunications Conference (GLOBECOM)*, (San Francisco, California, USA), Dec. 2003.

[50] F. Vacirca, A. D. Vendictis, and A. Baiocchi, "Optimal design of hybrid FEC/ARQ schemes for TCP over wireless links with Rayleigh fading," *IEEE Transactions on Mobile Computing*, vol. 5, pp. 289–302, Apr. 2006.

[51] J. Dunlop and M. Mzyece, "Influence of incremental redundancy on the performance of TCP in enhanced general packet radio service," *IET Communication*, vol. 1, pp. 1048–1055, Oct. 2007.

[52] I. S. Gradshteyn and M. Ryzhik, *Tables of integrals, series, and products*. Florida, USA: Academic Press, 6th ed., 2000.

[53] J. P. Buzen, "Computational algorithms for closed queuing networks with exponential servers," *Communication of the ACM*, vol. 16, pp. 527–531, Apr. 1973.

[54] P. Frenger, S. Parkvall, and E. Dahlman, "Performance comparison of HARQ with Chase combining and incremental redundancy for HSDPA," in *IEEE Vehicular Technology Conference (VTC)*, (Atlantic City, New Jersey, USA), Nov. 2001.

[55] V. Hajovsky, K. Kotuliakova, and I. Kotuliak, "HARQ schemes for HSDPA - analysis and simulation," in *ELMAR*, (Zadar, Croatia), Sept. 2008.

[56] A.-N. Assimi, C. Poulliat, and I.Fijalkow, "Diversity Techniques for Single-Carrier Packet Retransmissions over Frequency-Selective Channels," *EURASIP Journal on Wireless Communications and Networking*.

[57] M. Levorato and M. Zorzi, "Performance analysis of type II hybrid ARQ with Low-Density Parity-Check codes," in *International Symposium on Communications Control and Signal Processing (ISCCSP)*, (Saint-Julians, Malta), Mar. 2008.

[58] J. Kim, H. Woonhaing, A. Ramamoorthy, and S. W. McLaughlin, "Design of rate-compatible irregular LDPC codes for Incremental Redundancy Hybrid ARQ systems," in *IEEE International Symposium on Information Theory*, (Seattle, Washington, USA), July 2006.

[59] G. Richter, S. Stiglmayr, and M. Bossert, "Optimized asymptotic puncturing distributions for different LDPC code constructions," in *IEEE International Symposium on Information Theory*, (Seoul, Korea), July 2009.

[60] L. Weidnog, Y. Hongwen, and Y. Dacheng, "Approximation formulas for the symmetric capacity of M-ary modulations," *Science Paper Online*, Apr. 2007.

[61] D. Yang and W. C. Lee, "Adaptive hybrid Automatic Repeat reQuest (ARQ) with a novel packet reuse scheme for wireless communications," in *International Conference on Advanced Communication Technology (ICACT)*, (Gangwon-Do, Korea), Feb. 2007.

[62] A. E. Mohr, E. A. Riskin, and R. E. Ladner, "Unequal loss protection: graceful degradation of image quality over packet erasure channels through Forward Error Correction," *Journal on Selected Areas in Communications*, vol. 18, pp. 819–828, June 2000.

[63] C. Mingyu and M. N. Murthi, "Optimized unequal error protection for Voice over IP," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, (Montreal, Quebec, Canada), May 2004.

[64] X. Zhang and X. Peng, "An unequal packet loss protection scheme for H.264/AVC video transmission," in *IEEE International Conference on Information Networking ICOIN*, (Busan, Korea), Jan. 2009.

[65] R. Massin, C. Lamy-Bergot, C. J. L. Martret, and R. Fracchia, "OMNeT++ based cross-layer simula-tor for content transmission over wireless ad hoc networks," *EURASIP Journal on Wireless Communications and Networking*, Nov. 2009.