Security for wireless communications

Sarah Kamel

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

2017-ENST-0011

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

Télécom-ParisTech

Spécialité « Communications et Électronique »

présentée et soutenue publiquement par

Sarah KAMEL

le 10 Mars 2017

Sécurité pour les réseaux sans fil

Directeur de thèse : **Ghaya REKAYA-BEN OTHMAN**
Encadrant de thèse : **Mireille SARKISS**

Mme. Mari KOBAYASHI, Professeur, Centrale Supélec	Rapporteur
M. Cong LING, Professeur, Imperial College	Rapporteur
M. Daniel AUGOT, Directeur de Recherche, INRIA	Examinateur
M. Mérouane DEBBAH, Professeur, Huawei Technologies - Centrale Supélec	Examinateur
M. Ligong WANG, Chargé de Recherche, ENSEA Cergy-Pontoise	Examinateur
Mme. Michèle WIGGER, Maître de Conférences HDR, Télécom-ParisTech	Examinateur
Mme. Ghaya REKAYA-BEN OTHMAN, Professeur, Télécom-ParisTech	Directeur de thèse
Mme. Mireille SARKISS, Docteur Ingénieur-Chercheur, CEA LIST	Encadrant de thèse

Télécom-ParisTech
Grande école de l'Institut Mines-Télécom - membre fondateur de ParisTech
46 rue Barrault 75013 Paris - (+33) 1 45 81 77 77 - www.telecom-paristech.fr

THÈSE

Acknowledgements

I would like to take the opportunity to thank all those who supported me during this experience.

First, I would like to thank my PhD advisors, Professor Ghaya Rekaya Ben Othman and Doctor Mireille Sarkiss, for their consistent guidance and support throughout this thesis. I have an immense respect, appreciation and gratitude towards Doctor Michèle Wigger for sharing her invaluable knowledge, experience and advice which have been greatly beneficial to the successful completion of my thesis. I am also very grateful to Professor Joseph Boutros for his appreciable help and valuable discussions.

I would like to extend my gratitude to the reviewers, Professor Mari Kobayashi and Professor Cong Ling for accepting to evaluate my thesis. My earnest thanks are also due to Doctor Daniel Augot, Professor Mérouane Debbah and Doctor Ligong Wang for accepting to be a member of the jury.

Thanks to all my colleagues in LSC team, especially Babis, Wael, Mathias, Ibrahim, Nourhene, Kim, Siwar, Sami, Mounir, Antoine, Pierre and Mohamed, for creating a great work atmosphere. Special thanks to Alexis for his endless support, valuable help and technical and non-technical discussions. Thank you Alexis for being a source of motivation, optimism and positive energy.

Thanks also to my colleagues in Télécom ParisTech Elie, Asma, Joe, Hussein, Abir, Wiem, Taghrid, Chadi, Antonio, Achraf, Alaa, Louise, Mohamed, Ehsan and all those I forgot to mention.

I would also like to thank my friends Laure, Jad, Hadi, Marcelino and Damien for their support and all our happy get-togethers and for uplifting my spirit during the challenging times faced during this thesis.

I am also grateful to my cousins Rima, Nour and Sami, and to my friends, Nathalie,

Alex, Nancy, Serge and Roy for their priceless friendship and precious support, and for continuously giving me the courage to make this thesis better.

Last but definitely not least, I am forever indebted to my awesomely supportive parents and my wonderful brother Joseph for inspiring and guiding me throughout the course of my life, and hence this thesis.

Abstract

Over the past few years, the internet has started taking an ever growing place in our daily lives to become today the most used communication mean. This amplifies the need to strengthen the communication security by finding new techniques to protect the confidentiality of data during transfer and storage. These techniques should also anticipate the development in quantum computing and the eventual attacks arising from it.

Cryptography techniques are the basis for security protocols covering most layers of the communication system except for the physical layer. Physical layer security techniques rely on different principles than cryptography. Indeed, physical layer security exploits the specific nature of the used physical medium to protect data confidentiality while cryptography techniques rely on the computational complexity of mathematical problems.

This thesis on wireless communication security is divided into two parts. In the first part of this work, we focus on lattice-based public-key cryptography which is among the most promising techniques for the post-quantum cryptography systems. In particular, we focus on the Goldreich-Goldwasser-Halevi (GGH) cryptosystem which has been widely developed due to its simple encryption and decryption procedures. Despite many existing GGH improvements, its huge public key size remains its main drawback, which prevents the system from being used in practice. In order to overcome this drawback, we propose a new GGH cryptosystem using generalized low density lattices. Indeed, we show that this new GGH reduces the key size by one order of magnitude. In addition, we show that the key generation complexity as well as those of the encryption and decryption phases are significantly decreased. The security of this new GGH system is highlighted through a security analysis that reviews all known attacks on GGH systems. This allows us to conclude that our scheme does not add any new vulnerability as compared with the existing GGH schemes.

In the second part of this work, we study the security of multi-user cache-aided wiretap broadcast channels (BCs) against an external eavesdropper under two secrecy constraints: individual secrecy constraint and joint secrecy constraint. The former requires the messages to be individually secured from the intruder; whereas the latter requires the messages to be jointly secured. For both constraints, we derive lower and upper bounds on the secure capacity-memory tradeoff of the K-user ($K \geq 2$) wiretap erasure broadcast channel where K_w receivers are weak and have cache memories of equal size, and K_s receivers are strong and have no cache. The lower bounds exhibit that cache memories provide larger gains under a secrecy constraint than without such a constraint. Moreover, we propose different joint cache-channel coding schemes achieving these lower bounds. These schemes simultaneously exploit the cache contents and the channel statistics and leverage on storing secret keys, independent of cached data, in the weak receivers' caches under joint secret constraint only. For comparison, we compute lower bounds achieved following the best separate cache-channel coding scheme for the two-user scenario and prove that the joint design yields significant gains over the separation-based design. To justify our choice of cache distribution, we also compute lower and upper bounds on the secure capacity-memory tradeoff for the two-user scenario with equal cache distribution for both receivers and show that for a large set of parameters the capacity-memory tradeoff is larger when only the weaker receiver has cache memory than when this cache memory is split equally among both receivers. For the joint secrecy setup, we also compare with the case of two-sided asymmetric cache assignment using joint cache-channel coding schemes for the two-user scenarios. This cache assignment is beneficial only when the erasure probabilities of both receivers are close. The gain comes from allocating to strong receivers a small cache memory allowing them at least to store secret keys.

Contents

List of Figures

List of Tables

List of Abbreviations

AD	Ajtai-Dwork
BC	Broadcast Channel
BKZ	Block Korkine-Zolotarev
CVP	Closest Vector Problem
DMS	Discrete Memoryless Channel
GGH	Goldreich-Goldwasser-Halevi
GLD	Generalized Low Density
HNF	Hermite Normal Form
LB	Lower bound
LDLC	Low Density Lattice Codes
LDPC	Low-Density Parity-Check
LLL	Lenstra Lenstra Lovàsz
LWE	Learning With Errors
MDPC	Moderate-Density Parity-Check
MIMO	Multiple-Input Multiple-Output
OD	Orthogonality Defect
OSI	Open Systems Interconnection
pdf	probability distribution function
QC-LDPC	Quasi-Cyclic Low-Density Parity-Check
QC-MDPC	Quasi-Cyclic Moderate-Density Parity-Check
SVP	Shortest Vector Problem
UB	Upper bound

Résumé détaillé de la thèse

De nos jours, la place que l'Internet occupe dans notre vie s'accroit de manière significative. Par ailleurs, l'Internet est utilisé pour envoyer et stocker des données privées telles que les informations personnelles, les dossiers médicaux, etc. Cependant, en raison sa nature ouverte et globalisée, l'Internet est exposé à toutes sortes d'attaques qui deviennent actuellement de plus en plus sophistiquées. Pour cette raison, le renforcement de la sécurité des systèmes de communications devient une nécessité. Plus précisément, c'est la confidentialité des données transmises qu'il est nécessaire de garantir. En outre, le développement des ordinateurs quantiques et les nouvelles attaques qui en découleront nécessitent des techniques de sécurité post-quantiques.

Ce besoin de sécurité a poussé les chercheurs à considérer les couches du système de communication jusqu'alors non sécurisées comme la couche physique. De plus, la sécurité couche physique ne s'appuyant pas sur la résolution de problèmes mathématiques complexes mais sur l'exploitation des caractéristiques du canal de transmission, le gain de sécurité offert n'est pas corrélé à celui offert par les mécanismes classiques de sécurité.

Un autre aspect à prendre en considération est la non homogénéité de l'utilisation de l'Internet au cours de la journée. Récemment, les techniques de caching ont été proposées pour équilibrer le débit Internet transmis au cours de le journée. Les techniques de caching offrant le meilleur débit sont appliquées au niveau de la couche physique mais ne considèrent aucune problématique de sécurité.

Cette thèse explore deux techniques complémentaires permettant d'assurer la confidentialité des données transmises sur des liens sans-fils. Dans la première partie de ce travail, nous nous concentrons sur la cryptographie à clé publique basée sur les réseaux de points, qui est l'une des techniques les plus prometteuses de cryptographie post-quantique. Dans la seconde partie de ce travail, nous étudions la sécurité au niveau de la couche physique des systèmes de communication dans lesquels les utilisateurs ont accès à des mémoires de caches.

Dans ce travail, le chapitre 1 est dédié à l'exploration de l'état de l'art relatif aux deux aspects de sécurité étudiés. Le chapitre 2 présente le premier résultat de cette thèse: un nouveau système GGH qui utilise les réseaux de points "generalized low density" GLD. Nous commençons par présenter le premier système GGH et ses améliorations. Ensuite, nous présentons notre nouveau système et nous étudions sa sécurité et sa complexité en comparant avec les systèmes existants.

La seconde partie du travail, relative au caching, est présentée dans les chapitre 3 et 4, qui considèrent respectivement la sécurité individuelle et la sécurité jointe. Nous commençons par considérer le cas de deux utilisateurs. Nous dérivons des bornes supérieures et inférieures du compromis sécurisé capacité-mémoire en considérant différentes distributions de cache. Nous proposons des schémas de codage permettant d'atteindre nos bornes inférieures. Nous comparons les différents bornes obtenues pour mettre en évidences les meilleures distributions de cache et schémas de codage. Finalement, nous étendons nos résultats pour le cas de K utilisateurs.

Chapitre 1: Sécurité des systèmes de communication

La protection des données contre l'espionnage peut être appliquée sur différentes couches du système OSI. Les approches de sécurités appliquées sur les couches deux à sept se basent sur des primitives cryptographiques, tel que AES. Ces schémas se basent sur des problèmes mathématiques complexes ou bien des programmes de clés complexes. Donc, leur sécurité résulte du fait que l'espion a des capacités de calcul limitées.

D'autre part, la sécurité de la première couche, la couche physique, n'utilise pas les primitives cryptographique. Contrairement aux techniques cryptographique qui sont inconscient de la nature physique du canal de transmission, la sécurité couche physique exploite les caractéristiques de ce canal, tel que le bruit et les interférences, pour protéger la confidentialité des données transmises. Cette sécurité n'impose aucune contrainte sur la capacité de calcul des espions. Cette indépendance entre la sécurité au niveau de la couche physique et la cryptographie peut idéalement être exploitée pour créer un système de sécurité plus renforcé.

Dans cette thèse, nous nous intéressons à la cryptographie et à la sécurité couche physique dans les systèmes de transmission sans fil. Du point de vue cryptographie, nous nous concentrons sur les techniques basées sur les réseaux de points et nous étudions un schéma composé d'un émetteur, un récepteur et un espion. Du point de vue sécurité couche physique, nous considérons le schéma de caching avec un émetteur, plusieurs récepteurs et un espions et où les récepteurs ont accès à des mémoires caches. Nous présentons ici l'état de l'art lié au deux thématiques étudiées.

Cryptographie basée sur les réseaux de points

Les systèmes de cryptographie basés sur les réseaux de points sont des systèmes à clé publique basés sur des problèmes difficiles à résoudre dans ces réseaux. Ces cryptosystèmes ont attiré l'attention des chercheurs car ils sont supposés robustes contre les attaques d'ordinateurs quantiques. En 1996, Ajtai a proposé le premier cryptosystème de ce type, nommé Ajtai-Dwork (AD) [39]. Cependant, l'intérêt de ce système est principalement théorique et ne peut pas être utilisé en pratique.

Inspiré par AD, plusieurs systèmes ont été proposés comme Goldreich-Goldwasser-Halevi (GGH) [1] et NTRU [40]. GGH est une alternative plus pratique du système AD. La sécurité du GGH est basée sur le problème du *vecteur le plus proche (CVP)* dans les réseaux de points. D'autre part, NTRU est un cryptosystème à clé publique basé sur les anneaux. Il peut être vu comme un problème de CVP ou de SVP.

En 2005, Regev a introduit le problème de "learning with errors (LWE)" [41]. Il a proposé un système basé sur LWE qui est équivalent à un problème de décodage d'un code linéaire aléatoire. Plusieurs schémas de cryptages ont été conçus basés sur le problème LWE (eg. [42, 43]).

Parmi ces systèmes, GGH est le seul dans lequel on gère explicitement les réseaux de points. Mais malgré ces caractéristiques intéressantes et les efforts pour son amélioration [7, 3], l'inconvénient majeur du GGH reste la taille très grande de sa clé publique. Dans cette thèse, nous résolvons ce problème en remplaçant les réseaux de points aléatoires du GGH par des réseaux de points à faible densité, plus précisément les réseaux GLD.

Sécurité basé sur la théorie d'information

Wyner a introduit le concept de sécurité couche physique qui exploite l'aléa du bruit dans les canaux de transmission pour sécuriser [45]. Il a modélisé le canal wiretap avec un émetteur, un récepteur et un espion qui écoute la transmission. Il a introduit la notion de capacité de secret qui correspond au débit maximal de transmission garantissant un décodage fiable au récepteur légitime tout en empêchant l'espion d'avoir aucune information du message. Dans son schéma, le canal de l'espion doit être dégradé par rapport à celui du récepteur légitime afin d'avoir une capacité de secret positive. Wyner a proposé de coder les bits d'informations avec des bits aléatoires afin d'atteindre cette capacité. Après la découverte de Wyner, la détermination de cette capacité de secret a été intensivement traité pour différentes classes de canaux tel que canal à effacement, canal Gaussien et canal à diffusion [47].

Récemment, les problèmes de sécurité ont commencé a être exploré dans les scénarios de caching. Le concept du caching est de stocker des données dans des mémoires cache à

coté des utilisateurs préalablement à la communication dans le but d'augmenter le débit de transmission. En effet, le trafic dans les systèmes de communication varie en fonction du temps. On a des périodes dans la journée pendant lesquels le réseau est fortement utilisé, induisant des délais de transmissions, et d'autres périodes pendant lesquels il est à peine utilisé. L'idée est de profiter de ces dernières pour stocker les données dans les mémoires caches. Ensuite, quand les utilisateurs demandent des fichiers pendant les périodes de congestion, une partie de leurs fichiers est déjà présente dans leur mémoire et il suffit de leur transmettre la partie manquante. Donc, la communication est divisée en deux phases: la phase de caching et la phase de transmission.

Maddah-Ali et Niesen ont introduit le premier schéma de codage de cache qui a permis d'ajouter un gain de cache au delà du gain évident provenant de la présence locale d'une partie des fichiers [10]. Ensuite, Saeedi, Timo, et Wigger ont montré que ce gain peut être encore augmenté en appliquant un codage joint de cache et canal [97, 9]. En d'autres termes, dans ce dernier, le codeur et décodeur exploitent simultanément le contenu du cache et les statistiques du canal. Ils ont proposé un schéma de codage joint qu'ils ont nommé le codage piggyback.

L'aspect de sécurité de ces systèmes a été adressé ultérieurement et des codes de caching sécurisés ont été proposés en [89, 90, 91]. Dans ces schémas, la sécurité profite du fait que l'espion n'a pas accès aux mémoires caches dans lesquels des clés secrètes sont sauvegardées dans la phase de caching et utilisées dans la phase de transmission. Cependant, ces travaux ne considèrent pas le canal de transmission et se concentrent seulement sur le design de codeur/décodeurs de cache. Mais, cette approche a été montrée sous-optimale pour le scénario sans contrainte de sécurité [97, 9]. Pour cela, dans ce travail, nous explorons la sécurité des systèmes de caching au niveau de la couche physique.

Chapitre 2: Schéma de cryptographie basé sur les réseau de points GLD

Dans cette partie, nous analysons l'utilisation des réseaux de points à faible densité appelés *generalized low density (GLD)* dans le cryptosystème Goldreich-Goldwasser-Halevi (GGH). Nous nous sommes intéressés aux GLD grâce à leur faible complexité de génération du réseau et de décodage, ce qui constitue un facteur très important pour les systèmes de cryptographie, manquant dans les systèmes GGH existant. Notre but est de diminuer la complexité du système GGH pour qu'il devienne un candidat pour remplacer les systèmes de cryptage à clé publique traditionnels. Nous montrons que cette réduction en complexité ne réduit pas la sécurité du système.

Schéma GGH et ses améliorations

Dans le système GGH original [1], la clé privée R est une bonne base d'un réseau de point Λ de dimension n, défini par $R = \sqrt{n}I + Q$, où I est la matrice d'identité et Q est une matrice aléatoire dont les éléments sont choisis dans l'intervalle $\{-4, \ldots, 4\}$. La clé publique B est une mauvaise base du même réseau de point Λ. B est générée en appliquant des combinaisons linéaires sur les vecteurs de base de la clé privée R. Pour crypter un message $\mathbf{m} \in \mathbb{Z}^n$, $\mathbf{m}$ est multiplié par la clé publique B pour générer $\mathbf{x}$, qui est un point du réseau Λ. Ensuite, $\mathbf{x}$ est sécurisé en lui additionnant un vecteur de bruit $\mathbf{e}$ de dimension n qui est choisi aléatoirement dans l'ensemble $\{-\sigma, \sigma\}^n$. Le texte chiffré obtenu est

$$\mathbf{c} = \mathbf{m}B + \mathbf{e} = \mathbf{x} + \mathbf{e}. \tag{1}$$

Pour décrypter le texte chiffré et obtenir le message, il suffit de calculer

$$\mathbf{m} = \lfloor \mathbf{c}R^{-1} \rceil RB^{-1}. \tag{2}$$

Ce système peut être facilement cassé pour les petites dimensions à cause de la forme particulière du bruit utilisé [2]. Et pour les grandes dimensions, la taille de clé du GGH devient très grande et donc le système devient inutilisable en pratique.

Dans le but de surmonter les inconvénients du GGH original, plusieurs améliorations ont été proposées. Une amélioration intéressante a été proposée par Micciancio qui a suggéré de choisir le bruit $\mathbf{e}$ uniformément dans l'intervalle $[-\sigma, \sigma]$. Le choix de bruit uniforme a rendu ce système robuste contre l'attaque de Nguyen [2]. De plus, Micciancio a proposé de choisir pour la clé publique la base ayant la forme *Hermite normal form (HNF)*. En fait, on dit qu'une base M est en HNF si elle est triangulaire inférieure vérifiant les conditions suivantes : $m_{i,j} = 0 \quad \forall\ i < j$,, $m_{i,i} > 0 \quad \forall\ i$, et $0 \leq m_{i,j} < m_{j,j} \quad \forall\ i > j$. Il a prouvé que cette forme de matrice réduit la taille de la clé de $O(n^3 log_2(n))$ à $O(n^2 log_2(n))$. Cependant, son système a des désavantages. En effet, la procédure de génération de la clé publique est très lente et sa procédure de décodage est aussi lente et instable. Ce système, donc, n'est pas non plus utilisable en pratique.

Récemment, Hooshmad et Aref ont proposé de remplacer les réseaux de points aléatoires utilisés dans les cryptosystèmes GGH par des réseaux de points à faible densité, notamment les *low density lattice codes (LDLC)* [3]. Dans leur schéma, ils ont suivi la proposition de Micciancio et ont choisi de prendre la matrice génératrice du LDLC qui est en HNF comme clé publique. Par contre, ils ont choisi de générer leur vecteur de bruit suivant une distribution gaussienne de moyenne nulle et de variance σ^2, qui est choisi proche de la borne de Poltyrev [4]. L'utilisation des LDLC a permis de fixer le problème de décryptage puisqu'ils utilisent un décodage itératif qui offre des performances proches de l'optimal mais à faible complexité. Mais l'inconvénient de ce système est le fait que les LDLC sont des réseaux de points réels, ce qui augmente la complexité de génération de la matrice HNF.

Réseaux de points GLD

Les réseaux de points GLD sont des réseaux entiers définis dans $\mathbb{Z}^n$ et ils ont des matrices de parité de faible densité [5]. Ils sont décodés à l'aide du décodage itératif qui est de faible complexité permettant l'utilisation facile de ces réseaux à grande dimensions.

Un réseau de point GLD de dimension n est généré suivant la construction A à partir d'un code linéaire GLD $C_{\mathrm{GLD}}[n,k]$ de longueur n et dimension k, de la façon suivante:

$$\Lambda = C_{\mathrm{GLD}} + p\mathbb{Z}^n, \tag{3}$$

où p est un nombre premier.

Dans cette construction, on commence par considérer un code élémentaire linéaire $C_0[n_0,k_0]$ défini sur le corps fini $\mathbb{F}_p$ de matrice de parité H_{C_0}. On génère un deuxième code C_1 obtenue comme la somme directe de L copies de C_0, de matrice de parité

$$H_{C_1} = \begin{bmatrix} H_{C_0} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & H_{C_0} \end{bmatrix}. \tag{4}$$

Ensuite, on génère le code GLD $C_{\mathrm{GLD}}[n,k]$, défini par $C_{\mathrm{GLD}} = \bigcap_{j=1}^{J} \pi_j(C_1) = \bigcap_{j=1}^{J} \pi_j(C_0^{\oplus L})$, où $\pi_1 = \mathrm{id}$ et $\pi_2, \dots, \pi_J$ sont J permutations de $\{1,2,\dots,n\}$ telles que $\pi_j(x_1,x_2,\dots,x_n) = (x_{\pi_j(1)}, x_{\pi_j(2)}, \dots, x_{\pi_j(n)})$. On trouve la forme systématique $H_{C,\mathrm{syst}} = [I|-B^t]$ de la matrice de parité H_C du code GLD en utilisant l'élimination de Gauss. Ensuite, la matrice génératrice du code est donnée par $G_C = [B|I]$. Finalement, suivant la construction A (3), la matrice génératrice de Λ, qui est de forme HNF, est donnée par

$$G_\Lambda = \begin{bmatrix} pI & 0 \\ B & I \end{bmatrix}. \tag{5}$$

Schéma de cryptographie basé sur les GLD

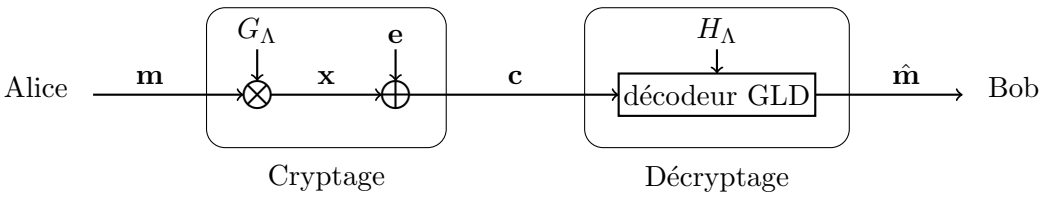

Figure 1: Cryptosystème GGH basé sur les GLD

Notre schéma de cryptographie basé sur les GLD est décrit dans Figure 1. Dans ce système, la clé privée est la matrice de parité H_C du code GLD et la clé publique est la matrice génératrice G_Λ du réseau de point. La dimension choisie du réseau est $n \approx 1000$. Le bruit $\mathbf{e}$ est choisi uniformément dans un intervalle $[-A, A]$ où A est la valeur maximale tolérée par le décodage itératif du réseau GLD. Cette valeur peut être définie théoriquement par $A < \sqrt{3p^{2(1-R)}/(2\pi \mathrm{e})}$. Nous avons aussi calculé expérimentalement la valeur de A pour différents codes élémentaires C_0. Quelques exemples sont montrés dans Tableau 1.

Table 1: Valeur de A pour une dimension $n \approx 1000$.

C_0	p	n	A
$C_0[3, 2]$	17	999	2
$C_0[3, 2]$	29	999	3
$C_0[8, 6]$	53	1000	2

Analyse de sécurité

Nous analysons la robustesse de notre système contre les attaques connues appliquées à ce genre de cryptosystèmes.

Attaque par force brute

Dans cette attaque, l'espion essaye de trouver la clé privée en essayant toutes les possibilités. Donc, la complexité de cette attaque dépend du nombre de clés possibles qui est dans notre cas

$$n! \times (J-1) \sim \sqrt{2\pi n}\Big(\frac{n}{e}\Big)^n, \tag{6}$$

où l'approximation suit la formule de Stirling. Puisque ce nombre est exponentiel en n, l'attaque n'est pas faisable pour une dimension $n = 1000$.

Attaque de décodage

Dans cette attaque l'espion essaye de décrypter le texte chiffré et de retrouver le texte clair, ce qui est équivalent à trouver le point du réseau le plus proche du texte chiffré. L'espion commence par réduire la base publique du réseau pour trouver une base plus orthogonale dans laquelle le décodage est plus efficace. Pour cela, il applique une des deux techniques de réduction de bases suivantes: Lenstra-Lenstra-Lovász (LLL) ou Block Korkine-Zolotarev (BKZ). Ensuite, il applique un des deux algorithmes de décodage de Babai [6] suivants: *round-off algorithm* ou *nearest plane algorithm.*

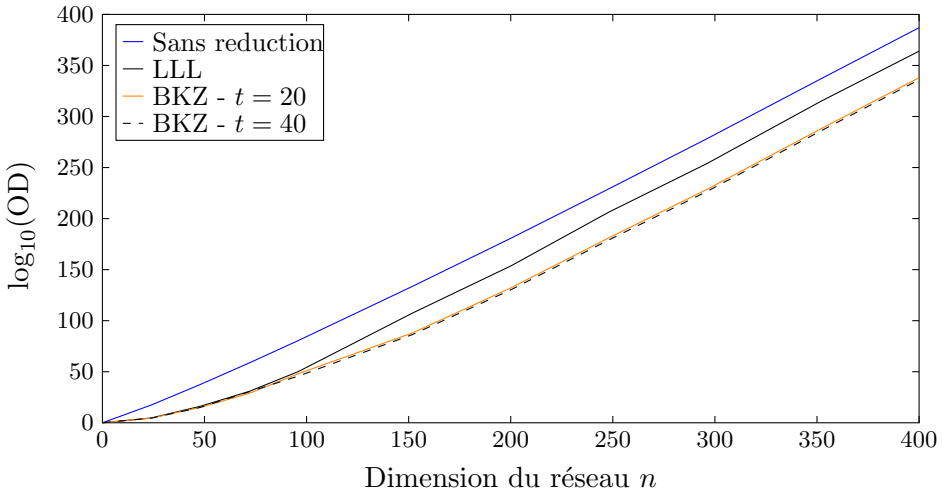

Figure 2: *Orthogonality defect* de la clé publique avec et sans reduction pour $C_0(8,6)_{17}$.

Le succès ou l'échec de ces attaques dépendent de deux facteurs: l'orthogonalité de la base du réseau et le vecteur de bruit. En fait, si la base utilisée pour le décodage n'est pas orthogonale et si le vecteur de bruit est suffisamment grand, le décodage ne peut qu'échouer. Pour mesurer l'orthogonalité de la base, nous utilisons l'*orthogonality defect* défini par [1]

$$\mathrm{OD}(G_\Lambda) = \frac{\prod_{i=1}^{n} \|g_i\|}{|\det(G_\Lambda)|}. \tag{7}$$

$\mathrm{OD}(G_\Lambda) = 1$ si la base est orthogonale. Plus $\mathrm{OD}(G_\Lambda)$ est grand, plus la base est loin de l'orthogonalité et plus le décodage est difficile, voire impossible. Nous avons conduit des simulations pour calculer $\mathrm{OD}(G_\Lambda)$ pour des réseaux GLD générés à partir du code élémentaire $C_0(8,6)_{17}$ avant et après l'application de méthodes de réduction. Les résultats sont montrés dans Figure 2 avec une échelle logarithmique en fonction de la dimension n. Nous remarquons que le $\mathrm{OD}(G_\Lambda)$ reste très grand, et donc la base reste mauvaise, même après réduction.

Le deuxième facteur qui peut faire échouer cette attaque est le vecteur de bruit. Nous avons aussi tourné des simulations d'attaques en appliquant les deux algorithmes de réduction et les deux algorithmes de décodage et en considérant les réseaux de points et les valeurs de bruit correspondant du Tableau 1. Dans chacun des cas, le décryptage ne donnait pas le bon résultat. Nous déduisons que le bruit considéré est suffisant pour sécuriser notre système contre les attaques de décodage.

Analyse de complexité

Nous examinons la taille de notre clé publique, la complexité de sa génération et la complexité de décryptage et nous nous comparons aux systèmes GGH précédents.

Taille de la clé publique

L'espace nécessaire pour stocker la clé publique du GLD est de $k \times (n-k) \times \log_2(p)$ bits. Tableau 2 présente la taille de clé publique pour plusieurs exemples de codes élémentaires. On voit que pour une dimension de réseau $n \approx 1000$, la taille de la clé est autour de 100 KBs. La taille de clé du GGH original en dimension $n = 400$ est de 2.3 MBs [7]. La taille de clé du système de Micciancio en dimension $n = 800$ est de 1 MB [8]. Donc, nous pouvons clairement conclure que notre système basé sur les réseaux de points réduit dix fois la taille de clé.

Table 2: Taille de la clé publique pour $n \approx 1000$.

C_0	p	n	Taille (KBs)
$C_0[3, 2]$	17	999	110.66
$C_0[8, 6]$	17	1000	124.74
$C_0[8, 6]$	53	1000	174.8
$C_0[16, 12]$	17	992	122.75

Génération de la clé

Dans notre schéma, la clé publique est en HNF mais elle est générée en utilisant la réduction de Gauss. Dans Micciancio et dans le système basé sur LDLC, la clé est générée en utilisant un des algorithmes de génération de HNF. Pour comparer la complexité de génération de clé de notre système avec les précédent, nous avons tourné des simulations sur un Intel i5 3320M (2.6 GHz). Nous avons trouvé que pour une dimension $n \approx 1000$, le génération de clé pour le système de Micciancio prend 5 heures, celle du GGH original prend 10 minutes, alors que pour le GLD, cela ne prend que quelques secondes.

Décryptage

Le décodage itératif des GLD offre des performances qui s'approchent du décodage ML tout en ayant une faible complexité. Les algorithmes de décodage utilisés dans les systèmes GGH précédents sont moins efficaces et plus compliqués. Par exemple, pour

une dimension $n = 1000$, le temps de décodage des GLD varie de quelques ms à quelques secondes (selon le code C_0 choisi), tandis que celui de Micciancio est autour de 2 minutes.

Chapitre 3: Sécurité individuelle pour le caching

Dans ce chapitre, nous étudions la sécurité des systèmes de communications où les récepteurs ont accès à des mémoires caches en imposant la contrainte de sécurité individuelle.

Définition du problème

Nous considérons un canal wiretap de diffusion (BC) avec un émetteur, K récepteurs et un espion. Il est modélisé par un canal BC à effacement par blocs sans mémoire, ayant pour alphabet d'entrée $\mathcal{X} := \{0,1\}^F$, où F est un entier positif. L'alphabet de sortie $\mathcal{Y} := \mathcal{X} \cup \Delta$ est le même pour tous les utilisateurs et l'espion. Δ indique l'effacement d'un bloc à la réception.

Les K récepteurs sont divisés en deux groupes. Le premier groupe $\mathcal{K}_w := \{1, \ldots, K_w\}$ est formé par K_w récepteurs faibles qui ont de mauvais canaux. Le deuxième groupe $\mathcal{K}_s := \{K_w + 1, \ldots, K\}$ est formé par $K_s = K - K_w$ récepteurs forts qui ont de bons canaux. Nous supposons que les trois probabilités d'effacement δ_w, δ_s et δ_z aux récepteurs faibles, récepteurs forts et l'espion, respectivement, vérifient $0 \leq \delta_s \leq \delta_w \leq \delta_z \leq 1$.

Nous supposons aussi que chaque récepteur faible a accès à une mémoire de cache de taille $n\mathcal{M}$ bits, tandis que les récepteurs forts n'ont pas de cache. L'émetteur a accès à une librarie de $D > K$ messages indépendants $W_1, \ldots, W_D$ de débit $R_s \geq 0$ chacun. Soit $\mathcal{D} := \{1, \ldots, D\}$. Pour $d \in \mathcal{D}$, chaque message W_d est uniformément distribué sur l'ensemble $\left\{1, \ldots, \lfloor 2^{nR_s} \rfloor\right\}$, où n est la longueur du block transmis.

Chaque récepteur $k \in \mathcal{K} := \{1, \ldots, K\}$ demande un seul message W_{d_k} de la librairie. Nous notons par $d_k \in \mathcal{D}$ la demande d'un récepteur et par $\mathbf{d} := \{d_1, \ldots, d_K\} \in \mathcal{D}^K$ le vecteur de demande de tous les récepteurs. La communication est effectuée en deux phases: la phase de caching dans laquelle des fragments de messages sont stockés dans les mémoires caches et la phase de transmission dans laquelle les messages demandés sont transmis au récepteurs.

Durant la phase de caching, le vecteur de demande $\mathbf{d}$ n'est pas encore connu. Donc, le contenu du cache V_i pour chaque récepteur faible $i \in \mathcal{K}_w$ est une fonction de toute la librairie $V_i := g_i(W_1, \ldots, W_D)$, pour une certaine fonction de caching $g_i : \left\{1, \ldots, \lfloor 2^{nR_s} \rfloor\right\}^D \to \mathcal{V}$ et un alphabet de cache $\mathcal{V} := \left\{1, \ldots, \lfloor 2^{n\mathcal{M}} \rfloor\right\}$.

Avant la phase de transmission, $\mathbf{d}$ est communiqué à l'émetteur et aux récepteurs légitimes. Selon $\mathbf{d}$, l'émetteur transmet $X^n := f_{\mathbf{d}}(W_1, \ldots, W_D)$, pour une certaine fonction $f_{\mathbf{d}} : \left\{1, \ldots, \lfloor 2^{nR_s} \rfloor\right\}^D \to \mathcal{X}^n$.

Chaque récepteur faible $i \in \mathcal{K}_w$ utilise son vecteur reçu Y_i^n et le contenu de son cache V_i pour décoder $\hat{W}_i := \varphi_{i,\mathbf{d}}(Y_i^n, V_i)$, pour une certaine fonction $\varphi_{i,\mathbf{d}} : \mathcal{Y}^n \times \mathcal{V} \to \left\{1, \ldots, \lfloor 2^{nR_s} \rfloor\right\}$. Chaque récepteur fort $j \in \mathcal{K}_s$ utilise seulement son vecteur reçu Y_j^n pour décoder $\hat{W}_j := \varphi_{j,\mathbf{d}}(Y_j^n)$, pour une certaine fonction $\varphi_{j,\mathbf{d}} : \mathcal{Y}^n \to \left\{1, \ldots, \lfloor 2^{nR_s} \rfloor\right\}$.

Une paire débit-mémoire $(R_s, \mathcal{M})$ est atteinte en sécurité, en vérifiant la *contrainte de sécurité individuelle*, si pour chaque $\epsilon > 0$ et longueur de bloc n suffisamment large, il existe des fonctions de caching, codage, et décodage telles que

$$\mathrm{P}_e^{\mathrm{Worst}} := \max_{\mathbf{d} \in \mathcal{D}^K} \mathrm{P}\Big[\bigcup_{k=1}^{K} \left\{\hat{W}_k \neq W_{d_k}\right\}\Big] \leq \epsilon, \qquad \text{et} \qquad \frac{1}{n} I(W_{d_k}; Z^n) < \epsilon, \ \forall k \in \mathcal{K}. \quad (8)$$

Pour chaque taille de mémoire cache $\mathcal{M}$, nous définissons le *compromis sécurisé entre capacité et mémoire* $C_s(\mathcal{M})$ comme la borne supérieure de tous les débits R_s tel que la paire $(R_s, \mathcal{M})$ qui peuvent être atteints en sécurité:

$$C_s(\mathcal{M}) := \sup\left\{R_s : \ \ (R_s, \mathcal{M}) \text{ atteint en sécurité}\right\}. \quad (9)$$

Bornes supérieure et inférieure pour le scénario avec deux récepteurs et cache pour le récepteur faible

Nous considérons le scénario avec un utilisateur faible et un utilisateur fort où l'utilisateur faible seulement a accès à une mémoire cache de taille $\mathcal{M}$.

La borne supérieure du compromis sécurisé capacité-mémoire $C_s(\mathcal{M})$ pour ce scénario est donnée par:

$$C_s(\mathcal{M}) \leq (\delta_z - \delta_1)F + \mathcal{M}, \quad (10a)$$

$$C_s(\mathcal{M}) \leq (\delta_z - \delta_2)F, \quad (10b)$$

$$C_s(\mathcal{M}) \leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F + \frac{\mathcal{M}}{D}. \quad (10c)$$

La borne inférieure du compromis sécurisé capacité-mémoire $C_s(\mathcal{M})$ est formée par les débits R_s vérifiant:

$$R_s \leq (\delta_z - \delta_2)F, \quad (11a)$$

$$R_s \leq \frac{(1-\delta_2)(\delta_z-\delta_2)}{1+\delta_z-2\delta_2}F + \frac{1-\delta_2}{1+\delta_z-2\delta_2}\frac{\mathcal{M}}{D}, \quad (11b)$$

$$R_s \leq \frac{(1-\delta_1)(\delta_z-\delta_2)}{1-\delta_1+\delta_z-\delta_2}F + \frac{\mathcal{M}}{D}, \tag{11c}$$

$$R_s \leq \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F + \frac{D(\delta_z-\delta_2)+(\delta_z-\delta_1)}{2\delta_z-\delta_1-\delta_2}\frac{\mathcal{M}}{D}, \tag{11d}$$

$$R_s \leq \frac{\delta_z-\delta_2}{2}F + \frac{D}{2}\frac{\mathcal{M}}{D}, \tag{11e}$$

$$R_s \leq \frac{D}{D+1}(\delta_z-\delta_2)F + \frac{D}{D+1}\frac{\mathcal{M}}{D}. \tag{11f}$$

Schéma de codage qui atteint cette borne inférieure

Cette borne inférieure est atteinte par le schéma de codage joint combinant le codage de cache et le codage canal, décrit ci-dessous:

Préparations: Pour chaque $d \in \mathcal{D}$, on divise le message W_d en deux sous-messages, tels que $W_d = \left[W_d^{(0)}, W_d^{(1)}\right]$, de débits $R^{(0)}$ et $R^{(1)}$, respectivement. Le débit total de W_d est $R_s = R^{(0)} + R^{(1)}$. Si $R^{(0)} > (D-2)R^{(1)}$, on divise $W_d^{(0)}$ en deux sous-messages, tels que $W_d^{(0)} = \left[W_d^{(0,1)}, W_d^{(0,2)}\right]$, de débits $(D-2)R^{(1)}$ et $R^{(0)}-(D-2)R^{(1)}$, respectivement. Sinon, $W_d^{(0,1)} = W_d^{(0)}$ est de débit $R^{(0)}$ et $W_d^{(0,2)}$ est de débit nul.

Génération du codebook piggyback: On génère codebook $\mathcal{C}_1$ ayant $\Gamma_1 := \lfloor 2^{nR^{(0)}} \rfloor \cdot \lfloor 2^{nR^{(1)}} \rfloor \cdot \lfloor 2^{nR'} \rfloor$ mots de code de longueur αn. Il est généré en choisissant chaque élément de chaque mot de code aléatoirement et indépendamment suivant une distribution Bernoulli-1/2. $\mathcal{C}_1$ est partitionné en $\lfloor 2^{nR^{(0)}} \rfloor \cdot \lfloor 2^{nR^{(1)}} \rfloor$ blocs, contenant chacun $\lfloor 2^{nR'} \rfloor$ mots de codes. Ces blocs sont arrangés en une matrice ayant $\lfloor 2^{nR^{(0)}} \rfloor$ lignes et $\lfloor 2^{nR^{(1)}} \rfloor$ colonnes, comme on voit dans Figure 3 où chaque carré représente un bloc. Le bloc dans la ligne $\tilde{w}_1$ et la colonne $\tilde{w}_2$ est noté par $\mathcal{C}_1(\tilde{w}_1, \tilde{w}_2)$.

Phase de caching: On sauvegarde $W_1^{(1)}, \ldots, W_D^{(1)}$ dans la mémoire cache du récepteur 1. Cela est faisable si: $R^{(1)} \leq \frac{\mathcal{M}}{D}$.

Phase de transmission: La transmission est divisée en deux périodes de longueurs αn et $(1-\alpha)n$, pour $\alpha \in [0,1]$.

Durant la première période, l'émetteur envoie le message $W_{d_1}^{(0)}$ au récepteur 1 et le message $W_{d_2}^{(1)}$ au récepteur 2. Il choisit aléatoirement un ensemble d'indices ι, tels que $\left\{j_1, j_2, \ldots, j_\iota\right\} \in \left(\mathcal{D} \setminus \{d_1, d_2\}\right)$, où $\iota = \max\left\{1, \lceil R^{(0)}/R^{(1)} \rceil\right\}$ et génère

$$W_{\text{XOR}} := W_{d_1}^{(0,1)} \oplus \left[W_{j_1}^{(1)}, W_{j_2}^{(1)}, \ldots, W_{j_\iota}^{(1)}\right]. \tag{12}$$

Ensuite, il choisit aléatoirement un indice J_1 de $\left[1 : \lfloor 2^{nR'} \rfloor\right]$ et envoie le mot de code

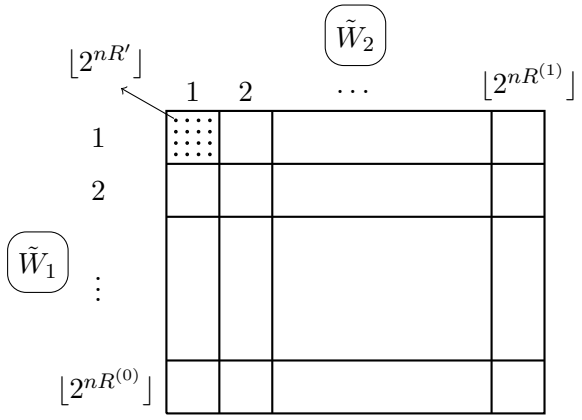

Figure 3: *Codebook piggyback sécurisé* $\mathcal{C}_1$.

numéro J_1 du codebook $\mathcal{C}_1\left(\tilde{W}_1, \tilde{W}_2 = W_{d_2}^{(1)}\right)$, où $\tilde{W}_1 := \left[W_{\text{XOR}}, W_{d_1}^{(0,2)}\right]$.

Durant la deuxième période, l'émetteur envoie le message $W_{d_2}^{(0)}$ au récepteur 2 en utilisant un code wiretap.

Décodage au récepteur 1*:* Récepteur 1 récupère le message $W_{d_1}^{(1)}$ de sa mémoire cache et décode seulement le message $W_{d_1}^{(0)}$ en se basant sur son vecteur reçu dans la première phase $y_1^{\alpha n}$ et sa mémoire cache V_1. Connaissant $W_{d_1}^{(1)}$, il réduit sa recherche à la colonne correspondante et donc, aux mots de code dans $\mathcal{C}_1(\tilde{W}_2 = W_{d_2}^{(1)}) \in \mathcal{C}_1$.

Décodage au récepteur 2*:* Récepteur 2 n'a pas de mémoire cache et doit donc décoder les deux phases de transmissions en se basant seulement sur son vecteur reçu y_2^n et en considérant tous les mots de codes des codebooks.

Analyse: Les deux récepteurs décodent leurs messages avec une probabilité d'erreur négligeable si

$$R^{(0)} + R' \leq \alpha(1-\delta_1)F, \tag{13a}$$
$$R_s + R' \leq \alpha(1-\delta_2)F, \tag{13b}$$
$$R^{(0)} + R'' \leq (1-\alpha)(1-\delta_2)F. \tag{13c}$$

De plus, la contrainte de sécurité individuelle est respectée si

$$(D-1)R^{(1)} + R' \geq \alpha(1-\delta_z)F, \tag{14a}$$
$$R^{(0)} + R' \geq \alpha(1-\delta_z)F, \tag{14b}$$
$$R'' \geq (1-\alpha)(1-\delta_z)F. \tag{14c}$$

En combinant ces contraintes et en éliminant R', R'' et α, on obtient la borne inférieure dans 11.

Borne inférieure obtenue par schéma de codage cache-canal séparé

Pour montrer l'importance d'un schéma de codage cache-canal joint, nous calculons une borne inférieure du compromis capacité-mémoire obtenue par le meilleur schéma de codage cache-canal séparé connu, qu'on note $C_{s,\mathrm{Sep}}(\mathcal{M})$. La borne inférieure de $C_{s,\mathrm{Sep}}(\mathcal{M})$ est formée par les débits $R_{s,\mathrm{Sep}}$ vérifiant:

$$
\begin{aligned}
R_{s,\mathrm{Sep}} &\leq (\delta_z-\delta_2)F, && \text{(15a)}\\
R_{s,\mathrm{Sep}} &\leq \frac{(1-\delta_1)(\delta_z-\delta_2)}{1+\delta_z-\delta_1-\delta_2}F+\frac{\delta_z-\delta_2}{1+\delta_z-\delta_1-\delta_2}\frac{\mathcal{M}}{D}, && \text{(15b)}\\
R_{s,\mathrm{Sep}} &\leq \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F+\frac{(\delta_z-\delta_2)\big[(D-1)(1-\delta_z)+(\delta_z-\delta_1)\big]}{(1-\delta_1)(2\delta_z-\delta_1-\delta_2)}\frac{\mathcal{M}}{D}. && \text{(15c)}
\end{aligned}
$$

Bornes supérieure et inférieure pour le scénario avec deux récepteurs ayant des mémoires de cache égales

Pour montrer l'intérêt de notre choix d'affectation de cache pour le récepteur faible, nous étudions aussi le cas où la mémoire de cache disponible est divisée également entre les deux utilisateurs. Dans ce cas, chaque récepteur a donc accès à une mémoire de taille $\mathcal{M}/2$. On note par $C_{s,\mathrm{Sym}}(\mathcal{M})$ le compromis sécurisé capacité-mémoire dans ce cas.

La borne supérieure du compromis sécurisé capacité-mémoire $C_{s,\mathrm{Sym}}(\mathcal{M})$ est donnée par:

$$
\begin{aligned}
C_{s,\mathrm{Sym}}(\mathcal{M}) &\leq (\delta_z-\delta_1)F+\frac{\mathcal{M}}{2}, && \text{(16a)}\\
C_{s,\mathrm{Sym}}(\mathcal{M}) &\leq (1-\delta_1)F+\frac{\mathcal{M}}{2D}, && \text{(16b)}\\
C_{s,\mathrm{Sym}}(\mathcal{M}) &\leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F+\frac{\mathcal{M}}{D}. && \text{(16c)}
\end{aligned}
$$

La borne inférieure du compromis sécurisé capacité-mémoire $C_{s,\mathrm{Sym}}(\mathcal{M})$ est formée par les $R_{s,\mathrm{Sym}}$ vérifiant:

$$
\begin{aligned}
R_{s,\mathrm{Sym}} &\leq 2(1-\delta_1)F, && \text{(17a)}\\
R_{s,\mathrm{Sym}} &\leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F+\frac{3-2\delta_1-\delta_2}{2(2-\delta_1-\delta_2)}\frac{\mathcal{M}}{D}, && \text{(17b)}\\
R_{s,\mathrm{Sym}} &\leq \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F+\Bigg[\frac{(\delta_z-\delta_1)(\delta_z-\delta_2)(3-2\delta_1-\delta_2)}{2(1-\delta_1)(1-\delta_2)(2\delta_z-\delta_1-\delta_2)}
\end{aligned}
$$

$$+\frac{D(1-\delta_z)\big[(1-\delta_1)(\delta_z-\delta_1)+(1-\delta_2)(\delta_z-\delta_2)\big]}{2(1-\delta_1)(1-\delta_2)(2\delta_z-\delta_1-\delta_2)}\Bigg]\frac{\mathcal{M}}{D}. \tag{17c}$$

Discussion et résultats numériques

Figure 4 montre nos bornes supérieure et inférieure du compromis sécurisé capacité-mémoire $C_s(\mathcal{M})$, celle du cas non sécurisé $C(\mathcal{M})$ [9] et la borne inférieure obtenu avec le schéma de codage cache-canal séparé. Nous observons les points suivants:

- Pour une mémoire cache $\mathcal{M} \leq \mathcal{M}_1 = \frac{D(1-\delta_z)(\delta_z-\delta_2)}{(D-1)(1+\delta_z-\delta_1-\delta_2)}$, notre codage cache-canal joint atteint le débit suivant:

$$R_s = R_0 + \left[\frac{\delta_z-\delta_2}{2\delta_z-\delta_1-\delta_2}+\frac{\delta_z-\delta_1}{D(2\delta_z-\delta_1-\delta_2)}\right]\mathcal{M}. \tag{18}$$

  Nous remarquons que pour des mémoires de petites tailles, la pente de $C_s(\mathcal{M})$ ne diminue pas rapidement avec D, contrairement au cas non sécurisé où la pente est fonction de $\frac{1}{D}$ [9, Corollary 7.1]. Nous pouvons déduire que, quand $\mathcal{M}$ est petit, le caching est plus avantageux dans le scénario avec contrainte de sécurité. Cela est dû au fait que dans ce cas, la mémoire est utilisée non seulement pour augmenter l'efficacité mais aussi pour sécuriser, ce qui augmente le gain du caching.

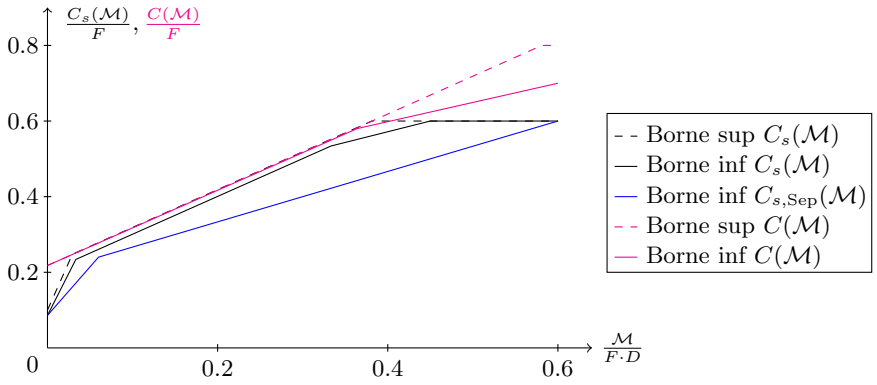

Figure 4: Bornes inférieures et supérieures de $C(\mathcal{M})/C_s(\mathcal{M})$ pour $K = 2$, $\delta_1 = 0.7$, $\delta_2 = 0.2$, $\delta_z = 0.8$, $F = 5$, et $D = 5$.

- Pour une mémoire

$$\mathcal{M} \geq \mathcal{M}_3 = F \cdot \max\left\{D\frac{(\delta_z-\delta_2)^2}{1-\delta_2}, (\delta_z-\delta_2)\right\}, \tag{19}$$

  le compromis sécurisé capacité-mémoire est $C_s(\mathcal{M}) = (\delta_z - \delta_2)F$. Cette capacité correspond à la capacité du récepteur 2 en l'absence du récepteur 1.

- Nos bornes sont très proches pour tous les paramètres et sont exactes pour des tailles de $\mathcal{M} \geq \mathcal{M}_3$, définie en (19).
- Notre schéma joint de codage cache-canal atteint une borne inférieure meilleure que celle atteinte par le schéma séparé. Cela est vrai pour tous les paramètres de canal et toutes les tailles de mémoires cache.

Figure 5 compare les bornes supérieures et inférieures pour le cas où l'utilisateur faible a toute la mémoire cache avec le cas où la mémoire est divisée également entre les deux utilisateurs. Nous constatons que:

- Quand $\mathcal{M}$ est petit, la borne inférieure de $C_s(\mathcal{M})$ est meilleure que celle pour le cas du cache symétrique pour les deux utilisateurs $C_{s,\text{Sym}}(\mathcal{M})$. Cela est vrai pour tous les paramètres de canal.
- Quand $\mathcal{M}$ est grand, si la différence entre les canaux du récepteur faible et fort est grande, il est plus avantageux que la mémoire cache soit pour l'utilisateur faible. Mais, si δ_1 et δ_2 ont des valeurs proches, $C_{s,\text{Sym}}(\mathcal{M})$ dépasse $C_s(\mathcal{M})$.

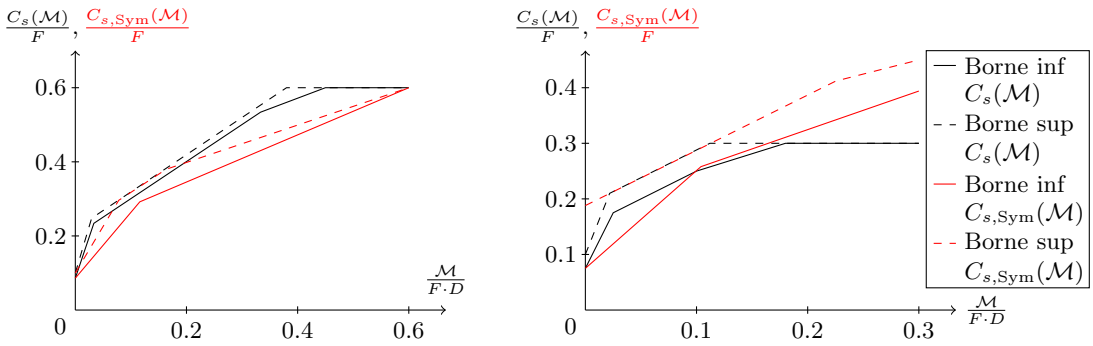

Figure 5: Bornes inférieures et supérieures de $C_s(\mathcal{M})/C_{s,\text{Sym}}(\mathcal{M})$ pour $K = 2$, $D = 5$, $\delta_1 = 0.7$, $\delta_z = 0.8$. Dans la figure à gauche $\delta_2 = 0.2$ et dans la figure à droite $\delta_2 = 0.5$.

Bornes supérieures et inférieures pour le scénario avec K récepteurs

Nous considérons le cas général avec K_w récepteurs faibles et K_s récepteurs forts. Les récepteurs faibles seulement ont accès à des mémoires caches de taille $\mathcal{M}$. La borne supérieure du compromis sécurisé capacité-mémoire $C_s^{(K)}(\mathcal{M})$ est donnée par:

$$C_s^{(K)}(\mathcal{M}) \leq F(\delta_z - \delta_w) + \mathcal{M}, \tag{20a}$$

$$C_s^{(K)}(\mathcal{M}) \leq F\frac{\delta_z - \delta_s}{K_s}, \tag{20b}$$

$$C_s^{(K)}(\mathcal{M}) \;\leq\; F\left(\frac{i}{1-\delta_w}+\frac{K_s}{1-\delta_s}\right)^{-1}+\frac{i\mathcal{M}}{D},\quad i\in\{1,\dots,K_w\}. \tag{20c}$$

Pour la borne inférieure, nous considérons les cinq paires débit-mémoire suivantes:

- $R_0^{(K)} := \left(\frac{K_w}{\delta_z-\delta_w}+\frac{K_s}{\delta_z-\delta_s}\right)^{-1}F, \qquad \mathcal{M}_0^{(K)} := 0;$ (21a)
- $R_3^{(K)} := \frac{(\delta_z-\delta_s)}{K_s}F, \qquad \mathcal{M}_3^{(K)} := \frac{DK_w(\delta_z-\delta_s)^2}{K_s\big[K_s(1-\delta_z)+K_w(\delta_z-\delta_s)\big]}F;$ (21b)
- $R_4^{(K)} := \frac{(\delta_z-\delta_s)}{K_s}F, \qquad \mathcal{M}_4^{(K)} := D\frac{(\delta_z-\delta_s)}{K_s}F;$ (21c)

Si $K_s(1-\delta_z)-(D-K_w)(\delta_w-\delta_s)\leq 0$,

- $R_1^{(K)} := \frac{2(1-\delta_w)(\delta_z-\delta_s)\big[(D-K_w)(1-\delta_w)+K_w(1-\delta_z)\big]F}{\beta_1},$ (21d)

  $\mathcal{M}_1^{(K)} := \frac{2D(1-\delta_w)(\delta_z-\delta_s)(1-\delta_z)F}{\beta_1};$ (21e)
- $R_2^{(K)} := \frac{2(1-\delta_w)(\delta_z-\delta_s)\big[K_s(1-\delta_w)+K_w(\delta_w-\delta_s)\big]F}{\beta_2},$ (21f)

  $\mathcal{M}_2^{(K)} := \frac{2D(1-\delta_w)(\delta_z-\delta_s)(\delta_w-\delta_s)F}{\beta_2};$ (21g)

Sinon, si $K_s(1-\delta_z)-(D-K_w)(\delta_w-\delta_s)>0$,

- $R_1^{(K)} := \frac{2(1-\delta_w)(\delta_z-\delta_s)\big[K_s(\delta_z-\delta_w)+D(\delta_w-\delta_s)\big]F}{\beta_3},$ (21h)

  $\mathcal{M}_1^{(K)} := \frac{2D(1-\delta_w)(\delta_z-\delta_s)(\delta_w-\delta_s)F}{\beta_3};$ (21i)
- $R_2^{(K)} := \frac{2(1-\delta_w)(\delta_z-\delta_s)\big[(D-K)(1-\delta_w)+K_w(1-\delta_z-\delta_w+\delta_s)\big]F}{\beta_4},$ (21j)

  $\mathcal{M}_2^{(K)} := \frac{2D(1-\delta_w)(\delta_z-\delta_s)(1-\delta_z-\delta_w+\delta_s)F}{\beta_4};$ (21k)

où

$$\begin{aligned}
\beta_1 &= 2K_w(D-K_w)(1-\delta_w)(\delta_z-\delta_s)+K_w(K_w-1)(1-\delta_z)(\delta_z-\delta_s)\\
&\quad +2K_s(D-K_w)(1-\delta_w)^2,\\
\beta_2 &= 2K_sK_w(1-\delta_w)(\delta_z-\delta_s)+K_w(K_w-1)(\delta_w-\delta_s)(\delta_z-\delta_s)+2K_s^2(1-\delta_w)^2,\\
\beta_3 &= 2K_wK_s(1-\delta_w)(\delta_z-\delta_s)+K_w(K_w-1)(\delta_w-\delta_s)(\delta_z-\delta_s)\\
&\quad +2K_s(1-\delta_w)\big[K_s(\delta_z-\delta_w)+(D-K_w)(\delta_w-\delta_s)\big],
\end{aligned}$$

$$\beta_4 = 2K_w(D-K)(1-\delta_w)(\delta_z-\delta_s) + K_w(K_w-1)(\delta_z-\delta_s)(1-\delta_z-\delta_w+\delta_s) + 2(1-\delta_w)\big[K_s(D-K)(1-\delta_w) + K_wK_s(1-\delta_z) - K_w(D-K_w)(\delta_w-\delta_s)\big].$$

La borne inférieure du compromis sécurisé capacité-mémoire $C_s^{(K)}(\mathcal{M})$ est donnée par:

$$C_s^{(K)}(\mathcal{M}) \geq \text{upper hull}\big\{(R_\ell^{(K)}, \mathcal{M}_\ell^{(K)}) : \ \ell \in \{0,\dots,4\}\big\}. \tag{22}$$

Figure 6 montre ces bornes pour un exemple avec $K = 20$ récepteurs.

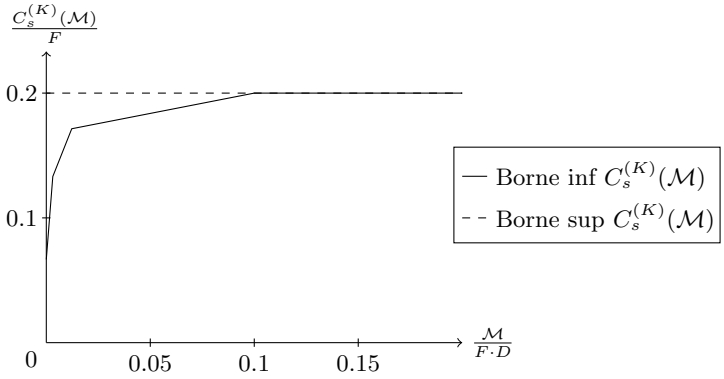

Figure 6: Bornes inférieure et supérieure de $C_s^{(K)}(\mathcal{M})$ pour $\delta_w = 0.7$, $\delta_s = 0.2$, $\delta_z = 0.8$, $F = 5$, $D = 30$, $K_w = 5$ et $K_s = 15$.

Chapitre 4: Sécurité jointe pour le caching

Dans ce chapitre, nous considérons le même modèle de canal que dans le chapitre précédent mais avec une contrainte de sécurité plus forte, qui est la contrainte de sécurité jointe. Cette contrainte impose que l'espion ne doit savoir aucune information à propos de tous les messages de la librairie conjointement à partir de son vecteur reçu Z^n:

$$\lim_{n\to\infty} \frac{1}{n} I(W_1,\dots,W_D; Z^n) < \epsilon. \tag{23}$$

Afin de satisfaire cette contrainte, l'émetteur a accès à un générateur d'aléa θ, défini sur un alphabet Θ. Cela induit un changement aux fonctions de caching et codage définies dans le chapitre précédent. Le contenu de la mémoire cache de chaque récepteur faible $i \in K_w$ devient $V_i := g_i(W_1,\dots,W_D,\theta)$, pour une certaine fonction de caching $g_i : \big\{1,\dots,\lfloor 2^{nR_s}\rfloor\big\}^D \times \Theta \to \mathcal{V}$. De plus, l'émetteur produit le vecteur à transmettre $X^n := f_{\mathbf{d}}(W_1,\dots,W_D,\theta)$, pour une certaine fonction de codage $f_{\mathbf{d}} : \big\{1,\dots,\lfloor 2^{nR_s}\rfloor\big\}^D \times \Theta \to \mathcal{X}^n$.

Bornes supérieure et inférieure pour le scénario avec deux récepteurs et cache pour le récepteur faible

Dans cette section, nous considérons le scénario avec un utilisateur faible et un utilisateur fort où l'utilisateur faible seulement a accès à une mémoire cache de taille $\mathcal{M}$.

La borne supérieure du compromis sécurisé capacité-mémoire $C_s(\mathcal{M})$ est donnée par:

$$C_s(\mathcal{M}) \leq \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F+\frac{\delta_z-\delta_2}{2\delta_z-\delta_1-\delta_2}\mathcal{M}, \tag{24a}$$

$$C_s(\mathcal{M}) \leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F+\frac{\mathcal{M}}{D}, \tag{24b}$$

$$C_s(\mathcal{M}) \leq (\delta_z-\delta_2)F. \tag{24c}$$

Pour la borne inférieure, nous considérons les six paires débit-mémoire suivantes:

- $R_0 := \dfrac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F, \qquad \mathcal{M}_0 := 0;$ (25a)

- $R_1 := \dfrac{(1-\delta_1)(\delta_z-\delta_2)}{1+\delta_z-\delta_1-\delta_2}F, \qquad \mathcal{M}_1 := \dfrac{(1-\delta_z)(\delta_z-\delta_2)}{1+\delta_z-\delta_1-\delta_2}F;$ (25b)

- $R_2 := (1-\delta_2)\min\left\{\dfrac{\delta_z-\delta_1}{1-\delta_1}, \dfrac{1-\delta_1}{2-\delta_1-\delta_2}\right\}F, \qquad \mathcal{M}_2 := (1-\delta_z)F;$ (25c)

- $R_3 := \dfrac{(1-\delta_2)(\delta_z-\delta_2)}{1+\delta_z-\delta_1-\delta_2}F, \qquad \mathcal{M}_3 := \dfrac{(\delta_z-\delta_2)\big[(\delta_1-\delta_2)D+(1-\delta_z)\big]}{1+\delta_z-\delta_1-\delta_2}F;$ (25d)

- $R_4 := (\delta_z-\delta_2)F, \qquad \mathcal{M}_4 := \dfrac{(\delta_z-\delta_2)\big[(\delta_z-\delta_2)D+(1-\delta_z)\big]}{1-\delta_2}F;$ (25e)

- $R_5 := (\delta_z-\delta_2)F, \qquad \mathcal{M}_5 := D(\delta_z-\delta_2)F.$ (25f)

La borne inférieure du compromis sécurisé capacité-mémoire $C_s(\mathcal{M})$ est donnée par:

$$C_s(\mathcal{M}) \geq \text{upper hull}\big\{(R_\ell,\mathcal{M}_\ell):\ \ell=0,\ldots,5\big\}. \tag{26}$$

Il suffit de prouver que les paires $(R_\ell,\mathcal{M}_\ell)$ sont atteignables et la preuve du *upper convex hull* suit par des arguments de partage de mémoire et temps comme dans [10].

Schéma de codage atteignant $(R_1, \mathcal{M}_1)$

On divise la phase de transmission en deux périodes de longueurs αn et $(1-\alpha)n$, pour $\alpha = \frac{\delta_z - \delta_s}{1+\delta_z-\delta_w-\delta_s}$. On génère aléatoirement une clé K_1 de débit $\alpha(1-\delta_z)F$ et on la sauvegarde dans la mémoire cache du récepteur 1. Dans la première période, l'émetteur envoie le message W_{d_1} au récepteur 1 en utilisant un code wiretap avec clé secrète K_1 [11, (22.7)]. Dans la deuxième période, l'émetteur envoie le message W_{d_2} au récepteur 2 en utilisant un code wiretap sans clé secrète.

Schéma de codage atteignant $(R_2, \mathcal{M}_2)$

On divise la phase de transmission en deux périodes de longueurs αn et $(1-\alpha)n$, pour $\alpha = \max\left\{0, \frac{(1-\delta_s)(\delta_z-\delta_w)-(1-\delta_z)(1-\delta_w)}{(\delta_z-\delta_w)(2-\delta_w-\delta_s)}\right\}$. On divise chaque message W_d, pour $d \in \mathcal{D}$, en deux sous-messages $W_d = [W_d^{(0)}, W_d^{\oplus}]$ de débits $\alpha(1-\delta_w)F$ et $(1-\alpha)(1-\delta_z)F$. On génère deux clés aléatoires K_1 et K_2 de débits $\alpha(1-\delta_z)F$ et $(1-\alpha)(1-\delta_z)F$ et on les sauvegarde dans la mémoire cache du récepteur 1. Dans la première période, l'émetteur envoie le message $W_{d_1}^{(0)}$ au récepteur 1 en utilisant un code wiretap avec clé secrète K_1.

Pour la communication dans la deuxième période, on génère un codebook de superposition avec un centre cloud ayant $2^{n(1-\alpha)(1-\delta_z)F}$ mots de code, et avec chaque codebook satellite contenant 2^{nR_2} mots de code. L'émetteur code $W_{d_1}^{\oplus} \oplus K_2$ dans le centre cloud et W_{d_2} dans le satellite. Récepteur 1 décode seulement le message $W_{d_1}^{\oplus} \oplus K_2$ tandis que récepteur 2 décode les deux messages.

Schéma de codage atteignant $(R_3, \mathcal{M}_3)$

Soit $\alpha = \frac{\delta_z - \delta_s}{1+\delta_z-\delta_w-\delta_s}$. On divise chaque message W_d, pour $d \in \mathcal{D}$, en trois sous-messages $W_d = [W_d^{(0)}, W_d^{(1)}, W_d^{\oplus}]$ de débits $\alpha(\delta_z - \delta_w)F$, $\alpha(\delta_w - \delta_s)F$ et $\alpha(1-\delta_z)F$. On génère une clé aléatoire K_1 de débit $\alpha(1-\delta_z)F$. On sauvegarde K_1 et $W_1^{(1)}, \ldots, W_D^{(1)}$ dans la mémoire cache du récepteur 1.

On génère un piggyback codebook $\mathcal{C}_1$ formé de $\lfloor 2^{n\alpha(\delta_z-\delta_1)F} \rfloor \cdot \lfloor 2^{n\alpha(\delta_1-\delta_2)F} \rfloor$ blocs ayant $\lfloor 2^{n\alpha(1-\delta_z)F} \rfloor$ mots de codes chacun. Ces blocs sont arrangés en une matrice de $\lfloor 2^{n\alpha(\delta_z-\delta_1)F} \rfloor$ lignes $\lfloor 2^{n\alpha(\delta_1-\delta_2)F} \rfloor$ colonnes. La phase de transmission est divisée en deux périodes de longueurs αn et $(1-\alpha)n$. Dans la première période, l'émetteur transmet les messages $W_{d_1}^{(0)}$ et $W_{d_1}^{\oplus}$ au récepteur 1 et $W_{d_2}^{(1)}$ au récepteur 2. Il envoie donc le mot de code correspondant à $W_{\text{XOR}} = W_{d_1}^{\oplus} \oplus K_1$ du bloc $\mathcal{C}_1(W_{d_1}^{(0)}, W_{d_2}^{(1)})$. Dans la deuxième période, il transmet le message $W_{d_2}^{(0),\oplus} = [W_{d_2}^{(0)}, W_{d_2}^{\oplus}]$ au récepteur 2 en utilisant un code wiretap.

Schéma de codage atteignant $(R_4, \mathcal{M}_4)$

On applique le même schéma de codage décrit pour $(R_3, \mathcal{M}_3)$ avec les changements suivant: on annule le débit de $W_d^{(0)}$, on change le débit de $W_d^{(1)}$ à $R^{(1)} = \alpha(\delta_z - \delta_2)F$ et on choisit $\alpha(1-\delta_z)F \leq (1-\alpha)(\delta_z - \delta_2)F$.

Bornes supérieure et inférieure pour le scénario avec deux récepteurs ayant des mémoires de cache égales

Comme dans le chapitre précédent, nous étudions le cas où la mémoire de cache disponible est divisée également entre les deux utilisateurs. La borne supérieure du compromis sécurisé capacité-mémoire $C_{s,\mathrm{Sym}}(\mathcal{M})$ est donnée par:

$$C_{s,\mathrm{Sym}}(\mathcal{M}) \leq \frac{(\delta_z - \delta_1)(\delta_z - \delta_2)}{2\delta_z - \delta_1 - \delta_2} F + \frac{\mathcal{M}}{2}, \tag{27a}$$

$$C_{s,\mathrm{Sym}}(\mathcal{M}) \leq (1-\delta_1)F + \frac{\mathcal{M}}{2D}, \tag{27b}$$

$$C_{s,\mathrm{Sym}}(\mathcal{M}) \leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2} F + \frac{\mathcal{M}}{D}. \tag{27c}$$

Pour la borne inférieure, nous considérons les trois paires débit-mémoire suivantes:

- $$R_{0,\mathrm{Sym}} := \frac{(\delta_z - \delta_1)(\delta_z - \delta_2)}{2\delta_z - \delta_1 - \delta_2} F, \qquad \mathcal{M}_{0,\mathrm{Sym}} := 0; \tag{28a}$$

- $$R_{1,\mathrm{Sym}} := \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2} F, \qquad \mathcal{M}_{1,\mathrm{Sym}} := \frac{2(1-\delta_z)(1-\delta_2)}{2-\delta_1-\delta_2} F; \tag{28b}$$

- $$R_{2,\mathrm{Sym}} := \min\big\{2(1-\delta_1)F, (1-\delta_2)F\big\}, \tag{28c}$$
  $$\mathcal{M}_{2,\mathrm{Sym}} := \min\Bigg\{2\big[(1-\delta_z) + D(1-\delta_1)\big]F, \frac{2(1-\delta_2)\big[2(1-\delta_z) + D(1-\delta_2)\big]}{2(1-\delta_1) + (1-\delta_2)} F\Bigg\}. \tag{28d}$$

La borne inférieure du compromis sécurisé capacité-mémoire $C_{s,\mathrm{Sym}}(\mathcal{M})$ est donnée par:

$$C_{s,\mathrm{Sym}}(\mathcal{M}) \geq \text{upper hull}\big\{(R_{\ell,\mathrm{Sym}}, \mathcal{M}_{\ell,\mathrm{Sym}}) \colon\ \ell = 0,1,2\big\}. \tag{29}$$

Bornes supérieure et inférieure pour le scénario avec deux récepteurs avec des mémoires caches non égales

Dans cette partie, nous étudions une distribution de cache pour les deux utilisateurs qui tient compte de leurs canaux. Récepteur 1 a accès à une mémoire de taille $\mathcal{M}_{R_1}$ et récepteur 2 a accès à une mémoire de taille $\mathcal{M}_{R_2}$, tel que $\mathcal{M}_{R_1} + \mathcal{M}_{R_2} = \mathcal{M}$.

La borne supérieure du compromis sécurisé capacité-mémoire $C_{s,\text{Asym}}(\mathcal{M})$ est donnée par:

$$C_{s,\text{Asym}}(\mathcal{M}) \leq \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F + \frac{\delta_z-\delta_2}{2\delta_z-\delta_1-\delta_2}\mathcal{M}, \tag{30a}$$

$$C_{s,\text{Asym}}(\mathcal{M}) \leq \frac{1}{2}(\delta_z-\delta_2)F + \frac{\mathcal{M}}{2}, \tag{30b}$$

$$C_{s,\text{Asym}}(\mathcal{M}) \leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F + \frac{\mathcal{M}}{D}, \tag{30c}$$

$$C_{s,\text{Asym}}(\mathcal{M}) \leq \frac{(2-\delta_1-\delta_2)}{2}F + \frac{\mathcal{M}}{2D}. \tag{30d}$$

Pour la borne inférieure, nous considérons les cinq paires débit-mémoire suivantes:

- $R_{0,\text{Asym}} := \dfrac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F, \qquad \mathcal{M}_{0,\text{Asym}} := 0;$ (31a)
- $R_{1,\text{Asym}} := \dfrac{(1-\delta_1)(\delta_z-\delta_2)}{1+\delta_z-\delta_1-\delta_2}F, \qquad \mathcal{M}_{1,\text{Asym}} := \dfrac{(1-\delta_z)(\delta_z-\delta_2)}{1+\delta_z-\delta_1-\delta_2}F;$ (31b)
- $R_{2,\text{Asym}} := \dfrac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F, \qquad \mathcal{M}_{2,\text{Asym}} := (1-\delta_z)F;$ (31c)
- $R_{3,\text{Asym}} := \dfrac{(1-\delta_2)^2}{2-\delta_1-\delta_2}F,$ (31d)

  $\mathcal{M}_{3,\text{Asym}} := \left[(1-\delta_z) + \dfrac{D(1-\delta_2)(\delta_1-\delta_2)}{2-\delta_1-\delta_2}\right]F;$ (31e)
- $R_{4,\text{Asym}} := \left[(2\delta_z-\delta_1-\delta_2) + \dfrac{1-\delta_z}{2}\right]F,$ (31f)

  $\mathcal{M}_{4,\text{Asym}} := \big[D(2\delta_z-\delta_1-\delta_2) + (1-\delta_z)\big]F;$ (31g)

La borne inférieure du compromis sécurisé capacité-mémoire $C_{s,\text{Asym}}(\mathcal{M})$ est donnée par:

$$C_{s,\text{Asym}}(\mathcal{M}) \geq \text{upper hull}\big\{(R_{\ell,\text{Asym}}, \mathcal{M}_{\ell,\text{Asym}}) \colon\ \ell = 0,\ldots,4\big\}. \tag{32}$$

Comme dans la preuve de (11), il suffit de prouver que les paires $(R_{\ell,\text{Asym}}, \mathcal{M}_{\ell,\text{Asym}})$ sont atteignables. $(R_{1,\text{Asym}}, \mathcal{M}_{1,\text{Asym}})$ est similaire à $(R_1, \mathcal{M}_1)$.

Schéma de codage atteignant $(R_{2,\mathbf{Asym}}, \mathcal{M}_{2,\mathbf{Asym}})$

On divise la phase de transmission en deux périodes de longueurs αn et $(1-\alpha)n$, pour $\alpha = \frac{\delta_z - \delta_s}{1+\delta_z-\delta_w-\delta_s}$. On génère deux clés aléatoires K_1 et K_2 de débits $\alpha(1-\delta_z)F$ et $(1-\alpha)(1-\delta_z)F$. Pour $i \in \{1,2\}$, on sauvegarde K_i dans la mémoire cache du récepteur i. Dans chaque période $i \in \{1,2\}$, l'émetteur envoie le message W_{d_i} au récepteur i en utilisant un code wiretap avec clé secrète K_i.

Schéma de codage atteignant $(R_{3,\mathbf{Asym}}, \mathcal{M}_{3,\mathbf{Asym}})$

Cette paire est obtenue suivant un schéma similaire à $(R_3, \mathcal{M}_3)$ sauf que dans ce cas, on génère une deuxième clé K_2 de débit $(1-\alpha)(1-\delta_z)F$ et on la sauvegarde dans la mémoire cache du récepteur 2. Elle est utilisée pour sécuriser les messages $W_{d_2}^{(0)}$ et $W_{d_2}^{\oplus}$ transmis au récepteur 2 pendant la deuxième période.

Schéma de codage atteignant $(R_{4,\mathbf{Asym}}, \mathcal{M}_{4,\mathbf{Asym}})$

On divise chaque message W_d, $d \in \mathcal{D}$, en trois sous-messages: $W_d = [W_d^{(0)}, W_d^{(1)}, W_d^{\oplus}]$, tel que $R^{(0)} = (\delta_z - \delta_1)F$, $R^{(1)} = (\delta_z - \delta_2)F$ et $R^{\oplus} = \frac{(1-\delta_z)F}{2}$. Ensuite, on génère deux clés aléatoires K_1 et K_2 de débit $\frac{(1-\delta_z)F}{2}$ chacun. On sauvegarde K_1 et $W_1^{(1)}, \ldots, W_D^{(1)}$ dans la mémoire du récepteur 1 et on sauvegarde K_2 et $W_1^{(0)}, \ldots, W_D^{(0)}$ dans la mémoire du récepteur 2.

On génère un piggyback codebook $\mathcal{C}_1$ formé de $\lfloor 2^{n(\delta_z-\delta_1)F} \rfloor \cdot \lfloor 2^{n(\delta_z-\delta_2)F} \rfloor$ blocs contenant chacun $\lfloor 2^{n(1-\delta_z)F} \rfloor$ mots de codes. Ces blocs sont arrangés en une matrice ayant $\lfloor 2^{n(\delta_z-\delta_1)F} \rfloor$ lignes et $\lfloor 2^{n(\delta_z-\delta_2)F} \rfloor$ colonnes. Dans la phase de transmission, l'émetteur envoie le mot de code représentant $W_{\text{XOR}} = [W_{d_1}^{\oplus} \oplus K_1, W_{d_2}^{\oplus} \oplus K_2]$ du bloc $\mathcal{C}_1(W_{d_1}^{(0)}, W_{d_2}^{(1)})$.

Discussion et résultats numériques

Figure 7 montre nos bornes supérieures et inférieures du compromis $C_s(\mathcal{M})$ pour la contrainte de sécurité jointe. Elle compare aussi avec les bornes du chapitre précédent. Nous pouvons voir que la contrainte jointe qui apporte une sécurité plus forte n'induit qu'une petite perte de débit.

Pour la sécurité jointe, nos bornes sont exactes pour les petites et grandes mémoires caches. Pour une grande mémoire cache $\mathcal{M} \geq \mathcal{M}_4$, où $\mathcal{M}_4$ est défini dans (25e), nous avons $C_s(\mathcal{M}) = (\delta_z - \delta_2)F$. Pour une petite mémoire cache $\mathcal{M} \leq \mathcal{M}_1$, où $\mathcal{M}_1$ est défini

dans (25b), nous avons:

$$C_s(\mathcal{M}) \quad = \quad \frac{(\delta_z - \delta_1)(\delta_z - \delta_2)}{2\delta_z - \delta_1 - \delta_2} F + \frac{\delta_z - \delta_2}{2\delta_z - \delta_1 - \delta_2} \mathcal{M}. \tag{33}$$

Quand la mémoire est petite, nous l'utilisons pour seulement sauvegarder des clés secrète. Cela induit une croissance rapide de $C_s(\mathcal{M})$ puisque les clés sont utiles quelles que soient les demandes des utilisateurs.

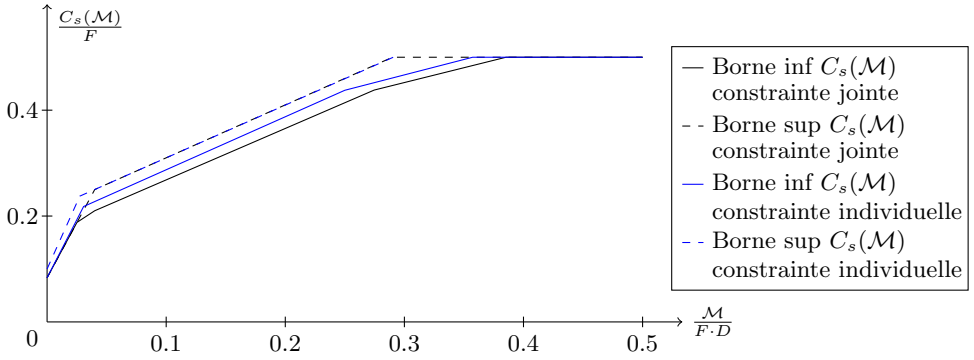

Figure 7: Bornes inférieures et supérieures de $C_s(\mathcal{M})$ pour les contraintes de sécurité jointes et individuelles pour $K = 2$, $\delta_1 = 0.7, \delta_2 = 0.3, \delta_z = 0.8, F = 5$ et $D = 5$.

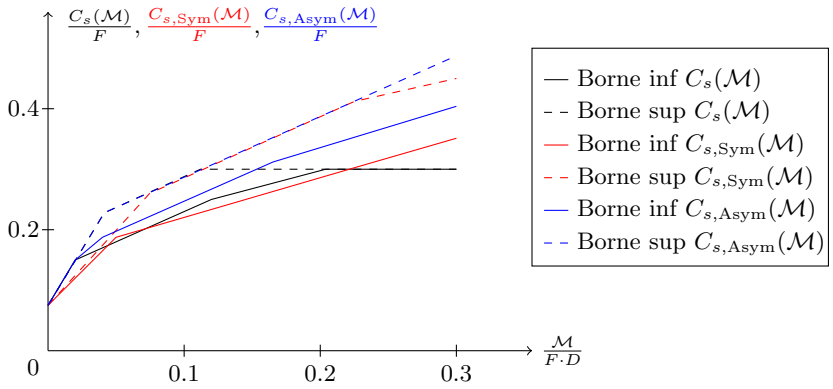

Figure 8: Bornes inférieures et supérieures de $C_s(\mathcal{M})/C_{s,\text{Sym}}(\mathcal{M})/C_{s,\text{Asym}}(\mathcal{M})$ pour $K = 2$, $\delta_1 = 0.7$, $\delta_2 = 0.5$, $\delta_z = 0.8$, $F = 5$ et $D = 5$.

Figure 8 compare les bornes obtenues pour les différentes distributions de cache considérées. Nous remarquons que pour une mémoire totale petite, il est toujours préférable d'allouer celle-ci à l'utilisateur faible. Pour des mémoires de tailles moyennes, nous voyons que dans la plupart des cas, allouer la mémoire à l'utilisateur faible est plus avantageux que de la répartir également entre les deux utilisateurs. Nous voyons aussi que

répartir la mémoire d'une façon asymétrique est le meilleur choix puisque, contrairement à l'allocation unilatérale, cela permet à l'utilisateur fort de sécuriser sa transmission.

Bornes supérieure et inférieure pour le scénario avec K récepteurs

Nous considérons le cas général avec K_w récepteurs faibles et K_s récepteurs forts. Les récepteurs faibles seulement ont accès à des mémoires caches de taille $\mathcal{M}$. La borne supérieure du compromis sécurisé capacité-mémoire $C_s^{(K)}(\mathcal{M})$ est donnée par:

$$C_s^{(K)}(\mathcal{M}) \leq \frac{\delta_z-\delta_s}{K_s}F, \tag{34a}$$

$$C_s^{(K)}(\mathcal{M}) \leq \left(\frac{j}{1-\delta_w}+\frac{K_s}{1-\delta_s}\right)^{-1}F+\frac{j\mathcal{M}}{D},\ \ j\in\{1,\dots,K_w\}, \tag{34b}$$

$$C_s^{(K)}(\mathcal{M}) \leq \max_{\alpha_i\in[0,1]}\min\Big\{\frac{\alpha_i(\delta_z-\delta_w)+(1-\alpha_i)(\delta_z-\delta_s)}{i+K_s}F+\frac{i}{i+K_s}\mathcal{M}, \alpha_i\frac{\delta_z-\delta_w}{i}F+\mathcal{M}\Big\},\quad i\in\{1,\dots,K_w\}. \tag{34c}$$

Pour la borne inférieure, nous considérons les six paires débit-mémoire suivantes:

- $R_0^{(K)} := \Big(\frac{K_w}{\delta_z-\delta_w}+\frac{K_s}{\delta_z-\delta_s}\Big)^{-1}F, \qquad \mathcal{M}_0^{(K)} := 0;$ (35a)
- $R_1^{(K)} := \frac{(1-\delta_w)(\delta_z-\delta_s)F}{K_w(\delta_z-\delta_s)+K_s(1-\delta_w)}, \quad \mathcal{M}_1^{(K)} := \frac{(1-\delta_z)(\delta_z-\delta_s)F}{K_w(\delta_z-\delta_s)+K_s(1-\delta_w)};$ (35b)
- $R_2^{(K)} := \min\Big\{\frac{(1-\delta_s)(\delta_z-\delta_w)}{K_s(1-\delta_w)}, \frac{(1-\delta_s)(1-\delta_w)}{K_w(1-\delta_s)+K_s(1-\delta_w)}\Big\}F,$ (35c)

  $\mathcal{M}_2^{(K)} := \frac{(1-\delta_z)}{K_w}F;$ (35d)
- $R_3^{(K)} := \frac{2(1-\delta_w)(\delta_z-\delta_s)\big[K_s(1-\delta_w)+K_w(\delta_w-\delta_s)\big]F}{K_w(\delta_z-\delta_s)\big[(K_w-1)(\delta_w-\delta_s)+2K_s(1-\delta_w)\big]+2K_s^2(1-\delta_w)^2},$ (35e)

  $\mathcal{M}_3^{(K)} := \frac{2D(\delta_z-\delta_s)(1-\delta_w)(\delta_w-\delta_s)F}{K_w(\delta_z-\delta_s)\big[(K_w-1)(\delta_w-\delta_s)+2K_s(1-\delta_w)\big]+2K_s^2(1-\delta_w)^2}$
  $+\frac{2(\delta_z-\delta_s)(1-\delta_z)\big[(K_w-1)(\delta_w-\delta_s)+K_s(1-\delta_w)\big]F}{K_w(\delta_z-\delta_s)\big[(K_w-1)(\delta_w-\delta_s)+2K_s(1-\delta_w)\big]+2K_s^2(1-\delta_w)^2}.$ (35f)
- $R_4^{(K)} := \frac{(\delta_z-\delta_s)}{K_s}F, \qquad \mathcal{M}_4^{(K)} := \frac{K_s(\delta_z-\delta_s)(1-\delta_z)+DK_w(\delta_z-\delta_s)^2}{K_s\big[K_s(1-\delta_z)+K_w(\delta_z-\delta_s)\big]}F;$ (35g)
- $R_5^{(K)} := \frac{(\delta_z-\delta_s)}{K_s}F, \qquad \mathcal{M}_5^{(K)} := D\frac{(\delta_z-\delta_s)}{K_s}F.$ (35h)

La borne inférieure du compromis sécurisé capacité-mémoire $C_s^{(K)}(\mathcal{M})$ est donnée par:

$$C_s^{(K)}(\mathcal{M}) \geq \text{upper hull}\big\{(R_\ell^{(K)}, \mathcal{M}_\ell^{(K)}) : \ \ell \in \{0, \ldots, 5\}\big\}. \tag{36}$$

Conclusion

Dans cette thèse, nous avons étudié deux méthodes de sécurité pour les transmissions sans fil. Dans la première partie de ce travail, nous avons proposé une amélioration au cryptosystème GGH en utilisant les réseaux de points GLD. Cela a réduit la complexité de ce système tout en garantissant sa sécurité. En effet, nous avons réduit de 10 fois la taille de la clé publique et de 1300 fois la complexité de génération de la clé. De plus, le décodage itératif des GLD est plus efficace que les algorithmes précédemment utilisés.

Dans la seconde partie de ce travail, nous avons étudié la sécurité des canaux de diffusion multi-utilisateur, ayant accès à des mémoires de caches, en présence d'un espion. Nous avons considéré les deux contraintes de sécurité individuelle et jointe. Nous avons considéré le scénario avec $K \geq 2$ récepteurs, parmi lesquels K_w récepteurs sont faibles et ont accès à des mémoires de caches de même taille, et K_s récepteurs sont forts et n'ont pas accès à des mémoires de cache. Nous avons dérivé des bornes supérieures et inférieures du compromis sécurisé capacité-mémoire en considérant différentes distributions de cache. Pour chaque contrainte, nous avons proposé des schémas de codage cache-canal joints qui atteignent les bornes inférieures, dont nous avons montré la pertinence en les comparant avec des schémas de codage séparé. De plus, pour $K = 2$, nous avons étudié une distribution symétrique de cache pour les deux utilisateurs ainsi qu'une distribution bilatérale asymétrique. Nous avons trouvé que la distribution symétrique est la plus mauvaise dans la plupart des cas. Pour les petites mémoires de cache, la distribution unilatérale est la meilleure. Pour des mémoires de cache plus grandes, la distribution bilatérale qui prend en considération les canaux des utilisateurs est le meilleur choix.

Introduction

The place that the Internet takes in our everyday lives is significantly growing over the years. In Europe, the proportion of households with Internet access increased from 30% to 84% over the last 10 years, according to the International Telegraph Union (ITU) [12]. The fact that the Internet has become the main communication medium becomes apparent if one considers throughput numbers. For instance, a recent Cisco study revealed that the global traffic reached 1.1 ZB per year in the end of 2016 and is expected to reach 2.3 ZB per year by the end of 2020 [13].

Accordingly, people get accustomed to use the Internet for more various purposes with less consideration for the privacy of the data they are exposing. Indeed, personal data is being daily exposed online through the Internet while achieving some critical private activities such as paying bills, banking and completing governmental processes. In addition, the Internet is also regularly being used as a mean to transmit personal data towards remote servers in order to perform secure storage, controlled sharing, among wider types of operations that are getting allowed by cloud-based computing power and storage means.

However, the Internet is highly vulnerable because of its open and global nature. Attacks carried out through the Internet are becoming more numerous and more sophisticated. The Identity Theft Resource Center reports show that 980 reported data breaches of companies and governmental agencies records occurred in 2016 in the US only [14]. These breaches exposed the personal data of millions of people including their names, addresses, credit card numbers, social security numbers, medical records and many other private information. At global level, Symantec reports that half a billion personal records were stolen or lost worldwide in 2015, including 78 million medical patient records [15].

There is therefore an urgent need to strengthen the Internet security. Among the security properties, there is a special need to enforce confidentiality of personal data

while they are transmitted. Data security protocols are used for that purpose. However, due to advances in quantum computing technology, new attacks are about to emerge that will require changes in these protocol designs. Hence, post-quantum security solutions are being extensively studied. Yet, current post-quantum cryptography algorithms are too heavy to be usable, mainly because of their huge key sizes compared to classical cryptosystems.

This problem is further aggravated if one considers the device constraints. Indeed, devices with low processing power, low memory space and low transmission throughput will have to transmit secure data in the near future, over an Internet exposed to quantum computer threats. This emphasizes the necessity for low complexity and small key size post-quantum algorithms.

The need to enforce the Internet security urged researches to explore beyond the existing notions of security and consider unsecured layers of the communication system. A promising technique suggests to combine security protocols with physical layer security. This new type of security improves the system robustness against quantum computer attacks since it is not based on complex computational problems as in classical cryptography. Instead, physical layer security makes use of the characteristics of the transmission channel to ensure its secrecy unlike traditional cryptography where the physical channel is never considered. Thus, applying physical layer security techniques with higher layers security protocols provides a new layer of security uncorrelated with the others and greatly increases the overall system security.

Besides security related problems, another aspect of future networks that has to be taken into consideration is the way the Internet usage pattern is evolving. Cisco study also shows that the Internet traffic during peak times is increasing much faster than the average Internet traffic. Indeed, traffic during the busiest hour of the day increased by 51% in 2015 while average traffic only increased by 29% during the same period. This calls for solutions, beyond the classical ones which increase the network throughput in general, in order to moderate the network usage during high traffic periods by taking advantage of stiller periods.

Recently, caching emerged as a promising technique to balance the network load between peak and off-peak periods. Its concept relies on pre-storing data during periods when the network is barely used and benefiting from this data during high network traffic times. Caching methods are currently used on the application layer of the communication systems. However, recent studies show that caching can be more beneficial if its design considers also the particular medium over which the transmission will occur. These studies did not consider physical layer security aspects of caching systems. Thus, security solutions for caching systems taking into account the physical transmission layer have yet to be explored.

The objective of this thesis is to study new security solutions for confidentiality of

data transmission. In a first phase, we focus on post-quantum cryptography solutions and in particular, lattice-based cryptography. Our aim is to design a lattice-based cryptosystem with a reduced key size that can be actually used in practice. In a second phase, we study physical layer secrecy of cache-aided communication systems. In these systems, security can be provided by exploiting both the nature of the transmission channel and the users' cache memories.

In this manuscript, Chapter 1 is dedicated to review the state of the art related to these two security aspects. We start by introducing post-quantum cryptography and in particular, lattice-based and code-based cryptography in Section 1.1. We then explore the advances in physical layer security and especially in cache-aided transmission systems in Section 1.2.

As a first part of the report, Chapter 2 presents our first thesis result which is a new GGH lattice-based cryptosystem using generalized low density lattices. First, we describe in detail the original GGH scheme and discuss its advantages and disadvantages in Section 2.1 and we review GGH-based improved schemes and discuss their drawbacks in Section 2.2. We then present in Section 2.3 our proposed public-key cryptography scheme based on generalized low density lattices. To demonstrate the robustness of our scheme, we carry out security analysis of all the known attacks against the GGH schemes and prove their failure against our proposed system. Our results are validated by some experimental results and are presented in Section 2.4. In Section 2.5, the advantage of our cryptosystem is emphasized by a complexity study and simulation results showing the effectiveness of the new scheme compared to existing ones.

The second part of the thesis report is devoted to securing cache-aided networks in Chapters 3 and 4 under different secrecy constraints. In Chapter 3, we study the security of the cache-aided packet-erasure broadcast channels under an individual secrecy constraint. We start by formally defining our problem in Section 3.1. Then, we present in Section 3.2 our joint cache-channel coding scheme and compute the lower bound on the secure capacity-memory tradeoff for the two-user scenario with cache only at the weaker of both receivers. The corresponding upper bound is proved in Section 3.3. To justify our choice for cache assignment, we compute the lower and upper bounds on the secure capacity-memory tradeoff for the two-user scenario under a symmetrical cache distribution in Sections 3.4 and 3.5. In Section 3.6, we compute the securely achievable lower bound using a separate cache-channel coding approach for the two-receiver case under one-sided cache assignment with the aim of emphasizing the profit of our joint coding scheme. In Section 3.7, we discuss and compare all of the bounds obtained in the previous sections. Finally, we extend our results to the K-receiver scenario and compute the general lower and upper bounds on the secure capacity-memory tradeoff in Sections 3.8 and 3.9.

In Chapter 4, we consider the joint secrecy constraint for the same scenario studied in the previous chapter. In Section 4.1, we present the modifications of the problem

definition imposed by the new constraint. Then, for the two-receiver one-sided cache assignment, we present and prove a lower bound on the secure capacity-memory tradeoff in Section 4.2 and compute the corresponding upper bound in Section 4.3. We devote Sections 4.4 and 4.5 to the symmetric cache assignment case where we demonstrate its lower and upper bounds. After that, we study the case of asymmetric two-sided cache distribution and give lower and upper bounds on the secure capacity-memory tradeoff in Sections 4.6 and 4.7 respectively. We dedicate Section 4.8 to the interpretation and comparison of the obtained bounds. In the previous sections, we consider the special case with two receivers. We study the general case with K receivers with cache at weak receivers only and provide lower and upper bounds on the generalized secure capacity-memory tradeoff in Sections 4.9 and 4.10, respectively.

Finally, we summarize the results of this work and present some future research perspectives in a general conclusion.

Chapter 1

Security in Communication Systems

As the need to strengthen the internet security increases, new techniques have to be developed to protect the confidentiality of data during transfer and storage. These techniques should also anticipate the development in quantum computing and the eventual attacks that will arise from this evolution.

Before delving into details on the specific security schemes that we study in this thesis, we find it interesting to position them relative to other security approaches within the communication system.

Protecting data security against eavesdroppers can be done at various layers of the classical open systems interconnection (OSI) model. Figure 1.1 depicts the seven layers of the OSI model with some examples of security approaches applied at each of the layers. According to the OSI model, transmitted data go through encapsulation and encoding operations as they travel down the sender communication stack. Symmetrically, data are decapsulated and decoded upon reception, traveling upward the receiver communication stack. In parallel with encapsulation/decapsulation and encoding/decoding, ciphering/deciphering operations may be applied.

Applicative security solutions are used to secure traffic relative to a specific application only. Moving downward the OSI model, security solutions become applicable to a greater number of applications. Reaching the network layer, the Internet protocol security (IPsec) can be used to secure any IP packet. In addition, the network layer is the last layer where end-to-end protocols can be applied. Indeed, data link layer, which consists of the media access control (MAC) and logical link control (LLC) sub-layers, provides node-to-node data transfer between two nodes that are directly connected to each other. Finally, at the physical layer, data are encoded and modulated before being

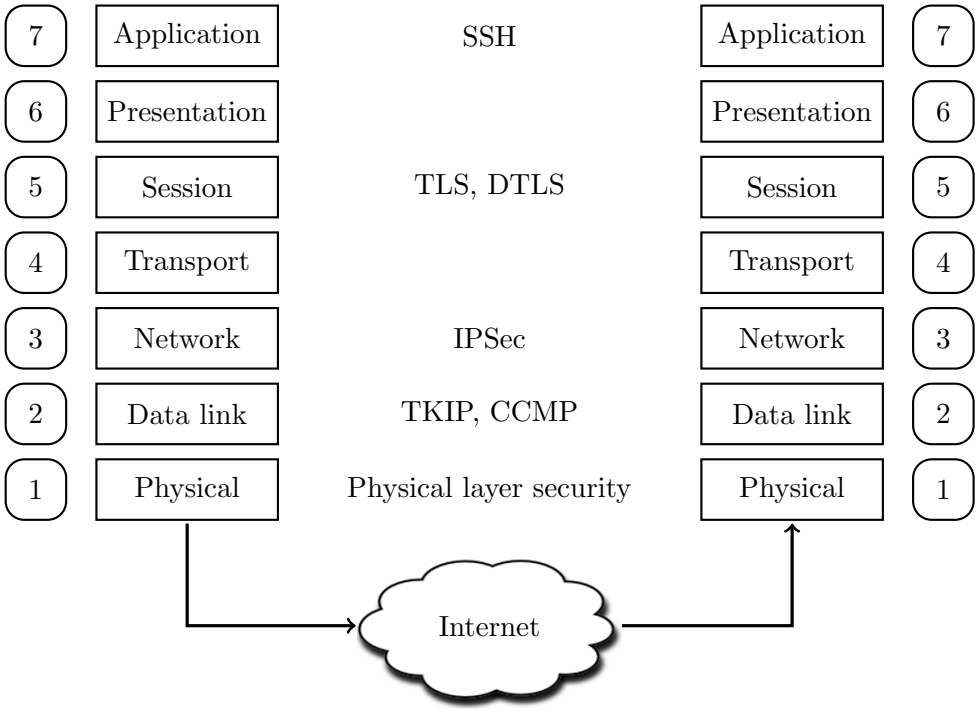

Figure 1.1: Communication security in the OSI model.

transmitted over the physical channel.

The security approaches employed at layers two to seven rely on cryptographic schemes. For example, the AES symmetric cryptosystem can be used in almost all protocol examples provided within Figure 1.1. These schemes ensure data security by means of complex mathematical problems or by reliance on complex key schedules. Thus, their security arises from the fact that the attacker has limited computational capacity.

On the other hand, physical layer security does not employ cryptosystems. While cryptography techniques are insensitive to the physical nature of the transmission channel, physical layer security exploits the characteristics of the wireless medium, such as multi-path fading and interference, to protect the confidential information against eavesdropping and other attacks. Since these security techniques do not make the same assumptions on the attacker as those relying on cryptographic primitives, they can be used to reinforce security of the higher communication layers.

Note that data security consists of confidentiality, integrity and availability properties. In this thesis, we focus on the confidentiality property only. Unless stated otherwise, we use the term "security" to refer to this confidentiality property as in most of the physical layer security literature.

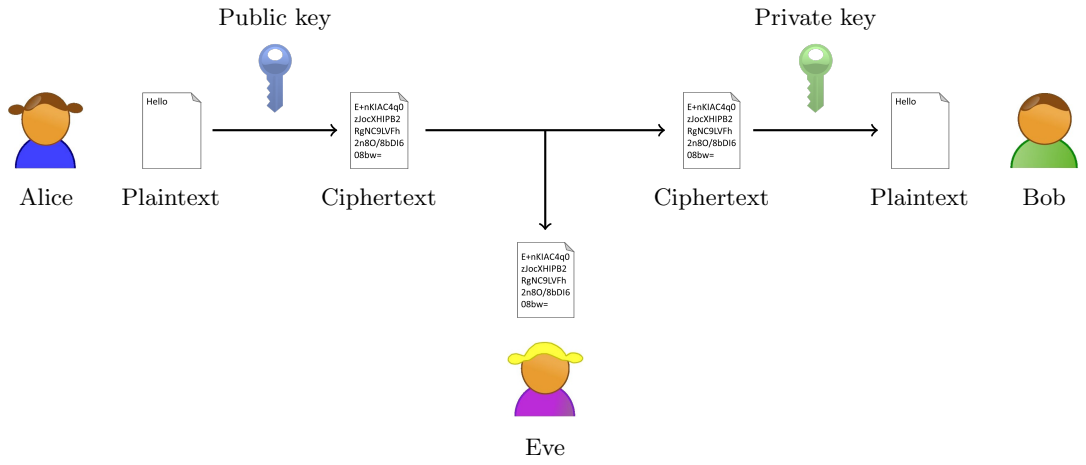

Figure 1.2: Cryptography scheme.

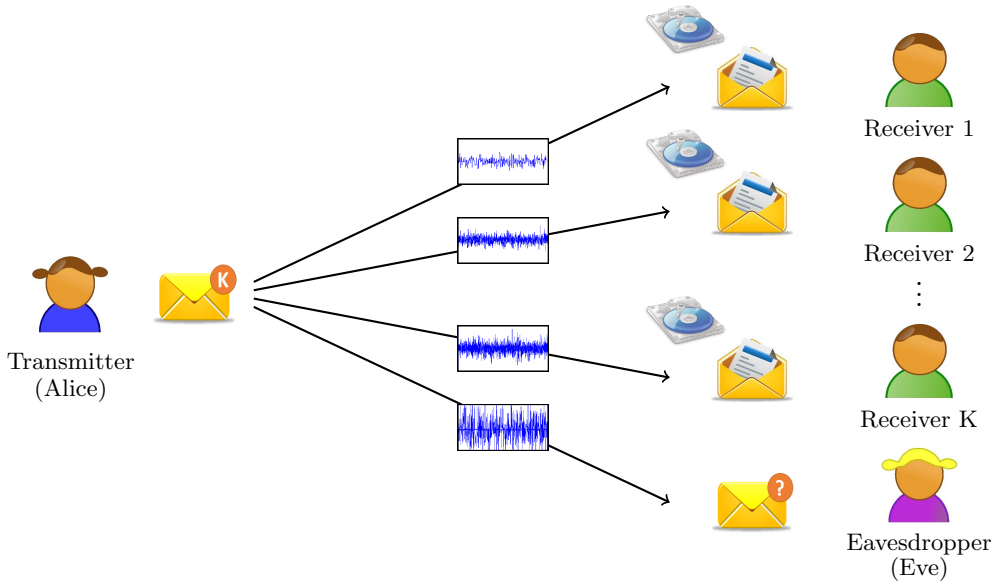

Figure 1.3: Caching scenario.

In this thesis, we will explore both cryptography and physical layer security of wireless communications. From a cryptographic perspective, we will study the scheme in Figure 1.2 where Alice wants to communicate a message, called *plaintext*, to Bob. She encrypts the plaintext with the public key generating the *ciphertext* to be transmitted. Upon message reception, Bob decrypts the ciphertext using its private key and obtains the plaintext. The ciphertext is also received by Eve who is eavesdropping the communication. However, Eve does not have access to the private key and thus, applies attack

algorithms that may allow her to decrypt without the private key and retrieve some confidential information. The main goal here is to design an encryption method that guarantees a failed decryption to Eve using any known attack algorithm.

From a physical layer security perspective, we will study the caching scenario depicted in Figure 1.3. In this scheme, the transmitter conveys messages to multiple users through a broadcast channel. Legitimate receivers have access to cache memories where messages can be pre-stored. Some receivers have good channels and others have worse channels. The broadcasted message is also received by an eavesdropper who does not have any cache memory and has the most degraded channel. The purpose here is to code the messages in a way to be decoded reliably by the legitimate receivers while being perfectly secured from the intruder.

1.1 Cryptography

Cryptography is the art of hiding information using a secret key. A plaintext, which is understood by everybody, is encrypted using an encryption key into an unreadable form called ciphertext. Only those who have access to the decryption key can decrypt the ciphertext and extract the message. The fundamental purpose of cryptography is to allow two people to exchange information through a channel in a way that the transmitted messages cannot be understood by any intruder listening to their communication.

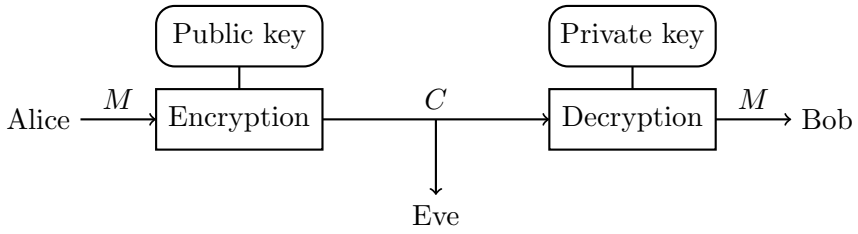

Figure 1.4: Public-key cryptography.

The process of designing cryptography systems involves three phases. First, cryptographers design schemes that they believe to be secure. Then, cryptanalysts try to break these schemes. Finally, the fastest of the unbroken schemes are chosen to be implemented and used in practice.

Cryptographic systems can be grouped into two main categories: private-key ciphers and public-key ciphers. In private-key cryptography, also called symmetric-key cryptography, the same private key is used for both encryption and decryption. Conversely, public-key cryptosystems use asymmetric algorithms where a public key serves for encryption and another private key serves for decryption. The public key is known to

everyone whereas the private key is possessed only by the legitimate receiver.

Nowadays, public-key encryption schemes are mainly based on three hard mathematical problems: the integer factorization problem, the discrete logarithm problem and the elliptic-curve discrete logarithm problem. No attack algorithm can solve these problems using a classical computer. However, Shor presented in [16] a quantum algorithm for integer factorization that runs in polynomial time making today's public-key cryptosystems easily breakable by a powerful quantum computer.

Most cryptography protocols that are adopted today to protect the Internet rely on public-key schemes, such as RSA and DSA. They are therefore proved to be broken by quantum computers according to Shor's algorithm. The day the construction of a powerful quantum computer succeeds, all of these protocols will be broken and the Internet will no longer be secure. Even though these computers do not exist today, cryptographers have to start preparing for this new era.

Quantum attacks do not present the same danger to symmetric cryptography algorithms. In fact, Grover designed another quantum algorithm that finds the input to a black box function based on its outputs [17]. However, it does not speed up attacks on symmetric ciphers as much as Shor's algorithm does on asymmetric ones. Therefore, doubling the key size of symmetric cryptography algorithms is enough to protect them against quantum computer attacks.

These facts directed the attention of cryptographers towards the design of post-quantum asymmetric cryptosystems secure against quantum algorithms attacks. Four categories of public-key ciphers are believed to resist against Shor's algorithms: multivariate quadratic equations cryptography, hash-based cryptography, code-based cryptography and lattice-based cryptography. We explore in Sections 1.1.1 and 1.1.2 the last two fields that are of interest to us in this thesis.

1.1.1 Code-based cryptography

Code-based cryptography systems are public-key encryption schemes that use error correction codes. McEliece developed the first code-based scheme using binary irreducible Goppa codes [18]. Encryption and decryption of this scheme are fast and efficient. However, it is not used in practice because of its large public and private keys sizes. The McEliece cryptosystem remains unbroken till today and it is believed to remain secure against attacks by quantum computers. It is described as follows:

- **System parameters:**
  - n : Length of the code.
  - t : Number of errors that the code can correct.

- **Key generation:** Generate the following matrices
  - G : $k \times n$ generator matrix of a binary irreducible Goppa code $C(n, k)$ of length n, dimension k and can correct t errors.
  - P : $n \times n$ randomly chosen permutation matrix.
  - S : $k \times k$ randomly chosen non-singular matrix.
- **Public key:** (G', t) where $G' = SGP$.
- **Private key:** (G, S, P).
- **Encryption:** Compute the ciphertext $\mathbf{c}$ as follows

$$\mathbf{c} = \mathbf{m}G' + \mathbf{e}, \tag{1.1}$$

  where $\mathbf{m}$ is the message of length k and $\mathbf{e}$ is an error vector of length n and weight t.
- **Decryption:** To decrypt the ciphertext, start by computing

$$\mathbf{c}P^{-1} = \mathbf{m}SG + \mathbf{e}P^{-1}. \tag{1.2}$$

  Then, apply the decoding algorithm of the code C to eliminate the erroneous part $\mathbf{e}P^{-1}$ and obtain $\mathbf{m}SG$. Finally, find $\mathbf{m}$ as

$$\mathbf{m} = (\mathbf{m}SG)G^{-1}S^{-1}. \tag{1.3}$$

A variant of the McEliece system is the Niederreiter cryptosystem [19]. Niederreiter proposed to encode the message into the error vector instead of representing it as a codeword. Originally, Niederreiter scheme was proposed with generalized Reed-Solomon codes (GRS) but these codes were proven to be insecure [20]. However, when designed with binary Goppa codes, Niederreiter has the same security as the McEliece system [21].

McEliece's main limitation is the size of its public key which is considerably larger than that of today's used publi-key encryption scheme, such as RSA. An interesting suggestion was to replace Goppa codes with low-density parity-check (LDPC) codes to overcome this drawback [22]. In fact, LDPC codes have sparse parity-check matrices with a storage size that increases linearly with the code length n [23]. However, this sparsity induces a vulnerability on the security of the system and hence cannot be exploited. To prevent this weakness, neither the public code nor its dual code should be too sparse. Yet by loosing the sparsity, the reduction in the key size is also lost. An alternative solution was to employ quasi-cyclic LDPC (QC-LDPC) codes, whose parity check matrices can be described by a single row of them and to increase their density resulting in quasi-cyclic moderate-density parity-check (QC-MDPC) codes [23].

Other codes were also investigated with the same aim of reducing the key size. Some examples of these codes are: Reed-Muller codes [24], Gabidulin codes [25] and BCH codes [26]. Many of these codes were broken, e.g. [27, 20]. The generalization of McEliece system with non-binary Goppa codes was also studied, e.g. [28],[29], and attacks on these systems were examined [30].

The most secure system remains the original McEliece based on Goppa codes. However, it is not secure against chosen-ciphertext attacks [31, 32]. In this attack, the attacker can enter one or more known ciphertexts into the system and obtain the resulting plaintext. Then, from these results he can attempt to recover the secret key used for decryption. Many works focused on the design of McEliece systems secure against adaptive chosen-ciphertext attack (CCA2), e.g. [33, 34, 35, 36].

1.1.2 Lattice-based cryptography

In this section, we introduce lattice codes and review the main results in lattice-based cryptography.

1.1.2.1 Lattice preliminaries

A real n-dimensional *lattice* Λ is a discrete subgroup of the Euclidean space $\mathbb{R}^n$. Λ can be represented by a set of n linearly independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$ in $\mathbb{R}^n$, called *basis vectors*. The matrix

$$B = [\mathbf{b}_1, \ldots, \mathbf{b}_n]^T \in \mathbb{R}^{n \times n}, \tag{1.4}$$

having the basis vectors as rows is called a *generator matrix* for the lattice. Thus, the lattice is obtained by taking all integral linear combinations of the basis vectors

$$\Lambda = \{\lambda = \mathbf{x}B : \mathbf{x} \in \mathbb{Z}^n\}. \tag{1.5}$$

Note that a lattice basis is not unique. There is an infinite number of bases for the same lattice but there exists only one basis that is of Hermite normal form (HNF).

A basis H is in Hermite normal form if it satisfies the following conditions:

1. H is lower-triangular , i.e. $h_{i,j} = 0 \ \ \forall\ i < j$.
2. $h_{i,i} > 0 \ \ \forall\ i$.
3. $0 \leq h_{i,j} < h_{j,j} \ \ \forall\ i > j$.

For every basis B of Λ, there exists a unimodular matrix U that, multiplied by B, generates the HNF form matrix H, as follows

$$H = UB. \tag{1.6}$$

The orthogonality of a basis B can be evaluated by its *orthogonality defect*. As introduced by Schnor in [37], it is the product of the basis vector lengths divided by the matrix determinant, defined as

$$\mathrm{OD}(B) = \frac{\prod_{i=1}^{n} ||\mathbf{b}_i||}{|\det(B)|}, \tag{1.7}$$

where $||\mathbf{b}_i||$ is the Euclidean norm of the i'th row in B.

If $\mathrm{OD}(B) = 1$, this means that B is an orthogonal basis of Λ. Conversely, when B is a bad basis, its $\mathrm{OD}(B)$ is high.

The n-dimensional space can be divided into an infinite number of similar regions, such that each copy of the region contains only one lattice point. This region is called *fundamental region* of the lattice. Its volume is denoted by the *fundamental volume* and is equal to the square of the determinant of B

$$\mathrm{vol}(\Lambda) = (\det(B))^2. \tag{1.8}$$

The fundamental region of a lattice is not unique but its fundamental volume is. An example of a fundamental region is the *fundamental parallelotope* which consists of the points

$$\theta_1\mathbf{b}_1 + \cdots + \theta_n\mathbf{b}_n \quad (0 \leq \theta_i < 1). \tag{1.9}$$

It is clear that the fundamental parallelotope's form depends on the choice of the lattice basis.

1.1.2.2 Lattice-based cryptography schemes

Lattice-based cryptography systems are public-key systems based on some hard lattice problems. Lattices were first used in cryptanalysis to break various cryptographic schemes. In 1996, Ajtai showed a connection between the average-case complexity and the worst-case complexity of some lattice problems [38]. This discovery opened the door for a different use of lattices in cryptography. From this point, lattices were greatly studied in the design of cryptosystems.

Lattice-based cryptosystems are believed to remain secure against quantum computers and hence they received a wide attention from the cryptography community. The first lattice-based cryptography scheme is the Ajtai-Dwork (AD) cryptosystem [39]. The security of AD is related to the hardness of the shortest vector problem (SVP). SVP

refers to the computation of a vector $\mathbf{u}$ in a lattice Λ, defined by a basis B, with minimum non-zero Euclidean length $\min\{ \|\mathbf{u}\| \ ; \ \mathbf{u} \in \Lambda, \mathbf{u} \neq \mathbf{0}\}$. However, AD is mainly of theoretical interest and far from being a practical cryptosystem.

Many ciphers were inspired from the AD system, such as Goldreich-Goldwasser-Halevi cryptosystem (GGH) [1] and NTRU cryptosystem [40]. The GGH scheme was proposed in [1] as a much more practical lattice-based alternative than the AD cryptosystem. GGH can be seen as the lattice-based analog of the McEliece cryptosystem, which uses linear codes for encryption [18]. GGH security relies on the hardness of the closest vector problem (CVP) in a lattice, which consists in finding the vector $\mathbf{u} \in \Lambda$ that minimizes the distance to a given vector $\mathbf{v} \in \mathbb{R}^n$. On the other hand, NTRU is a ring-based public-key cryptosystem whose encryption functions are designed using convolution polynomial rings and elementary probability theory. It can be interpreted as a SVP or CVP instance.

In 2005, Regev introduced the learning with errors (LWE) problem and proposed an LWE-based public-key cryptosystem that can be viewed as a decoding problem from a random linear code [41]. Later, multiple public-key encryption schemes were also proposed based on LWE (eg. [42, 43]).

Among these schemes, GGH is the only one that works explicitly with lattices. Despite some interesting properties and some efforts to improve it [7, 3], the drawback of GGH lattice-based cryptosystem remains the huge size of its public key. In this thesis, we propose to solve this issue by replacing random lattices used so far in GGH with structured low density lattices such as generalized low density (GLD) lattices. GLD lattices, GGH cryptosystems and the proposed GLD lattice-based GGH cryptosystem are the main focus of Chapter 2.

In the next section, we present physical layer security and the principal results in this domain. We also discuss in more detail security of the caching scenario on which we focus in the second part of this thesis.

1.2 Information-theoretic security

In 1949, Shannon introduced the principle of measuring the secrecy level of a communication system by a quantitative value [44]. He considered the model depicted in Figure 1.5 where a transmitter, Alice, wants to convey messages to a legitimate receiver, Bob, through a perfect channel, while an eavesdropper, Eve, is wiretapping the communication. He supposed that Eve also observes an error-free copy of all the transmitted messages.

He defined the notion of perfect secrecy which is achieved if the eavesdropper's equiv-

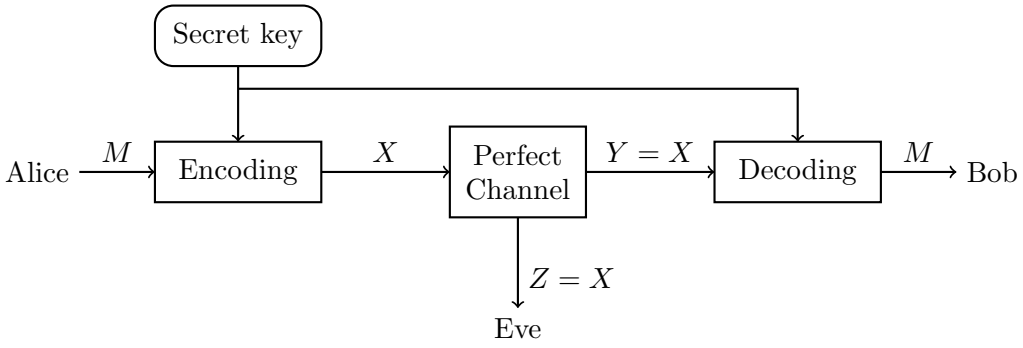

Figure 1.5: Shannon's model.

ocation of the message remains the same with and without the knowledge of its received vector

$$H(M|Z) = H(M). \tag{1.10}$$

This is equivalent to having zero mutual information between the source message and the eavesdropper's received vector

$$I(M;Z) = 0. \tag{1.11}$$

In his considered model, this perfect secrecy is achieved if the transmitted codeword is statistically independent of the message. It can be guaranteed by means of a one-time pad where the message is secured using a secret key known only by the transmitter and the legitimate receiver, and the key is used only once. For instance, the codeword can be computed as the binary addition (XOR) of the message and the secret key. However, the problem is that, since each key cannot be used more than once, the transmitter and the legitimate receiver have to store long sequences of random keys and share them over a secure channel. That caused information-theoretic security to be regarded as a theoretical concept unfeasible in practice.

In 1975, Wyner was the first to introduce the concept of creating security by exploiting the randomness of the noise present in all communication systems [45], which is known today as *physical layer security*. He considered the channel model depicted in Figure 1.6 which he called the *wiretap channel*. In this model, Alice conveys messages to Bob through a discrete memoryless channel (DMS). Eve observes the communication through a second discrete memoryless channel. The channel between Alice and Eve is called the wiretap channel and is considered to be degraded with respect to the main channel.

Wyner also introduced the notion of *secrecy capacity* as the maximal transmission rate that ensures a reliable decoding for Bob while preventing Eve from finding any information about the messages. It is equal to the difference between the capacity of the

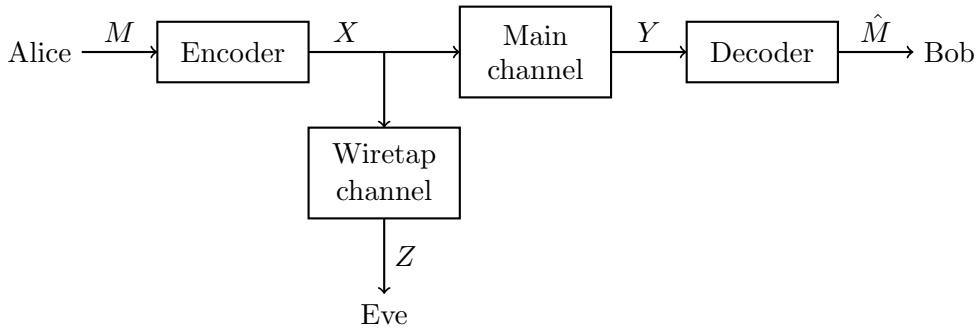

Figure 1.6: The wiretap channel.

channel between Alice and Bob and that of the channel between Alice and Eve

$$C_s = \max_{p_X} I(X;Y|Z) = \max_{p_X} \big[I(X;Y) - I(X;Z)\big], \tag{1.12}$$

where, X is the channel input, Y is the channel output at the legitimate receiver, Z is the channel output at the eavesdropper and p_X is the probability distribution of the channel inputs X.

Wyner proposed to use coset codes to encode the information bits with some random bits to achieve the secrecy capacity. The rate of the random bits should be at least equal to the eavesdropper channel capacity in order to sink it in total ambiguity.

Moreover, since the notion of perfect secrecy is too strict, two other secrecy notions were defined: strong secrecy condition [46] and weak secrecy condition [45]. The *strong secrecy condition* requires that the information leaked to the eavesdropper vanishes as the blocklength n of the codeword goes to infinity

$$\lim_{n\to\infty} I(M;Z^n) = 0. \tag{1.13}$$

The *weak secrecy condition* is less stringent. It requires that the rate of the information leaked to the eavesdropper vanishes as n goes to infinity

$$\lim_{n\to\infty} \frac{1}{n} I(M;Z^n) = 0. \tag{1.14}$$

1.2.1 Main results in physical layer security

After Wyner's discovery, determining this secrecy capacity has been addressed extensively and has led to a plethora of information-theoretical results for many classes of channels such as erasure, Gaussian, MIMO, broadcast, interference and relay channels (see [47] and references therein). The Gaussian wiretap channel was introduced in [48] and was then investigated in different scenarios [49, 50, 51]. Secure communications

over fading channels were also investigated [52, 53, 54]. The point-to-point scenario was extended to multi-user systems for different channels, e.g. multiple access channel [55, 56, 57], broadcast channel [58, 59, 60, 61] and relay channel [62, 63, 64].

In order to achieve this secrecy capacity, practical wiretap codes were designed based on the application of classic and modern coding techniques [65, 66]. First approaches considered capacity-achieving graph-based codes such as LDPC codes [67] and extended their applications for secrecy [68]. Then, some researches examined secrecy rate using structured nested lattice codes for wiretap coding [69, 70, 71]. Polar codes were also considered as secrecy achieving codes for some wiretap channels [72, 73, 74]. Physical layer security issues were also studied based on network coding theory in [75].

Secrecy capacities of multiple antenna channels were extensively studied. The main challenge in securing multiple-input multiple-output (MIMO) channels is that the wiretap MIMO channel is not degraded in general. Secret communications in MIMO channels where multiple antennas are used for transmission and reception were first studied by Hero in [76]. He introduced two constraints, called low probability of intercept and low probability of detection, and studied them in different scenarios where the transmitter, the legitimate receiver and the eavesdropper are either informed or uninformed about their channel states. The secrecy capacity of the single-input multiple-output (SIMO) wiretap channel where the eavesdropper has more than one antenna was analyzed in [77]. The secrecy capacity of the multiple-input single-output (MISO) wiretap channel was also investigated when the eavesdropper has a single antenna [78, 51] or multiple antennas [79, 80]. These studies were then generalized to the MIMO channel with single or multiple eavesdropper antennas [81, 82, 83, 84].

Secure communication over broadcast channels with receivers side information was also considerably investigated. In these channels, a legitimate receiver is given access to messages of other legitimate receivers as side information. This side information along with some random binning are used to secure the broadcasted messages. It is assumed that the eavesdropper does not have access to any side information. Security in these channels was studied under weak individual secrecy constraints [85, 86, 87] or weak joint secrecy constraint [87, 88]. Individual (respectively joint) secrecy constraint imposes that the messages should be individually (respectively jointly) secured from the eavesdropper.

Recently, security issues have started to be explored from an information-theoretic angle in caching systems. Caching is the process of storing data in local memories close to the users with the aim of reducing network congestion. Security in caching takes advantage of the fact that the eavesdropper does not have access to the users' cache memories. Secure caching codes have been designed in [89, 90, 91] without considering transmission channels. The employed security techniques are inspired from Shannon's one-time-pad scheme [44]. Hence, secure caching remains an interesting domain unexplored from a physical layer perceptive.

In the following section, we introduce cache-aided communication systems and review some secure coding methods designed for these systems.

1.2.2 Secrecy in caching scenario

Traffic in communication systems varies as a function of the time of the day resulting in periods where network congestion is high, causing packet loss, delivery delays and unsatisfied users, and other periods where the network is barely used. Lately, caching has emerged as a promising technique to reduce the network load and latency in such dense wireless networks. The main idea of caching is to benefit from the low network traffic periods to pre-store popular content in cache memories distributed across users. Then, when users request specific files during periods of peak-traffic, they are served partly from their cache memories and partly from the server, reducing thus the network load.

In such scenarios, communication is divided into two phases: the caching phase and the delivery phase. The caching phase occurs during the off-peak periods of the network. In this phase, fragments of popular contents are stored in users' cache memories or on nearby servers. The delivery phase occurs during the peak-traffic periods of the network. In this phase, the servers convey to the users their demanded files. The technical challenge in these networks is that in the caching phase the servers do not know exactly which files the receivers will demand during the delivery phase. They are thus obliged to store information about *all possibly requested files*, namely the *library*, in the receivers' cache memories.

First attempts to investigate the caching problem focused either on optimizing the cache content for a fixed delivery method or on optimizing the delivery phase for a fixed cache content and fixed users' demands. In the former case, the focus was directed towards studying the statistics of the users' demands in order to identify the most popular files [92, 93, 94]; whereas in the latter case, the delivery problem was studied with the aim of reducing the transmission rate while ensuring users' capacity to decode their requested files [95, 96]. In both cases, a caching gain is obtained since users can be locally served whenever they demand files present in their cache memories. These caching techniques were later referred to as *uncoded caching schemes* and their gain was called the *local caching gain*.

Afterwards, Maddah-Ali and Niesen showed in their seminal work [10] that the delivery (high-traffic) communication can benefit from the cache memories more than the obvious local caching gain arising from locally retrieving parts of the requested files. They assumed that the delivery phase takes place over an error-free broadcast channel (BC) and that all the receivers have equal cache sizes. The additional gain, termed *global caching gain*, is obtained through carefully designing the cached contents and applying a new *coded caching scheme* where the transmitter can simultaneously serve multiple

receivers.

Saeedi, Timo, and Wigger showed in [97, 9] that further global caching gain can be achieved by means of their new *piggyback coding scheme*, when the delivery phase is modeled as a packet-erasure BC and different receivers have different channel strengths. Piggyback coding is a *joint cache-channel coding scheme* where the encoder and the decoders simultaneously exploit the cache contents and the channel statistics. This is in contrast to *separate cache-channel coding* as in [10] where the cache encoder/decoders depend only on the cache content and the channel encoder/decoders depend only on the receivers' channel statistics, and thus are designed separately.

A different line of research has addressed security issues in cache-aided BCs [89, 90]. In [89], an external eavesdropper is not allowed to learn any information about the messages. More precisely, a *joint secrecy constraint*, where the eavesdropper is not allowed to learn anything about the entire library, is considered. A noiseless BC for all legitimate receivers as well as for the eavesdropper is studied. Moreover, it is assumed that all legitimate receivers have cache memories of equal size, while the external eavesdropper does not have access to these caches. In the caching phase, random keys are stored in addition to the cached messages in users' cache memories. In the delivery phase, the stored keys are used to secure the transmitted messages in the form of a one-time pad scheme.

In [90], the security between users themselves without an external eavesdropper is studied. As in [89], it is assumed that delivery communication takes place over a noise-free BC, and that all legitimate receivers have the same cache memory size. The same joint secrecy constraint is imposed, however in this scheme, any legitimate receiver also acts as an eavesdropper. It is thus not allowed to learn anything about the files requested by the other receivers from its cache content or the broadcasted message. Therefore, uncoded fragments of the messages cannot be stored in users' caches and it is proposed to cache random keys and combinations of the messages XORed with random keys. In the delivery phase, messages (or combination of messages) XORed with some random keys are transmitted in a way that each message is decoded only by its intended receiver.

Moreover, security in device to device cache-aided networks was studied in [91] where the transmission between the users is secured from an external eavesdropper. The problem of secure caching using maximum distance separable (MDS) codes at the wireless edge in heterogeneous networks was also addressed in [98, 99].

All of the previously described works assume a separate cache-channel coding approach as in [10], and focus only on the design of the cache encoder and decoders while considering that the BC is a noise-free pipe from the transmitter to all receivers. However, this approach was shown in [97, 9] to be highly sub-optimal when there is no secrecy constraint. In this thesis, we investigate security in cache-aided networks from physical layer security perspective in Chapters 3 and 4.

1.3 Conclusion

In this chapter, we have introduced two important aspects of communications security, namely post-quantum cryptography and physical layer security, that will help to secure future networks. We have reviewed the state of the art of these two domains. This analysis have led us to identify a promising post-quantum cryptography solution, which is GGH cryptosystem. However, this cryptosystem suffers from drawbacks preventing its practical use. In the Chapter 2, we propose to improve GGH cryptosystem in a way to eliminate its disadvantages.

Regarding physical layer security, our state of the art analysis highlighted the need to study security in cache-aided networks by jointly considering the communication channel and the cache design in order to optimize the secure transmission rate. In Chapters 3 and 4, we investigate communication in cache-aided networks under individual and joint secrecy constraints, respectively.

Chapter 2

GLD Lattice-Based Cryptosystem

Inspired by LDPC and MDPC code-based cryptography, we investigate in this chapter the use of *generalized low density* lattices in the GGH cryptosystem. GLD lattices caught our attention because of their low complexity lattice generation and decoding which is a major factor in defining the practicality of cryptography systems and a weak point in the previous GGH schemes. Our main goal is to reduce the complexity of the GGH cryptosystem making it a candidate to replace the traditional public-key encryption schemes while verifying that this reduction in complexity does not affect the security of the scheme.

In the sequel, we start by reviewing some existing versions of GGH systems related to our work. Then, we introduce GLD lattices and describe our GLD lattice-based cryptosystem. Finally, we analyze the security and complexity of the proposed scheme and provide experimental results.

2.1 Original GGH scheme

The Goldreich-Goldwasser-Halevi cryptosystem stemmed from the lattice closest vector problem and was proposed originally in [1]. The authors took advantage of the fact that it is easy to generate a random vector close to a lattice point; however, it is hard to find the lattice point which is the closest to this random vector. This is the main idea on which is based the GGH trapdoor function.

Lattice-based GGH scheme uses also the lattice properties of having an infinite number of bases and generating easily a lattice point using any basis of the lattice. However, after adding a noise to the lattice point and obtaining a "close-to-lattice" point, only a near orthogonal basis can be used to recover the initial lattice point. Thus, the GGH

private key is chosen as a good orthogonal basis allowing simple recovery of the encrypted message. For the public key, a bad non-orthogonal basis is chosen rendering the decryption by any attacker nearly impossible. The badness of the basis is measured by its orthogonality defect, defined in (1.7).

In fact, in the original GGH cryptosystem, the private key R, which is a good basis of an n-dimensional random lattice Λ, is an $n \times n$ non-singular integer matrix defined by

$$R = \sqrt{n}I + Q, \tag{2.1}$$

where I is the identity matrix and Q is a random matrix with elements uniformly chosen from the set $\{-4, \ldots, 4\}$. The public key B is a bad basis of the same lattice Λ, hence, it is also an $n \times n$ integer matrix. B is generated by transforming the good basis of the lattice into a bad one. It can be obtained by applying some elementary linear combinations on the basis vectors of the private key R. In order to encrypt a message $\mathbf{m} \in \mathbb{Z}^n$, $\mathbf{m}$ is multiplied by the public basis B generating a lattice point $\mathbf{x}$. Then, $\mathbf{x}$ is secured by adding an n-dimensional error vector $\mathbf{e}$ chosen uniformly from the set $\{-A, A\}^n$. The ciphertext $\mathbf{c}$ is given by

$$\begin{aligned} \mathbf{c} &= \mathbf{m}B + \mathbf{e} \\ &= \mathbf{x} + \mathbf{e}. \end{aligned} \tag{2.2}$$

To decrypt the ciphertext $\mathbf{c}$, it is first multiplied by the inverse of the private basis R^{-1} since it is the good basis capable of canceling the error vector. Once the noise is removed, the result is multiplied by R and then by the inverse of the public basis used for encryption to recover the message $\mathbf{m}$ as

$$\mathbf{m} = \lfloor \mathbf{c}R^{-1} \rceil RB^{-1}. \tag{2.3}$$

The value of A was defined by studying the decryption errors at the legitimate receiver. Indeed, a decryption error occurs if $\lfloor \mathbf{e}R^{-1} \rceil \neq \bar{0}$. The authors in [1] considered two possible noise definitions. The first one allows zero error probability by taking $A < 1/(2\rho)$ where ρ is the maximal L_1 norm of the columns of R^{-1}. In the second one, a threshold on the error probability is fixed and the maximal A is computed with respect to this threshold. The second method relaxes the constraint on the error probability yielding a higher A value in order to have better security.

Advantages: The most appealing property in the GGH scheme is its low complexity encryption and decryption procedures in comparison with other cryptographic systems, for instance RSA and ElGamal encryption schemes. Indeed, its encryption time increases linearly with the key size. This makes the GGH lattice-based scheme interesting and deserving further studies.

Disadvantages: The major flaw in this scheme is the particular form of the noise which makes the system vulnerable and hence, easily breakable by the embedding attack

[2]. Indeed, the elements of the noise vector $\mathbf{e}$ are chosen from two exact values $\pm A$. Nguyen showed in [2] that for this noise distribution, the CVP is reduced to a simpler CVP instance for which the noise vector is much smaller $\mathbf{e}' \in \{-1/2, 1/2\}$. By exploiting this weakness, he was able to break four of the five challenges published by the GGH authors from dimension $n = 200$ until $n = 350$. The only challenge that remained unbroken was for dimension $n = 400$, where the key size is 2 MB. Nguyen concluded that GGH requires working in high lattice dimensions to be secure. However, for high dimensions, the GGH has a huge public key size. Therefore, major improvements were needed on the original GGH cryptosystem in order for it to be considered as a serious alternative to the existing public-key cryptosystems.

2.2 GGH improvements

Despite Nguyen's conclusion on the practicality of the GGH system, its fast encryption procedure kept it an interesting scheme. Many GGH-based improvements were proposed in order to overcome the original GGH drawbacks. These works, on one side, focused on reducing the public key size resulting in new designs of public lattice bases. On the other side, they aimed at improving GGH security by proposing new methods of noise generation.

2.2.1 Micciancio's scheme

In [7], Micciancio applied some changes to both aspects, public key generation and noise vector choice. He considered two encoding methods. In the first method, the message is encoded in the lattice point and the noise vector is chosen uniformly from an interval $[-A, A]$ whereas in the second one, the lattice point is chosen at random and the message is embedded in the error vector. Since Nguyen's embedding attack relies on the fact that the noise entries have all the same absolute value, the uniform noise provides the GGH scheme with the necessary robustness against this attack. Note that the first method is analog to the McEliece cryptosystem and the second one is analog to the Niederreiter system which is a variant of McEliece.

For the public key, Micciancio proposed to use the HNF lattice basis. He proved that this HNF basis reduces the public key size while guaranteeing, at least, the same security level as the original GGH [7]. Indeed, from a complexity perspective, he showed that the lower triangular form of the HNF matrix induces a reduction in the key storage size from $O(n^3 log_2(n))$ to $O(n^2 log_2(n))$. And from a security perspective, since for a lattice Λ there exists only one basis in the form of an HNF matrix, the HNF public key does not give any information about the private key from which it was generated. Moreover, HNF is a bad basis with high orthogonality defect. Thus, using the HNF for the public

key can only improve the security of the scheme.

Advantages: The decrease in the public key size induced by the HNF matrix makes the GGH system more likely to be used in practice. Moreover, the uniform noise allows to overcome the GGH weakness to the embedding attack.

Disadvantages: The security and efficiency of Micciancio's cryptosystem were analyzed by Ludwig in [8]. He highlighted the main disadvantages of Micciancio's system. On one hand, the decryption is slow and suffers from instability since it employs Babai's nearest plane algorithm. On the other hand, the public key generation process has high running time for the proposed lattice dimensions due to the non-existence of low-complexity algorithms for computing the HNF matrix.

Moreover, in Ludwig's results [8], the public keys turned out to have larger size than in Micciancio's experiments. Ludwig also found that the minimal dimension for which the system is secure in practice is around 800, whereas Micciancio presumed this dimension to be 500. All these findings [8] led to conclude that Micciancio's cryptosystem is still far from being practical.

2.2.2 LDLC scheme

Recently, Hooshmand and Aref suggested to replace random lattices by latin square low density lattice codes (LDLC) [3]. They chose these lattices because they can be decoded by means of an iterative low complexity decoding algorithm. To generate their private key, they start by defining a set of generating sequence

$$\mathcal{H} = \{h_1, h_2, \ldots, h_d\}, \tag{2.4}$$

with d rational elements, such that $1 \geq h_1 \geq h_2 \geq \ldots \geq h_d \geq 0$. d is the degree of the parity check matrix H of the lattice, i.e. the number of non-zero elements in each column and each row of H. Then, they generate a set of d indices

$$\mathcal{P} = \{p_1, p_2, \ldots, p_d\}, \tag{2.5}$$

such that $1 \leq p_i \leq n, \forall i \in \{1, \ldots, d\}$. $\mathcal{H}$ and $\mathcal{P}$ are used to generate the $n \times n$ parity check matrix H of the used LDLC and they are chosen in a way to obtain $|\det(H) = 1|$. The private generator matrix is then computed as $G = H^{-1}$. The public key G' is defined as the HNF of the LDLC generator matrix G following Micciancio's proposition. The encryption procedure of the LDLC scheme is similar to the previous schemes except for the choice of the error vector $\mathbf{e}$. It follows a Gaussian distribution with zero mean and variance σ^2 upper bounded by the Poltyrev limit $\sigma^2 < 1/(2\pi\mathrm{e})$ [4].

Advantages: Introducing low density lattices in GGH scheme helps fixing the ciphertext decryption problem. Indeed, the decoding of random lattices uses Babai's algo-

rithms which are sub-optimal lattice decoding algorithms. On the contrary, iterative decoding algorithms offer faster and close to optimal decoding.

Disadvantages: The main problem of this system is that the used LDLCs are rational lattices, which induces many drawbacks. First, the public key size is still very large because rational HNF matrices are employed. Second, the complexity of the HNF algorithm is further increased in this scheme. This is due to the fact that HNF cannot be applied on rational numbers and has to be applied on integer numbers. Thus, an additional procedure is applied before and after the HNF algorithm to transform rational numbers into integers and transforms them back at the end.

2.2.3 Other GGH improvements

Based on Micciancio's system, [100] focused on proposing a larger noise vector $\mathbf{e}$ to obtain a harder CVP problem. In fact, the original GGH defines $\mathbf{e}$ such that its product by the inverse of the private key generates a null vector, i.e. $\lfloor \mathbf{e}R^{-1} \rceil = \mathbf{0}$. In [100], the error vector $\mathbf{e}$ has $(n-k)$ coordinates chosen from a set of integers $I_1 = \{-s, \ldots, -1, 1, \ldots s\}$ and the remaining k from a second set of integers $I_2 = \{-h, h\}$ with $h > s$. In this way, the product $\lfloor \mathbf{e}R^{-1} \rceil$ results in $(n-k)$ zero elements and k elements with value $\{\pm 1\}$.

$$\lfloor \mathbf{e}R^{-1} \rceil = (v_1, \ldots, v_n), \tag{2.6}$$

where

$$v_i = \begin{cases} 0 & \text{for } (n-k) \text{ values of } i, \\ \pm 1 & \text{for } k \text{ values of } i. \end{cases} \tag{2.7}$$

It was stated in [100] that this larger error vector fixes the GGH flaws showed by Nguyen [2]. However, it was shown in [101] that the GGH version of [100] has the same behavior as the original GGH and its security can be improved by optimizing the parameters s and h defining the sets I_1 and I_2. The complexity and the key size of this GGH version [101] were studied in [102] and the use of polynomial rings was proposed.

Another independent attempt to reduce the public-key size of the GGH scheme using polynomial representations was proposed in [103] but then broken in [104].

2.3 GLD lattice-based cryptosystem

Motivated by the benefits of low density lattices and to overcome the drawbacks of the LDLC-based scheme, we propose in the sequel to use the generalized low density lattices to design an improved version of GGH cryptosystem. In this section, we start first by introducing the construction of GLD lattices. We then present our GLD lattice-based cryptosystem.

2.3.1 GLD lattices

GLD lattices were proposed by Boutros *et al.* in [5]. They are integer lattices in $\mathbb{Z}^n$ and have sparse parity-check matrices. These properties enable low complexity iterative decoding, hence allowing the codes to be used in dimensions up to 1 million. It was shown that these lattices achieve asymptotically Poltyrev limit [4].

GLD lattices can be generated using construction A from linear GLD codes $C_{\mathrm{GLD}}[n,k]$ with length n and dimension k as follows

$$\Lambda = C_{\mathrm{GLD}} + p\mathbb{Z}^n, \tag{2.8}$$

where n is the lattice dimension and p is a prime number.

This construction considers an elementary small linear code $C_0[n_0,k_0]$ defined over the finite field $\mathbb{F}_p$. Denote by H_{C_0} the parity-check matrix of C_0. A second code C_1 is generated as the direct sum of L copies of C_0

$$C_1 = C_0^{\oplus L}. \tag{2.9}$$

The parity-check matrix of C_1 is given by

$$H_{C_1} = \begin{bmatrix} H_{C_0} & 0 & \dots & 0 \\ 0 & H_{C_0} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & H_{C_0} \end{bmatrix}. \tag{2.10}$$

Now, let $\pi_1 = \mathrm{id}$ and $\pi_2, \dots, \pi_J$ be J permutations of $\{1, 2, \dots, n\}$. A GLD code $C_{\mathrm{GLD}}[n,k]$ is defined as

$$C_{\mathrm{GLD}} = \bigcap_{j=1}^{J} \pi_j(C_1) = \bigcap_{j=1}^{J} \pi_j(C_0^{\oplus L}), \tag{2.11}$$

where $\pi_j(x_1, x_2, \dots, x_n) = (x_{\pi_j(1)}, x_{\pi_j(2)}, \dots, x_{\pi_j(n)})$.

The matrix H_C of the GLD code is a non-square matrix obtained as

$$H_C = \begin{bmatrix} H_{C_1} \\ \pi_2(H_{C_1}) \\ \vdots \\ \pi_J(H_{C_1}) \end{bmatrix}. \tag{2.12}$$

Note that H_C has $n - k = JL(n_0 - k_0)$ rows and $n = Ln_0$ columns.

In order to define the generator matrix G_C of the GLD code, the systematic form of the parity-check matrix H_C is computed using the Gaussian elimination algorithm. $H_{C,\text{syst}}$ has the following form

$$H_{C,\text{syst}} = [\ I \mid -B^t\ ]. \tag{2.13}$$

Then, the $k \times n$ generator matrix of the code is defined by

$$G_C = [\ B \mid I\ ], \tag{2.14}$$

Based on construction A (2.8), the generator matrix of Λ is given by

$$G_\Lambda = \begin{bmatrix} pI & 0 \\ B & I \end{bmatrix}. \tag{2.15}$$

B is a dense matrix whose elements belong to the finite field $\mathbb{F}_p$. We can see that G_Λ is a lower triangular matrix. Its column elements are modulo the diagonal elements. Therefore, G_Λ is in Hermite normal form.

Remark 2.1. For our experiments, we fix the number of permutations to $J = 2$. However, a bigger value can be used with a slight increase in the system's complexity.

2.3.2 Proposed public-key scheme

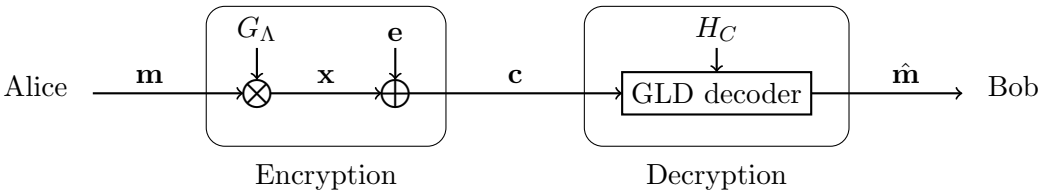

Figure 2.1: GLD-based GGH cryptosystem

Our GLD-based GGH cryptosystem, depicted in Figure 2.1, is described as follows:

- The private key is the parity-check matrix H_C of a GLD code. The non-zero elements of H are chosen from the field $\mathbb{F}_p \setminus \{0\}$ where p is a prime number.

  Note that, to store the private key, it is enough to save the parity-check matrix H_0 of the elementary code and the $(J-1)$ random permutations $\pi_2, \ldots, \pi_J$.

- The public key is the generator matrix G_Λ of the GLD lattice which is an HNF matrix.

- To encrypt a message $\mathbf{m} \in \mathbb{Z}^n$, $\mathbf{m}$ is multiplied by the public key G_Λ and an error vector $\mathbf{e}$ is added to obtain the ciphertext

$$\begin{aligned} \mathbf{c} &= \mathbf{m}G_\Lambda + \mathbf{e} \\ &= \mathbf{x} + \mathbf{e}. \end{aligned} \tag{2.16}$$

  The noise vector $\mathbf{e}$ is a random noise chosen uniformly from an interval $[-A, A]$.

- To decrypt $\mathbf{c}$ and recover the message $\mathbf{m}$, the GLD iterative decoding is applied.

2.3.2.1 Choice of the noise interval $[-A, A]$

In our scheme, we consider the uniform noise proposed by Micciancio since it is enough to ensure the system's security. The value of A defines the interval from which the noise elements are chosen. This value should not exceed the maximal noise variance for which the iterative decoder succeeds. On the other hand, it should be large enough to create sufficient ambiguity to prevent the attacker from decrypting the message.

We first start by finding the maximal value A tolerated by the GLD lattices decoder. It depends on the parameters, n_0, k_0 and p of the elementary code C_0. We describe how to compute A theoretically and experimentally.

Theoretical definition: The Poltyrev limit of the noise variance expressed as function of the parameters of the GLD lattices is defined as

$$\sigma^2_{\text{Polt}} = \frac{\text{vol}(\Lambda)^{2/n}}{2\pi\text{e}} = \frac{p^{\frac{2(n-k)}{n}}}{2\pi\text{e}} = \frac{p^{2(1-R)}}{2\pi\text{e}}, \tag{2.17}$$

where R is the rate of the GLD code C_{GLD}.

GLD lattices can be decoded with a negligible decoding error probability if the noise variance is smaller than the Poltyrev limit. Let b be a real noise value chosen uniformly from the interval $\mathcal{B} = [-A, A]$. For large n, the variance of the noise b is given by

$$\sigma^2_b = \int_{-A}^{A} \frac{1}{2A} b^2 db = \frac{A^2}{3}. \tag{2.18}$$

This variance and thus the noise interval, should satisfy

$$n\sigma^2_b < n\sigma^2_{\text{Polt}} \quad \Rightarrow \quad A^2 < 3\frac{p^{2(1-R)}}{2\pi\text{e}}. \tag{2.19}$$

Therefore, we choose the noise $\mathcal{B} = [-A, A]$ for a certain GLD by taking the maximal value of A that satisfies (2.19).

Experimental results: We also determine experimentally the maximum noise interval tolerated by the iterative decoding for different elementary codes C_0 with different n_0, k_0 and p. The results of this experimentation are shown in Table 2.1. As we will see later, these A values are sufficient to ensure the security of our encryption scheme.

Table 2.1: Maximal value A defining the noise interval for lattice dimension $n \approx 1000$.

C_0	p	n	A
$C_0[3,2]$	11	999	1
$C_0[3,2]$	17	999	2
$C_0[3,2]$	29	999	3
$C_0[8,6]$	17	1000	1
$C_0[8,6]$	53	1000	2

2.3.2.2 Choice of the lattice dimension n

In our scheme, we consider GLD lattices with dimension $n \approx 1000$. This choice is inspired by previous works and based on the security analysis of the proposed GLD lattice-based GGH cryptosystem that we conducted. In the following sections, we prove through experimental results that our system is both secure and practical for this dimension.

2.4 Security analysis

In this section, the security of our proposed scheme is studied from three perspectives. First, we investigate the complexity of the exhaustive attack to recover the private key. Then, we analyze the decoding attacks to decrypt the transmitted ciphertext. Finally, we discuss the dual code attack to find the private matrix.

2.4.1 Brute-force attack

The most naive attack that can be applied is the brute-force attack. In this attack, the eavesdropper attempts to find the private key by exhaustively testing all the possibilities. Hence, the complexity of this attack depends on the number of private keys that are likely to be used.

In our case, the search space of the private key depends on the number of possible matrices H_C. Recall that H_C is generated by the concatenation of $H_{C_0^{\oplus L}}$ with $(J-1)$

of its permuted versions. The number of possible versions is $n!$ since they are generated by permuting the n columns of $H_{C_0^{\oplus L}}$. Therefore, the search space for the private key is

$$n! \times (J-1). \tag{2.20}$$

The formula is dominated by $n!$, which can be approximated using Stirling's formula into

$$n! \sim \sqrt{2\pi n}\Big(\frac{n}{e}\Big)^n. \tag{2.21}$$

Since it is exponential in n, then for $n \geq 1000$, the search space becomes huge, making this attack unfeasible.

2.4.2 Decoding attacks

In the decoding attacks, the eavesdropper aims at decrypting the ciphertext, which is equivalent to finding the closest lattice vector to this ciphertext. In fact, the GGH system is essentially a closest vector problem instance which makes the decoding attacks the most obvious attacks on such cryptosystems. These approaches to attack GGH apply Babai's algorithms [6], namely round-off and nearest plane algorithms, resulting in the well-known round-off attack and nearest plane attack.

The performance of these attacks can be improved by applying bases reduction techniques, typically Lenstra-Lenstra-Lovász (LLL) and Block Korkine-Zolotarev (BKZ), before running the decoding algorithms.

2.4.2.1 The round-off attack

The round-off attack [6] consists in multiplying the ciphertext $\mathbf{c}$ by the inverse of the public basis G_Λ^{-1} which results in a noisy message as follows:

$$\mathbf{c}G_\Lambda^{-1} = \mathbf{m} + \mathbf{e}G_\Lambda^{-1}. \tag{2.22}$$

By rounding this result, the message $\mathbf{m}$ can be found only if $\lfloor \mathbf{e}G_\Lambda^{-1} \rceil = 0$. Thus, the feasibility of this attack depends mainly on the orthogonality of the public basis G_Λ and the variance of the noise $\mathbf{e}$.

2.4.2.2 The nearest plane attack

The nearest plane attack [6] is an improvement of the round-off attack that uses a better approximation for the CVP. The idea is to find the nearest lattice point by considering one dimension after the other. For every basis vector $\mathbf{g}_i$, $i = 1, \ldots, n$, the nearest plane

algorithm finds the closest hyperplane to the ciphertext $\mathbf{c}$. It starts by finding the integer multiple k_n of $\mathbf{g}_n$ that minimizes the distance to the hyperplane spanned by the basis vectors $\{\mathbf{g}_1, \ldots, \mathbf{g}_{n-1}\}$ by projecting $\mathbf{c} - k_n\mathbf{g}_n$, $k_n \in \mathbb{Z}$, onto this $(n-1)$-dimensional hyperplane. Then, the algorithm proceeds recursively until dimension 1. Finally, the lattice vector is obtained by the sum of all vectors $k_i\mathbf{g}_i$.

Various improvements of the nearest plane attack have been proposed to increase its success probability. Some applied basis reduction procedures to the remaining basis vectors at each level of the recursion. Others considered more than one basis vector in each level of the recursion or examined more than one closest hyperplane.

Note that both Round-off attack and nearest plane attacks can be seen as Zero-forcing (ZF) and ZF-decision feedback equalizer (ZF-DFE) decoders, respectively. These decoders are known to be sub-optimal for lattice decoding. Though they have low complexity, they are prone to error propagation that can degrade considerably the performance for high dimensions.

2.4.2.3 Lattice reductions

Lattice reduction approaches are applied to generate good basis with short, nearly orthogonal vectors. They can reduce the decoding errors and thus improve the decryption. In high dimensions, these techniques need a huge running time to provide a shorter basis. Yet, they may fail to produce this basis rendering the decryption unfeasible.

Indeed, the LLL algorithm succeeds to generate a reduced basis for small dimensions. For high dimensions, BKZ, which is an improved variant of LLL, needs to be used. While LLL operates on each basis vector alone, BKZ works on blocks of t basis vectors. The complexity of BKZ algorithm is $O(n^3 t^{t+o(t)} + n^4)$ [8]. For small t, BKZ will not result in a reduced basis for high dimensions. Increasing t will increase the complexity and cannot guarantee a good basis. In our scheme, we consider lattices of dimension $n \approx 1000$. For this dimension, t should be very large to increase the probability of obtaining a good basis making the BKZ reduction algorithm impractical.

2.4.2.4 Discussion and experimental results

The success of the previously described decoding attacks depends on two factors: the lattice public basis and the noise vector.

a - Public basis: After applying the reduction algorithms, if the resulting basis is quasi-orthogonal these attacks will be able to decrypt the ciphertext and recover the message. Conversely, if the reduction fails, the attacks have to be applied on a bad basis, and thus, the decryption will certainly fail. Therefore, to strengthen the system's

security, the public key should be a lattice basis as bad as possible with large dimension in order to prevent reduction success.

In our scheme, the generator matrix G_Λ is in HNF. To evaluate its orthogonality, we compute the orthogonality defect of G_Λ as

$$\mathrm{OD}(G_\Lambda) = \frac{\prod_{i=1}^{n} \|g_i\|}{|\det(G_\Lambda)|}, \tag{2.23}$$

where $\|g_i\|$ is the Euclidean norm of the i'th row in G_Λ. Since G_Λ is given by construction A in (2.15), the first $(n-k)$ rows have norm p and the determinant is $\det(G_\Lambda) = p^{n-k}$, then

$$\begin{aligned} \mathrm{OD}(G_\Lambda) &= \frac{p^{n-k} \prod_{i=n-k+1}^{n} \|g_i\|}{p^{n-k}} \\ &= \prod_{i=n-k+1}^{n} \|g_i\| \\ &= \prod_{i=n-k+1}^{n} \sqrt{1 + \|b_i\|^2}, \end{aligned}$$

where $\|b_i\|$ is the Euclidean norm of the i'th row in B.

B is a dense matrix with elements in the finite field $\mathbb{F}_p$. Thus, $\|b_i\|^2 \gg 1 \Rightarrow \|g_i\| \gg \sqrt{2}$, yielding an orthogonality defect $\mathrm{OD}(G_\Lambda) \gg (\sqrt{2})^k \gg 1$, showing hence that G_Λ is a bad basis.

We have carried out some experiments to confirm this result. Table 2.2 presents the average length of the basis vectors and the orthogonality defect of G_Λ in dimension $n \approx 1000$ for different elementary codes C_0. These values indicate that for this dimension, the orthogonality defect ranges approximately from $O(10^{500})$ to $O(10^{1000})$. Figure 2.2 presents the $\mathrm{OD}(G_\Lambda)$ in logarithmic scale as function of the GLD lattice dimension n for the elementary code $C_0(8,6)_{17}$. The curves are illustrated for the initial GLD public basis as well as the reduced bases by LLL and BKZ for several block sizes t. We can see that $\mathrm{OD}(G_\Lambda)$ increases significantly with the dimension. Thus, applying either basis reduction is not efficient and cannot yield reduced basis. We also notice that increasing the block size for the BKZ algorithm from $t = 20$ to $t = 40$ does not improve the reduction while it increases considerably its running time.

b - Noise vector: The second factor that can lead to decryption failure is the added noise vector $\mathbf{e}$. If $\mathbf{e}$ is small, the noisy point will not be translated far from the lattice point, which makes the decoding easier. Many methods were proposed to define the error vector and improve the security of the original GGH by defeating its attacks. In [7], Micciancio proposed to use a random noise uniformly chosen from an interval $[-A, A]$. In [3], Hooshmand *et al.* suggested to choose a Gaussian noise $\mathcal{N}(0, \sigma^2)$, where

Table 2.2: Average value of the norm of g_i and the orthogonality defect of G_Λ for lattice dimension $n \approx 1000$.

C_0	p	n	k	$\|g_i\|$	OD(G_Λ)
$C_0[3,2]$	11	999	333	39.9	$O(10^{531})$
$C_0[3,2]$	17	999	333	63.2	$O(10^{598})$
$C_0[3,2]$	29	999	333	109.8	$O(10^{678})$
$C_0[8,6]$	17	1000	500	116.8	$O(10^{1034})$
$C_0[8,6]$	53	1000	500	365.3	$O(10^{1281})$
$C_0[16,12]$	17	992	496	145, 6	$O(10^{1073})$

the variance σ^2 is bounded by Poltyrev limit $\sigma^2 < 1/(2\pi e)$. In [100], a larger noise vector was generated by selecting its elements from two different sets as described in (2.7).

GLD lattices were first proposed for channel coding and their performance was studied for a Gaussian channel with noise variance close to the Poltyrev limit. However, this does not prevent us from using these codes with any other noise distribution. In our GLD lattice-based scheme, we consider a uniform noise in $[-A, A]$. As mentioned in Section 2.3.2.1, A should not exceed the maximal noise value for which the iterative decoder succeeds. At the same time, it should be large enough to prevent the attacker from decrypting the message.

The maximal noise values A with the corresponding elementary codes C_0 were identified in Table 2.1. Considering these values, we have run both the round-off and nearest plane decoding attacks, in dimension $n \approx 1000$, in an attempt to decrypt ciphertexts obtained as $\boldsymbol{c} = \boldsymbol{m}G_\Lambda + \boldsymbol{e}$ for $\boldsymbol{e} \in [-A, A]$. Before running these decoding algorithms, we have reduced G_Λ using LLL and BKZ with different block sizes. For each code C_0 of Table 2.1 and its corresponding A, the decryption attempt fails regardless of the attack scheme and reduction method. We can therefore conclude that these A values are sufficient to ensure the security of our scheme against previous decoding attacks in dimension $n \approx 1000$.

2.4.3 Dual code attacks

Dual code attacks are dangerous attacks applied on code-based cryptography systems using low density parity-check (LDPC) codes. These attacks exploit the low density property of the LDPC parity-check matrix H that corresponds to the generator matrix of the dual code. The main idea is to search for low weight codewords belonging to the dual code and recover H. One solution to avoid such attacks consists in increasing the density of the parity check-matrices. In this case, the resulting codes are called moderate

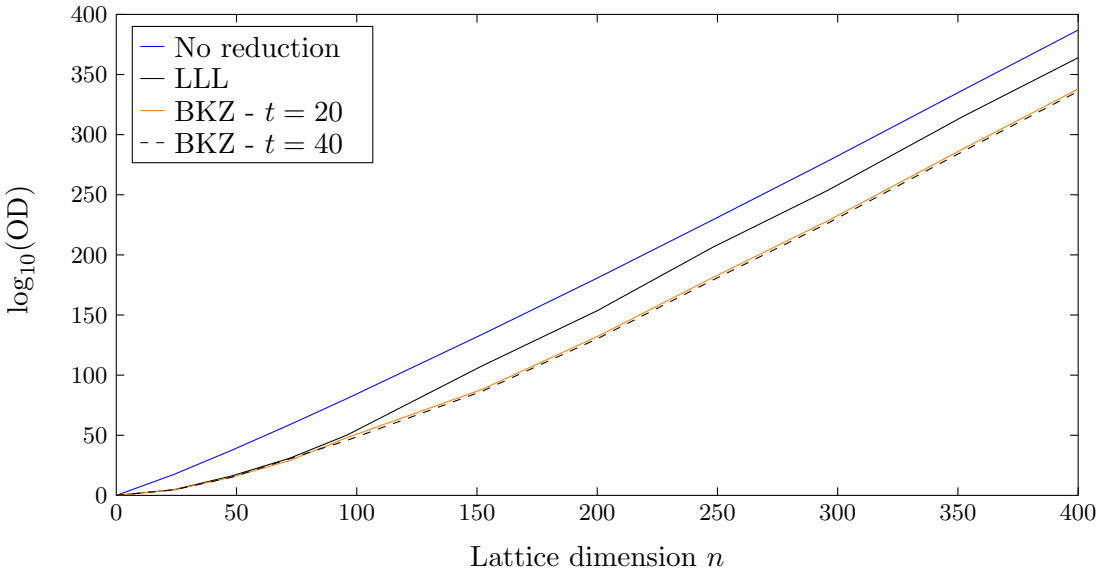

Figure 2.2: Public key orthogonality defect w/wo reduction for $C_0(8, 6)_{17}$.

density parity-check (MDPC) codes [105].

Similarly, the robustness of the GLD lattice-based scheme against these attacks depends on the density of the lattice. This parameter can be managed by the selection of C_0. To improve the security of our scheme, the density of the considered lattices should be increased by taking a code C_0 of higher length n_0 and working in a larger finite field $\mathbb{F}_p$. At the same time, this density should not exceed a given value which enables successful iterative decoding. The codes $C_0[8, 6]_{17}$ and $C_0[16, 12]_{17}$ satisfy this double requirement.

2.5 Complexity analysis

For the complexity analysis of our scheme, we will examine three key features, namely the public key size, the public key generation and the decryption and compare them to previous GGH schemes.

2.5.1 Public key size

Recall that the GLD lattice generator matrix has the following form

$$G_\Lambda = \begin{bmatrix} pI & 0 \\ B & I \end{bmatrix}.$$

We can notice that only part B of G_Λ needs to be stored. It is a $k \times (n-k)$ matrix with elements smaller than p by construction. Hence, the space needed to store this matrix is

$$k \times (n-k) \times \log_2(p) \text{ bits.} \tag{2.24}$$

Hence, for a fixed lattice dimension, the key size depends on the elementary code's parameters and the finite field over which it is defined. In order to analyze the GLD public key complexity, we consider some examples of GLD lattices Λ of dimension $n \approx 1000$ and we compute their key size.

As a first example, let Λ be generated from the elementary code C_0 of length $n_0 = 3$ and dimension $k_0 = 2$ defined over the field $\mathbb{F}_{11}$. This elementary lattice is given by

$$\Lambda_0 = [3, 2]_{11} + 11\mathbb{Z}^3. \tag{2.25}$$

Then, the generated GLD lattice Λ is of dimension $n = n_0 L = 3L$. The matrix B in this case has $k = 333$ rows and $n - k = 666$ columns. Thus, the necessary space to store B is $333 \times 666 \times \log_2(11) = 93.7$ KBs.

In Table 2.3, we present the key size for other examples of elementary codes.

Table 2.3: Public key size for lattice dimension $n \approx 1000$.

C_0	p	n	Key size (KBs)
$C_0[3, 2]$	17	999	110.66
$C_0[3, 2]$	29	999	131.52
$C_0[8, 6]$	17	1000	124.74
$C_0[8, 6]$	53	1000	174.8
$C_0[16, 12]$	17	992	122.75

We can see that for the considered lattice dimension, the key size is always in the order of 100 KBs. Even for very large p values, the key size does not attain 200 KBs.

For the previously described GGH based schemes, upper bounds were derived on the key size of the original GGH and Micciancio cryptosystems [7]. Indeed, the size of the GGH public key is $O(n^2 \log_2(\det(\Lambda)))$ bits since the matrix has n^2 elements and they are bounded by the determinant of the lattice Λ. For the choice of GGH's private basis $R = \sqrt{n}I + Q$, described previously in Section 2.1, the determinant can be estimated by applying Hadamard inequality as

$$\det(\Lambda) \leq 2^{O(n \log_2(n))} \text{ bits,} \tag{2.26}$$

thus, the size of the public key is upper bounded by

$$O(n^3 \log_2(n)) \text{ bits.} \tag{2.27}$$

In Micciancio's cryptosystem, the public basis B is in HNF form. So, each column can be represented using $\log_2(\det(\Lambda))$ bits. Hence, the public key has size of

$$O(n \log_2(\det(\Lambda))) = O(n^2 \log_2(n)) \text{ bits}, \tag{2.28}$$

since the same private key is used as in the GGH system.

In [3], the size of the public key is claimed to be $O(n^2)$. However, this estimation is not justified in their paper. In addition, LDLCs have real generator matrices for the private and public keys. Thus, their public key size depends on the precision used to represent the real elements and should be defined accordingly.

The estimations of the public key sizes for the GGH cryptosystem and its improvements are summarized in Table 2.4.

Table 2.4: Public key size estimate for GGH-based cryptosystems.

Cryptosystem	Public key size (bits)
GGH	$O(n^3 \log_2(n))$
Micciancio	$O(n^2 \log_2(n))$
LDLC	$O(n^2)$
GLD	$k \times (n-k) \times \log_2(p)$

Micciancio's experimental results showed that the public key size in dimension $n = 400$ is 2.3 MBs for GGH and 140 KBs for his system [7]. In [8], Ludvig also computed the public key size of Micciancio's scheme and found that its size in dimension $n = 800$ is 1 MB. In our scheme, the key can be represented in dimension $n \approx 1000$ by approximately 100 KBs. Therefore, we can clearly state that the GLD lattice-based cryptosystem reduces at least by a factor of 10 the size of the public key.

2.5.2 Key generation

In the proposed GLD lattice-based cryptosystem, the public key is the HNF generator matrix of the GLD lattice. In order to compute this matrix, we apply the Gaussian elimination algorithm to represent the GLD code's parity-check matrix in its systematic form $H_{C,\text{syst}}$. Therefore, the key generation complexity is mainly due to the Gaussian elimination algorithm. This time complexity is in the order of $O(nm^{\omega-1})$ for an $m \times n$ matrix with $m \leq n$ and $\omega < 2.38$ the exponent of matrix multiplication [106]. In our case, this reduces to $O(n \times (n-k)^{\omega-1}) < O(n^3)$.

In Micciancio's system [7], the main problem is the complexity of the HNF algorithm that generates the public key. In fact, there exist different algorithms for generating HNF matrices. The most basic algorithm for generating HNF matrices perfoms a polynomial

number of arithmetic operations but has exponential space complexity due to the exponential increase of its matrix elements during the execution. Many improvements were proposed aiming to reduce the space complexity but resulted in increasing the running time. The most space efficient algorithm was designed in [107] with running time $O(n^5 \log_2(M))$ where M is the bound on the matrix entries. Therefore, it can be concluded that the public key generation in our case is at least $O(n^2)$ times faster than Micciancio's system.

Moreover, in [3], the LDLC-based cryptosystem used rational lattices. Thus, before proceeding to HNF, the rational matrix should be transformed into an integer matrix by multiplying its elements by the common denominator. This process increases the running time needed to generate the public key. Hence, for the LDLC-based proposition, generating public key increases the complexity compared to Micciancio's system, and our proposed scheme.

In order to compare the key generation running time of our GLD lattice-based scheme with those of the GGH and Micciancio systems, we experimentally measured the duration of these operations on an Intel i5 3320M (2.6 GHz) platform. The results are presented in Figure 2.3 for dimensions smaller than 1000. For $n \approx 1000$, Micciancio's key generation procedure lasts more than 5 hours and for the GGH system, it is around 10 minutes. On the contrary, our GLD scheme allows to generate the public key in the order of seconds as we can see in Table 2.5 for different elementary GLD codes.

Table 2.5: Public key generation running time of the GLD cryptosystem for lattice dimension $n \approx 1000$.

C_0	p	n	Running time (s)
$C_0[3,2]$	11	999	1.07
$C_0[3,2]$	17	999	1.17
$C_0[3,2]$	29	999	1.25
$C_0[8,6]$	17	1000	7.95
$C_0[8,6]$	53	1000	8.99
$C_0[16,12]$	17	992	13.95

2.5.3 Decryption

As mentioned in [108], GLD lattices are suitable for iterative decoding. In fact, the parity-check matrices H_C of the GLD code can be associated with a Tanner graph with n variable nodes and LJ check nodes. A variable node $x_j, j = 1, \ldots, n$, represents a lattice coordinate and a check node represents one copy of the elementary code C_0 or equivalently the elementary lattice Λ_0. An edge connects a variable node x_j and a check

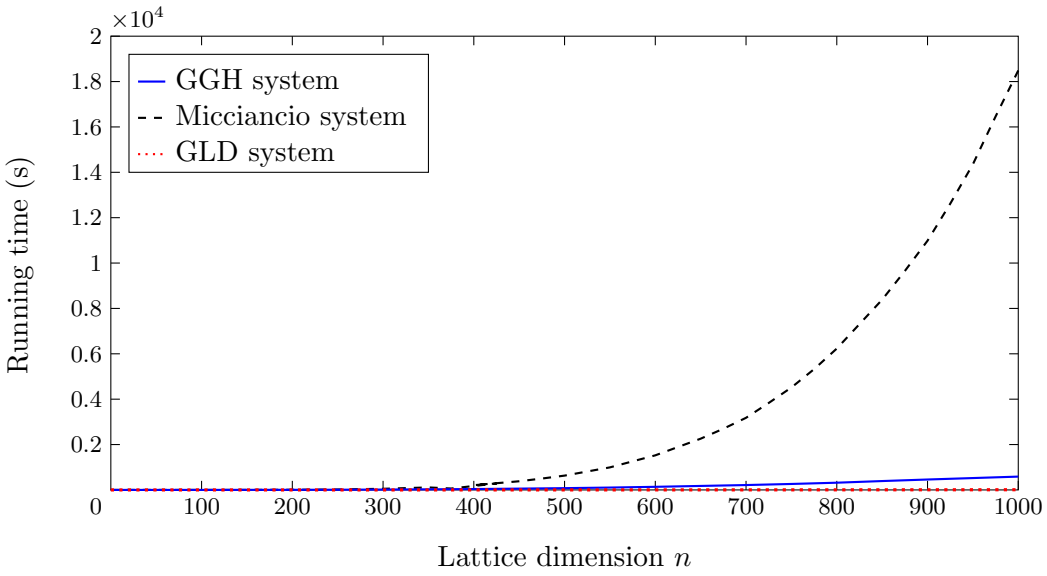

Figure 2.3: Key generation running time for GGH, Micciancio and GLD cryptosystem.

node C_0 if the corresponding position in the parity check matrix H_C has a non zero element. Since in our GLD construction all variable nodes have degree $J = 2$, the graph can be converted into a simple generalized Tanner graph with L check nodes on the right and L check nodes on the left each with degree n_0, as represented in Figure 2.4. The total number of edges $n = n_0 L$ represents the lattice dimension thus, one lattice coordinate is assigned to one graph edge.

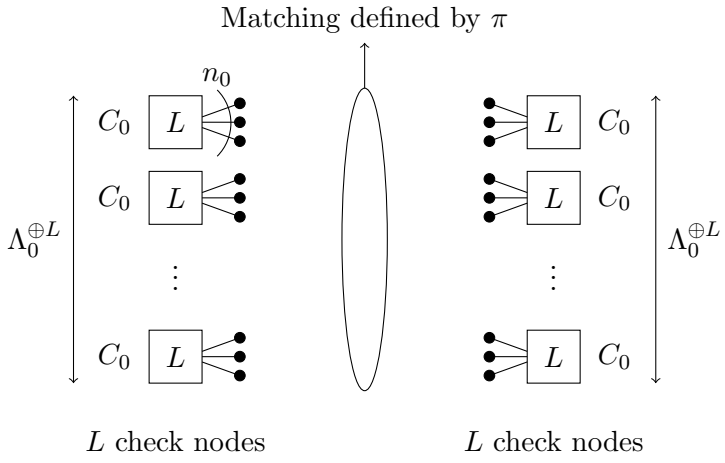

Figure 2.4: Generalized Tanner graph of GLD lattices.

Iterative decoding of GLD lattices is done via message passing along edges of the

generalized Tanner graph. Messages are computed locally by a check node using a soft-input soft-output decoder. This soft decoding is made via the forward-backward algorithm on the syndrome trellis that has $p^{(n_0-k_0+1)}$ transitions where $(n_0 - k_0 + 1)$ is small [109]. For instance, for the codes C_0 in Table 2.2, its maximal value is 5. Then, these messages are sent to the n_0 neighboring check nodes. The complexity of the iterative message passing is linear in the lattice dimension n and is given by $O(n \cdot t \cdot p^{(n_0-k_0+1)})$ where t is the number of iterations.

The LDLCs are also suitable for iterative decoding that is linear in the dimension n but the constant term multiplying n in $O(\cdot)$ is higher [110]. In fact, in this decoder, the messages computed and passed by the check nodes are continuous Gaussian probability distribution functions (pdfs). These real functions are sampled and quantized into discrete vectors which increases the complexity of the decoding. In addition, the number of pdfs grows exponentially with the number of iterations. In order to limit the Gaussians number to M values, other algorithms were proposed in [111, 112]. However, these algorithms add a new complexity related to the Gaussian mixture reduction procedure applied at each iteration. Therefore, this decoding procedure results in high computational complexity and large storage requirements. Choosing a low value of M can reduce the complexity but at the expense of significant performance degradation. Unlike LDLC decoding, GLD iterative decoding exchanges a fixed number p of discrete messages regardless of the number of iterations. Therefore, we can conclude that GLD iterative decoding is less complex than LDLC decoding.

For Micciancio's cryptosystem, Babai's nearest plane algorithm is applied to compute the closest lattice vector to the noisy ciphertext. This sub-optimal algorithm should be repeated many times to return the closest lattice point. In order to decrypt correctly without running the algorithm for a long time, the precision used to represent real numbers should be high but this induces large storage requirements. It was shown by simulations that this decryption procedure is very slow in high dimensions [8]. Conversely, GLD iterative decoding provides a performance close to the optimal decoding while keeping the complexity at a very low level.

Table 2.6: Decryption time of the GLD cryptosystem for lattice dimension $n \approx 1000$.

C_0	p	n	Decryption time
$C_0[3, 2]$	11	999	17 ms
$C_0[3, 2]$	17	999	0.13 s
$C_0[3, 2]$	29	999	0.4 s
$C_0[8, 6]$	17	1000	2.3 s
$C_0[8, 6]$	53	1000	53 s

By conducting another experiment on the time needed to decrypt one ciphertext, we

have observed that Micciancio scheme requires 2 minutes for $n \approx 1000$. For the GLD system, this time varies depending on C_0 as shown in Table 2.6.

We can deduce that, in addition to offering better performance, the GLD decryption algorithm is also faster than Micciancio's decryption algorithm.

2.6 Conclusion

We have proposed a new GGH cryptosystem based on GLD lattices [5]. In this scheme, the private key is the parity-check matrix H_C of the GLD code and the public key is the lattice generator matrix G_Λ which can be obtained by construction A and has a Hermite normal form. We have shown through analysis and experimental results that our cryptosystem reduces significantly the complexity compared to previous GGH schemes while guarantying the same level of security.

Indeed, a large reduction in the key size is induced by the HNF form of the public key G_Λ. The public key in dimension $n \approx 1000$ is represented using 100 KBs approximately, which is one order of magnitude smaller than the public keys of the existing GGH schemes. In addition, the public key is in HNF form by design; hence, no HNF algorithms are applied, which reduces the complexity of the key generation phase making it 1300 times faster than Micciancio and LDLC systems. Finally, the iterative decoding of the GLD lattices offers performance close to the optimal decoding algorithms while operating at low complexity.

Chapter 3

Individual Secrecy in Caching Scenario

In this chapter, we address security in cache-aided communication systems. We consider a wiretap erasure broadcast channel with one transmitter, $K \geq 2$ receivers and one eavesdropper. We partition the set of receivers into K_w weak receivers and K_s strong receivers, and assume that only the weak receivers have equal cache memories of size $\mathcal{M}$. In this communication scenario, the eavesdropper can intercept the transmitted messages. We require that this transmission be secured under an *individual secrecy constraint*, i.e. the eavesdropper cannot learn any information about any of the messages *individually*. However, it is allowed to learn, for example, the XOR of two messages.

We propose a new secure coding scheme and a new information-theoretic converse for the wiretap erasure BC with cache memory at only the weaker receivers. Our secure coding scheme extends the piggyback coding in [97, 9] to a wiretap scenario. The so obtained *secure piggyback coding* is a *joint cache-channel coding scheme* where the design of the encoder and decoders simultaneously exploits the cache content and the channel statistics. Most previous works [10, 89, 90] assume a separate cache-channel architecture (both under secrecy constraints and in the standard model), and focus only on the design of the cache encoder and decoders while assuming that the BC is a noise-free pipe from the transmitter to all receivers. This approach was shown to be highly suboptimal when there is no secrecy constraint [97, 9]; the same is proved here in the presence of such a constraint.

For more clarity, we present first our results for the two-user scenario with one weak receiver and one strong receiver. To study the effect of the secrecy constraint on the capacity-memory tradeoff, we compare our lower and upper bounds with the bounds obtained using also a joint cache-channel coding scheme but in the standard scenario without any secrecy constraint [9]. We also present lower and upper bounds on the secure

capacity-memory tradeoff when both receivers have cache memories of equal size. We study the effect of the cache memory assignment on the secure capacity memory-tradeoff. Moreover, we compare our joint cache-channel coding with the separate cache-channel scheme. Finally, we extend our results to the scenario with K receivers.

3.1 Problem definition

3.1.1 Channel model

We consider a wiretap broadcast channel with a single transmitter, K receivers and one eavesdropper, as shown in Figure 3.1. We model this channel by a memoryless packet-erasure BC with input alphabet

$$\mathcal{X} := \{0,1\}^F \tag{3.1}$$

and the same output alphabet at all receivers and the eavesdropper

$$\mathcal{Y} := \mathcal{X} \cup \Delta. \tag{3.2}$$

Here, F is a fixed positive integer, and an input symbol $x \in \mathcal{X}$ is an F-bit packet. The output erasure symbol Δ indicates the loss of a packet at the receiver. Hence, each receiver $k \in \mathcal{K} := \{1,\dots,K\}$ observes $y_k = \Delta$ with a probability $\delta_k \geq 0$, and it observes $y_k = x$ with probability $1-\delta_k$. The marginal transition laws of this BC channel are described by

$$\mathbb{P}[Y_k = y_k | X = x] = \begin{cases} 1-\delta_k & \text{if } \ y_k = x \\ \delta_k & \text{if } \ y_k = \Delta \\ 0 & \text{otherwise} \end{cases}, \quad \forall\ k. \tag{3.3}$$

The K receivers are partitioned into two sets. The first set

$$\mathcal{K}_w := \{1,\dots,K_w\} \tag{3.4}$$

is formed by K_w weak receivers which have bad channels. The second set

$$\mathcal{K}_s := \{K_w+1,\dots,K\} \tag{3.5}$$

is formed by $K_s = K - K_w$ strong receivers which have good channels.

We assume that the erasure probabilities of the receivers' channels and the eavesdropper are fixed to

$$\delta_k = \begin{cases} \delta_w & \text{if } \ k \in \mathcal{K}_w \\ \delta_s & \text{if } \ k \in \mathcal{K}_s \\ \delta_z & \text{if } \ k = \text{Eavesdropper}, \end{cases} \tag{3.6}$$

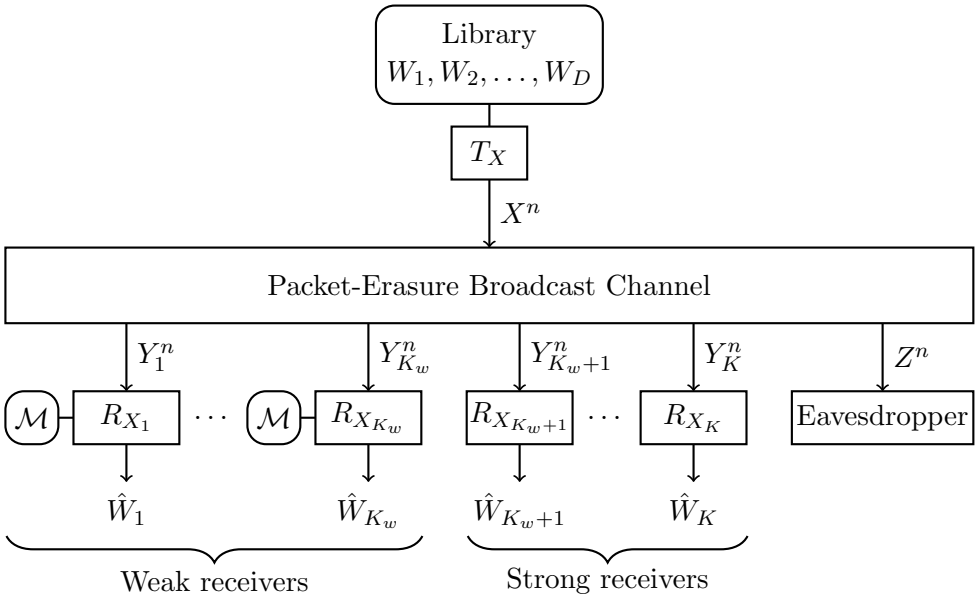

Figure 3.1: Packet-erasure BC with K legitimate receivers and an eavesdropper. The K_w weaker receivers have cache memories of size $\mathcal{M}$.

and verify

$$0 \leq \delta_s \leq \delta_w \leq \delta_z \leq 1. \tag{3.7}$$

Thus, the eavesdropper has statistically the worst channel and the weak receivers have worse channels than the strong ones.

Moreover, we consider that every weak receiver has access to a local cache memory of size $n\mathcal{M}$ bits, while the stronger receivers have no cache memory.

3.1.2 Message library and receiver demands

The transmitter can access a library of $D > K$ independent messages

$$W_1, \ldots, W_D \tag{3.8}$$

of rate $R_s \geq 0$ each. Let

$$\mathcal{D} := \{1, \ldots, D\}. \tag{3.9}$$

For $d \in \mathcal{D}$, every message W_d is uniformly distributed over the set $\left\{1, \ldots, \lfloor 2^{nR_s} \rfloor\right\}$, where n is the transmission blocklength.

Every receiver $k \in \mathcal{K} := \{1, \ldots, K\}$ demands exactly one message W_{d_k} from the library. We denote the demand of receiver k by $d_k \in \mathcal{D}$, and the *demand vector* of all receivers by

$$\mathbf{d} := \{d_1, \ldots, d_K\} \in \mathcal{D}^K. \tag{3.10}$$

The communication takes place in two phases: the caching phase where the transmitter sends caching information to be stored in weak receivers' cache memories and the delivery phase where the demanded messages W_{d_k}, for $k \in \mathcal{K}$, are conveyed to the receivers.

3.1.3 Caching phase

During the caching phase, the demand vector $\mathbf{d}$ is unknown to the transmitter and the receivers. Thus, the cache content V_i of every weak receiver $i \in \mathcal{K}_w$ will be a function of the entire library:

$$V_i := g_i(W_1, \ldots, W_D), \qquad i \in \mathcal{K}_w \tag{3.11}$$

for some caching function

$$g_i : \left\{1, \ldots, \lfloor 2^{nR_s} \rfloor\right\}^D \to \mathcal{V} \tag{3.12}$$

and cache memory alphabet

$$\mathcal{V} := \left\{1, \ldots, \lfloor 2^{n\mathcal{M}} \rfloor\right\}. \tag{3.13}$$

Since the caching phase occurs during low-network traffic periods, we assume that the transmission of the cache content is done via an error-free and erasure-free link. Thus, each weak receiver $i \in \mathcal{K}_w$ stores V_i in its cache memory.

3.1.4 Delivery phase

Prior to the delivery phase, the demand vector $\mathbf{d}$ is learned by the transmitter and the legitimate receivers. We can assume that the communication of the demand vector requires zero communication rate since it takes only $K \cdot \lceil \log(D) \rceil$ bits to describe $\mathbf{d}$.

Based on the demand vector, the transmitter sends

$$X^n := f_{\mathbf{d}}(W_1, \ldots, W_D), \tag{3.14}$$

for some function

$$f_{\mathbf{d}} : \left\{1, \ldots, \lfloor 2^{nR_s} \rfloor\right\}^D \to \mathcal{X}^n. \tag{3.15}$$

All the receivers attempt to decode their demanded messages based on the observed outputs Y_k^n, $k \in \mathcal{K}$. Every weak receiver $i \in \mathcal{K}_w$ uses its observed vector Y_i^n and its

cache content V_i to decode

$$\hat{W}_i := \varphi_{i,\mathbf{d}}(Y_i^n, V_i), \qquad i \in \mathcal{K}_w \tag{3.16}$$

for some function

$$\varphi_{i,\mathbf{d}} : \mathcal{Y}^n \times \mathcal{V} \to \big\{1, \dots, \lfloor 2^{nR_s} \rfloor\big\}. \tag{3.17}$$

Every strong receiver $j \in \mathcal{K}_s$ uses its observed vector Y_j^n to decode

$$\hat{W}_j := \varphi_{j,\mathbf{d}}(Y_j^n), \qquad j \in \mathcal{K}_s \tag{3.18}$$

for some function

$$\varphi_{j,\mathbf{d}} : \mathcal{Y}^n \to \big\{1, \dots, \lfloor 2^{nR_s} \rfloor\big\}. \tag{3.19}$$

3.1.5 Secure capacity-memory tradeoff

A decoding error occurs whenever $\hat{W}_k \neq W_{d_k}$, for $k \in \mathcal{K}$. We consider the worst-case probability of error over all feasible demand vectors

$$\mathrm{P}_e^{\mathrm{Worst}} := \max_{\mathbf{d} \in \mathcal{D}^K} \mathrm{P}\Big[\bigcup_{k=1}^{K} \big\{\hat{W}_k \neq W_{d_k}\big\}\Big]. \tag{3.20}$$

We aim to secure the communication over the BC channel from the eavesdropper and we consider in this chapter the individual secrecy constraints defined as follows:

Definition 3.1. The communication is considered secure under an *individual secrecy constraint* if the eavesdropper's channel outputs Z^n during the delivery phase provide no information about any of the demanded messages individually:

$$\begin{cases} \lim\limits_{n\to\infty} \dfrac{1}{n} I(W_{d_1}; Z^n) & < \quad \epsilon, \\ \lim\limits_{n\to\infty} \dfrac{1}{n} I(W_{d_2}; Z^n) & < \quad \epsilon, \\ \qquad\qquad \vdots & \\ \lim\limits_{n\to\infty} \dfrac{1}{n} I(W_{d_k}; Z^n) & < \quad \epsilon. \end{cases} \tag{3.21}$$

Definition 3.2. A rate-memory pair $(R_s, \mathcal{M})$ is *securely achievable* if for every $\epsilon > 0$ and sufficiently large blocklength n, there exist caching, encoding, and decoding functions as in (3.12), (3.15), (3.17) and (3.19) so that

$$\mathrm{P}_e^{\mathrm{Worst}} \leq \epsilon, \tag{3.22}$$

and

$$\frac{1}{n} I(W_{d_k}; Z^n) < \epsilon, \quad \forall k \in \mathcal{K}. \tag{3.23}$$

Therefore, we define the secure capacity-memory tradeoff as follows.

Definition 3.3. For a cache memory size $\mathcal{M}$, the *secure capacity-memory tradeoff* $C_s(\mathcal{M})$ is the supremum of all rates R_s so that the pair $(R_s, \mathcal{M})$ is securely achievable:

$$C_s(\mathcal{M}) := \sup\big\{R_s\colon \quad (R_s, \mathcal{M}) \text{ securely achievable}\big\}. \tag{3.24}$$

This secure capacity-memory tradeoff $C_s(\mathcal{M})$ is unknown even for the case when there is no cache, i.e. when $\mathcal{M} = 0$. We delimit $C_s(\mathcal{M})$ by computing lower and upper bounds on its values.

3.2 Secure joint cache-channel coding scheme under one-sided cache assignment for the two-user scenario

In this section, we consider the wiretap BC channel with only one weak receiver and one strong receiver. The weak receiver is provided with a cache memory of size $\mathcal{M}$. The scenario is illustrated in Figure 3.2. We propose a secure joint cache-channel coding scheme and compute the lower bound on the secure capacity-memory tradeoff $\mathcal{C}_s(\mathcal{M})$ for the two-receiver scenario under one-sided cache assignment. During the delivery phase, our joint cache-channel coding is based on *piggyback coding* [9]. We will see that this coding increases the transmission rate by piggybacking parts of the strong receiver message on the message intended for the weak receiver.

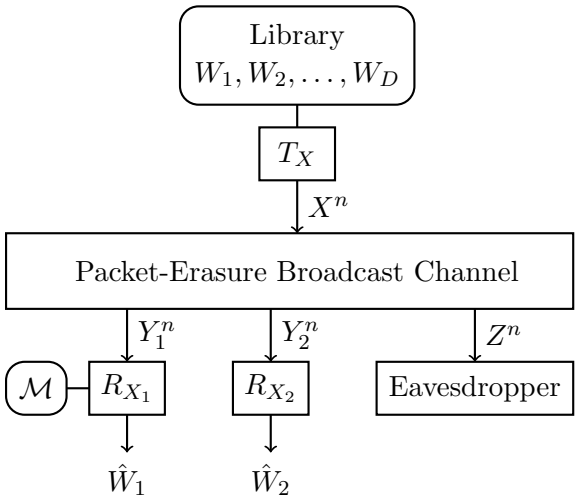

Figure 3.2: Packet-erasure BC with two legitimate receivers and an eavesdropper. Receiver 1 has cache memory of size $\mathcal{M}$.

The lower bound on the secure capacity-memory tradeoff $\mathcal{C}_s(\mathcal{M})$ is stated in the following theorem:

Theorem 3.1 (Lower Bound on $C_s(\mathcal{M})$)**.** *A rate-memory pair $(R_s, \mathcal{M})$ is securely achievable over the two-user wiretap erasure BC with cache memory $\mathcal{M}$ only at the weak receiver using a joint cache-channel coding scheme, if it satisfies the following six conditions:*

$$R_s \leq (\delta_z - \delta_2)F, \tag{3.25a}$$

$$R_s \leq \frac{(1-\delta_2)(\delta_z-\delta_2)}{1+\delta_z-2\delta_2}F + \frac{1-\delta_2}{1+\delta_z-2\delta_2}\frac{\mathcal{M}}{D}, \tag{3.25b}$$

$$R_s \leq \frac{(1-\delta_1)(\delta_z-\delta_2)}{1-\delta_1+\delta_z-\delta_2}F + \frac{\mathcal{M}}{D}, \tag{3.25c}$$

$$R_s \leq \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F + \frac{D(\delta_z-\delta_2)+(\delta_z-\delta_1)}{2\delta_z-\delta_1-\delta_2}\frac{\mathcal{M}}{D}, \tag{3.25d}$$

$$R_s \leq \frac{\delta_z-\delta_2}{2}F + \frac{D}{2}\frac{\mathcal{M}}{D}, \tag{3.25e}$$

$$R_s \leq \frac{D}{D+1}(\delta_z-\delta_2)F + \frac{D}{D+1}\frac{\mathcal{M}}{D}. \tag{3.25f}$$

Thus, any R_s satisfying (3.25) *forms a lower bound on $C_s(\mathcal{M})$.*

Proof. We derive the proof of this theorem in the next subsections by describing the joint cache-channel coding that can achieve this lower bound. □

Remark 3.1. The scenario which is of most interest to us is when receiver 2 is sufficiently stronger than receiver 1. In this case, constraints (3.25e) and (3.25f) are not active and the lower bound is defined only by (3.25a)–(3.25d).

3.2.1 Message splitting

For each $d \in \mathcal{D}$, split the message W_d into two sub-messages, such that

$$W_d = \left[W_d^{(0)}, W_d^{(1)}\right], \tag{3.26}$$

with rates $R^{(0)}$ and $R^{(1)}$, respectively. The total rate of W_d is

$$R_s = R^{(0)} + R^{(1)}. \tag{3.27}$$

If $R^{(0)} > (D-2)R^{(1)}$, divide $W_d^{(0)}$ into two further sub-messages, such that

$$W_d^{(0)} = \left[W_d^{(0,1)}, W_d^{(0,2)}\right], \tag{3.28}$$

with rates $(D-2)R^{(1)}$ and $R^{(0)} - (D-2)R^{(1)}$, respectively. Otherwise, $W_d^{(0,1)} = W_d^{(0)}$ has rate $R^{(0)}$ and $W_d^{(0,2)}$ has zero rate.

3.2.2 Codebook generation

Generate two codebooks that will be used later to securely encode the messages before their transmission. The first one is a piggyback codebook and the second one is a regular wiretap codebook.

Generate the first codebook $\mathcal{C}_1$ with

$$\Gamma_1 := \lfloor 2^{nR^{(0)}} \rfloor \cdot \lfloor 2^{nR^{(1)}} \rfloor \cdot \lfloor 2^{nR'} \rfloor \tag{3.29}$$

codewords of length αn,

$$\mathcal{C}_1 := \left\{ X_1^{(\alpha n)}(l_1) \right\}_{l_1=1}^{\Gamma_1}. \tag{3.30}$$

by drawing each entry of each codeword at random according to a Bernoulli-1/2 distribution independently of all other entries.

The codebook $\mathcal{C}_1$ is partitioned into $\lfloor 2^{nR^{(0)}} \rfloor \cdot \lfloor 2^{nR^{(1)}} \rfloor$ subcodebooks (bins) each with $\lfloor 2^{nR'} \rfloor$ codewords. The subcodebooks are arranged into an array with $\lfloor 2^{nR^{(0)}} \rfloor$ rows and $\lfloor 2^{nR^{(1)}} \rfloor$ columns, as depicted in Figure 3.3 where each square represents a subcodebook. The subcodebook in row $\tilde{w}_1$ and column $\tilde{w}_2$ is denoted $\mathcal{C}_1(\tilde{w}_1, \tilde{w}_2)$.

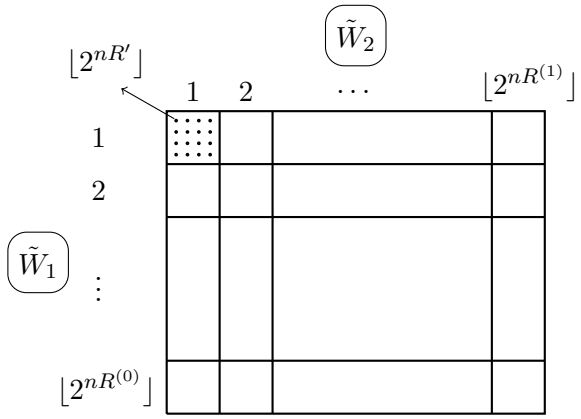

Figure 3.3: *Secure piggyback codebook* $\mathcal{C}_1$ where each dot symbolizes a codeword. Subcodebooks (bins) $\mathcal{C}_1(\tilde{w}_1, \tilde{w}_2)$ are depicted by the squares, each containing $\lfloor 2^{nR'} \rfloor$ codewords.

Then, generate the second codebook $\mathcal{C}_2$ with

$$\Gamma_2 := \lfloor 2^{nR^{(1)}} \rfloor \cdot \lfloor 2^{nR''} \rfloor \tag{3.31}$$

codewords of length $(1-\alpha)n$,

$$\mathcal{C}_2 := \left\{ X_2^{((1-\alpha)n)}(l_2) \right\}_{l_2=1}^{\Gamma_2}, \tag{3.32}$$

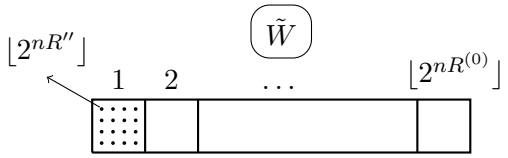

Figure 3.4: Wiretap codebook $\mathcal{C}_2$ where each dot symbolizes a codeword. Subcodebooks $\mathcal{C}_2(\tilde{w})$ are depicted by the squares, each containing $\lfloor 2^{nR''} \rfloor$ codewords.

$\mathcal{C}_2$ is also generated by drawing each entry of each codeword at random according to a Bernoulli-1/2 distribution independently of all the other entries.

We assume that both codebooks $\mathcal{C}_1$ and $\mathcal{C}_2$ are revealed to the transmitter, both receivers and the eavesdropper.

3.2.3 Caching phase

For each $d \in \mathcal{D}$, store $W_d^{(1)}$ in the cache memory of receiver 1. Thus, the cache content V_1 of receiver 1 is

$$V_1 = \left\{ W_1^{(1)}, W_2^{(1)}, \ldots, W_D^{(1)} \right\}. \tag{3.33}$$

This is feasible whenever the cache memory size of receiver 1 is larger than the size of the D sub-messages $W_d^{(1)}$:

$$R^{(1)} \leq \frac{\mathcal{M}}{D}. \tag{3.34}$$

3.2.4 Delivery phase

The delivery phase is divided into two periods of lengths αn and $(1-\alpha)n$, for some $\alpha \in [0,1]$.

During the first period, the transmitter conveys message $W_{d_1}^{(0)}$ to receiver 1 and message $W_{d_2}^{(1)}$ to receiver 2.

It randomly chooses a set of ι indices such that

$$\left\{ j_1, j_2, \ldots, j_\iota \right\} \in \left(\mathcal{D} \setminus \{d_1, d_2\} \right), \tag{3.35}$$

Those indices serve to select the messages from receiver 1's cache that will be XORed with the transmitted message $W_{d_1}^{(0)}$. Thus, if $R^{(0)} \leq R^{(1)}$, $\iota = 1$ and only a part of the chosen message $W_{d_{j_\iota}}^{(1)}$ is used in the XOR operation. Otherwise, if $R^{(1)} < R^{(0)} \leq (D-2)R^{(1)}$, then it needs $\iota = \lceil R^{(0)}/R^{(1)} \rceil$.

Now, if the messages in the cache are not enough to cover the whole message $W_{d_1}^{(0)}$, i.e. if $R^{(0)} > (D-2)R^{(1)}$, it XORs only the part $W_{d_1}^{(0,1)}$ of $W_{d_1}^{(0)}$ with the $(D-2)$ messages from the cache memory and it secures the remaining part $W_{d_1}^{(0,2)}$ by the binning. Thus, ι is defined by

$$\iota := \max\left\{1, \min\left\{\left\lceil \frac{R^{(0)}}{R^{(1)}} \right\rceil, (D-2)\right\}\right\}. \tag{3.36}$$

Afterwards, the transmitter forms

$$W_{\text{XOR}} := W_{d_1}^{(0,1)} \oplus \left[W_{j_1}^{(1)}, W_{j_2}^{(1)}, \ldots, W_{j_\iota}^{(1)}\right]. \tag{3.37}$$

It then uses the secure piggyback codebook $\mathcal{C}_1$ in Figure 3.3. Specifically, it picks an index J_1 uniformly at random from $\left[1 : \lfloor 2^{nR'} \rfloor\right]$ and transmits the

$$J_1\text{-th codeword of subcodebook } \mathcal{C}_1\left(\tilde{W}_1, \tilde{W}_2 = W_{d_2}^{(1)}\right) \tag{3.38}$$

where

$$\tilde{W}_1 := \left[W_{\text{XOR}}, W_{d_1}^{(0,2)}\right]. \tag{3.39}$$

During the second period, the transmitter conveys message $W_{d_2}^{(0)}$ to receiver 2 using the wiretap codebook $\mathcal{C}_2$. Specifically, it picks an index J_2 uniformly at random from $\left[1 : \lfloor 2^{nR''} \rfloor\right]$, and transmits the

$$J_2\text{-th codeword of subcodebook } \mathcal{C}_2\left(\tilde{W} = W_{d_2}^{(0)}\right). \tag{3.40}$$

3.2.5 Decoding at receiver 1

Receiver 1 demands the message W_{d_1}. But since $W_{d_1}^{(1)}$ is stored in its cache memory, it only needs to decode message $W_{d_1}^{(0)}$ based on its observed outputs $y_1^{\alpha n}$ in the first phase and its cache memory V_1. To decode $W_{d_1}^{(0)}$, it performs the following steps:

1. It retrieves the message $W_{d_2}^{(1)}$ from its cache memory.
2. It forms the column-subcodebook $\mathcal{C}_1(\tilde{W}_2 = W_{d_2}^{(1)}) \in \mathcal{C}_1$ that contains all the codewords that represent the retrieved message:

$$\mathcal{C}_1(W_{d_2}^{(1)}) := \left\{X_1^{(\alpha n)}(l_1|W_{d_2}^{(1)})\right\}_{l_1 \in \Gamma_1(W_{d_2}^{(1)})}, \tag{3.41}$$

   where $\Gamma_1(W_{d_2}^{(1)})$ is a subset of Γ_1 of cardinality $\lfloor 2^{nR^{(0)}} \rfloor \cdot \lfloor 2^{nR'} \rfloor$. $\mathcal{C}_1(W_{d_2}^{(1)})$ corresponds to the subcodebook in column $W_{d_2}^{(1)}$ of the secure piggyback codebook depicted in Figure 3.3.

3. It finds $\hat{\tilde{w}}_1$ by restricting its attention to this column-subcodebook $\mathcal{C}_1(W_{d_2}^{(1)})$. It looks for a unique $l_1 \in \Gamma_1(W_{d_2}^{(1)})$ so that $x_1^{(\alpha n)}(l_1|W_{d_2}^{(1)})$ is jointly typical with its observed outputs $y_1^{\alpha n}$:

$$\left(x_1^{(\alpha n)}(l_1|W_{d_2}^{(1)}), y_1^{\alpha n}\right) \in \mathcal{T}_\epsilon^{(\alpha n)}\big(p_X \cdot p_{Y_1|X}\big), \tag{3.42}$$

where p_X stands for the Bernoulli-1/2 distribution, $p_{Y_1|X}$ stands for the channel law to receiver 1, and $\mathcal{T}_\epsilon^{(\alpha n)}$ stands for the typical set [11].

If the desired unique index l_1 exists, it finds the bin $\mathcal{C}_1(\hat{\tilde{w}}_1, W_{d_2}^{(1)})$ containing l_1. Then, the value of $\hat{\tilde{w}}_1$ is determined as the row index of the $\mathcal{C}_1(\hat{\tilde{w}}_1, W_{d_2}^{(1)})$.

Otherwise, if the unique index does not exit, receiver 1 declares a decoding error.

4. If $\hat{\tilde{w}}_1$ is found in the previous step, receiver 1 splits it as follows

$$\hat{\tilde{w}}_1 = \left[\hat{w}_{\text{XOR}}, \hat{w}_{d_1}^{(0,2)}\right]. \tag{3.43}$$

Then, it retrieves messages $W_{j_1}^{(1)}, W_{j_2}^{(1)}, \ldots, W_{j_\iota}^{(1)}$ from its cache memory and forms

$$\hat{w}_{d_1}^{(0,1)} = \hat{w}_{\text{XOR}} \oplus \left[W_{j_1}^{(1)}, W_{j_2}^{(1)}, \ldots, W_{j_\iota}^{(1)}\right]. \tag{3.44}$$

It finally retrieves $W_{d_1}^{(1)}$ from its cache memory and declares its decoded message

$$\hat{w}_1 = \left[\hat{w}_{d_1}^{(0,1)}, \hat{w}_{d_1}^{(0,2)}, W_{d_1}^{(1)}\right]. \tag{3.45}$$

3.2.6 Decoding at receiver 2

Receiver 2 does not have any cache memory and hence, it decodes both transmission periods using its observed outputs only. It decodes $W_{d_2}^{(1)}$ based on its outputs $y_2^{\alpha n}$ in the first period, and it decodes $W_{d_2}^{(0)}$ based on its outputs $y_2^{(1-\alpha)n}$ in the second period. It proceeds as follows:

1. To decode $W_{d_2}^{(1)}$, it considers the whole codebook $\mathcal{C}_1$. It looks for a unique $l_1 \in \Gamma_1$ such that $x_1^{(\alpha n)}(l_1)$ is jointly typical with its observed outputs $y_2^{\alpha n}$:

$$\left(x_1^{(\alpha n)}(l_1), y_2^{\alpha n}\right) \in \mathcal{T}_\epsilon^{(\alpha n)}\big(p_X \cdot p_{Y_2|X}\big), \tag{3.46}$$

where p_X stands for the Bernoulli-1/2 distribution and $p_{Y_2|X}$ stands for the channel law to receiver 2.

If the desired unique index l_1 exists, receiver 2 finds the bin $\mathcal{C}_1(\hat{\tilde{w}}_1, \hat{w}_{d_2}^{(1)})$ containing l_1 and it determines the value of $\hat{w}_{d_2}^{(1)}$ as the column index of this bin. Otherwise, receiver 2 declares a decoding error.

2. To decode $W_{d_2}^{(0)}$, it considers the codebook $\mathcal{C}_2$. It looks for a unique $l_2 \in \Gamma_2$ such that $x_2^{((1-\alpha)n)}(l_2)$ is jointly typical with its observed outputs $y_2^{(1-\alpha)n}$:

$$\left(x_2^{((1-\alpha)n)}(l_2), y_2^{(1-\alpha)n}\right) \in \mathcal{T}_\epsilon^{(\alpha n)}\big(p_X \cdot p_{Y_2|X}\big). \tag{3.47}$$

   If the desired unique index l_2 exists, receiver 2 finds the bin $\mathcal{C}_2(\hat{w}_{d_2}^{(0)})$ containing l_2 and it determines the value of $\hat{w}_{d_2}^{(0)}$ as the index of this bin. Otherwise, receiver 2 declares a decoding error.

3. If the decoding of both periods succeeds, receiver 2 declares its decoded message

$$\hat{w}_2 = \left[\hat{w}_{d_2}^{(0)}, \hat{w}_{d_2}^{(1)}\right]. \tag{3.48}$$

3.2.7 Analysis of the error probability

As seen in Section 3.2.5, receiver 1 decodes only the first transmission period and its decoding is restricted to the column-subcodebook $\mathcal{C}_1(W_{d_2}^{(1)})$. Hence, the probability of error at receiver 1 averaged over the codebooks $\mathcal{C}_1, \mathcal{C}_2$, the channel realizations, and the messages,

$$\mathrm{P}_{e,1}^{\mathrm{Worst}} \xrightarrow[n\to\infty]{} 0,$$

if the following condition is satisfied:

$$R^{(0)} + R' \leq \alpha(1-\delta_1)F. \tag{3.49}$$

Receiver 2 decodes both transmission periods without the help of any cache memory. Thus, the probability of error at receiver 2, averaged over the codebooks $\mathcal{C}_1, \mathcal{C}_2$, the channel realizations, and the messages,

$$\mathrm{P}_{e,2}^{\mathrm{Worst}} \xrightarrow[n\to\infty]{} 0,$$

if the following two conditions are satisfied:

$$R_s + R' \leq \alpha(1-\delta_2)F, \tag{3.50a}$$

$$R^{(0)} + R'' \leq (1-\alpha)(1-\delta_2)F. \tag{3.50b}$$

3.2.8 Analysis of the information leakage

In this section, we analyze the individual secrecy of each transmitted message W_{d_1} and W_{d_2}.

Secrecy of receiver 1*'s message* W_{d_1} :

For the secrecy of the message W_{d_1}, we derive the mutual information between W_{d_1} and the received vector Z^n at the eavesdropper knowing the codebooks $\mathcal{C}_1$ and $\mathcal{C}_2$ as follows

$$\begin{aligned} I(W_{d_1}; Z^n|\mathcal{C}_1, \mathcal{C}_2) &= I\big(W_{d_1}^{(0,1)}, W_{d_1}^{(0,2)}, W_{d_1}^{(1)}; Z^n|\mathcal{C}_1, \mathcal{C}_2\big) \\ &\overset{(a)}{=} I\big(W_{d_1,1}^{(0)}, W_{d_1}^{(0,2)}; Z^{\alpha n}|\mathcal{C}_1\big) \\ &= I\big(W_{d_1}^{(0,1)}; Z^{\alpha n}|W_{d_1}^{(0,2)}, \mathcal{C}_1\big) + I\big(W_{d_1}^{(0,2)}; Z^{\alpha n}|\mathcal{C}_1\big), \end{aligned} \tag{3.51}$$

where (a) holds because $W_{d_1}^{(1)}$ is in the cache memory of receiver 1 and it is not sent over the channel. Furthermore,

$$\begin{aligned} I(W_{d_1}^{(0,1)}; Z^{\alpha n}|W_{d_1}^{(0,2)}, \mathcal{C}_1) &\leq I\big(W_{d_1}^{(0,1)}; Z^{\alpha n}, W_{\text{XOR}}, W_{d_1}^{(0,2)}|\mathcal{C}_1\big) \\ &\overset{(a)}{=} I\big(W_{d_1}^{(0,1)}; W_{\text{XOR}}, W_{d_1}^{(0,2)}|\mathcal{C}_1\big) \\ &\overset{(b)}{=} 0, \end{aligned} \tag{3.52}$$

where (a) holds because of the Markov chain $W_{d_1}^{(0,1)} \to (W_{\text{XOR}}, W_{d_1}^{(0,2)}) \to Z^n$; and (b) holds because $W_{d_1}^{(0,1)}$ is independent of the pair $\big(W_{\text{XOR}}, W_{d_1}^{(0,2)}\big)$.

The secrecy of $W_{d_1}^{(0,2)}$ is proved as follows

$$\begin{aligned} I(W_{d_1}^{(0,2)}; Z^{\alpha n}|\mathcal{C}_1) &= H(W_{d_1}^{(0,2)}|\mathcal{C}_1) - H(W_{d_1}^{(0,2)}|Z^{\alpha n}, \mathcal{C}_1) \\ &= n\big[R^{(0)} - (D-2)R^{(1)}\big] - H(W_{d_1}^{(0,2)}, L_1|Z^{\alpha n}, \mathcal{C}_1) + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)}, \mathcal{C}_1) \\ &\overset{(a)}{=} n\big[R^{(0)} - (D-2)R^{(1)}\big] - H(L_1|Z^{\alpha n}, \mathcal{C}_1) + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)}, \mathcal{C}_1) \\ &= n\big[R^{(0)} - (D-2)R^{(1)}\big] - H(L_1|\mathcal{C}_1) + I(L_1; Z^{\alpha n}|\mathcal{C}_1) + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)}, \mathcal{C}_1) \\ &= n\big[R^{(0)} - (D-2)R^{(1)}\big] - n\big[R^{(0)} + R^{(1)} + R'\big] + I(L_1; Z^{\alpha n}|\mathcal{C}_1) \\ &\quad + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)}, \mathcal{C}_1) \\ &\overset{(b)}{=} -n\big[(D-1)R^{(1)} + R'\big] + I(X_1^{\alpha n}, L_1; Z^{\alpha n}|\mathcal{C}_1) + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)}, \mathcal{C}_1) \\ &\overset{(c)}{\leq} -n\big[(D-1)R^{(1)} + R'\big] + I(X_1^{\alpha n}, L_1, \mathcal{C}_1; Z^{\alpha n}) + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)}, \mathcal{C}_1) \\ &\overset{(d)}{\leq} -n\big[(D-1)R^{(1)} + R'\big] + I(X_1^{\alpha n}; Z^{\alpha n}) + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)}, \mathcal{C}_1) \\ &\overset{(e)}{\leq} -n\big[(D-1)R^{(1)} + R'\big] + \alpha n I(X_1; Z) + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)}, \mathcal{C}_1), \\ &= -n\big[(D-1)R^{(1)} + R'\big] + \alpha n(1-\delta_z)F + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)}, \mathcal{C}_1), \end{aligned} \tag{3.53}$$

where (a) holds because the message $W_{d_1}^{(0,2)}$ is a function of the index L_1; (b) holds because $X_1^{\alpha n}$ is a function of L_1 and $\mathcal{C}_1$; (c) holds because

$$I(X_1^{\alpha n}, L_1; Z^{\alpha n}|\mathcal{C}_1) = I(X_1^{\alpha n}, L_1, \mathcal{C}_1; Z^{\alpha n}) - I(\mathcal{C}_1; Z^{\alpha n})$$

and $I(\mathcal{C}_1; Z^{\alpha n}) \geq 0$; (d) holds because of the Markov chain $(L_1, \mathcal{C}_1) \to X_1^{\alpha n} \to Z^{\alpha n}$; and (e) holds because by construction $p(x_1^{\alpha n}, z^{\alpha n}) = \prod_{i=1}^{\alpha n} p_{XZ}(x_{1,i}, z_i)$.

In order to satisfy the individual secrecy constraint

$$\lim_{n\to\infty} \frac{1}{n} I(W_{d_1}^{(0,2)}; Z^{\alpha n}|\mathcal{C}_1) \leq \delta(\epsilon),$$

for some function $\delta(\epsilon) \xrightarrow[n\to\infty]{} 0$, we need

$$\lim_{n\to\infty} \frac{1}{n} H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)}, \mathcal{C}_1) \leq (D-1)R^{(1)} + R' - \alpha(1-\delta_z)F + \delta(\epsilon). \tag{3.54}$$

Lemma 3.1. (3.54) *holds for some function* $\delta(\epsilon) \xrightarrow[n\to\infty]{} 0$ *if*

$$(D-1)R^{(1)} + R' \geq \alpha(1-\delta_z)F. \tag{3.55}$$

Proof. The proof of Lemma 3.1 is given in the appendix. □

Note that the condition in Lemma 3.1 is relevant only in case $R^{(0)} > (D-2)R^{(1)}$.

Secrecy of receiver 2's message W_{d_2} :

Following the same steps, we can prove the secrecy of message W_{d_2}. We can show that

$$\lim_{n\to\infty} \frac{1}{n} I(W_{d_2}^{(0)}; Z^n|\mathcal{C}_1, \mathcal{C}_2) \to 0$$

whenever

$$R^{(0)} + R' \geq \alpha(1-\delta_z)F, \tag{3.56}$$

and

$$\lim_{n\to\infty} \frac{1}{n} I(W_{d_2}^{(1)}; Z^n|\mathcal{C}_1, \mathcal{C}_2) \to 0$$

whenever

$$R'' \geq (1-\alpha)(1-\delta_z)F. \tag{3.57}$$

We can state finally that under constraints (3.34), (3.49), (3.50) and (3.55)–(3.57), when averaged over the random choice of the codebooks $\mathcal{C}_1$ and $\mathcal{C}_2$, the probabilities of error tend to zero and the information leakage constraints are satisfied. There must thus exist at least one choice of $\mathcal{C}_1$ and $\mathcal{C}_2$ with these properties.

3.2.9 Securely achievable rate-memory tuples

We consider all the constraints (3.34), (3.49), (3.50) and (3.55)–(3.57) of our scheme and compute the securely achievable transmission rate R_s as function of the cache memory size $\mathcal{M}$, as follows:

$$R^{(1)} \leq \frac{\mathcal{M}}{D}, \tag{3.58a}$$
$$R^{(0)} + R' \leq \alpha(1-\delta_1)F, \tag{3.58b}$$
$$R_s + R' \leq \alpha(1-\delta_2)F, \tag{3.58c}$$
$$R^{(0)} + R'' \leq (1-\alpha)(1-\delta_2)F, \tag{3.58d}$$
$$R^{(0)} + R' \geq \alpha(1-\delta_z)F, \tag{3.58e}$$
$$(D-1)R^{(1)} + R' > \alpha(1-\delta_z)F, \tag{3.58f}$$
$$R'' \geq (1-\alpha)(1-\delta_z)F. \tag{3.58g}$$

Since $R' \geq 0$ and $R'' \geq 0$, we apply the Fourier-Motzkin elimination to remove R' and R'' and obtain:

$$R^{(1)} \leq \frac{\mathcal{M}}{D}, \tag{3.59a}$$
$$R^{(0)} \leq \alpha(1-\delta_1)F, \tag{3.59b}$$
$$R_s \leq \alpha(1-\delta_2)F, \tag{3.59c}$$
$$1-\delta_z \leq 1-\delta_1, \tag{3.59d}$$
$$R^{(0)} - (D-1)R^{(1)} \leq \alpha(\delta_z-\delta_1)F, \tag{3.59e}$$
$$R^{(1)} \leq \alpha(\delta_z-\delta_2)F, \tag{3.59f}$$
$$R_s - (D-1)R^{(1)} \leq \alpha(\delta_z-\delta_2)F, \tag{3.59g}$$
$$R^{(0)} \leq (1-\alpha)(1-\delta_2)F, \tag{3.59h}$$
$$R^{(0)} \leq (1-\alpha)(\delta_z-\delta_2)F. \tag{3.59i}$$

Notice that the bound in (3.59i) is tighter than that of (3.59h). Hence, we can remove condition (3.59h) without any effect on the result. Moreover, constraint (3.59d) is always valid since by definition $\delta_z \geq \delta_1$. We replace $R^{(0)}$ by $R_s - R^{(1)}$ and remove $R^{(1)}$ using Fourier-Motzkin elimination since $0 \leq R^{(1)} \leq R$. We obtain:

$$R_s \leq \alpha(1-\delta_1)F + \frac{\mathcal{M}}{D}, \tag{3.60a}$$
$$R_s \leq \alpha(1-\delta_2)F, \tag{3.60b}$$
$$R_s \leq \alpha(\delta_z-\delta_1)F + D\frac{\mathcal{M}}{D}, \tag{3.60c}$$

$$R_s \leq \alpha(\delta_z - \delta_2)F + (D-1)\frac{\mathcal{M}}{D}, \tag{3.60d}$$
$$R_s \leq (1-\alpha)(\delta_z - \delta_2)F + \frac{\mathcal{M}}{D}, \tag{3.60e}$$
$$R_s \leq \alpha(1 - \delta_1 + \delta_z - \delta_2)F, \tag{3.60f}$$
$$R_s \leq \alpha\big[D(\delta_z - \delta_2) + (\delta_z - \delta_1)\big]F, \tag{3.60g}$$
$$R_s \leq \alpha D(\delta_z - \delta_2)F, \tag{3.60h}$$
$$R_s \leq (\delta_z - \delta_2)F. \tag{3.60i}$$

Finally, since $0 \leq \alpha \leq 1$, we apply Fourier-Motzkin elimination one last time to remove α and we get:

$$R_s \leq (1-\delta_1)F + \frac{\mathcal{M}}{D}, \tag{3.61a}$$
$$R_s \leq (1-\delta_2)F, \tag{3.61b}$$
$$R_s \leq (\delta_z - \delta_1)F + D\frac{\mathcal{M}}{D}, \tag{3.61c}$$
$$R_s \leq (\delta_z - \delta_2)F + (D-1)\frac{\mathcal{M}}{D}, \tag{3.61d}$$
$$R_s \leq (\delta_z - \delta_2)F + \frac{\mathcal{M}}{D}, \tag{3.61e}$$
$$R_s \leq (1 - \delta_1 + \delta_z - \delta_2)F, \tag{3.61f}$$
$$R_s \leq \big[D(\delta_z - \delta_2) + (\delta_z - \delta_1)\big]F, \tag{3.61g}$$
$$R_s \leq D(\delta_z - \delta_2)F, \tag{3.61h}$$
$$R_s \leq (\delta_z - \delta_2)F, \tag{3.61i}$$
$$R_s \leq \frac{(1-\delta_1)(\delta_z - \delta_2)}{1 - \delta_1 + \delta_z - \delta_2}F + \frac{\mathcal{M}}{D}, \tag{3.61j}$$
$$R_s \leq \frac{(1-\delta_2)(\delta_z - \delta_2)}{1 + \delta_z - 2\delta_2}F + \frac{1-\delta_2}{1 + \delta_z - 2\delta_2}\frac{\mathcal{M}}{D}, \tag{3.61k}$$
$$R_s \leq \frac{(\delta_z - \delta_1)(\delta_z - \delta_2)}{2\delta_z - \delta_1 - \delta_2}F + \frac{D(\delta_z - \delta_2) + (\delta_z - \delta_1)}{2\delta_z - \delta_1 - \delta_2}\frac{\mathcal{M}}{D}, \tag{3.61l}$$
$$R_s \leq \frac{\delta_z - \delta_2}{2}F + \frac{D}{2}\frac{\mathcal{M}}{D}, \tag{3.61m}$$
$$R_s \leq \frac{(\delta_z - \delta_2)(1 - \delta_1 + \delta_z - \delta_2)}{1 - \delta_1 + 2\delta_z - 2\delta_2}F + \frac{1 - \delta_1 + \delta_z - \delta_2}{1 - \delta_1 + 2\delta_z - 2\delta_2}\frac{\mathcal{M}}{D}, \tag{3.61n}$$
$$R_s \leq \frac{(\delta_z - \delta_2)\big[D(\delta_z - \delta_2) + (\delta_z - \delta_1)\big]}{(D+1)(\delta_z - \delta_2) + (\delta_z - \delta_1)}F + \frac{D(\delta_z - \delta_2) + (\delta_z - \delta_1)}{(D+1)(\delta_z - \delta_2) + (\delta_z - \delta_1)}\frac{\mathcal{M}}{D}, \tag{3.61o}$$
$$R_s \leq \frac{D}{D+1}\Big[(\delta_z - \delta_2)F + \frac{\mathcal{M}}{D}\Big]. \tag{3.61p}$$

Notice that (3.61i) is tighter than (3.61b), (3.61d), (3.61e), (3.61f),(3.61g) and (3.61h). Moreover, (3.61p) is tighter than (3.61o); (3.61k) is tighter than (3.61n); (3.61j)

is tighter than (3.61a); and (3.61l) is tighter than (3.61c). We can remove all the loose constraints without affecting the result. We conclude that the rate-memory tuples securely achievable by our joint cache-channel coding scheme satisfy the constraints given in Theorem 3.1.

3.3 Upper bound on the secure capacity-memory tradeoff under one-sided cache assignment for the two-user scenario

In this section, we provide the upper bound on the secure capacity-memory tradeoff $C_s(\mathcal{M})$ for the two-receiver channel under one-sided cache assignment. This upper bound is stated in the following theorem:

Theorem 3.2 (Upper Bound on $C_s(\mathcal{M})$)**.** *The secure capacity-memory tradeoff $C_s(\mathcal{M})$ of the two-user wiretap erasure BC with cache memory $\mathcal{M}$ only at the weaker receiver satisfies the following three conditions:*

$$C_s(\mathcal{M}) \leq (\delta_z - \delta_1)F + \mathcal{M}, \tag{3.62a}$$

$$C_s(\mathcal{M}) \leq (\delta_z - \delta_2)F, \tag{3.62b}$$

$$C_s(\mathcal{M}) \leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F + \frac{\mathcal{M}}{D}. \tag{3.62c}$$

3.3.1 Proof of the upper bound

Constraint (3.62c) follows from [9, Theorem 9] and by ignoring the secrecy constraints (3.21).

Constraint (3.62a) is proved as follows. For each blocklength n, we fix caching, encoding and decoding functions as in (3.12), (3.15), (3.17) and (3.19), so that both the probability of worst-case error and the secrecy leakage satisfy:

$$\mathrm{P}_e^{\mathrm{Worst}} \xrightarrow[n\to\infty]{} 0 \qquad \text{and} \qquad \frac{1}{n} I(W_{d_k}; Z^n) \xrightarrow[n\to\infty]{} 0, \quad \text{for } k \in \{1,2\},$$

where $\mathrm{P}_e^{\mathrm{Worst}}$ is defined in (3.20).

By Fano's inequality, there exists a sequence of real numbers $\{\epsilon_n\}_{n=1}^{\infty}$ with

$$\frac{\epsilon_n}{n} \xrightarrow[n\to\infty]{} 0,$$

so that

$$H(W_{d_1}|Y_1^n, V_1) \leq \frac{\epsilon_n}{2}. \tag{3.63}$$

Thus,

$$
\begin{aligned}
nR_s &= H(W_{d_1}) \\
&= H(W_{d_1}|Z^n) + I(W_{d_1}; Z^n) \\
&\leq H(W_{d_1}|Z^n) + \frac{\epsilon_n}{2} \\
&\leq I(W_{d_1}; Y_1^n, V_1) - I(W_{d_1}; Z^n) + H(W_{d_1}|Y_1^n, V_1) + \frac{\epsilon_n}{2} \\
&\leq I(W_{d_1}; Y_1^n, V_1) - I(W_{d_1}; Z^n) + \epsilon_n \\
&\leq I(W_{d_1}; Y_1^n|V_1) - I(W_{d_1}; Z^n|V_1) + I(W_{d_1}; V_1|Z^n) + \epsilon_n \\
&\overset{(a)}{=} \sum_{i=1}^{n} \Big[I(W_{d_1}; Y_{1,i}|V_1, Y_1^{i-1}) - I(W_{d_1}; Z_i|V_1, Z_{i+1}^n) \Big] + n\mathcal{M} + \epsilon_n \\
&\overset{(b)}{=} \sum_{i=1}^{n} \Big[I(W_{d_1}; Y_{1,i}|V_1, Y_1^{i-1}) - I(W_{d_1}; Z_i|V_1, Z_{i+1}^n) \Big] + n\mathcal{M} + \epsilon_n \\
&\quad + \sum_{i=1}^{n} \Big[I(Z_{i+1}^n; Y_{1,i}|Y_1^{i-1}, V_1, W_{d_1}) - I(Y_1^{i-1}; Z_i|Z_{i+1}^n, V_1, W_{d_1}) \Big] \\
&= \sum_{i=1}^{n} \Big[I(W_{d_1}, Z_{i+1}^n; Y_{1,i}|V_1, Y_1^{i-1}) - I(W_{d_1}, Y_1^{i-1}; Z_i|V_1, Z_{i+1}^n) \Big] + n\mathcal{M} + \epsilon_n \\
&\overset{(c)}{=} \sum_{i=1}^{n} \Big[I(W_{d_1}, Z_{i+1}^n; Y_{1,i}|V_1, Y_1^{i-1}) - I(W_{d_1}, Y_1^{i-1}; Z_i|V_1, Z_{i+1}^n) \Big] + n\mathcal{M} + \epsilon_n \\
&\quad - \sum_{i=1}^{n} \Big[I(Z_{i+1}^n; Y_{1,i}|Y_1^{i-1}, V_1) - I(Y_1^{i-1}; Z_i|Z_{i+1}^n, V_1) \Big] \\
&= \sum_{i=1}^{n} \Big[I(W_{d_1}; Y_{1,i}|V_1, Y_1^{i-1}, Z_{i+1}^n) - I(W_{d_1}; Z_i|V_1, Y_1^{i-1}, Z_{i+1}^n) \Big] + n\mathcal{M} + \epsilon_n \\
&\overset{(d)}{\leq} \sum_{i=1}^{n} \Big[I(W_{d_1}; Y_{1,i}|V_1, Y_1^{i-1}, Z_{i+1}^n) - I(W_{d_1}; Z_i|V_1, Y_1^{i-1}, Z_{i+1}^n) \Big] + n\mathcal{M} + \epsilon_n \\
&\quad + \sum_{i=1}^{n} \Big[I(V_1, Y_1^{i-1}, Z_{i+1}^n; Y_{1,i}) - I(V_1, Y_1^{i-1}, Z_{i+1}^n; Z_i) \Big] \\
&= \sum_{i=1}^{n} \Big[I(W_{d_1}, V_1, Y_1^{i-1}, Z_{i+1}^n; Y_{1,i}) - I(W_{d_1}, V_1, Y_1^{i-1}, Z_{i+1}^n; Z_i) \Big] + n\mathcal{M} + \epsilon_n \\
&\overset{(e)}{\leq} \sum_{i=1}^{n} \Big[I(W_{d_1}, V_1, Y_1^{i-1}, Z_{i+1}^n; Y_{1,i}) - I(W_{d_1}, V_1, Y_1^{i-1}, Z_{i+1}^n; Z_i) \Big] + n\mathcal{M} + \epsilon_n \\
&\quad + \sum_{i=1}^{n} \Big[I(X_i; Y_{1,i}|W_{d_1}, V_1, Y_1^{i-1}, Z_{i+1}^n) - I(X_i; Z_i|W_{d_1}, V, Y_1^{i-1}, Z_{i+1}^n) \Big]
\end{aligned}
$$

$$= \sum_{i=1}^{n} \Big[I(X_i, W_{d_1}, V_1, Y_1^{i-1}, Z_{i+1}^n; Y_{1,i}) - I(X_i, W_{d_1}, V_1, Y_1^{i-1}, Z_{i+1}^n; Z_i) \Big] + n\mathcal{M} + \epsilon_n$$

$$\overset{(f)}{=} \sum_{i=1}^{n} \Big[I(X_i; Y_{1,i}) - I(X_i; Z_i) \Big] + n\mathcal{M} + \epsilon_n, \tag{3.64}$$

where (a) holds because $I(W_{d_1}; V_1|Z^n)$ is limited by the entropy of V_1 which cannot exceed $n\mathcal{M}$; (b) and (c) follow by the chain rule of mutual information and by applying Csiszar's sum-identity [11, pp. 25]; (d) and (e) hold because the eavesdropper is degraded with respect to receiver 1; and (f) holds because of the Markov chain

$$(V_1, W_{d_1}, Y_1^{i-1}, Z_{i+1}^n) \to X_i \to (Y_{1,i}, Z_i).$$

Dividing by n, we get

$$R_s \leq \sum_{i=1}^{n} \frac{1}{n} \Big[I(X_i; Y_{1,i}) - I(X_i; Z_i) \Big] + \mathcal{M} + \frac{\epsilon_n}{n}. \tag{3.65}$$

Then, we define a random variable Q uniform over $\{1, \ldots, n\}$ and independent of all the previously defined random variables. Also, we define the following random variables:

$$X := X_Q, \tag{3.66}$$

$$Y_1 := Y_{1,Q}, \tag{3.67}$$

$$Z := Z_Q. \tag{3.68}$$

We can now rewrite the above inequality as

$$\begin{aligned} R_s &\leq \sum_{q=1}^{n} \Pr\{Q = q\} \Big[I(X_q; Y_{1,q}|Q = q) - I(X_q; Z_q|Q = q) \Big] + \mathcal{M} + \frac{\epsilon_n}{n} \\ &\leq I(X; Y_1|Q) - I(X; Z|Q) + \mathcal{M} + \frac{\epsilon_n}{n}. \end{aligned} \tag{3.69}$$

When $n \to \infty$, R_s is achievable if there exists a pmf p_{QX} that satisfies (3.69):

$$\begin{aligned} R_s &\leq \max_{p_{QX}} \Big[I(X; Y_1|Q) - I(X; Z|Q) \Big] + \mathcal{M} \\ &= \max_{p_{QX}} \Big[(1 - \delta_1) H(X|Q) - (1 - \delta_z) H(X|Q) \Big] + \mathcal{M} \\ &= \max_{p_{QX}} \Big[(\delta_z - \delta_1) H(X|Q) \Big] + \mathcal{M} \\ &\overset{(a)}{=} (\delta_z - \delta_1) F + \mathcal{M}, \end{aligned} \tag{3.70}$$

where (a) holds because the maximum is achieved for $Q = \emptyset$ and X uniform over $\{1, \ldots, 2^F\}$.

Constraint (3.62b) can be proved along similar lines, when index 1 is replaced by index 2; cache content V_1 by a constant, and thus, cache memory size $\mathcal{M}$ by 0.

3.4 Lower bound on the secure capacity-memory tradeoff under symmetric cache assignment for the two-user scenario

With the purpose of highlighting the interest of one-sided cache assignment, we study, in this section, the same wiretap BC channel model but with a symmetric cache assignment among users. We consider one weak receiver and one strong receiver and we assume that the total cache memory $\mathcal{M}$ is split equally among both receivers, irrespective of their channel statistics. Hence, each receiver gets a cache memory of size $\mathcal{M}/2$, as shown in Figure 3.5. We compute the lower bound on the securely achievable capacity-memory tradeoff for this scenario.

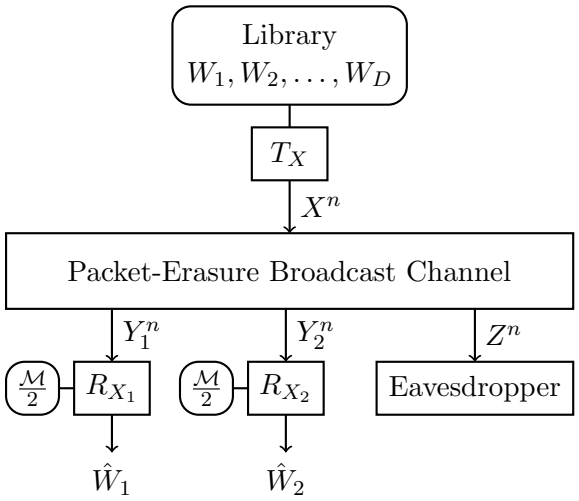

Figure 3.5: Packet-erasure BC with two legitimate receivers and an eavesdropper. Both receivers have equal cache memory of size $\frac{\mathcal{M}}{2}$.

We denote by $C_{s,\mathrm{Sym}}(\mathcal{M})$ the secure capacity-memory tradeoff for symmetric cache assignment in analogy to the secure capacity-memory tradeoff $C_s(\mathcal{M})$ for one-sided cache memory given in Definition 3.3. The lower bound on $C_{s,\mathrm{Sym}}(\mathcal{M})$ is stated in the following proposition:

Proposition 3.1 (Lower Bound on $C_{s,\mathrm{Sym}}(\mathcal{M})$)**.** *A rate-pair $(R_{s,Sym}, \mathcal{M})$ is securely achievable over the two-user wiretap erasure BC with symmetric cache assignment $\mathcal{M}/2$ at both receivers, if it satisfies the following three conditions:*

$$R_{s,Sym} \leq 2(1-\delta_1)F, \tag{3.71a}$$

$$R_{s,Sym} \leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F + \frac{3-2\delta_1-\delta_2}{2(2-\delta_1-\delta_2)}\frac{\mathcal{M}}{D}, \tag{3.71b}$$

$$R_{s,Sym} \leq \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F + \left[\frac{(\delta_z-\delta_1)(\delta_z-\delta_2)(3-2\delta_1-\delta_2)}{2(1-\delta_1)(1-\delta_2)(2\delta_z-\delta_1-\delta_2)} + \frac{D(1-\delta_z)\big[(1-\delta_1)(\delta_z-\delta_1)+(1-\delta_2)(\delta_z-\delta_2)\big]}{2(1-\delta_1)(1-\delta_2)(2\delta_z-\delta_1-\delta_2)}\right]\frac{\mathcal{M}}{D}. \quad (3.71c)$$

Thus, any $R_{s,Sym}$ satisfying (3.71) *forms a lower bound on $C_{s,Sym}(\mathcal{M})$.*

Proof. We derive the proof of this proposition in the next subsections by describing the coded caching scheme that can achieve this lower bound. □

3.4.1 Message splitting

For $d \in [1:D]$, split every message W_d into three sub-messages, such that

$$W_d = \left[W_d^{(0)}, W_d^{(1)}, W_d^{(2)}\right], \quad (3.72)$$

with rates $R^{(0)}$, $\frac{R^{(1)}}{2}$, and $\frac{R^{(1)}}{2}$, respectively. The total rate of W_d is

$$R_{s,\text{Sym}} = R^{(0)} + R^{(1)}. \quad (3.73)$$

If $R^{(0)} > (D-2)\frac{R^{(1)}}{2}$, divide $W_d^{(0)}$ into two further parts, such that

$$W_d^{(0)} = \left[W_d^{(0,1)}, W_d^{(0,2)}\right], \quad (3.74)$$

with rates $(D-2)\frac{R^{(1)}}{2}$ and $R^{(0)} - (D-2)\frac{R^{(1)}}{2}$, respectively. Otherwise, $W_d^{(0,1)} = W_d^{(0)}$ has rate $R^{(0)}$ and $W_d^{(0,2)}$ has zero rate.

3.4.2 Caching phase

Store sub-messages $\{W_d^{(k)}\}_{d=1}^D$ in cache memory V_k, $k \in \{1,2\}$. The cache contents of receivers 1 and 2 are

$$V_1 = \left\{W_1^{(1)}, W_2^{(1)}, \ldots, W_D^{(1)}\right\} \quad \text{and} \quad V_2 = \left\{W_1^{(2)}, W_2^{(2)}, \ldots, W_D^{(2)}\right\}. \quad (3.75)$$

Therefore, $R^{(1)}$ should satisfy

$$R^{(1)} \leq \frac{\mathcal{M}}{D}. \quad (3.76)$$

3.4.3 Delivery phase

If $R^{(0)} > (D-2)\frac{R^{(1)}}{2}$, the delivery phase takes place in five periods, otherwise it takes only three periods to transmit. For $\ell = 1, \ldots, 5$, let $\alpha_\ell n$ be the length of period ℓ, such that $0 \leq \alpha_\ell \leq 1$ and $\sum_{\ell=1}^{5} \alpha_\ell = 1$. If $R^{(0)} \leq (D-2)\frac{R^{(1)}}{2}$, α_4 and α_5 are null.

In the first and second periods, the transmitter randomly chooses a set of

$$\iota := \max\left\{1, \min\left\{\left\lceil \frac{2R^{(0)}}{R^{(1)}} \right\rceil, (D-2)\right\}\right\} \tag{3.77}$$

indices $\{j_1, j_2, \ldots, j_\iota\} \in (\mathcal{D} \setminus \{d_1, d_2\})$ and forms

$$W_{k,\mathrm{XOR}} := W_{d_k}^{(0,1)} \oplus \left[W_{j_1}^{(k)}, W_{j_2}^{(k)}, \ldots, W_{j_\iota}^{(k)}\right], \qquad k \in \{1,2\}. \tag{3.78}$$

It uses an optimal regular (non-wiretap) code to send $W_{1,\mathrm{XOR}}$ to receiver 1 in period 1 and $W_{2,\mathrm{XOR}}$ to receiver 2 in period 2. For each $k \in \{1,2\}$, receiver k first decodes the XOR message $W_{k,\mathrm{XOR}}$ and, with its cache content, it reconstructs the desired $W_{d_k}^{(0,1)}$.

In period 3, the transmitter sends the common message

$$W_{\mathrm{XOR}} = W_{d_1}^{(2)} \oplus W_{d_2}^{(1)}, \tag{3.79}$$

to both receivers using an optimal regular code. Each receiver k, $k \in \{1,2\}$, decodes the XOR message W_{XOR} and, with its cache content, it reconstructs the desired sub-message $W_{d_k}^{(3-k)}$.

In period 4, the transmitter sends message $W_{d_1}^{(0,2)}$ to receiver 1 and in period 5, it sends message $W_{d_2}^{(0,2)}$ to receiver 2 using optimal wiretap codes.

3.4.4 Analysis of the error probability

According to the defined delivery phase, receiver 1 decodes period 1, 3 and 4 and receiver 2 decodes period 2, 3 and 5. Hence, their error probabilities averaged over the codebooks $\mathcal{C}_1, \mathcal{C}_2$, the channel realizations, and the messages,

$$\mathrm{P}_{e,1}^{\mathrm{Worst}} \xrightarrow[n\to\infty]{} 0 \quad \text{and} \quad \mathrm{P}_{e,2}^{\mathrm{Worst}} \xrightarrow[n\to\infty]{} 0,$$

if the following six conditions are satisfied:

$$\min\left\{R^{(0)}, \frac{D-2}{2}R^{(1)}\right\} \leq \alpha_1(1-\delta_1)F, \tag{3.80a}$$

$$\min\left\{R^{(0)}, \frac{D-2}{2}R^{(1)}\right\} \leq \alpha_2(1-\delta_2)F, \tag{3.80b}$$

$$\frac{R^{(1)}}{2} \leq \alpha_3(1-\delta_1)F, \tag{3.80c}$$

$$\frac{R^{(1)}}{2} \leq \alpha_3(1-\delta_2)F, \tag{3.80d}$$

$$\max\left\{0, R^{(0)} - \frac{D-2}{2}R^{(1)}\right\} + R' \leq \alpha_4(1-\delta_1)F, \tag{3.80e}$$

$$\max\left\{0, R^{(0)} - \frac{D-2}{2}R^{(1)}\right\} + R'' \leq \alpha_5(1-\delta_2)F. \tag{3.80f}$$

3.4.5 Analysis of the information leakage

If $R^{(0)} \leq (D-2)\frac{R^{(1)}}{2}$, all the sent messages are secured by the XOR operation. Otherwise, phases 4 and 5 are secured by means of a random binning. In the later case, the security conditions in (3.21) are satisfied whenever

$$R' \geq \alpha_4(1-\delta_z)F, \tag{3.81a}$$

$$R'' \geq \alpha_5(1-\delta_z)F. \tag{3.81b}$$

Combining constraints (3.76), (3.80) and (3.81) yields the securely achievable rate-memory tuples for the symmetric caching case defined in Proposition 3.1.

3.5 Upper bound on the secure capacity-memory tradeoff under symmetric cache assignment for the two-user scenario

The upper bound on the secure capacity-memory tradeoff $C_{s,\mathrm{Sym}}(\mathcal{M})$ under symmetric cache assignment is stated in the following proposition:

Proposition 3.2 (Upper Bound on $C_{s,\mathrm{Sym}}(\mathcal{M})$)**.** *The secure capacity-memory tradeoff $C_{s,\mathrm{Sym}}(\mathcal{M})$ of the two-user wiretap erasure BC with symmetric cache memory $\mathcal{M}/2$ at both receivers satisfies the following three conditions:*

$$C_{s,\mathrm{Sym}}(\mathcal{M}) \leq (\delta_z - \delta_1)F + \frac{\mathcal{M}}{2}, \tag{3.82a}$$

$$C_{s,\mathrm{Sym}}(\mathcal{M}) \leq (1-\delta_1)F + \frac{\mathcal{M}}{2D}, \tag{3.82b}$$

$$C_{s,\mathrm{Sym}}(\mathcal{M}) \leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F + \frac{\mathcal{M}}{D}. \tag{3.82c}$$

Proof. Constraints (3.82b) and (3.82c) follow from [9, Theorem 9] and by ignoring the secrecy constraints in (3.21). The proof of constraint (3.82a) is analogous to that of (3.62a) in Section 3.3. □

3.6 Separate cache-channel coding under one-sided cache assignment for the two-user scenario

In order to emphasize the strength of the joint cache-channel coding approach, we characterize in this section the rates that are securely achievable under the separate cache-channel coding approach. In the separate scheme, the encoder consists of a *cache-encoder* that does not depend on the channel statistics followed by a *channel encoder* that does not depend on the cache content. In addition, the decoder consists of a *channel decoder* followed by a *cache decoder* that are subject to similar restrictions, as depicted in Figure 3.6. For comparison purposes, we consider the same scenario as in Section 3.2 where the total cache memory $\mathcal{M}$ is given to the weak receiver 1, while the strong receiver 2 has no cache memory.

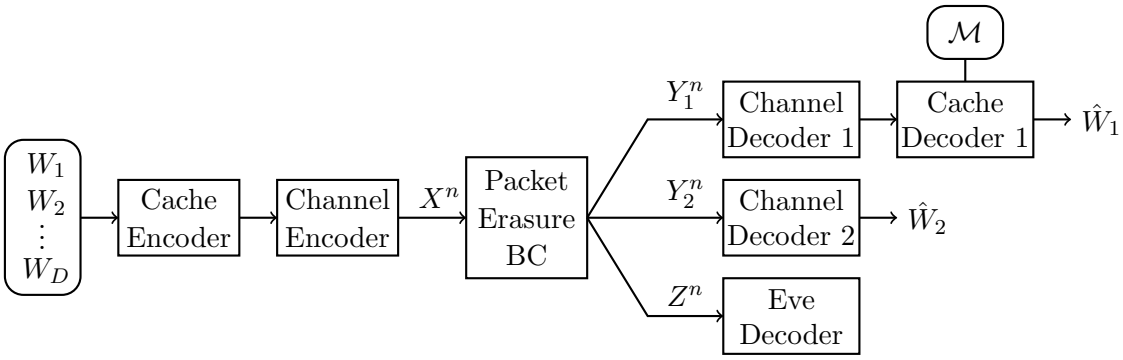

Figure 3.6: Separate cache-channel coding architecture.

The lower bound achieved by the best separate cache-channel coding scheme is stated in the following proposition:

Proposition 3.3 (Lower Bound on $C_{s,\text{Sep}}(\mathcal{M})$)**.** *A rate-memory pair $(R_{s,Sep}, \mathcal{M})$ is securely achievable over the two-user wiretap erasure BC with cache memory $\mathcal{M}$ only at the weak receiver using a separate cache-channel coding scheme, if it satisfies the following three conditions:*

$$R_{s,Sep} \leq (\delta_z - \delta_2)F, \tag{3.83a}$$

$$R_{s,Sep} \leq \frac{(1-\delta_1)(\delta_z-\delta_2)}{1+\delta_z-\delta_1-\delta_2}F + \frac{\delta_z-\delta_2}{1+\delta_z-\delta_1-\delta_2}\frac{\mathcal{M}}{D}, \tag{3.83b}$$

$$R_{s,Sep} \leq \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F + \frac{(\delta_z-\delta_2)\big[(D-1)(1-\delta_z)+(\delta_z-\delta_1)\big]}{(1-\delta_1)(2\delta_z-\delta_1-\delta_2)}\frac{\mathcal{M}}{D}. \quad (3.83c)$$

Thus, any $R_{s,Sep}$ satisfying (3.25) *forms a lower bound on $C_{s,Sep}(\mathcal{M})$.*

Proof. We derive the proof of this proposition in the next subsections by describing the separate cache-channel coding that can achieve this lower bound. □

3.6.1 Message splitting

As in Section 3.2.1, split every message W_d, $d \in \mathcal{D}$, into three sub-messages, such that

$$W_d = \left[W_d^{(0,1)}, W_d^{(0,2)}, W_d^{(1)}\right], \quad (3.84)$$

with rates $\min\{R^{(0)}, (D-2)R^{(1)}\}$, $\max\{0, R^{(0)}-(D-2)R^{(1)}\}$ and $R^{(1)}$, respectively. The total rate of W_d is

$$R_{s,\text{Sep}} = R^{(0)} + R^{(1)}. \quad (3.85)$$

3.6.2 Caching phase

The caching phase is similar to the case of joint cache-channel coding in Section 3.2.3. Here, we also store $\{W_d^{(1)}\}_{d=1}^{D}$ in cache memory of receiver 1. Hence, the cache content V_1 is given in (3.33) and $R^{(1)}$ has to satisfy (3.34).

3.6.3 Delivery phase

Let α_1, α_2 and α_3 be three values chosen from the interval $[0,1]$, such that

$$\alpha_1+\alpha_2+\alpha_3 = 1.$$

If $R^{(0)} > (D-2)R^{(1)}$, the delivery phase takes place in three periods of lengths $\alpha_1 n$, $\alpha_2 n$ and $\alpha_3 n$ respectively. Otherwise, we fix $\alpha_2 = 0$ and it takes two periods to transmit.

In the first period, the transmitter conveys message $W_{d_1}^{(0,1)}$ to receiver 1. It randomly chooses a set of ι indices $j_1, j_2, \ldots, j_\iota \in \big(\mathcal{D} \setminus \{d_1, d_2\}\big)$, where ι is defined in (3.36). It uses then these indices to select ι messages from receiver 1's cache and generate a key K_1 of the same length as message $W_{d_1}^{(0,1)}$ as:

$$K_1 = \left[W_{j_1}^{(1)}, W_{j_2}^{(1)}, \ldots, W_{j_\iota}^{(1)}\right], \quad (3.86)$$

Then, it sends $W_{d_1}^{(0,1)}$ using a wiretap code with secret key K_1 [113],[11, (22.7)].

In the second period, the transmitter conveys $W_{d_1}^{(0,2)}$ to receiver 1 using a wiretap code with random binning of rate R'.

In the third period, it conveys message W_{d_2} to receiver 2 using a wiretap code with random binning of rate R''.

3.6.4 Analysis of the probability of error

Receiver 1 decodes periods 1 and 2 and receiver 2 decodes period 3. Hence, their probabilities of error averaged over the codebooks $\mathcal{C}_1, \mathcal{C}_2$, the channel realizations, and the messages,

$$\mathrm{P}_{e,1}^{\mathrm{Worst}} \xrightarrow[n\to\infty]{} 0 \quad \text{and} \quad \mathrm{P}_{e,2}^{\mathrm{Worst}} \xrightarrow[n\to\infty]{} 0,$$

if the following three conditions are satisfied:

$$\min\left\{R^{(0)}, \frac{D-2}{2}R^{(1)}\right\} \leq \alpha_1(1-\delta_1)F, \tag{3.87a}$$

$$\max\left\{0, R^{(0)} - \frac{D-2}{2}R^{(1)}\right\} + R' \leq \alpha_2(1-\delta_1)F, \tag{3.87b}$$

$$R_{s,\mathrm{Sep}} + R'' \leq \alpha_3(1-\delta_2)F. \tag{3.87c}$$

3.6.5 Analysis of the information leakage

The message $W_{d_1}^{(0,1)}$ sent to receiver 1 is secured by the key K_1. If $R^{(0)} > (D-2)R^{(1)}$, the message $W_{d_1}^{(0,2)}$ is secured by means of random binning whenever

$$R' \geq \alpha_2(1-\delta_z)F, \tag{3.88}$$

and the message W_{d_2} is similarly secured whenever

$$R'' \geq \alpha_3(1-\delta_z)F. \tag{3.89}$$

Combining constraints (3.33) and (3.87)–(3.89) yields the securely achievable rate-memory tuples using the separate cache-channel coding under asymmetric cache assignment defined in Proposition 3.3.

3.7 Discussion and numerical results

In the sequel, we discuss our derived bounds in Sections 3.2 and 3.3 and compare them to the bounds proposed in [9] for the scenario without secrecy. We also compare these bounds to the ones derived for the symmetric cache assignment and the separate cache channel coding in Sections 3.4, 3.5 and 3.6, respectively.

We assume that receiver 2 is much stronger than receiver 1.

3.7.1 Impact of the secrecy constraint

In Figures 3.7 and 3.8, we compare the upper and lower bounds on the capacity-memory tradeoffs $C_s(\mathcal{M})$ and $C(\mathcal{M})$ with and without secrecy constraints.

Based on the lower and upper bounds on $C_s(\mathcal{M})$ defined in Theorems 3.1 and 3.2, and comparing them with the bound without secrecy [9], we can conclude the following:

- *Secure capacity without caching:*
  When no cache memory exists at the receivers, i.e. at $\mathcal{M} = 0$, the best lower bound on $C_s(\mathcal{M} = 0)$ can be obtained from Theorem 3.1 as

$$C_s(\mathcal{M} = 0) \geq R_0 := \frac{(\delta_z - \delta_1)(\delta_z - \delta_2)}{2\delta_z - \delta_1 - \delta_2} F. \tag{3.90}$$

  The right-hand side R_0 coincides with the secrecy capacity of the two-user wiretap BC without caching when the individual secrecy constraints in (3.21) are replaced by the stronger joint secrecy constraint [114]

$$\lim_{n\to\infty} \frac{1}{n} I(W_{d_1}, W_{d_2}; Z^n) = 0. \tag{3.91}$$

- *Small cache memories regime:*
  For cache memories $\mathcal{M}$ below a given threshold $\mathcal{M}_1$,

$$\mathcal{M} \leq \mathcal{M}_1 = \frac{D(1 - \delta_z)(\delta_z - \delta_2)}{(D-1)(1 + \delta_z - \delta_1 - \delta_2)}, \tag{3.92}$$

  our joint cache-channel coding scheme achieves rates

$$R_s = R_0 + \left[\frac{\delta_z - \delta_2}{2\delta_z - \delta_1 - \delta_2} + \frac{\delta_z - \delta_1}{D(2\delta_z - \delta_1 - \delta_2)} \right] \mathcal{M}. \tag{3.93}$$

  For small cache memories, the slope of the secure capacity-memory tradeoff $C_s(\mathcal{M})$ decreases negligibly with the library size D. In fact, the slope is given by the sum

of the two terms in brackets in (3.93), where the first term is constant and the second one decreases with D. This is different in a scenario without secrecy, where the slope is at most $\frac{1}{D}$ [9, Corollary 7.1]. Therefore, caching is more useful in a wiretap-communication scenario than in a standard scenario. The reason is that the cache memory does not only render the transmission more efficient, but also more secure (for example by means of a one-time pad) adding thus an additional gain, namely the *secrecy gain*, to the case without secrecy.

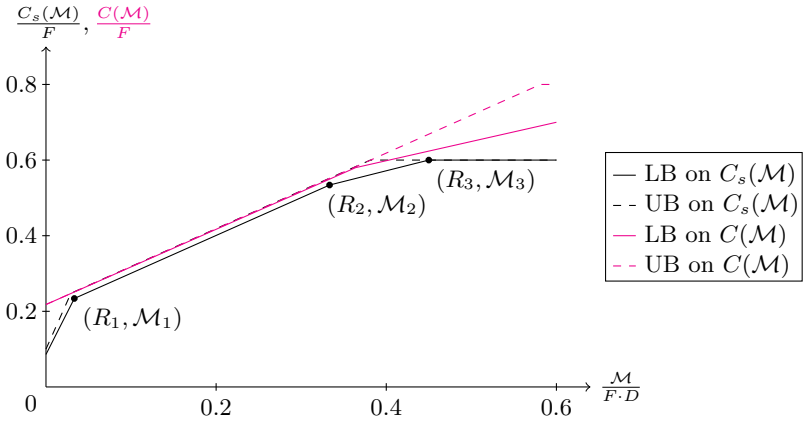

Figure 3.7: Lower and upper bounds on the capacity-memory tradeoffs $C(\mathcal{M})/C_s(\mathcal{M})$ wo/w secrecy constraint for the two-user wiretap erasure BC with erasure probabilities $\delta_1 = 0.7$, $\delta_2 = 0.2$, $\delta_z = 0.8$, $F = 5$, and for library size $D = 5$.

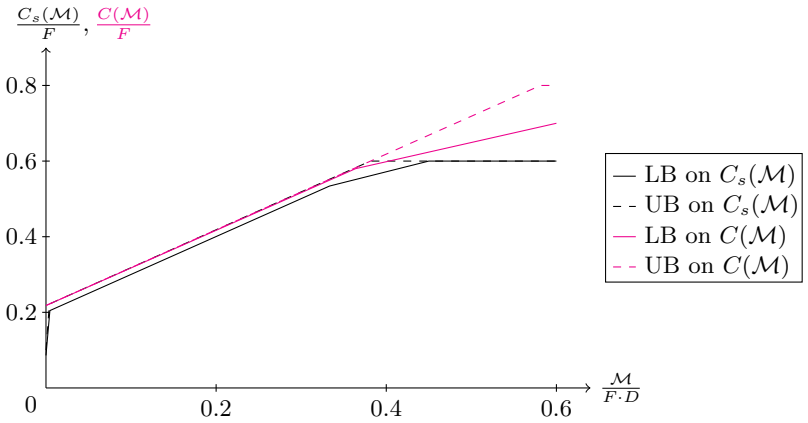

Figure 3.8: Lower and upper bounds on the capacity-memory tradeoffs $C(\mathcal{M})/C_s(\mathcal{M})$ wo/w secrecy constraint for the two-user wiretap erasure BC with erasure probabilities $\delta_1 = 0.7$, $\delta_2 = 0.2$, $\delta_z = 0.8$, $F = 5$, and for library size $D = 30$.

- *Intermediate regime:*
  In this regime, two changes in the slope of the lower bound on $C_s(\mathcal{M})$ occur at

points $(R_1, \mathcal{M}_1)$ and $(R_2, \mathcal{M}_2)$.

- $(R_1, \mathcal{M}_1)$:
  When the cache memory $\mathcal{M} = \mathcal{M}_1$, the securely achievable rate is

$$R_1 = \left[\frac{(1-\delta_1)(\delta_z - \delta_2)}{1+\delta_z-\delta_1-\delta_2} + \frac{(1-\delta_z)(\delta_z-\delta_2)}{(D-1)(1+\delta_z-\delta_1-\delta_2)} \right] F. \tag{3.94}$$

  At this point, the slope of $C_s(\mathcal{M})$ decreases to $\frac{1}{D}$. This change in the slope occurs when the secrecy gain saturates. Recall that, the first transmission period of our coding scheme is secured partly by the random binning of rate R' and partly by the W_{XOR} message which plays the role of a one time-pad key. When the cache memory is small, it is entirely exploited in our scheme: message $W_{d_1}^{(1)}$ is used to increase the efficiency, message $W_{d_2}^{(1)}$ serves for efficiency and security and the rest are XORed with the transmitted message to improve the security. Thus, the rate of W_{XOR} increases with the size of $\mathcal{M}$, decreasing with it the rate needed for random binning. At $(R_1, \mathcal{M}_1)$, W_{XOR} attains the capacity of the eavesdropper's channel, the secrecy gain saturates and the increase in cache memory is only used to improve the system's efficiency. Hence, the slope of the $C_s(\mathcal{M})$ becomes $\frac{1}{D}$ as in the standard scenario without secrecy.

- $(R_2, \mathcal{M}_2)$:

$$(R_2, \mathcal{M}_2) = \left(\frac{(1-\delta_2)(\delta_z-\delta_2)}{1+\delta_z-\delta_1-\delta_2} F, \frac{D(\delta_z-\delta_2)(\delta_1-\delta_2)}{1+\delta_z-\delta_1-\delta_2} F \right). \tag{3.95}$$

  At this point, the total rate of the transmitted messages attains the capacity of receiver 2's channel in both transmission periods. Within this rate, messages conveyed to receiver 1 attain also its channel capacity. After this point, the increase in the cache memory size is used in the first period to decrease the rate of messages meant for receiver 1 and replace them with messages for receiver 2.

- *Saturation regime:*
  At this point, the maximal secure rate is achieved. In fact, since the strong receiver has no cache memory, the secure capacity-memory tradeoff $C_s(\mathcal{M})$ is limited to $(\delta_z - \delta_2)F$. It can be trivially achieved when the weak receiver can store the entire library in its cache memory, i.e. when $\mathcal{M} = DF(\delta_z - \delta_s)$. In our scheme, this saturation capacity is achieved for a cache memory smaller than $DF(\delta_z - \delta_s)$, as we see in the following corollary.

Corollary 3.1. *The capacity-memory tradeoff $C_s(\mathcal{M})$ of the two-user wiretap erasure BC with cache memory $\mathcal{M}$ at the weaker receiver achieves*

$$C_s(\mathcal{M}) = (\delta_z - \delta_2)F, \tag{3.96}$$

when

$$\mathcal{M} \geq \mathcal{M}_3 = F \cdot \max\left\{ D\frac{(\delta_z - \delta_2)^2}{1-\delta_2}, (\delta_z - \delta_2) \right\}. \tag{3.97}$$

Proof. Under condition (3.97), constraints (3.25b)–(3.25f) are less stringent than constraint (3.25a). □

Note that in the scenario without secrecy, the maximal capacity is $(1-\delta_2)F$. This explains the difference between the bounds on $C_s(\mathcal{M})$ and $C(\mathcal{M})$ that we see in Figures 3.7 and 3.8.

- We also notice in Figures 3.7 and 3.8 that our lower and upper bounds are very close for many parameters. However, they seem to coincide only for large cache memories $\mathcal{M} \geq \mathcal{M}_3$, defined in (3.97).

Note that the observations stated above are not specific only to the considered example and are true for all channel parameters.

3.7.2 Impact of cache assignment

In Figures 3.9 and 3.10, we compare the capacity-memory tradeoff for one-sided and symmetric cache assignments.

Based on the lower and upper bounds on $C_s(\mathcal{M})$ defined in Theorems 3.1 and 3.2, and comparing them with the bound in Propositions 3.1 and 3.2, we can observe the following:

- For $\mathcal{M} = 0$, both schemes achieve the same rate R_0 in (3.90).
- For small cache memories, our joint cache-channel coding scheme for one-sided cache assignment improves over the best possible coding scheme for symmetric cache assignment.
- For large cache memories, when the difference of channel capacities between the weak and strong receivers is large, one-sided cache assignment is more efficient. Nevertheless, when δ_1 and δ_2 are close, the secure capacity-memory tradeoff is larger under a symmetric cache assignment than under a one-sided cache assignment. The reason is that in the former case the cache contents can be used to secure the communication to both receivers and thus, the capacity of receiver 2 can attain $F(1-\delta_2)$.

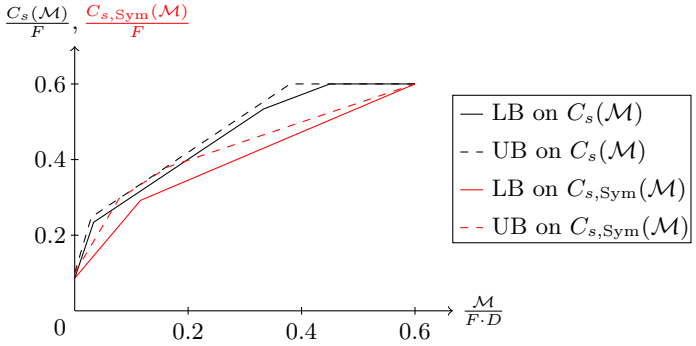

Figure 3.9: Lower and upper bounds on the secure capacity-memory tradeoffs $C_s(\mathcal{M})/C_{s,\mathrm{Sym}}(\mathcal{M})$ for the two-user wiretap erasure BC with erasure probabilities $\delta_1 = 0.7$, $\delta_2 = 0.2$, $\delta_z = 0.8$, $F = 5$, and library size $D = 5$.

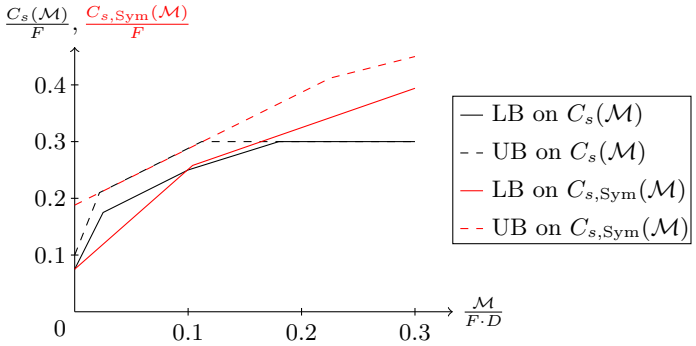

Figure 3.10: Lower and upper bounds on the secure capacity-memory tradeoffs $C_s(\mathcal{M})/C_{s,\mathrm{Sym}}(\mathcal{M})$ for the two-user wiretap erasure BC with erasure probabilities $\delta_1 = 0.7$, $\delta_2 = 0.5$, $\delta_z = 0.8$, $F = 5$, and library size $D = 5$.

3.7.3 Impact of joint cache-channel coding

In Figure 3.11, we compare the lower bounds obtained by means of the joint and separate cache-channel coding schemes. From the lower bounds in Theorem 3.1 and Proposition 3.3, we can observe the following:

- Without cache memory, $\mathcal{M} = 0$, both schemes achieve the same rate R_0 in (3.90).
- As in the case without secrecy [9], our joint cache-channel coding scheme achieves significantly larger rate-memory tradeoff than the equivalent separate cache-channel coding scheme for all channel parameters and cache memory sizes.

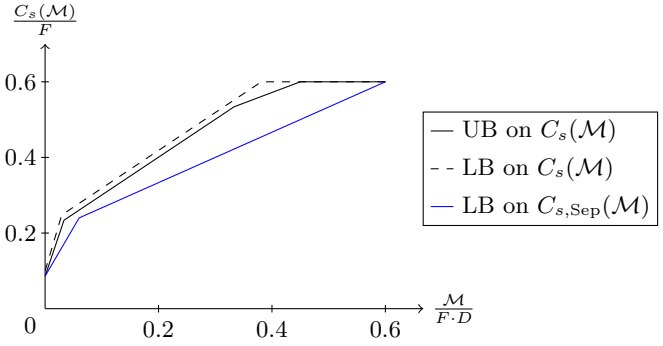

Figure 3.11: Lower and upper bounds on the secure capacity-memory tradeoffs $C_s(\mathcal{M})/C_{s,\text{Sep}}(\mathcal{M})$ for the two-user wiretap erasure BC with erasure probabilities $\delta_1 = 0.7$, $\delta_2 = 0.2$, $\delta_z = 0.8$, $F = 5$, and library size $D = 5$.

3.8 General lower bound on the secure capacity-memory tradeoff

In this section, we consider the general scheme, depicted in Figure 3.1, with K_w weak receivers and K_s strong receivers. Only the weak receivers are provided with cache memories of size $\mathcal{M}$. For this scheme, we compute the general lower bound on the secure capacity-memory tradeoff $C_s^{(K)}(\mathcal{M})$.

Consider the following five rate-memory pairs:

- $$R_0^{(K)} := \Big(\frac{K_w}{\delta_z - \delta_w} + \frac{K_s}{\delta_z - \delta_s}\Big)^{-1} F, \tag{3.98a}$$
  $$\mathcal{M}_0^{(K)} := 0; \tag{3.98b}$$
- $$R_3^{(K)} := \frac{(\delta_z - \delta_s)}{K_s} F, \tag{3.98c}$$
  $$\mathcal{M}_3^{(K)} := \frac{D K_w (\delta_z - \delta_s)^2}{K_s\big[K_s(1-\delta_z) + K_w(\delta_z - \delta_s)\big]} F; \tag{3.98d}$$
- $$R_4^{(K)} := \frac{(\delta_z - \delta_s)}{K_s} F, \tag{3.98e}$$
  $$\mathcal{M}_4^{(K)} := D\frac{(\delta_z - \delta_s)}{K_s} F; \tag{3.98f}$$

If $K_s(1-\delta_z) - (D - K_w)(\delta_w - \delta_s) \leq 0$,

- $$R_1^{(K)} := \frac{2(1-\delta_w)(\delta_z - \delta_s)\big[(D-K_w)(1-\delta_w) + K_w(1-\delta_z)\big]F}{\beta_1}, \tag{3.98g}$$

$$\mathcal{M}_1^{(K)} := \frac{2D(1-\delta_w)(\delta_z-\delta_s)(1-\delta_z)F}{\beta_1}; \tag{3.98h}$$

- $$R_2^{(K)} := \frac{2(1-\delta_w)(\delta_z-\delta_s)\big[K_s(1-\delta_w)+K_w(\delta_w-\delta_s)\big]F}{\beta_2}, \tag{3.98i}$$
$$\mathcal{M}_2^{(K)} := \frac{2D(1-\delta_w)(\delta_z-\delta_s)(\delta_w-\delta_s)F}{\beta_2}; \tag{3.98j}$$

Otherwise, if $K_s(1-\delta_z)-(D-K_w)(\delta_w-\delta_s)>0$,

- $$R_1^{(K)} := \frac{2(1-\delta_w)(\delta_z-\delta_s)\big[K_s(\delta_z-\delta_w)+D(\delta_w-\delta_s)\big]F}{\beta_3}, \tag{3.98k}$$
$$\mathcal{M}_1^{(K)} := \frac{2D(1-\delta_w)(\delta_z-\delta_s)(\delta_w-\delta_s)F}{\beta_3}; \tag{3.98l}$$

- $$R_2^{(K)} := \frac{2(1-\delta_w)(\delta_z-\delta_s)\big[(D-K)(1-\delta_w)+K_w(1-\delta_z-\delta_w+\delta_s)\big]F}{\beta_4} \tag{3.98m}$$
$$\mathcal{M}_2^{(K)} := \frac{2D(1-\delta_w)(\delta_z-\delta_s)(1-\delta_z-\delta_w+\delta_s)F}{\beta_4}; \tag{3.98n}$$

where

$$\begin{aligned}\beta_1 &= 2K_w(D-K_w)(1-\delta_w)(\delta_z-\delta_s)+K_w(K_w-1)(1-\delta_z)(\delta_z-\delta_s)\\ &\quad+2K_s(D-K_w)(1-\delta_w)^2,\end{aligned} \tag{3.99a}$$

$$\beta_2 = 2K_sK_w(1-\delta_w)(\delta_z-\delta_s)+K_w(K_w-1)(\delta_w-\delta_s)(\delta_z-\delta_s)+2K_s^2(1-\delta_w)^2, \tag{3.99b}$$

$$\begin{aligned}\beta_3 &= 2K_wK_s(1-\delta_w)(\delta_z-\delta_s)+K_w(K_w-1)(\delta_w-\delta_s)(\delta_z-\delta_s)\\ &\quad+2K_s(1-\delta_w)\big[K_s(\delta_z-\delta_w)+(D-K_w)(\delta_w-\delta_s)\big],\end{aligned} \tag{3.99c}$$

$$\begin{aligned}\beta_4 &= 2K_w(D-K)(1-\delta_w)(\delta_z-\delta_s)+K_w(K_w-1)(\delta_z-\delta_s)(1-\delta_z-\delta_w+\delta_s)\\ &\quad+2(1-\delta_w)\big[K_s(D-K)(1-\delta_w)+K_wK_s(1-\delta_z)-K_w(D-K_w)(\delta_w-\delta_s)\big].\end{aligned} \tag{3.99d}$$

Theorem 3.3 (Lower Bound on $C_s^{(K)}(\mathcal{M})$)**.** *The upper convex hull of the five rate-memory pairs* $\big\{(R_\ell^{(K)},\mathcal{M}_\ell^{(K)}) : \ \ell\in\{0,\dots,4\}\big\}$ *in* (3.98) *lower bounds the secure capacity-memory tradeoff of the* K*-receiver channel with* K_w *weak receivers and* K_s *strong receivers with cache memories only at the weak receivers:*

$$C_s^{(K)}(\mathcal{M}) \geq \text{upper hull}\big\{(R_\ell^{(K)},\mathcal{M}_\ell^{(K)}), \ell\in\{0,\dots,4\}\big\}. \tag{3.100}$$

Proof. It suffices to prove the achievability of the six rate-memory pairs $\{(R_\ell,\mathcal{M}_\ell)\colon \ \ell=0,\dots,5\}$. The achievability of the upper convex hull follows by time/memory sharing arguments as in [10]. □

Remark 3.2. When there is no cache, i.e., $\mathcal{M} = 0$, the secure capacity-memory tradeoff $C_s(\mathcal{M})$ was determined in [114] as:

$$C_s(\mathcal{M}) = \left(\sum_{k=1}^{K} \frac{1}{\delta_z - \delta_k} \right)^{-1} F, \tag{3.101}$$

showing the achievability of the pair $(R_0^{(K)}, \mathcal{M}_0^{(K)})$.

Remark 3.3. For the achievability of $(R_4^{(K)}, \mathcal{M}_4^{(K)})$, since the strong receivers have no cache memories, the secure capacity-memory tradeoff $C_s^{(K)}(\mathcal{M})$ cannot exceed $\frac{(\delta_z - \delta_2)F}{K_s}$. It is trivially achieved when the weak receivers can store the entire library in their cache memories, i.e. for $\mathcal{M} = D\frac{(\delta_z - \delta_s)F}{K_s}$.

Remark 3.4. Since the weak receivers can always choose to ignore parts of their cache memories, the secure capacity-memory tradeoff is monotonically non-decreasing and thus if for some $\tilde{\mathcal{M}}$ the maximum $\frac{(\delta_z - \delta_2)F}{K_s}$ is achieved, it is also achieved for all $\mathcal{M} \geq \tilde{\mathcal{M}}$:

$$C_s(\tilde{\mathcal{M}}) = \frac{(\delta_z - \delta_2)F}{K_s} \quad \Rightarrow \quad C_s(\mathcal{M}) = \frac{(\delta_z - \delta_2)F}{K_s}, \quad \forall \mathcal{M} \geq \tilde{\mathcal{M}}.$$

The achievability of the remaining rate-memory pairs is outlined in the following subsections.

3.8.1 Scheme achieving rate-memory pair $(R_1^{(K)}, \mathcal{M}_1^{(K)})$

- If $K_s(1 - \delta_z) - (D - K_w)(\delta_w - \delta_s) \leq 0$, apply the following scheme:

  Preparations: Let $\alpha_1, \alpha_2, \alpha_3 \in [0, 1]$, such that $\alpha_1 + \alpha_2 + \alpha_3 = 1$. For $d \in \mathcal{D}$, split each message into two sub-messages, such that

  $$W_d = \left[W_d^{(0)}, W_d^{(1)} \right], \tag{3.102}$$

  with rates

  $$R^{(0)} = \frac{\alpha_2 F(1 - \delta_w)}{K_w} \quad \text{and} \quad R^{(1)} = \frac{\alpha_2 F(1 - \delta_z)}{(D - K_w)}. \tag{3.103}$$

  Then, split every sub-message $W_d^{(1)}$ into K_w sub-messages

  $$W_d^{(1)} = \left\{ W_{d,i}^{(1)} : i \in \mathcal{K}_w \right\}, \tag{3.104}$$

  of rate $\frac{R^{(1)}}{K_w}$ each.

If $R^{(0)} > (D-K)R^{(1)}/K_w$, divide every sub-message $W_d^{(0)}$ into two further sub-messages, such that

$$W_d^{(0)} = \left[W_d^{(0,1)}, W_d^{(0,2)}\right], \tag{3.105}$$

with rates $(D-K)R^{(1)}/K_w$ and $R^{(0)} - (D-K)R^{(1)}/K_w$, respectively. Otherwise, $W_d^{(0,1)} = W_d^{(0)}$ has rate $R^{(0)}$ and $W_d^{(0,2)}$ has zero rate.

Caching phase: For every weak receiver $i \in \mathcal{K}_w$, store the cache content

$$V_i = \left\{W_{d,i}^{(1)} : d \in \{1, \ldots, D\}\right\}. \tag{3.106}$$

Thus, the cache memory size is

$$\mathcal{M}_1^{(K)} = D\frac{R^{(1)}}{K_w}. \tag{3.107}$$

Delivery phase: The delivery phase is divided into three sub-phases of lengths $\alpha_1 n$, $\alpha_2 n$ and $\alpha_3 n$.

In the first sub-phase, the transmitter conveys message $W_{d_i}^{(1)}$ to every weak receiver $i \in \mathcal{K}_w$ by time-sharing over $\binom{K_w}{2}$ periods. In each period, it sends $W_{d_{i_1},i_2}^{(1)} \oplus W_{d_{i_2},i_1}^{(1)}$ to receivers i_1 and i_2.

In the second sub-phase, the transmitter conveys message $W_{d_i}^{(0)}$ to every weak receiver $i \in \mathcal{K}_w$ and message $W_{d_j}^{(1)}$ to every strong receiver $j \in \mathcal{K}_s$ by time-sharing over K_w periods. For each period, it generates a piggyback codebook $\mathcal{C}$ with $\lfloor 2^{nR^{(0)}} \rfloor \cdot \lfloor 2^{nK_sR^{(1)}/K_w} \rfloor$ subcodebooks $\mathcal{C}\big(\tilde{W}_1, \tilde{W}_2\big)$ with one codeword each.

For each period, it chooses a set of indices $\{j_1, \ldots, j_\iota\} \in \mathcal{D} \setminus \mathbf{d}$, where

$$\iota := \max\left\{1, \min\left\{\left\lceil \frac{K_wR^{(0)}}{R^{(1)}} \right\rceil, (D-K)\right\}\right\}, \tag{3.108}$$

and generates

$$W_{\mathrm{XOR},i} = W_{d_i}^{(0,1)} \oplus \left[W_{d_{j_1},i}^{(1)}, \ldots, W_{d_{j_\iota},i}^{(1)}\right]. \tag{3.109}$$

Then, it transmits the codeword of $\mathcal{C}\big(\tilde{W}_1, \tilde{W}_2\big)$ for $\tilde{W}_1 = \left[W_{\mathrm{XOR},i}, W_{d_i}^{(0,2)}\right]$ and $\tilde{W}_2 = \left\{W_{d_j,i}^{(1)} : j \in \mathcal{K}_s\right\}$.

In the third sub-phase, the transmitter conveys message $W_{d_j}^{(0)}$ to every strong receiver $j \in \mathcal{K}_s$ by time-sharing over K_s periods. In each period, it sends $W_{d_j}^{(0)}$ to receiver j using a wiretap code.

Analysis: Only the weak receivers decode the first transmission sub-phase. Thus, it is reliably decoded if

$$\frac{\binom{K_w}{2}\frac{R^{(1)}}{K_w}}{F(1-\delta_w)} \le \alpha_1 \quad \Rightarrow \quad \frac{\frac{K_w-1}{2}R^{(1)}}{F(1-\delta_w)} \le \alpha_1. \tag{3.110}$$

Only the strong receivers decode the third transmission sub-phase. Thus, it is reliably decoded if

$$\frac{K_s R^{(0)}}{F(\delta_z - \delta_s)} \leq \alpha_3. \tag{3.111}$$

Every weak receiver $i \in \mathcal{K}_w$ decodes only message $W_{d_i}^{(0)}$ transmitted in the second sub-phase, whereas every strong receiver $j \in \mathcal{K}_s$ decodes all the transmitted messages. Thus, the second sub-phase is reliably decoded whenever

$$\max\left\{\frac{K_w R^{(0)}}{F(1-\delta_w)}, \frac{K_w R^{(0)} + K_s R^{(1)}}{F(1-\delta_s)}\right\} \leq \alpha_2. \tag{3.112}$$

Moreover, the messages conveyed in the first sub-phase are secured by the XOR operation and the messages conveyed in the third sub-phase are secured by the random binning. In the second sub-phase, the messages are secure because $(D - K_w)R^{(1)} = \alpha_2(1-\delta_z)F$.

Maximizing constraints (3.110), (3.111) and (3.112) achieves the rate-memory pair $(R_1^{(K)}, \mathcal{M}_1^{(K)})$ in (3.98g) and (3.98h), for

$$\alpha_1 = \frac{K_w(K_w-1)(1-\delta_z)(\delta_z-\delta_s)}{\beta_1}, \tag{3.113a}$$

$$\alpha_2 = \frac{2K_w(D-K_w)(1-\delta_w)(\delta_z-\delta_s)}{\beta_1}, \tag{3.113b}$$

$$\alpha_3 = \frac{2K_s(D-K_w)(1-\delta_w)^2}{\beta_1}, \tag{3.113c}$$

where β_1 is defined in (3.99a).

- Otherwise, if $K_s(1-\delta_z) - (D-K_w)(\delta_w - \delta_s) > 0$, apply the following scheme:

  Preparations: Let $\alpha_1, \alpha_2, \alpha_3 \in [0,1]$, such that $\alpha_1 + \alpha_2 + \alpha_3 = 1$. Let

  $$R' = \frac{\alpha_2 F\big[K_s(1-\delta_z) - (D-K_w)(\delta_w - \delta_s)\big]}{K_w K_s}. \tag{3.114}$$

  The messages are split similarly to the previous case but with a change in the message rates. In this case,

  $$R^{(0)} = \frac{\alpha_2 F(1-\delta_w)}{K_w} - R' \quad \text{and} \quad R^{(1)} = \frac{\alpha_2 F(\delta_w - \delta_s)}{K_s}. \tag{3.115}$$

  Caching phase: The caching phase is also similar to the previous case.

  Delivery phase: The delivery phase is divided into three sub-phases of lengths $\alpha_1 n$, $\alpha_2 n$ and $\alpha_3 n$. The first and third sub-phases are similar to the previous case. However, the second one is different.

In the second sub-phase, the transmitter conveys message $W_{d_i}^{(0)}$ to every weak receiver $i \in \mathcal{K}_w$ and message $W_{d_j}^{(1)}$ to every strong receiver $j \in \mathcal{K}_s$ by time-sharing over K_w periods. For each period, it generates a piggyback codebook $\mathcal{C}$ with $\lfloor 2^{nR^{(0)}} \rfloor \cdot \lfloor 2^{nK_sR^{(1)}/K_w} \rfloor$ subcodebooks $\mathcal{C}\big(\tilde{W}_1, \tilde{W}_2\big)$. Each subcodebook has $\lfloor 2^{nR'} \rfloor$ codewords.

For each period, it chooses a set of indices $\{j_1, \ldots, j_\iota\} \in \mathcal{D} \setminus \mathbf{d}$, where ι is defined in (3.108), and generates $W_{\text{XOR},i}$ as in (3.109). Then, it picks an index $J_{1,i}$ uniformly at random from $\big[1 : \lfloor 2^{nR'} \rfloor\big]$, and transmits the

$$J_{1,i}\text{-th codeword of the subcodebook } \mathcal{C}\big(\tilde{W}_1, \tilde{W}_2\big), \tag{3.116}$$

for $\tilde{W}_1 = \big[W_{\text{XOR},i}, W_{d_i}^{(0,2)}\big]$ and $\tilde{W}_2 = \big\{W_{d_j,i}^{(1)} : j \in \mathcal{K}_s\big\}$.

Analysis: Since the first and third sub-phases do not change, the same conditions (3.110) and (3.111) should be satisfied in order to ensure reliable decoding in these phases.

As for the second sub-phase, it is reliably decoded whenever

$$\max\left\{\frac{K_w\big(R^{(0)}+R'\big)}{F(1-\delta_w)}, \frac{K_w\big(R^{(0)}+R'\big)+K_sR^{(1)}}{F(1-\delta_s)}\right\} \le \alpha_2. \tag{3.117}$$

Maximizing constraints (3.110), (3.111) and (3.117) achieves the rate-memory pair $(R_1^{(K)}, \mathcal{M}_1^{(K)})$ in (3.98k) and (3.98l), for

$$\alpha_1 = \frac{K_w(K_w-1)(\delta_w-\delta_s)(\delta_z-\delta_s)}{\beta_3}, \tag{3.118a}$$

$$\alpha_2 = \frac{2K_wK_s(1-\delta_w)(\delta_z-\delta_s)}{\beta_3}, \tag{3.118b}$$

$$\alpha_3 = \frac{2K_s(1-\delta_w)\big[K_s(\delta_z-\delta_w)+(D-K_w)(\delta_w-\delta_s)\big]}{\beta_3}, \tag{3.118c}$$

where β_3 is defined in (3.99c).

3.8.2 Scheme achieving rate-memory pair $(R_2^{(K)}, \mathcal{M}_2^{(K)})$

- If $K_s(1-\delta_z)-(D-K_w)(\delta_w-\delta_s) \le 0$, apply the same scheme as in the first case of Section 3.8.1 but with a change in the messages rates:

$$R^{(0)} = \frac{\alpha_2 F(1-\delta_w)}{K_w} \quad \text{and} \quad R^{(1)} = \frac{\alpha_2 F(\delta_w-\delta_s)}{K_s}. \tag{3.119}$$

Maximizing the rate achieves the rate-memory pair $(R_2^{(K)}, \mathcal{M}_2^{(K)})$ in (3.98i) and (3.98j), for

$$\alpha_1 = \frac{K_w(K_w-1)(\delta_w-\delta_s)(\delta_z-\delta_s)}{\beta_2}, \tag{3.120a}$$

$$\alpha_2 = \frac{2K_sK_w(1-\delta_w)(\delta_z-\delta_s)}{\beta_2}, \tag{3.120b}$$

$$\alpha_3 = \frac{2K_s^2(1-\delta_w)^2}{\beta_2}, \tag{3.120c}$$

where β_2 is defined in (3.99b).

- Otherwise, if $K_s(1-\delta_z) - (D-K_w)(\delta_w-\delta_s) > 0$, apply the following scheme:

  Preparations: Let $\alpha_1, \alpha_2, \alpha_3 \in [0,1]$, such that $\alpha_1+\alpha_2+\alpha_3 = 1$. For $d \in \mathcal{D}$, split each message into three sub-messages, such that

  $$W_d = \left[W_d^{(0)}, W_d^{(1)}, W_d^{(2)}\right], \tag{3.121}$$

  with rates

  $$R^{(0)} = \frac{\alpha_2 F(1-\delta_w)}{K_w}, \tag{3.122a}$$

  $$R^{(1)} = \frac{\alpha_2 F(\delta_w-\delta_s)}{K_s}, \tag{3.122b}$$

  $$R^{(2)} = \frac{\alpha_2 F\left[K_s(1-\delta_z) - (D-K_w)(\delta_w-\delta_s)\right]}{K_s(D-K)}. \tag{3.122c}$$

  Then, split every sub-message $W_d^{(1)}$ and $W_d^{(2)}$ into K_w sub-messages

  $$W_d^{(1)} = \left\{W_{d,i}^{(1)} : i \in \mathcal{K}_w\right\} \quad \text{and} \quad W_d^{(2)} = \left\{W_{d,i}^{(2)} : i \in \mathcal{K}_w\right\}, \tag{3.123}$$

  of rates $\frac{R^{(1)}}{K_w}$ and $\frac{R^{(2)}}{K_w}$, respectively.

  If $R^{(0)} > (D-K)R^{(1)}/K_w$, divide every sub-message $W_d^{(0)}$ into two further sub-messages, such that

  $$W_d^{(0)} = \left[W_d^{(0,1)}, W_d^{(0,2)}\right], \tag{3.124}$$

  with rates $(D-K)R^{(1)}/K_w$ and $R^{(0)} - (D-K)R^{(1)}/K_w$, respectively. Otherwise, $W_d^{(0,1)} = W_d^{(0)}$ has rate $R^{(0)}$ and $W_d^{(0,2)}$ has zero rate.

  Caching phase: For every weak receiver $i \in \mathcal{K}_w$, store the cache content

  $$V_i = \left\{W_{d,i}^{(1)} : d \in \{1,\dots,D\}\right\} \bigcup \left\{W_{d,i}^{(2)} : d \in \{1,\dots,D\}\right\}. \tag{3.125}$$

  Thus, the cache memory size is

  $$\mathcal{M}_2^{(K)} = D\frac{R^{(1)}+R^{(2)}}{K_w}. \tag{3.126}$$

  Delivery phase: The delivery phase is divided into three sub-phases of lengths $\alpha_1 n$, $\alpha_2 n$ and $\alpha_3 n$.

In the first sub-phase, the transmitter conveys messages $W_{d_i}^{(1)}$ and $W_{d_i}^{(2)}$ to every weak receiver $i \in \mathcal{K}_w$ by time-sharing over $\binom{K_w}{2}$ periods. In each period, it sends $W_{d_{i_1},i_2}^{(1)} \oplus W_{d_{i_2},i_1}^{(1)}$ and $W_{d_{i_1},i_2}^{(2)} \oplus W_{d_{i_2},i_1}^{(2)}$ to receivers i_1 and i_2.

In the second sub-phase, the transmitter conveys message $W_{d_i}^{(0)}$ to every weak receiver $i \in \mathcal{K}_w$ and message $W_{d_j}^{(1)}$ to every strong receiver $j \in \mathcal{K}_s$ by time-sharing over K_w periods. For each period, it generates a piggyback codebook $\mathcal{C}$ with $\lfloor 2^{nK_sR^{(1)}/K_w} \rfloor$ subcodebooks $\mathcal{C}\big(\tilde{W}\big)$ with $\lfloor 2^{nR^{(0)}} \rfloor$ codewords each.

For each period, it chooses a set of indices $\{j_1, \ldots, j_\iota\} \in \mathcal{D} \setminus \mathbf{d}$, where

$$\iota := \max\left\{1, \min\left\{\left\lceil \frac{K_w R^{(0)}}{R^{(1)}+R^{(2)}} \right\rceil, (D-K)\right\}\right\}, \tag{3.127}$$

and generates

$$W_{\text{XOR},i} = W_{d_i}^{(0,1)} \oplus \Big[W_{d_{j_1},i}^{(1)}, \ldots, W_{d_{j_\iota},i}^{(1)}, W_{d_{j_1},i}^{(2)}, \ldots, W_{d_{j_\iota},i}^{(2)}\Big]. \tag{3.128}$$

Then, it generates $W_i = \big[W_{\text{XOR},i}, W_{d_i}^{(0,2)}\big]$ and transmits the W_i-th codeword of the subcodebook $\mathcal{C}\big(\tilde{W}\big)$, with $\tilde{W} = \big\{W_{d_j,i}^{(1)} : j \in \mathcal{K}_s\big\}$.

In the third sub-phase, the transmitter conveys messages $W_{d_j}^{(0)}$ and $W_{d_j}^{(2)}$ to every strong receiver $j \in \mathcal{K}_s$ by time-sharing over K_s periods. In each period, it sends $W_{d_j}^{(0)}$ and $W_{d_j}^{(2)}$ to receiver j using a wiretap code without secret key.

Analysis: Only the weak receivers decode the first transmission sub-phase. Thus, it is reliably decoded if

$$\frac{\binom{K_w}{2}\frac{R^{(1)}+R^{(2)}}{K_w}}{F(1-\delta_w)} \le \alpha_1 \quad \Rightarrow \quad \frac{\frac{K_w-1}{2}\big(R^{(1)}+R^{(2)}\big)}{F(1-\delta_w)} \le \alpha_1. \tag{3.129}$$

Only the strong receivers decode the third transmission sub-phase. Thus, it is reliably decoded if

$$\frac{K_s\big(R^{(0)}+R^{(2)}\big)}{F(\delta_z-\delta_s)} \le \alpha_3. \tag{3.130}$$

Every weak receiver $i \in \mathcal{K}_w$ decodes only message $W_{d_i}^{(0)}$ transmitted in the second sub-phase, whereas every strong receiver $j \in \mathcal{K}_s$ decodes all the transmitted messages. Thus, the second sub-phase is reliably decoded whenever

$$\max\left\{\frac{K_w R^{(0)}}{F(1-\delta_w)}, \frac{K_w R^{(0)} + K_s R^{(1)}}{F(1-\delta_s)}\right\} \le \alpha_2. \tag{3.131}$$

Maximizing constraints (3.129), (3.130) and (3.131) achieves the rate-memory pair $(R_2^{(K)}, \mathcal{M}_2^{(K)})$ in (3.98m) and (3.98n), for

$$\alpha_1 = \frac{K_w(K_w-1)(\delta_z-\delta_s)(1-\delta_z-\delta_w+\delta_s)}{\beta_4}, \tag{3.132a}$$

$$\alpha_2 = \frac{2K_w(D-K)(1-\delta_w)(\delta_z-\delta_s)}{\beta_4}, \tag{3.132b}$$

$$\alpha_3 = \frac{2(1-\delta_w)\big[K_s(D-K)(1-\delta_w)+K_wK_s(1-\delta_z)-K_w(D-K_w)(\delta_w-\delta_s)\big]}{\beta_4}, \tag{3.132c}$$

where β_4 is defined in (3.99d).

3.8.3 Scheme achieving rate-memory pair $(R_3^{(K)}, \mathcal{M}_3^{(K)})$

Preparations: Let $\alpha \in [0,1]$. For $d \in \mathcal{D}$, split each message into two sub-messages, such that

$$W_d = \left[W_d^{(0)}, W_d^{(1)}\right], \tag{3.133}$$

with rates

$$R^{(0)} = \frac{\alpha F(1-\delta_z)}{K_w} \quad \text{and} \quad R^{(1)} = \frac{\alpha F(\delta_z-\delta_s)}{K_s}. \tag{3.134}$$

Caching phase: For every weak receiver $i \in \mathcal{K}_w$, store the D-tuple $W_1^{(1)}, \ldots, W_D^{(1)}$. Thus, the cache memory size is

$$\mathcal{M}_3^{(K)} = DR^{(1)}. \tag{3.135}$$

Delivery phase: The delivery phase is divided into two sub-phases of lengths αn and $(1-\alpha)n$.

In the first sub-phase, the transmitter conveys message $W_{d_i}^{(0)}$ to every weak receiver $i \in \mathcal{K}_w$ and message $W_{d_j}^{(1)}$ to every strong receiver $j \in \mathcal{K}_s$ following a similar scheme as the second sub-phase in Section 3.8.2.

In the second sub-phase, the transmitter conveys message $W_{d_j}^{(0)}$ to every strong receiver $j \in \mathcal{K}_s$ by time sharing over K_s periods. In each period, it sends $W_{d_j}^{(0)}$ to receiver j using a wiretap code.

Analysis: Every weak receiver $i \in \mathcal{K}_w$ decodes reliably its message $W_{d_i}^{(0)}$ conveyed in the first transmission sub-phase since $R^{(0)}$ rate is smaller than its channel capacity.

Strong receivers also decode reliably the first transmission sub-phase. They decode the second transmission sub-phase reliably whenever

$$R^{(0)} = \frac{\alpha(1-\delta_z)F}{K_w} \leq \frac{(1-\alpha)(\delta_z-\delta_s)F}{K_s}, \tag{3.136}$$

which is maximized for

$$\alpha = \frac{K_w(\delta_z-\delta_s)}{K_w(\delta_z-\delta_s)+K_s(1-\delta_z)}, \tag{3.137}$$

giving the rate-memory pair $(R_3^{(K)}, \mathcal{M}_3^{(K)})$ in (3.98c) and (3.98d).

3.9 General upper bound on the secure capacity-memory tradeoff

In this section, we provide the upper bound on the secure capacity-memory tradeoff of the more general setup with K_w weak receivers and K_s strong receivers where the K_w weak receivers have cache memories of size $\mathcal{M}$. This upper bound is stated in the following theorem:

Theorem 3.4 (Upper Bound on $C_s^{(K)}(\mathcal{M})$)**.** *The secure capacity-memory tradeoff $C_s^{(K)}(\mathcal{M})$ of the K-receiver channel with K_w weak receivers and K_s strong receivers with cache memories only at the weak receivers is upper bounded by the following K_w+2 conditions:*

$$C_s^{(K)}(\mathcal{M}) \leq F(\delta_z-\delta_w)+\mathcal{M}, \tag{3.138a}$$

$$C_s^{(K)}(\mathcal{M}) \leq F\frac{\delta_z-\delta_s}{K_s}, \tag{3.138b}$$

$$C_s^{(K)}(\mathcal{M}) \leq F\left(\frac{i}{1-\delta_w}+\frac{K_s}{1-\delta_s}\right)^{-1}+\frac{i\mathcal{M}}{D}, \quad i\in\{1,\ldots,K_w\}. \tag{3.138c}$$

Proof. The proof of constraint (3.138a) is similar to that of (3.62a) in Section 3.3. (3.138b) holds because strong receivers have no cache and their capacity cannot be larger than in the absence of weak receivers. Constraint (3.138a) follows from [9, Theorem 9] by ignoring the secrecy constraint. □

3.10 Examples

We illustrate in Figure 3.12 and 3.13 the lower and upper bounds derived for the general setup assuming a total of $K = 20$ users, a library size $D = 30$ and different channel parameters.

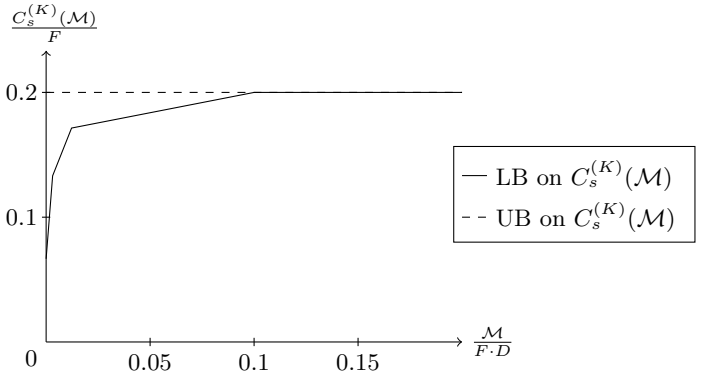

Figure 3.12: Lower and upper bounds on the secure capacity-memory tradeoff $C_s^{(K)}(\mathcal{M})$ under individual secrecy constraint for the K-user wiretap erasure BC with $\delta_w = 0.7$, $\delta_s = 0.2$, $\delta_z = 0.8$, $F = 5$, $D = 30$, $K_w = 5$ and $K_s = 15$.

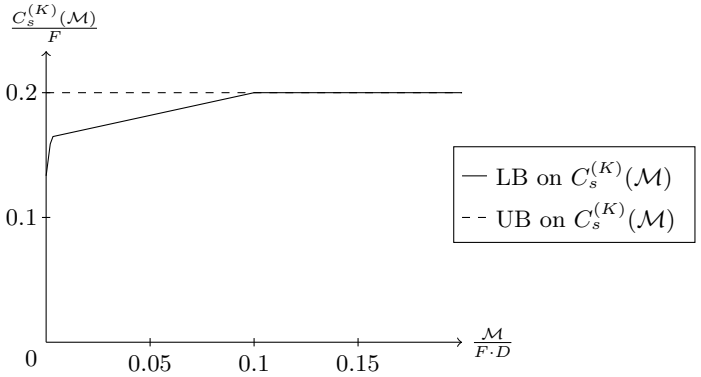

Figure 3.13: Lower and upper bounds on the secure capacity-memory tradeoff $C_s^{(K)}(\mathcal{M})$ under individual secrecy constraint for the K-user wiretap erasure BC with $\delta_w = 0.4$, $\delta_s = 0.2$, $\delta_z = 0.8$, $F = 5$, $D = 30$, $K_w = 5$ and $K_s = 15$.

3.11 Conclusion

In this chapter, we have derived lower and upper bounds on the securely achievable capacity-memory tradeoff of the two-user wiretap packet-erasure BC where the weaker receiver has a cache memory and where the eavesdropper is not allowed to learn any information about each of the delivered messages individually. The corresponding lower and upper bounds on the secure capacity-memory tradeoff are close for most scenarios and coincide when the weak receiver's cache memory exceeds a certain size. We have thus established the exact secure capacity-memory tradeoff for cache memory sizes above this threshold.

The lower bound under our secrecy constraint exhibits that cache memories provide larger gains than in the standard scenario without any secrecy constraints [9]. The reason being that the cache content can not only help to improve the communication rate, but also to make it more secure, for example, by means of a one-time pad.

For comparison, we have derived the lower bound on the secure capacity-memory tradeoff obtained by the best separate cache-channel coding scheme. We have found that the benefits of the cache memories are much more important when applying a joint cache-channel coding scheme that simultaneously leverages on the cache contents and the channel statistics.

Moreover, we have presented lower and upper bounds on the secure capacity-memory tradeoff when both receivers have cache memories of equal size. These bounds show that for a large range of parameters, the capacity-memory tradeoff is larger when all the cache memory is allocated to the weaker receiver instead of allocating half of it to each receiver. However, in contrast to the scenario without secrecy constraint, there exist situations where one receiver is weaker, but it is still better to allocate the cache memory symmetrically. The reason seems to be that a receiver can benefit from the cache memory at another receiver to increase its transmission rate, but it can only exploit its own cache memory to make it secure.

Finally, we have also computed the generalized lower and upper bounds for the general K-user scenario.

Chapter 4

Joint Secrecy in Caching Scenario

In this chapter, we consider the same cache-aided wiretap erasure BC as Chapter 3 under a joint secrecy constraint. We establish lower and upper bounds on the securely achievable capacity-memory tradeoff. To obtain the lower bound, we propose four different secure coding schemes that build on sophisticatedly combining wiretap coding, superposition coding and piggyback coding with random secret keys. Those keys are independent of all the data and are stored in the receivers' caches. The necessity for secret keys stems from the joint-secrecy constraint and had already been observed in [89, 90]. In such setups, the eavesdropper has no access to cache memories and is not allowed to learn anything about the set of all possible receivers' messages. In the previous chapter, we were able to use cached data as "secret keys" because only an individual secrecy-constraint had to be satisfied.

For the remaining of the chapter, we start by deriving our bounds and describing our coding schemes for the case of $K = 2$ users. We compare these bounds with the ones derived in Chapter 3 with the individual secrecy constraint. We also derive the lower and upper bounds on the secure capacity-memory tradeoff for the two-user scenario under a joint secrecy constraint with symmetric cache assignment and with asymmetric two-sided cache assignment. We compare these bounds with our scheme and determine the impact of the cache distribution on the secure capacity. Finally, we generalize our results for the K-user scenario.

4.1 Problem definition

We consider the same channel model described in Section 3.1 with a stronger joint secrecy constraint, i.e. we assume that the communication is secure if the eavesdropper does

not learn any information about the library messages from its outputs Z^n:

$$\lim_{n\to\infty} \frac{1}{n} I(W_1, \ldots, W_D; Z^n) < \epsilon. \tag{4.1}$$

In order to satisfy this secrecy constraint, the transmitter is given access to a source of randomness θ in addition to its access to the message library. θ is defined over some alphabet Θ. As a result, the caching and encoding functions are changed accordingly.

The caching content V_i at each weak receiver $i \in K_w$ becomes

$$V_i := g_i(W_1, \ldots, W_D, \theta), \qquad i \in \mathcal{K}_w \tag{4.2}$$

for some caching function

$$g_i : \left\{1, \ldots, \lfloor 2^{nR_s} \rfloor\right\}^D \times \Theta \to \mathcal{V}. \tag{4.3}$$

The transmitter will produce its channel inputs as

$$X^n := f_{\mathbf{d}}(W_1, \ldots, W_D, \theta), \tag{4.4}$$

for some encoding function

$$f_{\mathbf{d}} : \left\{1, \ldots, \lfloor 2^{nR_s} \rfloor\right\}^D \times \Theta \to \mathcal{X}^n. \tag{4.5}$$

Finally, the securely achievable rate-memory pairs are defined as follows.

Definition 4.1. A rate-memory pair $(R_s, \mathcal{M})$ is *securely achievable* if for every $\epsilon > 0$ and sufficiently large blocklength n, there exist caching, encoding, and decoding functions as in (4.3), (4.5), (3.17) and (3.19) so that

$$\mathrm{P}_e^{\mathrm{Worst}} \leq \epsilon \quad \text{and} \quad \frac{1}{n} I(W_1, \ldots, W_D; Z^n) < \epsilon. \tag{4.6}$$

4.2 Lower bound on the secure capacity-memory tradeoff under one-sided cache assignment for the two-user scenario

In this section, we consider only one weak and one strong receiver with erasure probabilities δ_1 and δ_2, respectively. We give the lower bound on the secure capacity-memory tradeoff $C_s(\mathcal{M})$ as defined in Definition 3.3 and prove its achievability.

Consider the six rate-memory pairs:

- $R_0 := \dfrac{(\delta_z - \delta_1)(\delta_z - \delta_2)}{2\delta_z - \delta_1 - \delta_2} F,$ (4.7a)

$$\mathcal{M}_0 := 0; \tag{4.7b}$$

- $$R_1 := \frac{(1-\delta_1)(\delta_z-\delta_2)}{1+\delta_z-\delta_1-\delta_2}F, \tag{4.7c}$$
$$\mathcal{M}_1 := \frac{(1-\delta_z)(\delta_z-\delta_2)}{1+\delta_z-\delta_1-\delta_2}F; \tag{4.7d}$$

- $$R_2 := (1-\delta_2)\min\left\{\frac{\delta_z-\delta_1}{1-\delta_1}, \frac{1-\delta_1}{2-\delta_1-\delta_2}\right\}F, \tag{4.7e}$$
$$\mathcal{M}_2 := (1-\delta_z)F; \tag{4.7f}$$

- $$R_3 := \frac{(1-\delta_2)(\delta_z-\delta_2)}{1+\delta_z-\delta_1-\delta_2}F, \tag{4.7g}$$
$$\mathcal{M}_3 := \frac{(\delta_z-\delta_2)\big[(\delta_1-\delta_2)D+(1-\delta_z)\big]}{1+\delta_z-\delta_1-\delta_2}F; \tag{4.7h}$$

- $$R_4 := (\delta_z-\delta_2)F, \tag{4.7i}$$
$$\mathcal{M}_4 := \frac{(\delta_z-\delta_2)\big[(\delta_z-\delta_2)D+(1-\delta_z)\big]}{1-\delta_2}F; \tag{4.7j}$$

- $$R_5 := (\delta_z-\delta_2)F, \tag{4.7k}$$
$$\mathcal{M}_5 := D(\delta_z-\delta_2)F. \tag{4.7l}$$

Theorem 4.1 (Lower Bound on $C_s(\mathcal{M})$)**.** *The upper convex hull of the six rate-memory pairs* $\big\{(R_\ell, \mathcal{M}_\ell);\ \ell \in \{0,1,\ldots,5\}\big\}$ *in* (4.7) *lower bounds the secure capacity-memory tradeoff under one-sided cache assignment:*

$$C_s(\mathcal{M}) \geq \text{upper hull}\big\{(R_\ell, \mathcal{M}_\ell)\colon\ \ell = 0,\ldots,5\big\}. \tag{4.8}$$

Proof. It suffices to prove the achievability of the six rate-memory pairs $\{(R_\ell, \mathcal{M}_\ell)\colon\ \ell = 0,\ldots,5\}$. The achievability of the upper convex hull follows by time/memory sharing arguments as in [10].

The achievability of the pairs $(R_0, \mathcal{M}_0)$ and $(R_5, \mathcal{M}_5)$ follows from Remarks 3.2 and 3.3, respectively.

The achievability of the remaining rate-memory pairs is outlined in the next subsections. □

Remark 4.1. For all $\mathcal{M} \geq \mathcal{M}_5$, the securely achievable capacity is $C_s(\mathcal{M}) = R_5$. This can be deduced from Remark 3.4.

4.2.1 Scheme achieving rate-memory pair $(R_1, \mathcal{M}_1)$

Preparations: Let $\alpha \in [0,1]$. Generate a random key K_1 of rate $\alpha(1-\delta_z)F$.

Caching phase: Store K_1 in receiver 1's cache memory. Thus, the cache memory size is

$$\mathcal{M}_1 = \alpha(1-\delta_z)F. \tag{4.9}$$

Delivery phase: The delivery phase is divided into two periods of lengths αn and $(1-\alpha)n$. In the first period, the transmitter sends W_{d_1} to receiver 1 using a wiretap code with secret key K_1 [113],[11, (22.7)]. In the second period, the transmitter sends W_{d_2} to receiver 2 using a wiretap code without secret key.

Analysis: Receiver 1 decodes message W_{d_1} sent in the first transmission period. The total decoded rate should not exceed its capacity, hence

$$R_1 \leq \alpha(1-\delta_1)F. \tag{4.10}$$

Receiver 2 decodes W_{d_2} conveyed in the second transmission period. It can reliably decode its message if its rate satisfies,

$$R_1 \leq (1-\alpha)(\delta_z - \delta_2)F. \tag{4.11}$$

Thus, R_1 is maximized when (4.10) and (4.11) are equal. The equality is obtained when

$$\alpha = \frac{\delta_z - \delta_2}{1 + \delta_z - \delta_1 - \delta_2}. \tag{4.12}$$

Replacing α by its value in (4.10) and (4.9) gives the rate-memory pair $(R_1, \mathcal{M}_1)$ in (4.7c) and (4.7d).

Moreover, considering a key's rate larger than the channel capacity at the eavesdropper guarantees the security of the transmitted message.

4.2.2 Scheme achieving rate-memory pair $(R_2, \mathcal{M}_2)$

Preparations: Let $\alpha \in [0,1]$ and $\gamma \in [0,1]$. For $d \in \mathcal{D}$, split each message W_d into two sub-messages, such that

$$W_d = \left[W_d^{(0)}, W_d^{\oplus}\right],$$

with rates $R^{(0)} = \alpha(1-\delta_1)F$, $R^{\oplus} = (1-\alpha)(1-\delta_z)F$, respectively.

Then, generate two random keys K_1 and K_2 of rates $\alpha(1-\delta_z)F$ and $(1-\alpha)(1-\delta_z)F$.

Caching phase: Store K_1 and K_2 in receiver 1's cache. Thus, the cache memory size is

$$\mathcal{M}_2 = (1-\delta_z)F. \tag{4.13}$$

Delivery phase: The delivery phase is divided into two periods of lengths αn and $(1-\alpha)n$. In the first period, the transmitter sends message $W_{d_1}^{(0)}$ to receiver 1 using a wiretap code with secret key K_1.

For the communication in the second period, generate a superposition codebook with a cloud center that contains $2^{n(1-\alpha)(1-\delta_z)F}$ codewords, and with each of the satellite codebooks containing 2^{nR_2} codewords. The transmitter encodes $W_{d_1}^{\oplus} \oplus K_2$ into the cloud center and W_{d_2} into the satellite.

Analysis: Receiver 1 decodes the first transmission period and only message $W_{d_1}^{\oplus} \oplus K_2$ from the second one. Hence, receiver 1 decodes reliably its messages whenever

$$R_2 \leq \alpha(1-\delta_1)F + (1-\alpha)(1-\delta_1)F \times (1-\gamma). \tag{4.14}$$

Receiver 2 decodes only the second transmission period. It decodes reliably its messages whenever

$$R_2 \leq (1-\alpha)(1-\delta_2)F \times \gamma. \tag{4.15}$$

From the second transmission period, receiver 1 decodes only message $W_{d_1}^{\oplus} \oplus K_2$ from the cloud center, thus $R^{\oplus}$ should satisfy

$$R^{\oplus} = (1-\alpha)(1-\delta_z)F \leq (1-\alpha)(1-\delta_1)F \times (1-\gamma). \tag{4.16}$$

Equality in (4.16) is obtained for

$$\gamma = \frac{\delta_z - \delta_1}{1-\delta_1}. \tag{4.17}$$

For this γ value, conditions (4.14) and (4.15) become

$$R_2 \leq \alpha(1-\delta_1) + (1-\alpha)(1-\delta_z)F, \tag{4.18a}$$

$$R_2 \leq (1-\alpha)\frac{(\delta_z-\delta_1)(1-\delta_2)}{1-\delta_1}F. \tag{4.18b}$$

If $(1-\delta_z)(1-\delta_1) \leq (1-\delta_2)(\delta_z-\delta_1)$, equality of (4.18a) and (4.18b) is achieved for

$$\alpha = \frac{(1-\delta_2)(\delta_z-\delta_1) - (1-\delta_z)(1-\delta_1)}{(\delta_z-\delta_1)(2-\delta_1-\delta_2)}, \tag{4.19}$$

yielding a rate

$$R_2 = \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F. \tag{4.20}$$

Otherwise, (4.18b) gives a smaller R_2 than (4.18a) $\forall\ \alpha \in [0,1]$ and the maximal rate, achieved for $\alpha = 0$, is

$$R_2 = F\frac{(\delta_z - \delta_1)(1-\delta_2)}{1-\delta_1}. \tag{4.21}$$

Secrecy of the proposed scheme can be proved by following the steps in Section 3.2.8, where in the proof of Lemma 3.1 we use the fact that the entire superposition codebook contains $2^{n(1-\alpha)(1-\delta_z)F}$ codewords that are compatible with a given satellite message W_{d_2}.

Combining (4.13), (4.20) and (4.21) gives the desired rate-memory pair $(R_2, \mathcal{M}_2)$ in (4.7e) and (4.7f).

4.2.3 Scheme achieving rate-memory pair $(R_3, \mathcal{M}_3)$

Preparations: Let $\alpha \in [0,1]$. For $d \in \mathcal{D}$, split each message W_d into three sub-messages, such that

$$W_d = \left[W_d^{(0)}, W_d^{(1)}, W_d^{\oplus}\right], \tag{4.22}$$

with rates $R^{(0)} = \alpha(\delta_z - \delta_1)F$, $R^{(1)} = \alpha(\delta_1 - \delta_2)F$ and $R^{\oplus} = \alpha(1-\delta_z)F$, respectively.

Then, generate a random key K_1 of rate $\alpha(1-\delta_z)F$.

Generate a piggyback codebook $\mathcal{C}_1$ with $\Gamma_1 := \lfloor 2^{n\alpha(1-\delta_z)F}\rfloor \cdot \lfloor 2^{n\alpha(\delta_z-\delta_1)F}\rfloor \cdot \lfloor 2^{n\alpha(\delta_1-\delta_2)F}\rfloor$ codewords of length αn,

$$\mathcal{C}_1 := \left\{X_1^{(\alpha n)}(l_1)\right\}_{l_1=1}^{\Gamma_1}, \tag{4.23}$$

by drawing each entry of each codeword at random according to a Bernoulli-1/2 distribution independently of all the other entries.

The codebook is partitioned into $\lfloor 2^{n\alpha(\delta_z-\delta_1)F}\rfloor \cdot \lfloor 2^{n\alpha(\delta_1-\delta_2)F}\rfloor$ subcodebooks (bins) each with $\lfloor 2^{n\alpha(1-\delta_z)F}\rfloor$ codewords. The subcodebooks are arranged into an array with $\lfloor 2^{n\alpha(\delta_z-\delta_1)F}\rfloor$ rows and $\lfloor 2^{n\alpha(\delta_1-\delta_2)F}\rfloor$ columns.

Caching phase: Store K_1 and the D-tuple $W_1^{(1)}, \ldots, W_D^{(1)}$ in receiver 1's cache. Thus, the cache memory size is

$$\mathcal{M}_3 = \alpha(1-\delta_z)F + D\cdot\alpha(\delta_1-\delta_2)F. \tag{4.24}$$

Delivery phase: The delivery phase is divided into two periods of lengths αn and $(1-\alpha)n$. In the first period, the transmitter conveys messages $W_{d_1}^{(0)}$ and $W_{d_1}^{\oplus}$ to receiver 1 and $W_{d_2}^{(1)}$ to receiver 2. It generates $W_{\text{XOR}} = W_{d_1}^{\oplus} \oplus K_1$ and transmits the W_{XOR}-th codeword of

the subcodebook $\mathcal{C}_1(W_{d_1}^{(0)}, W_{d_2}^{(1)})$ over the channel. In the second period, it sends message $W_{d_2}^{(0),\oplus} = [W_{d_2}^{(0)}, W_{d_2}^{\oplus}]$ to receiver 2 using a wiretap code without secret key.

Decoding at receiver 1: Receiver 1 retrieves message $W_{d_2}^{(1)}$ from its cache memory, and considers its outputs $y_1^{\alpha n}$ from the first period. It looks for a unique index-pair $(\hat{w}_{\text{XOR}}, \hat{w}_{d_1}^{(0)}) \in \big[1 : \lfloor 2^{n\alpha(1-\delta_z)F} \rfloor\big] \times \big[1 : \lfloor 2^{n\alpha(\delta_z-\delta_1)F} \rfloor\big]$ so that the $\hat{w}_{\text{XOR}}$-th codeword in subcodebook $\mathcal{C}_1(\hat{w}_{d_1}^{(0)}, W_{d_2}^{(1)})$, which we denote by $x_1^{(\alpha n)}\big(\hat{w}_{\text{XOR}}, \hat{w}_{d_1}^{(0)}, W_{d_2}^{(1)}\big)$, is jointly typical with its observed outputs:

$$\Big(x_1^{(\alpha n)}\big(\hat{w}_{\text{XOR}}, \hat{w}_{d_1}^{(0)}, W_{d_2}^{(1)}\big),\ y_1^{\alpha n}\Big) \in \mathcal{T}_\epsilon^{(\alpha n)}\big(p_X \cdot p_{Y_1|X}\big), \tag{4.25}$$

where p_X stands for the Bernoulli-1/2 distribution, $p_{Y_1|X}$ the channel law to receiver 1, and $\mathcal{T}_\epsilon^{(\alpha n)}$ the typical set [11].

If the desired unique pair of indexes $(\hat{w}_{\text{XOR}}, \hat{w}_{d_1}^{(0)})$ does not exist, receiver 1 declares an error. Otherwise, if the pair exists, receiver 1 retrieves the key K_1 from its cache memory and generates

$$\hat{w}_{d_1}^{\oplus} = \hat{w}_{\text{XOR}} \oplus K_1. \tag{4.26}$$

It finally retrieves $W_{d_1}^{(1)}$ from its cache memory and declares the tuple

$$\hat{w}_1 = \big(\hat{w}_{d_1}^{\oplus}, \hat{w}_{d_1}^{(0)}, W_{d_1}^{(1)}\big). \tag{4.27}$$

Decoding at receiver 2: Receiver 2 decodes $W_{d_2}^{(1)}$ based on its outputs $y_2^{\alpha n}$ in the first period, and it decodes $W_{d_2}^{(0),\oplus}$ based on its outputs $y_2^{(1-\alpha)n}$ in the second period.

It looks for a unique triple $(\hat{w}_{\text{XOR}}, \hat{w}_{d_1}^{(0)}, \hat{w}_{d_2}^{(1)})$ so that

$$\Big(x_1^{(\alpha n)}\big(\hat{w}_{\text{XOR}}, \hat{w}_{d_1}^{(0)}, \hat{w}_{d_2}^{(1)}\big), y_2^{\alpha n}\Big) \in \mathcal{T}_\epsilon^{(\alpha n)}\big(p_X \cdot p_{Y_2|X}\big). \tag{4.28}$$

Then, it looks for a unique pair $(\hat{w}_{d_2}^{(0),\oplus}, l)$ such that

$$\Big(x_2^{((1-\alpha)n)}\big(\hat{w}_{d_2}^{(0),\oplus}, l\big), y_2^{(1-\alpha)n}\Big) \in \mathcal{T}_\epsilon^{((1-\alpha)n)}\big(p_X \cdot p_{Y_2|X}\big). \tag{4.29}$$

If the desired triple and pair exist, receiver 2 declares

$$\hat{w}_2 = \big(\hat{w}_{d_2}^{(0)}, \hat{w}_{d_2}^{(1)}, \hat{w}_{d_2}^{\oplus}\big). \tag{4.30}$$

Otherwise it declares an error.

Analysis: Receiver 1 reliably decodes $W_{d_1}^{(0)}$ and $W_{d_1}^{\oplus}$ conveyed in the first transmission period whenever

$$R^{(0)} + R^{\oplus} = R_3 - R^{(1)} \leq \alpha(1-\delta_1)F \Rightarrow R_3 \leq \alpha(1-\delta_2)F. \tag{4.31}$$

Receiver 2 reliably decodes both transmission periods whenever

$$\begin{aligned} R_3 + R^{(0)} + R^{\oplus} &\leq \alpha(1-\delta_2)F + (1-\alpha)(\delta_z - \delta_2)F \\ \Rightarrow R_3 &\leq \alpha(\delta_1 - \delta_2)F + (1-\alpha)(\delta_z - \delta_2)F. \end{aligned} \tag{4.32}$$

Equality of (4.31) and (4.32) is obtained for

$$\alpha = \frac{\delta_z - \delta_2}{1 + \delta_z - \delta_1 - \delta_2}. \tag{4.33}$$

For this α value, the rate-memory pair $(R_3, \mathcal{M}_3)$ in (4.7g) and (4.7h) is achievable.

4.2.4 Scheme achieving rate-memory pair $(R_4, \mathcal{M}_4)$

We apply the same coding scheme described for $(R_3, \mathcal{M}_3)$ with the following changes: we cancel the rate of $W_d^{(0)}$, i.e. $R^{(0)} = 0$ and change $W_d^{(1)}$ rate to $R^{(1)} = \alpha(\delta_z - \delta_2)F$. We keep $R^{\oplus} = \alpha(1-\delta_z)F$. In this case, the cache memory becomes

$$\mathcal{M}_4 = \alpha(1-\delta_z)F + D \cdot \alpha(\delta_z - \delta_2)F. \tag{4.34}$$

Receiver 1 reliably decodes the message $W_{d_1}^{\oplus}$ of rate $R^{\oplus} < \alpha(1-\delta_1)F$. Receiver 2 decodes the first transmission phase without error since $R^{\oplus} + R^{(1)} = \alpha(1-\delta_2)F$. As for the second phase, the rate of $W_d^{\oplus}$ should satisfy

$$\alpha(1-\delta_z)F \leq (1-\alpha)(\delta_z - \delta_2)F. \tag{4.35}$$

Thus, the value of α that maximizes the transmission rate is

$$\alpha = \frac{\delta_z - \delta_2}{1 - \delta_2}, \tag{4.36}$$

giving the rate-memory pair $(R_4, \mathcal{M}_4)$ in (4.7i) and (4.7j).

4.3 Upper bound on the secure capacity-memory tradeoff under one-sided cache assignment for the two-user scenario

In this section, we derive the upper bound on the secure capacity-memory tradeoff for the two-receiver channel. This upper bound is stated in the following theorem:

Theorem 4.2 (Upper Bound on $C_s(\mathcal{M})$)**.** *The secure capacity-memory tradeoff $C_s(\mathcal{M})$ of the two-user wiretap erasure BC with cache memory $\mathcal{M}$ only at the weaker receiver under the joint secrecy constraint is upper bounded by the following three conditions:*

$$C_s(\mathcal{M}) \leq \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F + \frac{\delta_z-\delta_2}{2\delta_z-\delta_1-\delta_2}\mathcal{M}, \tag{4.37a}$$

$$C_s(\mathcal{M}) \leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F + \frac{\mathcal{M}}{D}, \tag{4.37b}$$

$$C_s(\mathcal{M}) \leq (\delta_z-\delta_2)F. \tag{4.37c}$$

4.3.1 Proof of the upper bound

Bound (4.37b) follows from [9] and by ignoring the secrecy constraint. Bound (4.37c) holds because receiver 2 has no cache, and its rate cannot be larger than in the absence of receiver 1.

Bound (4.37a) is proved as follows. For each blocklength n, we fix caching, encoding and decoding functions as in (4.3), (4.5), (3.17) and (3.19) so that both the probability of worst-case error and the secrecy leakage satisfy:

$$\mathrm{P}_e^{\mathrm{Worst}} \xrightarrow[n\to\infty]{} 0 \qquad \text{and} \qquad \frac{1}{n}I(W_1,\ldots,W_D;Z^n) \xrightarrow[n\to\infty]{} 0.$$

By Fano's inequality, there exists a sequence of real numbers $\{\epsilon_n\}_{n=1}^{\infty}$ with

$$\frac{\epsilon_n}{n} \xrightarrow[n\to\infty]{} 0,$$

so that

$$H(W_{d_1}|Y_1^n,V_1) \leq \frac{\epsilon_n}{2}. \tag{4.38}$$

Thus,

$$\begin{aligned}
nR_s &= H(W_{d_1}) = H(W_{d_1}|Z^n) + I(W_{d_1};Z^n)\\
&\leq H(W_{d_1}|Z^n) + \frac{\epsilon_n}{2}\\
&\leq I(W_{d_1};Y_1^n,V_1) - I(W_{d_1};Z^n) + H(W_{d_1}|Y_1^n,V_1) + \frac{\epsilon_n}{2}\\
&\leq I(W_{d_1};Y_1^n,V_1) - I(W_{d_1};Z^n) + \epsilon_n\\
&\leq I(W_{d_1};Y_1^n|V_1) - I(W_{d_1};Z^n|V_1) + I(W_{d_1};V_1|Z^n) + \epsilon_n\\
&\overset{(a)}{=} \sum_{i=1}^{n}\big[I(W_{d_1};Y_{1,i}|V_1,Y_1^{i-1},Z_{i+1}^n) - I(W_{d_1};Z_i|V_1,Y_1^{i-1},Z_{i+1}^n)\big] + I(W_{d_1};V_1|Z^n) + \epsilon_n\\
&\overset{(b)}{\leq} \sum_{i=1}^{n}\big[I(W_{d_1};Y_{1,i}|V_1,Y_1^{i-1},Z_{i+1}^n) - I(W_{d_1};Z_i|V_1,Y_1^{i-1},Z_{i+1}^n)\big]
\end{aligned}$$

$$
\begin{aligned}
&+\sum_{i=1}^{n}\left[I(V_1,Y_1^{i-1},Z_{i+1}^n;Y_{1,i})-I(V_1,Y_1^{i-1},Z_{i+1}^n;Z_i)\right]+I(W_{d_1};V_1|Z^n)+\epsilon_n\\
&\overset{(c)}{\leq}\sum_{i=1}^{n}\left[I(W_{d_1},V_1,Y_1^{i-1},Z_{i+1}^n;Y_{1,i})-I(W_{d_1},V_1,Y_1^{i-1},Z_{i+1}^n;Z_i)\right]+I(W_{d_1};V_1|Z^n)+\epsilon_n\\
&\quad+\sum_{i=1}^{n}\left[I(Y_2^{i-1};Y_{1,i}|W_{d_1},V_1,Y_1^{i-1},Z_{i+1}^n)-I(Y_2^{i-1};Z_i|W_{d_1},V_1,Y_1^{i-1},Z_{i+1}^n)\right]\\
&\overset{(d)}{\leq}\sum_{i=1}^{n}\left[I(W_{d_1},V_1,Y_1^{i-1},Y_2^{i-1},Z_{i+1}^n;Y_{1,i})-I(W_{d_1},V_1,Y_1^{i-1},Y_2^{i-1},Z_{i+1}^n;Z_i)\right]\\
&+n\mathcal{M}+\epsilon_n\\
&\overset{(e)}{\leq}\sum_{i=1}^{n}\left[I(W_{d_1},V_1,Y_2^{i-1},Z_{i+1}^n;Y_{1,i})-I(W_{d_1},V_1,Y_2^{i-1},Z_{i+1}^n;Z_i)\right]+n\mathcal{M}+\epsilon_n, \quad (4.39)
\end{aligned}
$$

where (a) follows by the chain rule of mutual information and by applying Csiszar's sum-identity [11, p. 25]; (b) and (c) hold because the eavesdropper is degraded with respect to receiver 1 and thus, for each $i \in \{1,\dots,n\}$:

$$I(V_1,Y_1^{i-1},Z_{i+1}^n;Y_{1,i})-I(V_1,Y_1^{i-1},Z_{i+1}^n;Z_i)\geq 0;$$

and

$$I(Y_2^{i-1};Y_{1,i}|W_{d_1},V_1,Y_1^{i-1},Z_{i+1}^n)-I(Y_2^{i-1};Z_i|W_{d_1},V_1,Y_1^{i-1},Z_{i+1}^n)\geq 0;$$

(d) holds because $I(W_{d_1};V_1|Z^n)$ is limited by the entropy of V_1 which cannot exceed $n\mathcal{M}$; and finally (e) holds because receiver 1 is degraded with respect to receiver 2 and thus the following Markov chain holds:

$$(V_1,W_{d_1},Z_{i+1}^n,Y_{1,i},Z_i)\to Y_2^{i-1}\to Y_1^{i-1}.$$

Let Q be a random variable uniform over $\{1,\dots,n\}$ and independent of all previously defined random variables. We define the random variables

$$Y_1 := Y_{1,Q}, \quad (4.40)$$
$$Z := Z_Q, \quad (4.41)$$
$$U_1 := (W_{d_1},V_1,Y_2^{Q-1},Z_{Q+1}^n). \quad (4.42)$$

Dividing by n, we can rewrite constraint (4.39) as

$$R_s \leq I(U_1;Y_1|Q)-I(U_1;Z|Q)+\mathcal{M}+\frac{\epsilon_n}{n}. \quad (4.43)$$

Accounting for receiver 2, we derive in a similar way:

$$
\begin{aligned}
2nR_s &= H(W_{d_1},W_{d_2})\\
&= H(W_{d_1},W_{d_2}|Z^n)+I(W_{d_1},W_{d_2};Z^n)
\end{aligned}
$$

$$
\begin{aligned}
&\leq H(W_{d_1}, W_{d_2}|Z^n) + \frac{\epsilon_n}{2} \\
&\overset{(a)}{\leq} I(W_{d_1}; Y_1^n, V_1) + I(W_{d_2}; Y_2^n, V_1|W_{d_1}) - I(W_{d_1}, W_{d_2}; Z^n) + \epsilon_n \\
&\overset{(b)}{=} I(W_{d_1}; Y_1^n, V_1) - I(W_{d_1}; Z^n) + I(W_{d_2}; Y_2^n, V_1|W_{d_1}) - I(W_{d_2}; Z^n|W_{d_1}) + \epsilon_n \\
&= I(W_{d_1}; Y_1^n|V_1) - I(W_{d_1}; Z^n|V_1) + I(W_{d_1}; V_1|Z^n) + I(W_{d_2}; Y_2^n|V_1, W_{d_1}) \\
&\quad - I(W_{d_2}; Z^n|V_1, W_{d_1}) + I(W_{d_2}; V_1|Z^n, W_{d_1}) + \epsilon_n \\
&\overset{(c)}{=} I(W_{d_1}; Y_1^n|V_1) - I(W_{d_1}; Z^n|V_1) + I(W_{d_2}; Y_2^n|V_1, W_{d_1}) \\
&\quad - I(W_{d_2}; Z^n|V_1, W_{d_1}) + n\mathcal{M} + \epsilon_n,
\end{aligned} \tag{4.44}
$$

where (a) follows by Fano's inequality and the chain rule of mutual information; (b) follows from the chain rule of mutual information; and (c) holds because

$$I(W_{d_1}; V_1|Z^n) + I(W_{d_2}; V_1|Z^n, W_{d_1}) = I(W_{d_2}, W_{d_1}; V_1|Z^n) \leq n\mathcal{M}$$

since $I(W_{d_2}, W_{d_1}; V_1|Z^n)$ is limited by the entropy of V_1 which cannot exceed $n\mathcal{M}$.

Dividing by n, (4.44) becomes

$$
\begin{aligned}
2R_s \leq \frac{1}{n}\Big[I(W_{d_1}; Y_1^n|V_1) - I(W_{d_1}; Z^n|V_1)\Big] + \mathcal{M} \\
+ \frac{1}{n}\Big[I(W_{d_2}; Y_2^n|V_1, W_{d_1}) - I(W_{d_2}; Z^n|V_1, W_{d_1})\Big] + \frac{\epsilon_n}{n},
\end{aligned} \tag{4.45}
$$

and following (4.39)–(4.42), we can prove that

$$\frac{1}{n}\Big[I(W_{d_1}; Y_1^n|V_1) - I(W_{d_1}; Z^n|V_1)\Big] \leq I(U_1; Y_1|Q) - I(U_1; Z|Q). \tag{4.46}$$

and

$$
\begin{aligned}
&\frac{1}{n}\Big[I(W_{d_2}; Y_2^n|V_1, W_{d_1}) - I(W_{d_2}; Z^n|V_1, W_{d_1})\Big] \\
&\overset{(a)}{=} \frac{1}{n}\sum_{i=1}^{n}\Big[I(W_{d_2}; Y_{2,i}|V_1, W_{d_1}, Y_2^{i-1}, Z_{i+1}^n) - I(W_{d_2}; Z_i|V_1, W_{d_1}, Y_2^{i-1}, Z_{i+1}^n)\Big] \\
&\overset{(b)}{\leq} \frac{1}{n}\sum_{i=1}^{n}\Big[I(W_{d_2}; Y_{2,i}|V_1, W_{d_1}, Y_2^{i-1}, Z_{i+1}^n) - I(W_{d_2}; Z_i|V_1, W_{d_1}, Y_2^{i-1}, Z_{i+1}^n)\Big] \\
&\quad + \frac{1}{n}\sum_{i=1}^{n}\Big[I(X_i; Y_{2,i}|V_1, W_{d_1}, W_{d_2}, Y_2^{i-1}, Z_{i+1}^n) - I(X_i; Z_i|V_1, W_{d_1}, W_{d_2}, Y_2^{i-1}, Z_{i+1}^n)\Big] \\
&= \frac{1}{n}\sum_{i=1}^{n}\Big[I(W_{d_2}, X_i; Y_{2,i}|V_1, W_{d_1}, Y_2^{i-1}, Z_{i+1}^n) - I(W_{d_2}, X_i; Z_i|V_1, W_{d_1}, Y_2^{i-1}, Z_{i+1}^n)\Big] \\
&\overset{(c)}{=} I(X; Y_2|Q, U_1) - I(X; Z|Q, U_1),
\end{aligned} \tag{4.47}
$$

where (a) follows by the chain rule of mutual information and by applying Csiszar's sum-identity; (b) hold because the eavesdropper is degraded with respect to receiver 2; and (c) follows from the Markov chain:

$$W_{d_2} \to X_i \to (Y_{2,i}, Z_i). \tag{4.48}$$

We define then the random variables:

$$X := X_Q, \tag{4.49}$$

$$Y_2 := Y_{2,Q}. \tag{4.50}$$

Combining (4.45), (4.46) and (4.47) gives

$$2R_s \leq I(U_1;Y_1|Q) - I(U_1;Z|Q) + I(X;Y_2|Q,U_1) - I(X;Z|Q,U_1) + \mathcal{M} + \frac{\epsilon_n}{n}. \tag{4.51}$$

Let $n \to \infty$. We derive constraint (4.43) and (4.51) for the erasure BC as follows:

$$\begin{aligned} R_s &\leq I(U_1;Y_1|Q) - I(U_1;Z|Q) + \mathcal{M} \\ &= I(U_1,X;Y_1|Q) - I(X;Y_1|U_1,Q) - I(U_1,X;Z|Q) + I(X;Z|U_1,Q) + \mathcal{M} \\ &\overset{(a)}{=} I(X;Y_1|Q) - I(X;Y_1|U_1,Q) - I(X;Z|Q) + I(X;Z|U_1,Q) + \mathcal{M} \\ &= (\delta_z - \delta_1)\big[H(X|Q) - H(X|U_1,Q)\big] + \mathcal{M} \\ &\overset{(a)}{=} (\delta_z - \delta_1)(\beta - \alpha) + \mathcal{M}, \end{aligned} \tag{4.52}$$

where (a) holds because of the Markov chain $U_1 \to X \to (Z, Y_1)$ and (b) follows by defining $\alpha := H(X|U_1,Q)$ and $\beta := H(X|Q)$, such that $0 \leq \beta \leq \alpha \leq F$.

$$\begin{aligned} R_s &\leq \frac{1}{2}\Big[I(U_1;Y_1|Q) - I(U_1;Z|Q) + I(X;Y_2|Q,U_1) - I(X;Z|Q,U_1) + \mathcal{M}\Big] \\ &\leq \frac{1}{2}\Big[(\delta_z - \delta_1)\big[H(X|Q) - H(X|U_1,Q)\big] + (\delta_z - \delta_2)H(X|U_1,Q) + \mathcal{M}\Big] \\ &= \frac{1}{2}\Big[(\delta_z - \delta_1)(\beta - \alpha) + (\delta_z - \delta_2)\alpha + \mathcal{M}\Big]. \end{aligned} \tag{4.53}$$

If R_s is securely achievable, there exist α and β satisfying (4.52) and (4.53). Since both constraints increase with β, $\beta = F$ is optimal. This is obtained by choosing $Q = \emptyset$ and X uniform over $\{1, \ldots, 2^F\}$. Optimizing the minimum of bounds (4.52) and (4.53) over $\alpha \in [0, F]$ gives the desired bound (4.37a) in Theorem 4.2.

4.4 Lower bound on the secure capacity-memory tradeoff under symmetric cache assignment for the two-user scenario

As in the previous chapter, here we also study the symmetric cache assignment for the two-user wiretap BC channel under a joint secrecy constraint. We compute the lower

bound on $C_{s,\text{Sym}}(\mathcal{M})$ when each receiver has a cache memory of size $\mathcal{M}/2$.

Consider the three rate-memory pairs:

$$\bullet \quad R_{0,\text{Sym}} := \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F, \tag{4.54a}$$

$$\mathcal{M}_{0,\text{Sym}} := 0; \tag{4.54b}$$

$$\bullet \quad R_{1,\text{Sym}} := \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F, \tag{4.54c}$$

$$\mathcal{M}_{1,\text{Sym}} := \frac{2(1-\delta_z)(1-\delta_2)}{2-\delta_1-\delta_2}F; \tag{4.54d}$$

$$\bullet \quad R_{2,\text{Sym}} := \min\big\{2(1-\delta_1)F, (1-\delta_2)F\big\}, \tag{4.54e}$$

$$\mathcal{M}_{2,\text{Sym}} := \min\Bigg\{2\big[(1-\delta_z)+D(1-\delta_1)\big]F, \frac{2(1-\delta_2)\big[2(1-\delta_z)+D(1-\delta_2)\big]}{2(1-\delta_1)+(1-\delta_2)}F\Bigg\}. \tag{4.54f}$$

Proposition 4.1 (Lower Bound on $C_{s,\text{Sym}}(\mathcal{M})$)**.** *The upper convex hull of the three rate-memory pairs* $\big\{(R_{\ell,Sym},\mathcal{M}_{\ell,Sym});\ \ell\in\{0,1,2\}\big\}$ *in* (4.54) *lower bounds the secure capacity-memory tradeoff under symmetric cache assignment:*

$$C_{s,Sym}(\mathcal{M}) \geq \text{upper hull}\big\{(R_{\ell,Sym},\mathcal{M}_{\ell,Sym})\colon\ \ell=0,1,2\big\}. \tag{4.55}$$

Proof. As in proof of Theorem 4.1, it suffices to prove the achievability of the three rate-memory pairs $\{(R_\ell,\mathcal{M}_\ell)\colon\ \ell=0,1,2\}$. At $\mathcal{M}_{0,\text{Sym}}$, $R_{0,\text{Sym}}=R_0$ since there is no cache. We prove the achievability of the two remaining points in the following subsections. □

Remark 4.2. Since both receivers can store messages in their cache memories, an increase of $\mathcal{M}$ in cache memory size implies at least an increase of $\frac{\mathcal{M}}{2D}$ in the achievable rate. Hence,

$$\text{if } (R_s,\mathcal{M}) \text{ is achievable} \quad\Rightarrow\quad \left(R_s+\frac{\mathcal{M}'-\mathcal{M}}{2D},\mathcal{M}'\right) \text{ is achievable.}$$

Therefore, for $\mathcal{M}>\mathcal{M}_{2,\text{Sym}}$,

$$R_{\text{Sym}} = R_{2,\text{Sym}} + \frac{\mathcal{M}-\mathcal{M}_{2,\text{Sym}}}{2D}. \tag{4.56}$$

4.4.1 Scheme achieving rate-memory pair $(R_{1,\mathbf{Sym}}, \mathcal{M}_{1,\mathbf{Sym}})$

Preparations: Let $\alpha \in [0,1]$. Generate two random keys K_1 and K_2 of rate $\alpha(1-\delta_z)F$ each.

Caching phase: Store K_1 in receiver 1's cache memory and K_2 in receiver 2's cache memory. Thus, the total cache memory size is

$$\mathcal{M}_{1,\text{Sym}} = 2\alpha(1-\delta_z)F. \tag{4.57}$$

Delivery phase: The delivery phase is divided into two periods of lengths αn and $(1-\alpha)n$. In the first period, the transmitter sends W_{d_1} to receiver 1 using a wiretap code with secret key K_1 and in the second period, it sends W_{d_2} to receiver 2 using a wiretap code with secret key K_2.

Analysis: Both receivers reliably decode their messages whenever

$$R_{1,\text{Sym}} \leq \alpha(1-\delta_1)F, \tag{4.58a}$$
$$R_{1,\text{Sym}} \leq (1-\alpha)(1-\delta_2)F. \tag{4.58b}$$

Thus, $R_{1,\text{Sym}}$ is maximized and the rate-memory pair $(R_{1,\text{Sym}}, \mathcal{M}_{1,\text{Sym}})$ in (4.54c) and (4.54d) is obtained when

$$\alpha = \frac{1-\delta_2}{2-\delta_1-\delta_2}. \tag{4.59}$$

4.4.2 Scheme achieving rate-memory pair $(R_{2,\mathbf{Sym}}, \mathcal{M}_{2,\mathbf{Sym}})$

Preparations: Let $0 \leq \alpha_1, \alpha_2 \leq 1$, such that $\alpha_1 + \alpha_2 = 1$. For $d \in \mathcal{D}$, split every message into three sub-messages, such that

$$W_d = \left[W_d^{(0)}, W_d^{(1)}, W_d^{(2)}\right],$$

with rates $R^{(0)}$, $\frac{R^{(1)}}{2}$ and $\frac{R^{(1)}}{2}$, respectively.

Moreover, generate three random keys K_1, K_2 and K_3 of rates $\alpha_1(1-\delta_z)F$, $\alpha_1(1-\delta_z)F$ and $(1-\alpha_1-\alpha_2)(1-\delta_z)F$, respectively.

Caching phase: Store K_1, K_3 and the D-tuple $W_1^{(1)}, \ldots, W_D^{(1)}$ in receiver 1's cache and K_2, K_3 and the D-tuple $W_1^{(2)}, \ldots, W_D^{(2)}$ in receiver 2's cache. Hence, the total cache memory size is

$$\mathcal{M}_{2,\text{Sym}} = 2(1-\alpha_2)(1-\delta_z)F + DR^{(1)}. \tag{4.60}$$

Delivery phase: The delivery phase is divided into three periods of lengths $\alpha_1 n$, $\alpha_2 n$ and $(1-\alpha_1-\alpha_2)n$. In the first period, the transmitter conveys message $W_{d_1}^{(0)}$ to receiver 1 using a wiretap code with secret key K_1. In the second period, it conveys message $W_{d_2}^{(0)}$ to receiver 2 using a wiretap code with secret key K_2. In the third period, it sends $W_{d_1}^{(2)} \oplus W_{d_2}^{(1)}$ using a wiretap code with secret key K_3.

Analysis: Receiver 1 decodes periods 1 and 3 and receiver 2 decodes periods 2 and 3. Periods 1 and 2 are decoded reliably if

$$R^{(0)} = \min\big\{\alpha_1(1-\delta_1), \alpha_2(1-\delta_2)\big\}F, \tag{4.61}$$

which is maximized for

$$\alpha_2 = \frac{1-\delta_1}{1-\delta_2}\alpha_1. \tag{4.62}$$

The third period is reliably decoded whenever

$$\begin{aligned}\frac{R^{(1)}}{2} &= \min\big\{(1-\alpha_1-\alpha_2)(1-\delta_1), (1-\alpha_1-\alpha_2)(1-\delta_2)\big\}F \\ &= (1-\alpha_1-\alpha_2)(1-\delta_1)F. \end{aligned}\tag{4.63}$$

The maximal rate in (4.54e) is obtained for

$$\alpha_1 = \max\left\{0, \frac{(1-\delta_2)\big[2(1-\delta_1)-(1-\delta_2)\big]}{(1-\delta_1)\big[2(1-\delta_1)+(1-\delta_2)\big]}\right\}. \tag{4.64}$$

Combining (4.60), (4.62), (4.63) and (4.64) gives the desired cache memory in (4.54f).

4.5 Upper bound on the secure capacity-memory tradeoff under symmetric cache assignment for the two-user scenario

The upper bound on the secure capacity memory tradeoff $C_{s,\mathrm{Sym}}(\mathcal{M})$ under symmetric cache assignment is stated in the following proposition:

Proposition 4.2 (Upper Bound on $C_{s,\mathrm{Sym}}(\mathcal{M})$)**.** *The secure capacity-memory tradeoff $C_{s,Sym}(\mathcal{M})$ of the two-user wiretap erasure BC with cache size $\mathcal{M}/2$ at both receivers under the joint secrecy constraint is upper bounded by the following three conditions:*

$$C_{s,Sym}(\mathcal{M}) \leq \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F + \frac{\mathcal{M}}{2}, \tag{4.65a}$$

$$C_{s,Sym}(\mathcal{M}) \leq (1-\delta_1)F + \frac{\mathcal{M}}{2D}, \tag{4.65b}$$

$$C_{s,Sym}(\mathcal{M}) \leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F + \frac{\mathcal{M}}{D}. \tag{4.65c}$$

Proof. Constraint (4.65a) is obtained following the same steps as in the proof of (4.37a). Constraints (4.65b) and (4.65c) follow from [9] by ignoring the secrecy constraint. □

4.6 Lower bound on the secure capacity-memory tradeoff under two-sided asymmetric cache assignment for the two-user scenario

In this section, we study a cache distribution between the one-sided cache assignment and the symmetric cache distribution. We consider the same two-user wiretap BC model with a total cache memory $\mathcal{M}$. Receiver 1 and 2 have access to a cache memory of size $\mathcal{M}_{R_1}$ and $\mathcal{M}_{R_2}$, respectively, such that $\mathcal{M}_{R_1} + \mathcal{M}_{R_2} = \mathcal{M}$. We choose the cache memories $\mathcal{M}_{R_1}$ and $\mathcal{M}_{R_2}$ based on the channel statistics of both users. We compute the lower and upper bounds on the secure capacity-memory tradeoff $C_{s,\text{Asym}}(\mathcal{M})$ for this cache distribution.

Consider the five rate-memory pairs:

- $$R_{0,\text{Asym}} := \frac{(\delta_z - \delta_1)(\delta_z - \delta_2)}{2\delta_z - \delta_1 - \delta_2} F, \tag{4.66a}$$
  $$\mathcal{M}_{0,\text{Asym}} := 0; \tag{4.66b}$$

- $$R_{1,\text{Asym}} := \frac{(1 - \delta_1)(\delta_z - \delta_2)}{1 + \delta_z - \delta_1 - \delta_2} F, \tag{4.66c}$$
  $$\mathcal{M}_{1,\text{Asym}} := \frac{(1 - \delta_z)(\delta_z - \delta_2)}{1 + \delta_z - \delta_1 - \delta_2} F; \tag{4.66d}$$

- $$R_{2,\text{Asym}} := \frac{(1 - \delta_1)(1 - \delta_2)}{2 - \delta_1 - \delta_2} F, \tag{4.66e}$$
  $$\mathcal{M}_{2,\text{Asym}} := (1 - \delta_z) F; \tag{4.66f}$$

- $$R_{3,\text{Asym}} := \frac{(1 - \delta_2)^2}{2 - \delta_1 - \delta_2} F, \tag{4.66g}$$
  $$\mathcal{M}_{3,\text{Asym}} := \left[(1 - \delta_z) + \frac{D(1 - \delta_2)(\delta_1 - \delta_2)}{2 - \delta_1 - \delta_2} \right] F; \tag{4.66h}$$

- $$R_{4,\text{Asym}} := \left[(2\delta_z - \delta_1 - \delta_2) + \frac{1 - \delta_z}{2} \right] F, \tag{4.66i}$$
  $$\mathcal{M}_{4,\text{Asym}} := \big[D(2\delta_z - \delta_1 - \delta_2) + (1 - \delta_z) \big] F; \tag{4.66j}$$

Proposition 4.3 (Lower Bound on $C_{s,\text{Asym}}(\mathcal{M})$)**.** *The upper convex hull of the five rate-memory pairs* $\big\{(R_{\ell,Asym}, \mathcal{M}_{\ell,Asym});\ \ell \in \{0,1,\ldots,4\}\big\}$ *in* (4.66) *lower bounds the secure capacity-memory tradeoff under two-sided asymmetric cache assignment:*

$$C_{s,Asym}(\mathcal{M}) \geq \text{upper hull}\big\{(R_{\ell,Asym}, \mathcal{M}_{\ell,Asym}):\ \ell = 0,\ldots,4\big\}. \tag{4.67}$$

Proof. As in proof of Theorem 4.1, it suffices to prove the achievability of the five rate-memory pairs $\{(R_\ell, \mathcal{M}_\ell):\ \ell = 0,\ldots,4\}$. At $\mathcal{M}_{0,\text{Asym}}$, $R_{0,\text{Asym}} = R_0$ since there is no cache. $(R_{1,\text{Asym}}, \mathcal{M}_{1,\text{Asym}})$ is similar to the first pair $(R_1, \mathcal{M}_1)$ of Section 4.2 since it is obtained by taking $\mathcal{M}_{R_1} = \mathcal{M}$ and $\mathcal{M}_{R_2} = 0$ and following the same analysis.

We prove the achievability of the remaining points in the following subsections. □

Remark 4.3. After $(R_{4,\text{Asym}}, \mathcal{M}_{4,\text{Asym}})$, the lower bound keeps increasing with a slope $\frac{1}{2D}$. This can be deduced from Remark 4.2

4.6.1 Scheme achieving rate-memory pair $(R_{2,\mathbf{Asym}}, \mathcal{M}_{2,\mathbf{Asym}})$

Preparations: Let $\alpha \in [0,1]$. Generate two random keys K_1 and K_2 of rates $\alpha(1-\delta_z)F$ and $(1-\alpha)(1-\delta_z)F$, respectively.

Caching phase: Store K_1 in the cache memory of receiver 1 and K_2 in the cache of receiver 2. Thus, the total cache memory size is

$$\mathcal{M}_{2,\text{Asym}} = (1-\delta_z)F.$$

Delivery phase: The delivery phase is divided into two periods of lengths αn and $(1-\alpha)n$, respectively. In the first period, the transmitter conveys message W_{d_1} to receiver 1 using wiretap code with secret key K_1. In the second period, it conveys message W_{d_2} to receiver 2 using wiretap code with secret key K_2.

Analysis: Both receivers decode reliably whenever

$$R_{2,\text{Asym}} \leq \alpha(1-\delta_1)F, \tag{4.68a}$$
$$R_{2,\text{Asym}} \leq (1-\alpha)(1-\delta_2)F. \tag{4.68b}$$

The maximal rate in (4.66e) is obtained for

$$\alpha = \frac{1-\delta_2}{2-\delta_1-\delta_2}. \tag{4.69}$$

4.6.2 Scheme achieving rate-memory pair $(R_{3,\mathbf{Asym}}, \mathcal{M}_{3,\mathbf{Asym}})$

This pair is obtained by following a similar analysis to the one in Section 4.2.3. The only difference is that we generate a second key K_2 of rate $(1-\alpha)(1-\delta_z)F$ and store it in receiver 2's cache. This key is used to secure messages $W_{d_2}^{(0)}$ and $W_{d_2}^{\oplus}$ conveyed to receiver 2 in the second period.

In this case, the total cache memory size is

$$\mathcal{M}_{3,\text{Asym}} = F(1-\delta_z) + D \cdot \alpha F(\delta_1 - \delta_2). \tag{4.70}$$

and the transmission rate should satisfy

$$R_{3,\text{Asym}} \leq \alpha(1-\delta_2)F, \tag{4.71}$$
$$R_{3,\text{Asym}} \leq (1-\alpha)(1-\delta_2)F + \alpha(\delta_1 - \delta_2)F. \tag{4.72}$$

The rate is maximized for

$$\alpha = \frac{1-\delta_2}{2-\delta_1-\delta_2}, \tag{4.73}$$

giving the rate memory pair $(R_{3,\text{Asym}}, \mathcal{M}_{3,\text{Asym}})$ in (4.66g) and (4.66h).

4.6.3 Scheme achieving rate-memory pair $(R_{4,\mathbf{Asym}}, \mathcal{M}_{4,\mathbf{Asym}})$

Preparations: For $d \in \mathcal{D}$, split each message W_d into three sub-messages, such that

$$W_d = \left[W_d^{(0)}, W_d^{(1)}, W_d^{\oplus}\right], \tag{4.74}$$

with rates $R^{(0)} = (\delta_z - \delta_1)F$, $R^{(1)} = (\delta_z - \delta_2)F$ and $R^{\oplus} = \frac{(1-\delta_z)F}{2}$, respectively. Thus, the total rate is

$$R_{4,\text{Asym}} = (2\delta_z - \delta_1 - \delta_2)F + \frac{1-\delta_z}{2}F.$$

Then, generate two random keys K_1 and K_2 of rate $\frac{(1-\delta_z)F}{2}$ each.

Moreover, generate a piggyback codebook $\mathcal{C}_1$ with $\Gamma_1 := \lfloor 2^{n(1-\delta_z)F} \rfloor \cdot \lfloor 2^{n(\delta_z-\delta_1)F} \rfloor \cdot \lfloor 2^{n(\delta_z-\delta_2)F} \rfloor$ codewords of length n,

$$\mathcal{C}_1 := \left\{ X_1^{(n)}(l_1) \right\}_{l_1=1}^{\Gamma_1}, \tag{4.75}$$

by drawing each entry of each codeword at random according to a Bernoulli-1/2 distribution independently of all other entries.

The codebook is partitioned into $\lfloor 2^{n(\delta_z-\delta_1)F} \rfloor \cdot \lfloor 2^{n(\delta_z-\delta_2)F} \rfloor$ subcodebooks (bins) each with $\lfloor 2^{n(1-\delta_z)F} \rfloor$ codewords. The subcodebooks are arranged into an array with $\lfloor 2^{n(\delta_z-\delta_1)F} \rfloor$ rows and $\lfloor 2^{n(\delta_z-\delta_2)F} \rfloor$ columns.

Caching phase: We store K_1 and the D-tuple $W_1^{(1)}, \ldots, W_D^{(1)}$ in receiver 1's cache. Moreover, we store K_2 and the D-tuple $W_1^{(0)}, \ldots, W_D^{(0)}$ in receiver 2's cache. Thus, the total cache memory size is

$$\mathcal{M}_{4,\text{Asym}} = (1-\delta_z)F + D(2\delta_z - \delta_1 - \delta_2)F. \tag{4.76}$$

Delivery phase: The transmitter uses the piggyback codebook $\mathcal{C}_1$ to encode messages $W_{d_1}^{(0)}$ and $W_{d_1}^{\oplus}$ to receiver 1 and messages $W_{d_2}^{(1)}$ and $W_{d_2}^{\oplus}$ to receiver 2. It generates

$$W_{\text{XOR}} = \left[W_{d_1}^{\oplus} \oplus K_1, W_{d_2}^{\oplus} \oplus K_2\right],$$

and transmits the W_{XOR}-th codeword of the subcodebook $\mathcal{C}_1(W_{d_1}^{(0)}, W_{d_2}^{(1)})$ over the channel.

Analysis: Receiver 1 reliably decodes messages W_{XOR} and $W_{d_1}^{(0)}$ since their total rate is $(1-\delta_1)F$. Receiver 2 also decodes reliably messages W_{XOR} and $W_{d_2}^{(1)}$ of total rate $(1-\delta_2)F$. Thus, the pair $(R_{4,\text{Asym}}, \mathcal{M}_{4,\text{Asym}})$ in (4.66i) and (4.66j) is securely achievable.

4.7 Upper bound on the secure capacity-memory tradeoff under two-sided asymmetric cache assignment for the two-user scenario

The upper bound on the secure capacity memory tradeoff $C_{s,\text{Sym}}(\mathcal{M})$ under two-sided asymmetric cache assignment is stated in the following proposition:

Proposition 4.4 (Upper Bound on $C_{s,\text{Asym}}(\mathcal{M})$)**.** *The secure capacity-memory tradeoff $C_{s,Asym}(\mathcal{M})$ of the two-user wiretap erasure BC with cache size $\mathcal{M}_{R_1}$ at receiver 1 and $\mathcal{M}_{R_2}$ at receiver 2, such that $\mathcal{M}_{R_1} + \mathcal{M}_{R_2} = \mathcal{M}$, under the joint secrecy constraint is upper bounded by the following four conditions:*

$$C_{s,Asym}(\mathcal{M}) \leq \frac{(\delta_z - \delta_1)(\delta_z - \delta_2)}{2\delta_z - \delta_1 - \delta_2}F + \frac{\delta_z - \delta_2}{2\delta_z - \delta_1 - \delta_2}\mathcal{M}, \tag{4.77a}$$

$$C_{s,Asym}(\mathcal{M}) \leq \frac{1}{2}(\delta_z - \delta_2)F + \frac{\mathcal{M}}{2}, \tag{4.77b}$$

$$C_{s,Asym}(\mathcal{M}) \leq \frac{(1-\delta_1)(1-\delta_2)}{2-\delta_1-\delta_2}F + \frac{\mathcal{M}}{D}, \tag{4.77c}$$

$$C_{s,Asym}(\mathcal{M}) \leq \frac{(2-\delta_1-\delta_2)}{2}F + \frac{\mathcal{M}}{2D}. \tag{4.77d}$$

Proof. Constraints (4.77a) and (4.77b) are obtained following the same proof of (4.37a). Constraints (4.77c) and (4.77d) follow from [9] by ignoring the secrecy constraint. □

4.8 Discussion and numerical results

4.8.1 Discussion on the obtained bounds

Figure 4.1 shows lower and upper bounds on the capacity-memory tradeoff $C_s(\mathcal{M})$ for a specific example.

From Theorem 4.2, upper bound (4.37a) is tight when the cache memory is small and upper bound (4.37c) is tight when the cache memory is sufficiently large. From Theorems 4.1 and 4.2, we can see that our lower and upper bounds coincide for small and large cache memory regimes.

Corollary 4.1. *When the cache memory is small:*

$$C_s(\mathcal{M}) = \frac{(\delta_z-\delta_1)(\delta_z-\delta_2)}{2\delta_z-\delta_1-\delta_2}F + \frac{\delta_z-\delta_2}{2\delta_z-\delta_1-\delta_2}\mathcal{M}, \qquad 0 \le \mathcal{M} \le \mathcal{M}_1, \quad (4.78)$$

where $\mathcal{M}_1$ *is defined in* (4.7d).

Corollary 4.2. *When the cache memory is large:*

$$C_s(\mathcal{M}) = (\delta_z-\delta_2)F, \qquad \mathcal{M} \ge \mathcal{M}_4, \quad (4.79)$$

where $\mathcal{M}_4$ *is defined in* (4.7j).

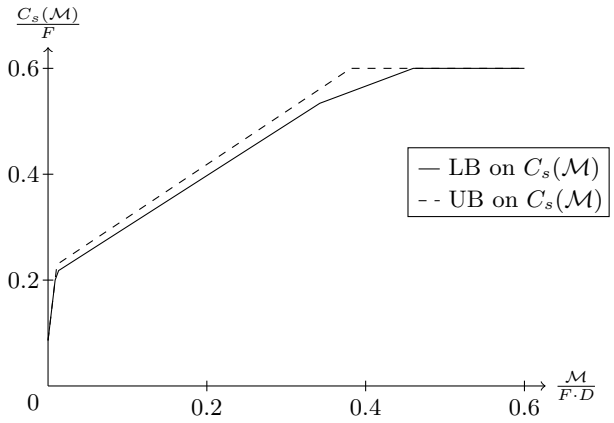

Figure 4.1: Lower and upper bounds on the secure capacity-memory tradeoff $C_s(\mathcal{M})$ under joint secrecy constraint for the two-user wiretap erasure BC with erasure probabilities $\delta_1 = 0.7, \delta_2 = 0.2, \delta_z = 0.8, F = 5$, and library size $D = 15$.

We observe from Corollary 4.1 and Figure 4.1 that for small cache memories, the rate-memory pairs $(R_0, \mathcal{M}_0)$, $(R_1, \mathcal{M}_1)$, and $(R_2, \mathcal{M}_2)$ determine the performance of our

lower bound in Theorem 4.1. The first point takes no cache memories. We achieve the other two points by storing only random keys in the cache memory, but no data. We thus conclude that for small cache memories it is not worth caching data, but only secret keys. The reason is that each piece of data will be useful for only a subset of all possible user demands, whereas a secret key serves with any demand. This also explains why in the regime of small $\mathcal{M}$, the secure capacity-memory tradeoff $C_s(\mathcal{M})$ can grow as a factor times $\mathcal{M}$, irrespective of the library size D. For larger values of $\mathcal{M}$, it grows at most like $\mathcal{M}/D$.

4.8.2 Impact of the joint secrecy constraint

We compare our bounds in Theorems 4.1 and 4.2 with the lower and upper bounds on the secure capacity-memory tradeoff under the individual secrecy constraint (see Chapter 3, Theorems 3.1 and 3.2).

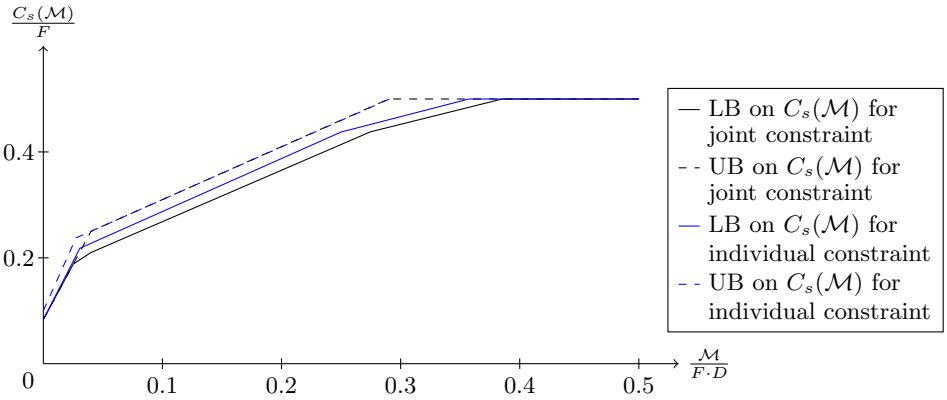

Figure 4.2: Lower and upper bounds on the secure capacity-memory tradeoff $C_s(\mathcal{M})$ under joint and individual secrecy constraints for the two-user wiretap erasure BC with erasure probabilities $\delta_1 = 0.7, \delta_2 = 0.3, \delta_z = 0.8, F = 5$, and library size $D = 5$.

Figure 4.2 shows that the upper bound on the capacity-memory tradeoff under an individual secrecy constraint is higher only for small cache memory regime and it coincides with the one under a joint constraint for large cache memories. This is trivial because in our upper bound's proof, we neglected the secrecy constraints for large cache memories.

We also observe that the lower bound under an individual secrecy constraint is higher than under a joint constraint for all parameters of $\mathcal{M}$. Therefore, we conclude that the additional security imposes loss in the transmission rate.

4.8.3 Impact of the cache assignment

Figures 4.3 and 4.4 depict the lower and upper bounds on the capacity memory tradeoffs for the different cases of cache assignment.

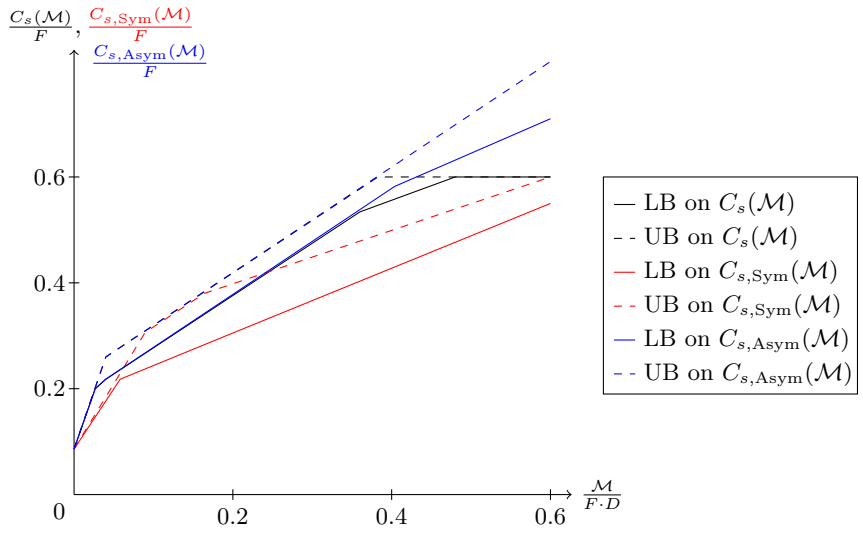

Figure 4.3: Lower and upper bounds on the secure capacity-memory tradeoffs $C_s(\mathcal{M})/C_{s,\text{Sym}}(\mathcal{M})/C_{s,\text{Asym}}(\mathcal{M})$ for the two-user wiretap erasure BC with erasure probabilities $\delta_1 = 0.7$, $\delta_2 = 0.2$, $\delta_z = 0.8$, $F = 5$, and library size $D = 5$.

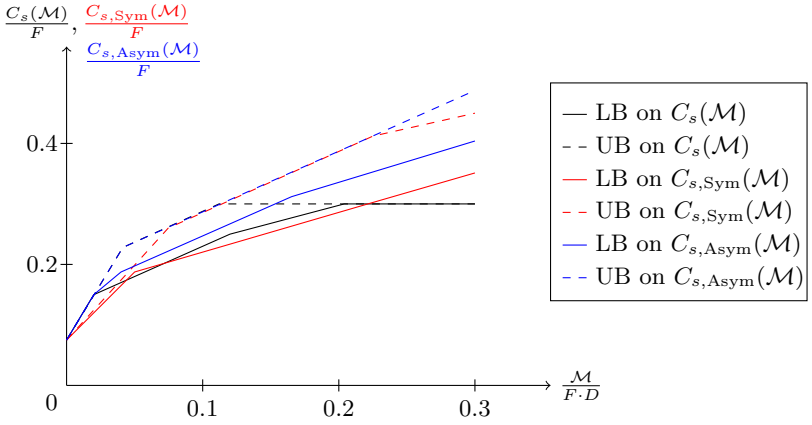

Figure 4.4: Lower and upper bounds on the secure capacity-memory tradeoffs $C_s(\mathcal{M})/C_{s,\text{Sym}}(\mathcal{M})/C_{s,\text{Asym}}(\mathcal{M})$ for the two-user wiretap erasure BC with erasure probabilities $\delta_1 = 0.7$, $\delta_2 = 0.5$, $\delta_z = 0.8$, $F = 5$ and library size $D = 5$.

By comparing Theorems 4.1 and 4.2 with Propositions 4.1 and 4.2, we observe that in the regime of small cache memory, our coding scheme for cache only at the weak receiver has a slope, namely $\frac{\delta_z-\delta_s}{2\delta_z-\delta_w-\delta_s}$, much steeper than the best possible coding scheme assuming that each receiver has the same cache memory size $\frac{\mathcal{M}}{2}$. In fact, by (4.65), the capacity in the latter case is upper bounded by $\frac{1}{2}$. So, for small cache memory regime, allocating all the cache memory to the weaker receiver results in a significantly higher performance than allocating the available cache memory equally between the two receivers.

We now compare Theorems 4.1 and 4.2 with Propositions 4.3 and 4.4. For the two-sided asymmetric cache assignment, the upper bound on $C_{s,\text{Asym}}(\mathcal{M})$ coincides with that of $C_s(\mathcal{M})$ in the one-sided cache assignment case for

$$\mathcal{M} \leq \frac{(1-\delta_2)(\delta_z-\delta_2)-(1-\delta_1)(1-\delta_z)}{2-\delta_1-\delta_2}F.$$

However, the lower bound $C_{s,\text{Asym}}(\mathcal{M})$ is better for the two-sided asymmetric cache assignment when storing only keys in the strong receiver's cache memory. This indicates that for the wiretap BC scenario, it is better to allocate a small cache memory even to the strong receiver to secure its communication. This is in contrast to the scenario without secrecy where no gain can be obtained from assigning cache memories to strong receiver.

For large cache memories, depending on the channel parameters, the bounds for equal cache sizes are sometimes higher than the bounds for cache only at receiver 1. However, distributing the cache asymmetrically to both receivers is always better than assigning it only to receiver 1. This is due to the fact that the capacity for the one-sided cache assignment saturates when the strong receiver does not have any cache memory; whereas the capacity for the two-sided cache assignment continues to increase with the memory size.

Moreover, since asymmetric cache distribution takes into consideration the channel parameters, it is always better than having symmetric caches regardless of the channel parameters and total cache sizes.

4.9 General lower bound on the secure capacity-memory tradeoff

We extend the results in Sections 4.2 and 4.3 to the general setup with K_w weak receivers and K_s strong receivers.

Consider the following six rate-memory pairs:

- $$R_0^{(K)} := \Big(\frac{K_w}{\delta_z-\delta_w}+\frac{K_s}{\delta_z-\delta_s}\Big)^{-1}F, \tag{4.80a}$$
  $$\mathcal{M}_0^{(K)} := 0; \tag{4.80b}$$

- $$R_1^{(K)} := \frac{(1-\delta_w)(\delta_z-\delta_s)}{K_w(\delta_z-\delta_s)+K_s(1-\delta_w)}F, \tag{4.80c}$$
  $$\mathcal{M}_1^{(K)} := \frac{(1-\delta_z)(\delta_z-\delta_s)}{K_w(\delta_z-\delta_s)+K_s(1-\delta_w)}F; \tag{4.80d}$$

- $$R_2^{(K)} := \min\Big\{\frac{(1-\delta_s)(\delta_z-\delta_w)}{K_s(1-\delta_w)}, \frac{(1-\delta_s)(1-\delta_w)}{K_w(1-\delta_s)+K_s(1-\delta_w)}\Big\}F, \tag{4.80e}$$
  $$\mathcal{M}_2^{(K)} := \frac{(1-\delta_z)}{K_w}F; \tag{4.80f}$$

- $$R_3^{(K)} := \frac{2(1-\delta_w)(\delta_z-\delta_s)\big[K_s(1-\delta_w)+K_w(\delta_w-\delta_s)\big]F}{K_w(\delta_z-\delta_s)\big[(K_w-1)(\delta_w-\delta_s)+2K_s(1-\delta_w)\big]+2K_s^2(1-\delta_w)^2}, \tag{4.80g}$$
  $$\begin{aligned}\mathcal{M}_3^{(K)} := &\frac{2D(\delta_z-\delta_s)(1-\delta_w)(\delta_w-\delta_s)F}{K_w(\delta_z-\delta_s)\big[(K_w-1)(\delta_w-\delta_s)+2K_s(1-\delta_w)\big]+2K_s^2(1-\delta_w)^2}\\ &+\frac{2(\delta_z-\delta_s)(1-\delta_z)\big[(K_w-1)(\delta_w-\delta_s)+K_s(1-\delta_w)\big]F}{K_w(\delta_z-\delta_s)\big[(K_w-1)(\delta_w-\delta_s)+2K_s(1-\delta_w)\big]+2K_s^2(1-\delta_w)^2}.\end{aligned} \tag{4.80h}$$

- $$R_4^{(K)} := \frac{(\delta_z-\delta_s)}{K_s}F, \tag{4.80i}$$
  $$\mathcal{M}_4^{(K)} := \frac{K_s(\delta_z-\delta_s)(1-\delta_z)+DK_w(\delta_z-\delta_s)^2}{K_s\big[K_s(1-\delta_z)+K_w(\delta_z-\delta_s)\big]}F; \tag{4.80j}$$

- $$R_5^{(K)} := \frac{(\delta_z-\delta_s)}{K_s}F, \tag{4.80k}$$
  $$\mathcal{M}_5^{(K)} := D\frac{(\delta_z-\delta_s)}{K_s}F. \tag{4.80l}$$

Theorem 4.3 (Lower Bound on $C_s^{(K)}(\mathcal{M})$)**.** *The upper convex hull of the six rate-memory pairs* $\big\{(R_\ell^{(K)},\mathcal{M}_\ell^{(K)}) : \ \ell\in\{0,\dots,5\}\big\}$ *in* (4.80) *lower bounds the secure capacity-memory tradeoff of the* K*-receiver channel with* K_w *weak receivers and* K_s *strong receivers with cache memories only at the weak receivers:*

$$C_s^{(K)}(\mathcal{M}) \geq \text{upper hull}\big\{(R_\ell^{(K)},\mathcal{M}_\ell^{(K)}), \ell\in\{0,\dots,5\}\big\}. \tag{4.81}$$

Proof. As in proof of Theorem 4.1, it suffices to prove the achievability of the six rate-memory pairs $\{(R_\ell^{(K)}, \mathcal{M}_\ell^{(K)}) \colon \ \ell = 0, \ldots, 5\}$. The achievability of the pairs $(R_0^{(K)}, \mathcal{M}_0^{(K)})$ and $(R_5^{(K)}, \mathcal{M}_5^{(K)})$ can be proved following the same analysis for $(R_0, \mathcal{M}_0)$ and $(R_5, \mathcal{M}_5)$.

The achievability of the remaining rate-memory pairs is outlined in the following subsections. □

Remark 4.4. For all $\mathcal{M} \geq \mathcal{M}_5^{(K)}$, the securely achievable capacity is $C_s(\mathcal{M}) = R_5^{(K)}$. This can be deduced from Remark 3.4.

4.9.1 Scheme achieving rate-memory pair $(R_1^{(K)}, \mathcal{M}_1^{(K)})$

Preparations: Let $\alpha \in [0, 1]$. For each weak receiver $i \in \mathcal{K}_w$, generate a random key K_i of rate $\alpha\frac{(1-\delta_z)F}{K_w}$ by drawing each entry randomly according to a Bernoulli-1/2 distribution independently of all other entries.

Caching phase: For each $i \in \mathcal{K}_w$, store K_i in the cache memory of receiver i, hence,

$$\mathcal{M}_1^{(K)} = \frac{\alpha(1-\delta_z)F}{K_w}. \tag{4.82}$$

Delivery phase: The delivery phase is divided into K periods such that the first K_w periods have length $\frac{\alpha n}{K_w}$ and the last K_s periods have length $\frac{(1-\alpha)n}{K_s}$. In each period $i \in \mathcal{K}_w$, the transmitter sends W_{d_i} to receiver i using a wiretap code with secret key K_i. In each period $j \in \mathcal{K}_s$, it sends W_{d_j} to receiver j using a wiretap code without secret key.

Analysis: Every weak receiver $i \in \mathcal{K}_w$ decodes its message reliably whenever

$$R_1^{(K)} \leq \frac{\alpha(1-\delta_w)F}{K_w}. \tag{4.83}$$

Every strong receiver $j \in \mathcal{K}_s$ decodes its message reliably whenever

$$R_1^{(K)} \leq \frac{(1-\alpha)(\delta_z-\delta_s)F}{K_s}. \tag{4.84}$$

Thus, the achievable rate is

$$R_1^{(K)} = \max_{\alpha\in[0,1]} \min\left\{\alpha\frac{(1-\delta_w)F}{K_w}, (1-\alpha)\frac{(\delta_z-\delta_s)F}{K_s}\right\}, \tag{4.85}$$

which is maximized for

$$\alpha = \frac{K_w(\delta_z-\delta_s)}{K_w(\delta_z-\delta_s) + K_s(1-\delta_w)}, \tag{4.86}$$

generating the rate-memory pairs $(R_1^{(K)}, \mathcal{M}_1^{(K)})$ in (4.80c) and (4.80d).

4.9.2 Scheme achieving rate-memory pair $(R_2^{(K)}, \mathcal{M}_2^{(K)})$

Preparations: Let $\alpha \in [0,1]$. For $d \in \mathcal{D}$, split each message into two sub-messages, such that

$$W_d = \left[W_d^{(0)}, W_d^{\oplus}\right],$$

with rates $R^{(0)} = \frac{\alpha(1-\delta_w)F}{K_w}$ and $R^{\oplus} = \frac{(1-\alpha)(1-\delta_z)F}{K_w}$, respectively.

For each weak receiver $i \in \mathcal{K}_w$, generate one random key $K_{i,1}$ of rate $\frac{\alpha(1-\delta_z)F}{K_w}$ and K_s random keys $K_{i,2}^{(j)}$, $j \in \mathcal{K}_s$, of rate $\frac{(1-\alpha)(1-\delta_z)F}{K_w K_s}$ each. The keys are generated by drawing each entry randomly according to a Bernoulli-1/2 distribution independently of all other entries.

Caching phase: For each $i \in \mathcal{K}_w$, store in receiver i's cache the following

$$V_i = K_{i,1} \bigcup \left\{K_{i,2}^{(j)},\ \forall j \in \mathcal{K}_s\right\}. \tag{4.87}$$

Hence, the cache memory size is

$$\mathcal{M}_2^{(K)} = \frac{(1-\delta_z)F}{K_w}. \tag{4.88}$$

Delivery phase: The delivery phase is divided into K periods such that the first K_w periods have length $\frac{\alpha n}{K_w}$ each and the last K_s periods have length $\frac{(1-\alpha)n}{K_s}$ each.

In the first K_w periods, the transmitter conveys messages $W_{d_i}^{(0)}$ to the weak receivers. In each period $i \in \mathcal{K}_w$, the transmitter sends message $W_{d_i}^{(0)}$ to receiver i using a wiretap code with secret key $K_{i,1}$.

In the last K_s periods, the transmitter conveys messages $W_{d_i}^{\oplus}$ to the weak receivers and messages W_{d_j} to the strong receivers. Each period $j \in \{K_w+1, \ldots, K\}$ is further divided into K_w sub-periods. Each message W_{d_j} is split into K_w parts $W_{d_j}^{(i)}$, $i = \{1, \ldots, K_w\}$ and each message $W_{d_i}^{\oplus}$ is split into K_s parts $W_{d_i}^{\oplus,(j)}$, $j = \{1, \ldots, K_s\}$. For each sub-period i of the period j, generate a superposition codebook with a cloud center that contains $2^{n(1-\alpha)(1-\delta_z)F/(K_s K_w)}$ codewords, and with each of the satellite codebooks containing $2^{nR_2^{(K)}/K_w}$ codewords. The transmitter encodes $W_{d_i}^{\oplus,(j)} \oplus K_{i,2}^{(j)}$ into the cloud center and $W_{d_j}^{(i)}$ into the satellite. Receiver i decodes only message $W_{d_i}^{\oplus} \oplus K_{i,2}^{(j)}$. Receiver j decodes both messages $W_{d_i}^{\oplus} \oplus K_{i,2}^{(j)}$ and $W_{d_j}^{(i)}$.

Analysis: Every weak receiver $i \in \mathcal{K}_w$ decodes reliably if

$$R_2^{(K)} \leq \alpha \frac{(1-\delta_w)F}{K_w} + (1-\alpha)\frac{(1-\delta_z)F}{K_w} \tag{4.89}$$

$$= \frac{(1-\delta_z)F}{K_w} + \alpha\frac{(\delta_z-\delta_w)F}{K_w}. \tag{4.90}$$

Moreover, every strong receiver $j \in \mathcal{K}_s$ decodes reliably if

$$R_2^{(K)} \leq (1-\alpha)\frac{(1-\delta_s)(\delta_z-\delta_w)F}{K_s(1-\delta_w)}. \tag{4.91}$$

Thus, the achievable rate is

$$R_2^{(K)} \leq \max_{\alpha\in[0,1]} \min\left\{\frac{(1-\delta_z)F}{K_w} + \alpha\frac{(\delta_z-\delta_w)F}{K_w}, (1-\alpha)\frac{(1-\delta_s)(\delta_z-\delta_w)F}{K_s(1-\delta_w)}\right\}, \tag{4.92}$$

which is maximized for

$$\alpha = \max\left\{0, \frac{K_w(1-\delta_s)(\delta_z-\delta_w) - K_s(1-\delta_z)(1-\delta_w)}{(\delta_z-\delta_w)[K_s(1-\delta_w)+K_w(1-\delta_s)]}\right\}, \tag{4.93}$$

generating the rate-memory pair $(R_2^{(K)}, \mathcal{M}_2^{(K)})$ in (4.80e) and (4.80f).

4.9.3 Scheme achieving rate-memory pair $(R_3^{(K)}, \mathcal{M}_3^{(K)})$

Preparations: Let $\alpha_1, \alpha_2, \alpha_3 \in [0,1]$, such that $\alpha_1+\alpha_2+\alpha_3 = 1$. For $d \in \mathcal{D}$, split each message into three sub-messages, such that

$$W_d = \left[W_d^{(0)}, W_d^{(1)}, W_d^{\oplus}\right]$$

with rates $R^{(0)} = \frac{\alpha_2(\delta_z-\delta_w)F}{K_w}$, $R^{(1)} = \frac{2\alpha_1 K_s(1-\delta_w)F}{K_w(K_w-1)}$ and $R^{\oplus} = \frac{\alpha_2(1-\delta_z)F}{K_w}$, respectively. Then, split every sub-message $W_d^{(1)}$ into K_w sub-messages

$$W_d^{(1)} = \left\{W_{d,i}^{(1)} : i \in \mathcal{K}_w\right\},$$

of rate $\frac{R^{(1)}}{K_w}$ each.

For each $i \in \mathcal{K}_w$, generate a random key $K_i^{(0)}$ of rate $R^{\oplus}$ by drawing each entry randomly according to a Bernoulli-1/2 distribution independently of all other entries. Similarly, generate $\binom{K_w}{2}$ random keys of rate $\frac{1-\delta_z}{1-\delta_w}\cdot\frac{R^{(1)}}{K_w}$:

$$K^{(1)} = \left\{K_{\{i_1,i_2\}}^{(1)} : \{i_1,i_2\} \subseteq \{1,\ldots,K_w\}\right\}. \tag{4.94}$$

Caching phase: For every weak receiver $i \in \mathcal{K}_w$, store the cache content

$$V_i = \left\{W_{d,i}^{(1)} : d \in \{1,\ldots,D\}\right\} \bigcup \left\{K_{\{i_1,i_2\}}^{(1)} : i \in \{i_1,i_2\}\right\} \bigcup K_i^{(0)}. \tag{4.95}$$

Thus, the cache memory size is

$$\mathcal{M}_3^{(K)} = D\frac{R^{(1)}}{K_w} + (K_w - 1)\frac{1-\delta_z}{1-\delta_w}\frac{R^{(1)}}{K_w} + R^{\oplus}. \tag{4.96}$$

Delivery phase: The delivery phase is divided into three sub-phases of lengths $\alpha_1 n$, $\alpha_2 n$ and $\alpha_3 n$.

In the first sub-phase, the transmitter conveys messages $W_{d_i}^{(1)}$ for every weak receiver $i \in \mathcal{K}_w$ by time-sharing over $\binom{K_w}{2}$ periods. In each period, it sends $W_{d_{i_1},i_2}^{(1)} \oplus W_{d_{i_2},i_1}^{(1)}$ to receivers i_1 and i_2 using a wiretap code with secret key $K_{\{i_1,i_2\}}^{(1)}$.

In the second sub-phase, the transmitter conveys messages $W_{d_i}^{(0)}$ and $W_{d_i}^{\oplus}$ for every weak receiver $i \in \mathcal{K}_w$ and message $W_{d_j}^{(1)}$ for every strong receiver $j \in \mathcal{K}_s$ by time-sharing over K_w periods. For each of the periods, it generates a piggyback codebook $\mathcal{C}$ with $\lfloor 2^{nR^{(0)}} \rfloor \cdot \lfloor 2^{nK_sR^{(1)}/K_w} \rfloor$ subcodebooks $\mathcal{C}\big(W_d^{(0)}, W_{K_s}^{(1)}\big)$. Each subcodebook has $\lfloor 2^{nR^{\oplus}} \rfloor$ codewords. It generates then $W_{\text{XOR}} = W_{d_i}^{\oplus} \oplus K_i^{(0)}$ and transmits the W_{XOR}-th codeword of the codebook $\mathcal{C}\big(W_{d_i}^{(0)}, W_{K_s}^{(1)}\big)$, where $W_{K_s}^{(1)} = \{W_{d_j,i}^{(1)} : j \in \mathcal{K}_s\}$.

In the third sub-phase, the transmitter conveys messages $W_{d_j}^{(0),\oplus} = [W_{d_j}^{(0)}, W_{d_j}^{\oplus}]$ for every strong receiver $j \in \mathcal{K}_s$ by time-sharing over K_s periods. In each of the periods, it sends $W_{d_j}^{(0),\oplus}$ to receiver j using a wiretap code without secret key.

Analysis: Weak receivers only decode the first transmission period. Thus, it is reliably decoded if

$$\frac{\binom{K_w}{2}\frac{R^{(1)}}{K_w}}{F(1-\delta_w)} \le \alpha_1 \quad \Rightarrow \quad \frac{\frac{K_w-1}{2}R^{(1)}}{F(1-\delta_w)} \le \alpha_1. \tag{4.97}$$

The third transmission period is reliably decoded only by strong receivers if

$$\frac{K_s\big(R^{(0)} + R^{\oplus}\big)}{F(\delta_z - \delta_s)} \le \alpha_3. \tag{4.98}$$

Every weak receiver $i \in \mathcal{K}_w$ decodes only messages $W_{d_i}^{(0)}$ and $W_{d_i}^{\oplus}$ transmitted in the second period; whereas every strong receiver $j \in \mathcal{K}_s$ decodes all the transmitted messages. Thus, the second period is reliably decoded whenever

$$\max\left\{\frac{K_w\big(R^{(0)} + R^{\oplus}\big)}{F(1-\delta_w)}, \frac{K_w\big(R^{(0)} + R^{\oplus}\big) + K_sR^{(1)}}{F(1-\delta_s)}\right\} \le \alpha_2. \tag{4.99}$$

The above inequality is maximized by equalizing both terms in the maximization yielding

$$R^{(1)} = \frac{K_w(\delta_w - \delta_s)}{K_w(\delta_w - \delta_s) + K_s(1-\delta_w)}R_3^{(K)}, \tag{4.100a}$$

$$R^{(0)} + R^{\oplus} = \frac{K_s(1-\delta_w)}{K_w(\delta_w-\delta_s)+K_s(1-\delta_w)} R_3^{(K)}. \tag{4.100b}$$

Combining (4.97)–(4.100) gives the rate $R_3^{(K)}$ in (4.80g) and the corresponding values:

$$\alpha_1 = \frac{K_w(K_w-1)(\delta_w-\delta_s)(\delta_z-\delta_s)}{\beta}, \tag{4.101a}$$

$$\alpha_2 = \frac{2K_wK_s(1-\delta_w)(\delta_z-\delta_s)}{\beta}, \tag{4.101b}$$

$$\alpha_3 = \frac{2K_s^2(1-\delta_w)^2}{\beta}, \tag{4.101c}$$

where

$$\beta = K_w(K_w-1)(\delta_w-\delta_s)(\delta_z-\delta_s) + 2K_wK_s(1-\delta_w)(\delta_z-\delta_s) + 2K_s^2(1-\delta_w)^2. \tag{4.102}$$

The cache memory size $\mathcal{M}_3^{(K)}$ is obtained as in (4.80h).

4.9.4 Scheme achieving rate-memory pair $(R_4^{(K)}, \mathcal{M}_4^{(K)})$

Preparations: Let $\alpha \in [0,1]$. For $d \in \mathcal{D}$, split each message into two sub-messages, such that

$$W_d = \left[W_d^{\oplus}, W_d^{(1)}\right], \tag{4.103}$$

with rates $R^{\oplus} = \frac{\alpha F(1-\delta_z)}{K_w}$ and $R^{(1)} = \frac{\alpha F(\delta_z-\delta_s)}{K_s}$, respectively.

Then, split every sub-message $W_d^{(1)}$ into K_w sub-messages

$$W_d^{(1)} = \left\{W_{d,i}^{(1)} : i \in \mathcal{K}_w\right\},$$

of rate $\frac{R^{(1)}}{K_w}$ each.

For each $i \in \mathcal{K}_w$, generate a random key K_i of rate $R^{\oplus}$ by drawing each entry randomly according to a Bernoulli-1/2 distribution independently of all other entries.

Caching phase: For every weak receiver $i \in \mathcal{K}_w$, store K_i and the D-tuple $W_1^{(1)}, \ldots, W_D^{(1)}$. Thus, the cache memory size is

$$\mathcal{M}_4^{(K)} = R^{\oplus} + DR^{(1)} = \alpha F\left[\frac{(1-\delta_z)}{K_w} + \frac{D(\delta_z-\delta_s)}{K_s}\right]. \tag{4.104}$$

Delivery phase: The delivery phase is divided into two sub-phases of lengths αn and $(1-\alpha)n$.

In the first sub-phase, the transmitter conveys messages $W_{d_i}^{\oplus}$ to every weak receiver $i \in \mathcal{K}_w$ and messages $W_{d_j}^{(1)}$ to every strong receiver $j \in \mathcal{K}_s$ by time sharing over $\mathcal{K}_w$ periods. For each of the periods, it generates a piggyback codebook $\mathcal{C}$ with $\lfloor 2^{nK_sR^{(1)}/K_w} \rfloor$ subcodebooks $\mathcal{C}(\tilde{W})$. Each subcodebook has $\lfloor 2^{nR^{\oplus}} \rfloor$ codewords. It generates then $W_{\text{XOR}} = W_{d_i}^{\oplus} \oplus K_i$ and transmits the W_{XOR}-th codeword of the codebook $\mathcal{C}(\tilde{W})$, where $\tilde{W} = \{W_{d_j,i}^{(1)} : j \in \mathcal{K}_s\}$.

In the second sub-phase, the transmitter conveys messages $W_{d_j}^{\oplus}$ to every strong receiver $j \in \mathcal{K}_s$ by time sharing over K_s periods using wiretap codes without secret key.

Analysis: Weak receivers decode reliably message $W_{d_i}^{\oplus}$ conveyed in the first transmission sub-phase since $R^{\oplus}$ is smaller than their channel capacity.

Strong receivers also decode reliably the first transmission sub-phase. They decode the transmission sub-phase reliably if

$$R^{\oplus} = \frac{\alpha(1-\delta_z)F}{K_w} \leq \frac{(1-\alpha)(\delta_z-\delta_s)F}{K_s}, \tag{4.105}$$

which is maximized for

$$\alpha = \frac{K_w(\delta_z-\delta_s)}{K_w(\delta_z-\delta_s)+K_s(1-\delta_z)}, \tag{4.106}$$

giving the rate-memory pair $(R_4^{(K)}, \mathcal{M}_4^{(K)})$ in (4.80i) and (4.80j).

4.10 General upper bound on the secure capacity-memory tradeoff

In this section, we provide the upper bound on the secure capacity-memory tradeoff of the general setup with K_w weak receivers and K_s strong receivers. This upper bound is stated in the following theorem:

Theorem 4.4 (Upper Bound on $C_s^{(K)}(\mathcal{M})$)**.** *The secure capacity-memory tradeoff $C_s^{(K)}(\mathcal{M})$ of the K-receiver channel with K_w weak receivers and K_s strong receivers with cache memories only at the weak receivers is upper bounded by the following $2K_w+1$ conditions:*

$$C_s^{(K)}(\mathcal{M}) \leq \frac{\delta_z-\delta_s}{K_s}F, \tag{4.107a}$$

$$C_s^{(K)}(\mathcal{M}) \leq \left(\frac{j}{1-\delta_w}+\frac{K_s}{1-\delta_s}\right)^{-1}F+\frac{j\mathcal{M}}{D}, \quad j \in \{1,\ldots,K_w\}, \tag{4.107b}$$

$$C_s^{(K)}(\mathcal{M}) \leq \max_{\alpha_i \in [0,1]} \min \left\{ \frac{\alpha_i(\delta_z-\delta_w)+(1-\alpha_i)(\delta_z-\delta_s)}{i+K_s}F+\frac{i}{i+K_s}\mathcal{M}, \right.$$

$$\alpha_i \frac{\delta_z - \delta_w}{i} F + \mathcal{M} \Bigg\}, \quad i \in \{1, \ldots, K_w\}. \quad (4.107c)$$

4.10.1 Proof of the general upper bound

Bound (4.107b) follows from [9] and by ignoring the secrecy constraint. Bound (4.107a) holds because the strong receivers have no cache, and their rate cannot be larger than in the absence of weak receivers.

Bound (4.107c) is proved as follows. For each blocklength n, we fix encoding, caching, and decoding functions as in (4.5), (4.3), (3.17) and (3.19) so that both the probability of worst-case error and the secrecy leakage satisfy:

$$\mathrm{P}_e^{\mathrm{Worst}} \xrightarrow[n\to\infty]{} 0 \qquad \text{and} \qquad \frac{1}{n} I(W_1, \ldots, W_D; Z^n) \xrightarrow[n\to\infty]{} 0.$$

By Fano's inequality and because conditioning can only reduce entropy, there exists a sequence of real numbers $\{\epsilon_n\}_{n=1}^{\infty}$ with

$$\frac{\epsilon_n}{n} \xrightarrow[n\to\infty]{} 0,$$

such that

$$\begin{cases} H(W_{d_1}|Y_1^n, V_1) & \leq & \frac{\epsilon_n}{2k}, \\ H(W_{d_2}|Y_2^n, V_1, V_2, W_{d_1}) & \leq & \frac{\epsilon_n}{2k}, \\ \qquad\vdots & & \\ H(W_{d_k}|Y_k^n, V_1, \ldots, V_k, W_{d_1}, \ldots, W_{d_{k-1}}) & \leq & \frac{\epsilon_n}{2k}. \end{cases}$$

We proved in Section 4.3 that the first inequality implies the following constraint

$$R_s \leq I(U_1; Y_1|Q) - I(U_1; Z|Q) + \mathcal{M}_1. \quad (4.108)$$

Accounting for receivers $1, \ldots, k$ among the K receivers, we derive:

$$\begin{aligned} kR_s &\leq \frac{1}{n} H(W_{d_1}, \ldots, W_{d_k}) \\ &= \frac{1}{n} H(W_{d_1}, \ldots, W_{d_k}|Z^n) + \frac{1}{n} I(W_{d_1}, \ldots, W_{d_k}; Z^n) \\ &\leq \frac{1}{n} H(W_{d_1}, \ldots, W_{d_k}|Z^n) + \frac{\epsilon_n}{2n} \\ &\overset{(a)}{\leq} \frac{1}{n} \big[H(W_{d_1}) + H(W_{d_2}|W_{d_1}) + \ldots + H(W_{d_k}|W_{d_{k-1}}, \ldots, W_{d_1}) \\ &\qquad - I(W_{d_1}, \ldots, W_{d_k}; Z^n) \big] + \frac{\epsilon_n}{2n} \end{aligned}$$

$$
\begin{aligned}
&\overset{(b)}{\leq} \frac{1}{n}\Big[I(W_{d_1};Y_1^n,V_1)+I(W_{d_2};Y_2^n,V_1,V_2|W_{d_1})+\dots \\
&\qquad +I(W_{d_k};Y_k^n,V_1,\dots,V_k|W_{d_1},\dots,W_{d_{k-1}})-I(W_{d_1},\dots,W_{d_k};Z^n)\Big]+\frac{\epsilon_n}{n} \\
&\overset{(c)}{=} \frac{1}{n}\Big[I(W_{d_1};Y_1^n,V_1)-I(W_{d_1};Z^n)\Big] \\
&+\frac{1}{n}\sum_{\ell=2}^{k}\Big[I(W_{d_\ell};Y_\ell^n,V_1,\dots,V_\ell|W_{d_1},\dots,W_{d_{\ell-1}})-I(W_{d_\ell};Z^n|W_{d_1},\dots,W_{d_{\ell-1}})\Big]+\frac{\epsilon_n}{n} \\
&=\frac{1}{n}\Big[I(W_{d_1};Y_1^n|V_1)-I(W_{d_1};Z^n|V_1)+I(W_{d_1};V_1|Z^n)\Big] \\
&+\frac{1}{n}\sum_{\ell=2}^{k}\Big[I(W_{d_\ell};Y_\ell^n|V_1,\dots,V_\ell,W_{d_1},\dots,W_{d_{\ell-1}}) \\
&\qquad -I(W_{d_\ell};Z^n|V_1,\dots,V_\ell,W_{d_1},\dots,W_{d_{\ell-1}}) \\
&\qquad\qquad +I(W_{d_\ell};V_1,\dots,V_\ell|W_{d_1},\dots,W_{d_{\ell-1}},Z^n)\Big]+\frac{\epsilon_n}{n}, \qquad (4.109)
\end{aligned}
$$

where (a) follows from the chain rule of mutual information; (b) follows from Fano's inequality; and (c) follows from the chain rule of mutual information.

In a similar way to (4.43), we can prove that

$$
\frac{1}{n}\Big[I(W_{d_1};Y_1^n|V_1)-I(W_{d_1};Z^n|V_1)\Big]\leq I(U_1;Y_1|Q)-I(U_1;Z|Q). \qquad (4.110)
$$

Then, we prove that for each $\ell\in\{1,\dots,k\}$, the following set of inequalities hold:

$$
\begin{aligned}
&I(W_{d_\ell};Y_\ell^n|V_1,\dots,V_\ell,W_{d_1},\dots,W_{d_{\ell-1}})-I(W_{d_\ell};Z^n|V_1,\dots,V_\ell,W_{d_1},\dots,W_{\ell-1}) \\
&\quad\overset{(a)}{=}\sum_{i=1}^{n}\big[I(W_{d_\ell};Y_{\ell,i}|V_1,\dots,V_\ell,W_{d_1},\dots,W_{d_{\ell-1}},Y_\ell^{i-1},Z_{i+1}^n) \\
&\qquad\qquad -I(W_{d_\ell};Z_i|V_1,\dots,V_\ell,W_{d_1},\dots,W_{d_{\ell-1}},Y_\ell^{i-1},Z_{i+1}^n)\big] \\
&\quad\overset{(b)}{=}\sum_{i=1}^{n}\big[I(W_{d_\ell};Y_{\ell,i}|V_1,\dots,V_\ell,W_{d_1},\dots,W_{d_{\ell-1}},Y_2^{i-1},\dots,Y_\ell^{i-1},Z_{i+1}^n) \\
&\qquad\qquad -I(W_{d_\ell};Z_i|V_1,\dots,V_\ell,W_{d_1},\dots,W_{d_{\ell-1}},Y_2^{i-1},\dots,Y_\ell^{i-1},Z_{i+1}^n)\big] \\
&\quad\overset{(c)}{\leq}\sum_{i=1}^{n}\big[I(W_{d_\ell};Y_{\ell,i}|V_1,\dots,V_\ell,W_{d_1},\dots,W_{d_{\ell-1}},Y_2^{i-1},\dots,Y_\ell^{i-1},Z_{i+1}^n) \\
&\qquad\qquad -I(W_{d_\ell};Z_i|V_1,\dots,V_\ell,W_{d_1},\dots,W_{d_{\ell-1}},Y_2^{i-1},\dots,Y_\ell^{i-1},Z_{i+1}^n)\big] \\
&\quad+\sum_{i=1}^{n}\big[I(V_\ell;Y_{\ell,i}|V_1,\dots,V_{\ell-1},W_{d_1},\dots,W_{d_{\ell-1}},Y_2^{i-1},\dots,Y_\ell^{i-1},Z_{i+1}^n) \\
&\qquad\qquad -I(V_\ell;Z_i|V_1,\dots,V_{\ell-1},W_{d_1},\dots,W_{d_{\ell-1}},Y_2^{i-1},\dots,Y_\ell^{i-1},Z_{i+1}^n)\big]
\end{aligned}
$$

$$
\begin{aligned}
&\overset{(d)}{\leq} \sum_{i=1}^{n} \big[I(W_{d_\ell}, V_\ell; Y_{\ell,i} | V_1, \ldots, V_{\ell-1}, W_{d_1}, \ldots, W_{d_{\ell-1}}, Y_2^{i-1}, \ldots, Y_\ell^{i-1}, Z_{i+1}^n) \\
&\qquad\qquad - I(W_{d_\ell}, V_\ell; Z_i | V_1, \ldots, V_{\ell-1}, W_{d_1}, \ldots, W_{d_{\ell-1}}, Y_2^{i-1}, \ldots, Y_\ell^{i-1}, Z_{i+1}^n) \big] \\
&+ \sum_{i=1}^{n} \big[I(Y_{\ell+1}^{i-1}; Y_{\ell,i} | V_1, \ldots, V_\ell, W_{d_1}, \ldots, W_{d_\ell}, Y_2^{i-1}, \ldots, Y_\ell^{i-1}, Z_{i+1}^n) \\
&\qquad\qquad - I(Y_{\ell+1}^{i-1}; Z_i | V_1, \ldots, V_\ell, W_{d_1}, \ldots, W_{d_\ell}, Y_2^{i-1}, \ldots, Y_\ell^{i-1}, Z_{i+1}^n) \big] \\
&= \sum_{i=1}^{n} \big[I(W_{d_\ell}, V_\ell, Y_{\ell+1}^{i-1}; Y_{\ell,i} | V_1, \ldots, V_{\ell-1}, W_{d_1}, \ldots, W_{d_{\ell-1}}, Y_2^{i-1}, \ldots, Y_\ell^{i-1}, Z_{i+1}^n) \\
&\qquad\qquad - I(W_{d_\ell}, V_\ell, Y_{\ell+1}^{i-1}; Z_i | V_1, \ldots, V_{\ell-1}, W_{d_1}, \ldots, W_{d_{\ell-1}}, Y_2^{i-1}, \ldots, Y_\ell^{i-1}, Z_{i+1}^n) \big],
\end{aligned} \tag{4.111}
$$

where (a) follows from the chain rule of mutual information and by applying Csiszar's sum-identity; (b) because receivers $1, \ldots, \ell-1$ are degraded with respect to receiver ℓ, and so the following Markov chain holds:

$$
(W_{d_\ell}; Y_{\ell,i} | V_1, \ldots, V_k, W_{d_1}, \ldots, W_{d_{\ell-1}}, Z_{i+1}^n) \to Y_\ell^{i-1} \to (Y_1^{i-1}, \ldots, Y_\ell^{i-1}); \tag{4.112}
$$

and (c) and (d) hold because the eavesdropper is degraded with respect to receiver ℓ, implying the following Markov chain:

$$
(V_1, \ldots, V_\ell, W_{d_1}, \ldots, W_{d_\ell}, Y_2^{i-1}, \ldots, Y_{\ell+1}^{i-1}, Z_{i+1}^n) \to Y_{\ell,i} \to Z_i. \tag{4.113}
$$

We define for each $k \in \{2, \ldots, K\}$ the random variables

$$
Y_k := Y_{k,Q} \tag{4.114}
$$

$$
U_k := (W_{d_k}, V_k, Y_{k+1}^{Q-1}, U_{k-1}). \tag{4.115}
$$

Dividing by n, we can rewrite constraint (4.111) as

$$
\begin{aligned}
&\frac{1}{n} \Big[I(W_\ell; Y_\ell^n | V_1, \ldots, V_\ell, W_{d_1}, \ldots, W_{d_{\ell-1}}) - I(W_\ell; Z^n | V_1, \ldots, V_\ell, W_{d_1}, \ldots, W_{d_{\ell-1}}) \Big] \\
&\leq \sum_{\ell=2}^{k} I(U_\ell; Y_\ell | U_{\ell-1}, Q) - I(U_\ell; Z | U_{\ell-1}, Q).
\end{aligned} \tag{4.116}
$$

Finally, we bound the following sum:

$$
\begin{aligned}
&I(W_{d_1}; V_1 | Z^n) + \sum_{\ell=2}^{k} I(W_{d_\ell}; V_1, \ldots, V_\ell | W_{d_1}, \ldots, W_{d_\ell}, Z^n) \\
&\qquad \leq I(W_{d_1}; V_1 \ldots, V_k | Z^n) + \sum_{\ell=2}^{k} I(W_{d_\ell}; V_1, \ldots, V_k | W_{d_1}, \ldots, W_{d_\ell}, Z^n)
\end{aligned}
$$

$$= I(W_{d_1}, \ldots, W_{d_k}; V_1, \ldots, V_k | Z^n)$$
$$\leq n \sum_{\ell=1}^{k} \mathcal{M}_\ell. \tag{4.117}$$

Taking into consideration constraints (4.110), (4.116) and (4.117), we can rewrite constraint (4.109) as:

$$kR_s \leq I(U_1; Y_1|Q) - I(U_1; Z|Q) + \sum_{\ell=1}^{k} \mathcal{M}_\ell + \epsilon_n$$
$$+ \sum_{\ell=2}^{k} \Big[I(U_\ell; Y_\ell | U_{\ell-1}, Q) - I(U_\ell; Z | U_{\ell-1}, Q) \Big]. \tag{4.118}$$

Let $n \to \infty$, from constraints (4.108) and (4.118), we conclude that if a rate-memory tuple $(R_s, \mathcal{M}_1, \ldots, \mathcal{M}_K)$ is securely-achievable (under the joint secrecy condition), then there exist auxiliaries $(U_1, U_2, \ldots, U_K, Q)$ so that for each realization of $Q = q$ the following Markov chain holds:

$$U_1 \to U_2 \to \cdots \to U_K \to X \to Y_K \to Y_{K-1} \to \cdots Y_1 \to Z \tag{4.119}$$

and so that the following K inequalities are satisfied:

$$R_s < I(U_1; Y_1|Q) - I(U_1; Z|Q) + \mathcal{M}_1; \tag{4.120a}$$

and for $k \in \{2, \ldots, K\}$:

$$kR_s \leq I(U_1; Y_1|Q) - I(U_,1; Z|Q) + \sum_{\ell=2}^{k} I(U_\ell; Y_\ell|U_{\ell-1}, Q) - I(U_\ell; Z|Q) + \sum_{\ell=1}^{k} \mathcal{M}_\ell. \tag{4.120b}$$

For the erasure BC, it is not hard to show that the weakest constraints are obtained by choosing $U_1, \ldots, U_K$ Bernoulli random variables and Q a constant.

So, constraints (4.120) become

$$R_s \leq \alpha_1(\delta_z - \delta_1) + \mathcal{M}_1, \tag{4.121a}$$
$$R_s \leq \frac{1}{k} \sum_{\ell=1}^{k} \big[\alpha_\ell(\delta_z - \delta_\ell) + \mathcal{M}_\ell \big], \qquad \forall k \in \{1, \ldots, K\} \tag{4.121b}$$

for a tuple $\alpha_1, \ldots, \alpha_K \geq 0$ summing up to 1.

Let i be the number of weak receivers among the k considered ones, and $j = k - i$ be the number of strong receivers. We can rewrite constraints (4.121) as follows:

$$R_s \leq \max_{\alpha_i \in [0,1]} \min \left\{ \frac{\alpha_i}{i}(\delta_z - \delta_w) + \mathcal{M}, \frac{1}{i+j} \Big[\alpha_i(\delta_z - \delta_w) + (1 - \alpha_i)(\delta_z - \delta_s) + i\mathcal{M} \Big] \right\},$$

$$\forall k \in \{1, \ldots, K\}, k = i + j. \tag{4.122}$$

We notice that for $i = K_s$, (4.122) generates a constraint tighter than for $i < K_s$. Thus, we can remove all constraints for $i < K_s$ without affecting the result and we retain only the K_w constraints in (4.107c).

4.11 Examples

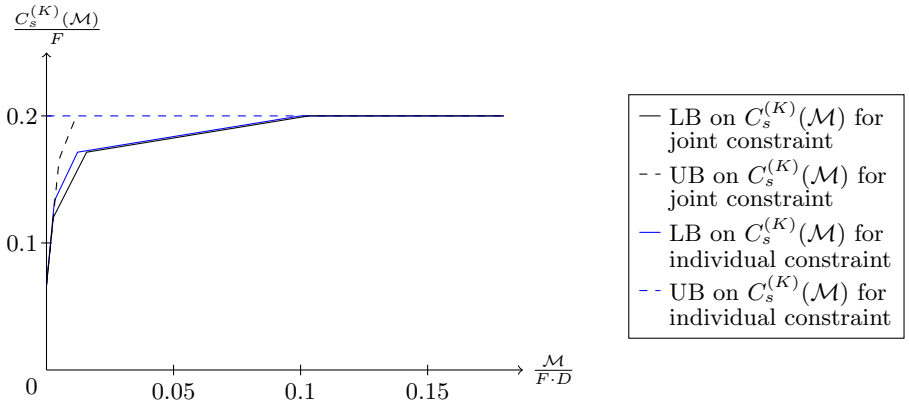

Figure 4.5: Lower and upper bounds on the secure capacity-memory tradeoff $C_s^{(K)}(\mathcal{M})$ under individual and joint secrecy constraints for the K-user wiretap erasure BC with $\delta_w = 0.7$, $\delta_s = 0.2$, $\delta_z = 0.8$, $F = 5$, $D = 30$, $K_w = 5$ and $K_s = 15$.

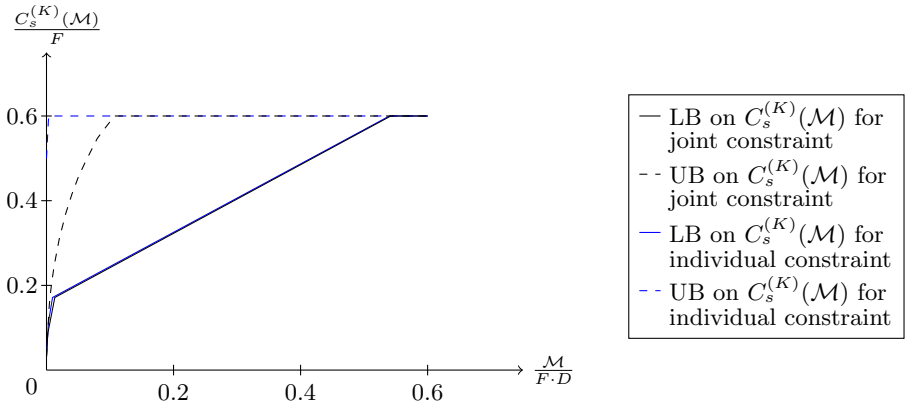

Figure 4.6: Lower and upper bounds on the secure capacity-memory tradeoff $C_s^{(K)}(\mathcal{M})$ under individual and joint secrecy constraints for the K-user wiretap erasure BC with $\delta_w = 0.7$, $\delta_s = 0.2$, $\delta_z = 0.8$, $F = 5$, $D = 30$, $K_w = 15$ and $K_s = 5$.

We illustrate in Figure 4.5 and 4.6 the lower and upper bounds derived for the general setup under individual and joint secrecy constraints assuming a total of $K = 20$ users and different choices of K_w and K_s. We can see that the lower bounds under both secrecy constraints are close.

4.12 Conclusion

In this chapter, we have derived lower and upper bounds on the secure capacity-memory tradeoff of the two-user wiretap packet-erasure BC with cache memory only at the weaker receiver under a joint secrecy constraint. Our bounds match for small and large cache memories. For small cache memory regime, they achieve optimality when only storing secret keys in the caches, but no data. The reason being that a cached secret key serves in securing any possible user's demand, whereas cached data serves only for a subset of demands. As a consequence, in the low cache memory regime, the capacity-memory tradeoff grows proportionally with the cache memory size, irrespective of the number of possible messages in the library.

For comparison, we have provided lower and upper bounds for the two-user scenario on the secure capacity-memory tradeoff under a joint secrecy constraint when both receivers have equal cache size. We observe that in the regime of small cache memory, it is highly beneficial to allocate all the available cache to the weaker receiver compared to allocating the cache memory uniformly across receivers. In fact, in our proposed coding schemes, the data and secret keys cached at the weak receivers are also used to secure the communication to the stronger receivers and to make it more efficient. This situation can however be reversed for larger cache sizes depending on the channel parameters.

Moreover, we have studied the two-user scenario with an asymmetric cache distribution for both receivers depending on their channel capacities. We have computed lower and upper bounds on the secure capacity-memory tradeoff in this case showing that this cache assignment is always better than the symmetric distribution. However, it is beneficial compared to the one-sided cache distribution only when both receivers have close erasure probabilities. In fact, in the two-sided distribution, a small cache memory can be allocated to strong receivers allowing them to secure their messages, removing thus the need for a random binning.

We have also compared with the bounds derived in the previous chapter under an individual secrecy constraint. The results reveal that the loss in performance due to the stronger secrecy constraint is small.

Finally, we have extended the results to K users, where K_w receivers are weak and have cache memories of equal size, and K_s receivers are strong and have no cache.

Conclusion

In this thesis, we have considered two lines of research in communication security. We have started by studying lattice-based cryptosystems. We have suggested improving the GGH scheme to make it an efficient post-quantum cryptography candidate. Then, we have investigated physical layer security in cache-aided wiretap broadcast channels.

In the first part of this thesis, we have proposed an improvement to the lattice-based GGH cryptosystem using generalized low density lattices. While guarantying the same level of security, our cryptosystem significantly reduces the complexity compared with previous GGH schemes. In fact, the specific form of the lattice generator matrix G_Λ induces a huge reduction in the key size. In dimension $n \approx 1000$, the key can be represented by approximately 100 KBs whereas the key size of previous GGH systems was in the order of MBs. Moreover, since G_Λ is by design in HNF, the complexity of the public key generation reduces by 1300 times compared to Micciancio and LDLC systems. Finally, the GLD lattice iterative decoding offers a better performance than Babai's algorithms and decreases the decryption complexity compared to LDLCs iterative decoding.

Next steps will consist in investigating other noise models. While uniform noise was considered in this study, further work will be needed to define the best noise distribution in our lattice-based system, through deeper theoretical and experimental studies of the error vector. Moreover, the dual code attack needs further in-depth study to determine the optimal density for GLD lattices and the best elementary codes to be employed.

In the second part of this thesis, we have focused on securing multi-user packet-erasure broadcast channels against eavesdropping attacks. Our scheme consists of a transmitter, K receivers and an eavesdropper. Among the K receivers, K_w are weak receivers and are provided with cache memories of size $\mathcal{M}$ each; whereas the remaining K_s are strong receivers and have no cache. We have considered two secrecy constraints: the individual constraint where each message should be individually secured from the eavesdropper and the joint constraint where all the messages are jointly secured against

the eavesdropper. We have computed lower and upper bounds on the secure capacity-memory tradeoff $C_s(\mathcal{M})$ for both constraints. For each secrecy condition, we have proposed a joint cache-channel coding scheme that can achieve the lower bound on $C_s(\mathcal{M})$. We have found that the individual secrecy constraint allows for a better capacity-memory tradeoff, while the joint secrecy constraint is more interesting from a security point of view.

To motivate our choice of cache distribution, we have provided lower and upper bounds on the secure capacity-memory tradeoff for the symmetric cache distribution for the two-user case and for both constraints. For the joint secrecy constraint, we have also computed bounds with an asymmetric cache distribution at both receivers depending on their channel states. We have compared the different cache distribution and have concluded that, for most cache memory sizes and channel parameters, cache only at weak receiver is the most beneficial case. Moreover, the pertinence of the joint cache-channel coding scheme is proved when compared with the best separate cache-channel coding scheme for the two-user case. This comparison clearly showed that our joint cache-channel coding achieves better lower bounds for all cache memory sizes and channel parameters for both secrecy constraints.

For future work, it would be interesting to replace our theoretical joint cache-channel coding approach with a practical coding scheme. In this purpose, the design of nested lattice coding schemes could be investigated. It would be also interesting to consider the case where the eavesdropper does not have the most degraded channel. In this case, the cache memory can still help to secure the messages for the receiver with the worst channel.

On a more general level, it would also be worth exploring the ways physical layer security and upper layer security could interact. These security schemes could benefit from each other especially to strengthen the overall system security. For instance, shared secret keys needed for network layer security protocols can be established at the physical layer by exploiting the channel characteristics. Symmetrically, network layer could securely convey secret keys to be used later in cache-aided transmissions. A well-optimized combination between physical layer security and network layer security could actually help building an efficient, secure and easily implementable system.

Appendix: Proof of Lemma 3.1

We bound $H(L_1|Z^{\alpha n}, W_{d_1,2}^{(0)}, \mathcal{C}_1)$ for every $w_{d_1}^{(0,2)}$. For a given codebook $\mathcal{C}_1$, let $\mathcal{C}_1(w_{d_1}^{(0,2)})$ be the subcodebook of $\mathcal{C}_1$ that contains all the codewords that are compatible with $w_{d_1}^{(0,2)}$:

$$\mathcal{C}_1(w_{d_1}^{(0,2)}) := \left\{x_1^{(\alpha n)}(l_1|w_{d_1}^{(0,2)})\right\}_{l_1 \in \Gamma_1(w_{d_1}^{(0,2)})} \tag{4.123}$$

where $\Gamma_1(w_{d_1}^{(0,2)})$ is a subset of Γ_1 defined in (3.29) such that $|\Gamma_1(w_{d_1}^{(0,2)})| = \lfloor 2^{n(D-2)R^{(1)}} \rfloor \cdot \lfloor 2^{nR^{(1)}} \rfloor \cdot \lfloor 2^{nR'} \rfloor$. Define for each $l_1 \in [1 : \Gamma_1']$ and for each sequence $z^{\alpha n} \in \mathcal{T}_\epsilon^{(\alpha n)}(p_Z)$, the set:

$$N_{\mathcal{C}_1}(l_1, z^{\alpha n}) := \left|\left\{\tilde{l}_1 \neq l_1 \colon \left(x_1^{(\alpha n)}(\tilde{l}_1|w_{d_1}^{(0,2)}), z^{\alpha n}\right) \in \mathcal{T}_\epsilon^{(\alpha n)}(p_X p_{Z|X})\right\}\right|, \tag{4.124}$$

where $p_{Z|X}$ stands for the channel law to the eavesdropper.

We are interested in the expectation and the variance of $N_{\mathcal{C}_1}(l_1, z^{\alpha n})$ over the choices of the index l_1, the sequence $z^{\alpha n}$, and the random code construction.

Lemma 4.1. *The desired expectation and variance satisfy*

$$\mathrm{E}_{\mathcal{C}_1, l_1, z^{\alpha n}}\left[N_{\mathcal{C}_1}(l_1, z^{\alpha n})\right] \geq 2^{n((D-1)R^{(1)} + R' - \alpha I(X_1;Z) - \delta(\epsilon) - \epsilon_n)}, \tag{4.125}$$

$$\mathrm{E}_{\mathcal{C}_1, l_1, z^{\alpha n}}\left[N_{\mathcal{C}_1}(l_1, z^{\alpha n})\right] \leq 2^{n((D-1)R^{(1)} + R' - \alpha I(X_1;Z) + \delta(\epsilon) - \epsilon_n)} \tag{4.126}$$

and

$$\mathrm{Var}(N_{\mathcal{C}_1}(l_1, z^{\alpha n})) \leq 2^{n((D-1)R^{(1)} + R' - \alpha I(X_1;Z) + \delta(\epsilon) - \epsilon_n)}, \tag{4.127}$$

where ϵ_n tends to 0 as $n \to \infty$.

Proof. The upper bound on the expectation of $N_{\mathcal{C}_1}(l_1, z^{\alpha n})$ is computed as follows

$$\mathrm{E}_{\mathcal{C}_1, l_1, z^{\alpha n}}\left[N_{\mathcal{C}_1}(l_1, z^{\alpha n})\right] = \mathrm{E}_{l_1}\left[\mathrm{E}_{\mathcal{C}_1, z^{\alpha n}}\left[N_{\mathcal{C}_1}(l_1, z^{\alpha n})|l_1\right]\right]$$

$$
\begin{aligned}
&= \mathrm{E}_{l_1}\Big[\mathrm{E}_{\mathcal{C}_1,z^{\alpha n}}\Big[\sum_{l_1'\neq l_1}\mathbb{1}_{(x_1^{(\alpha n)}(l_1'|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_Xp_{Z|X})}\Big]\Big]\\
&\overset{(a)}{=} \mathrm{E}_{l_1}\Big[\sum_{l_1'\neq l_1}\mathrm{E}_{\mathcal{C}_1,z^{\alpha n}}\Big[\mathbb{1}_{(x_1^{(\alpha n)}(l_1'|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_Xp_{Z|X})}\Big]\Big]\\
&\overset{(b)}{=} \mathrm{E}_{l_1}\Big[\sum_{l_1'\neq l_1}\Pr\Big[(x_1^{(\alpha n)}(l_1'|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_Xp_{Z|X})\Big]\Big]\\
&\overset{(c)}{\leq} \mathrm{E}_{l_1}\Big[\sum_{l_1'\neq l_1}2^{-\alpha n(I(X_1;Z)-\delta(\epsilon))}\Big]\\
&= (2^{n[(D-1)R^{(1)}+R']}-1)2^{-\alpha n(I(X_1;Z)-\delta(\epsilon))}\\
&= 2^{n[(D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n]},
\end{aligned}
\tag{4.128}
$$

where (a) holds because of the total law of expectations; (b) holds because $\mathrm{E}[\mathbb{1}_A] = \Pr(A)$; and (c) holds because of the upper bound of the joint typicality lemma.

The lower bound on the expectation follows the same proof except for the step (c) where we should use the lower bound of the joint typicality lemma to get the desired result.

As for the variance,

$$
\begin{aligned}
\mathrm{Var}(N_{\mathcal{C}_1}(l_1,z^{\alpha n})) &= \mathrm{E}_{\mathcal{C}_1,l_1,z^{\alpha n}}\Big[\big(N_{\mathcal{C}_1}(l_1,z^{\alpha n})\big)^2\Big]-\Big(\mathrm{E}_{\mathcal{C}_1,l_1,z^{\alpha n}}\Big[N_{\mathcal{C}_1}(l_1,z^{\alpha n})\Big]\Big)^2\\
&\leq \mathrm{E}_{\mathcal{C}_1,l_1,z^{\alpha n}}\Big[\big(N_{\mathcal{C}_1}(l_1,z^{\alpha n})\big)^2\Big]-2^{2n[(D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n]},
\end{aligned}
\tag{4.129}
$$

Now, let us bound the first term

$$
\begin{aligned}
&\mathrm{E}_{\mathcal{C}_1,l_1,z^{\alpha n}}\Big[\big(N_{\mathcal{C}_1}(l_1,z^{\alpha n})\big)^2\Big]\\
&= \mathrm{E}_{l_1}\Big[\mathrm{E}_{\mathcal{C}_1,z^{\alpha n}}\Big[\big(N_{\mathcal{C}_1}(l_1,z^{\alpha n})|l_1\big)^2\Big]\Big]\\
&= \mathrm{E}_{l_1}\Big[\mathrm{E}_{\mathcal{C}_1,z^{\alpha n}}\Big[\sum_{l_1'\neq l_1}\mathbb{1}_{(x_1^{(\alpha n)}(l_1'|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_XZ)}\sum_{l_1''\neq l_1}\mathbb{1}_{(x_1^{(\alpha n)}(l_1''|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_XZ)}\Big]\Big]\\
&= \mathrm{E}_{l_1}\Big[\sum_{l_1'\neq l_1}\sum_{l_1''\neq l_1}\mathrm{E}_{\mathcal{C}_1,z^{\alpha n}}\Big[\mathbb{1}_{(x_1^{(\alpha n)}(l_1'|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_XZ)}\mathbb{1}_{(x_1^{(\alpha n)}(l_1''|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_XZ)}\Big]\Big]\\
&= \mathrm{E}_{l_1}\Big[\sum_{l_1'\neq l_1}\sum_{\substack{l_1''\neq l_1\\ l_1''\neq l_1'}}\mathrm{E}_{\mathcal{C}_1,z^{\alpha n}}\Big[\mathbb{1}_{(x_1^{(\alpha n)}(l_1'|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_XZ)}\mathbb{1}_{(x_1^{(\alpha n)}(l_1''|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_XZ)}\Big]\Big]\\
&\quad+\mathrm{E}_{l_1}\Big[\sum_{l_1'\neq l_1}\mathrm{E}_{\mathcal{C}_1,z^{\alpha n}}\Big[\big(\mathbb{1}_{(x_1^{(\alpha n)}(l_1'|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_XZ)}\big)^2\Big]\Big]
\end{aligned}
$$

$$
\begin{aligned}
&\overset{(a)}{=} \mathrm{E}_{l_1}\Big[\sum_{l_1'\neq l_1} \mathrm{E}_{\mathcal{C}_1,z^{\alpha n}}\Big[\mathbb{1}_{(x_1^{(\alpha n)}(l_1'|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_{XZ})}\Big]\Big] \\
&\quad\times \mathrm{E}_{l_1}\Big[\sum_{\substack{l_1''\neq l_1\\ l_1''\neq l_1'}} \mathrm{E}_{\mathcal{C}_1,z^{\alpha n}}\Big[\mathbb{1}_{(x_1^{(\alpha n)}(l_1''|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_{XZ})}\Big]\Big] \\
&\quad+ \mathrm{E}_{l_1}\Big[\sum_{l_1'\neq l_1} \mathrm{E}_{\mathcal{C}_1,z^{\alpha n}}\Big[\mathbb{1}_{(x_1^{(\alpha n)}(l_1'|w_{d_1}^{(0,2)}),z^{\alpha n})\in\mathcal{T}_\epsilon^{(\alpha n)}(p_{XZ})}\Big]\Big] \\
&\leq 2^{2n[(D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n]} + 2^{n[(D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n]}, \qquad (4.130)
\end{aligned}
$$

where (a) holds because $\mathrm{E}[AB] = \mathrm{E}[A]\,\mathrm{E}[B]$ if A and B are independent, and $\mathrm{E}[(\mathbb{1}_A)^2] = \mathrm{E}[\mathbb{1}_A]$.

Finally, by combining (4.129) and (4.130), we get the desired upper bound in (4.127). □

We define the random process

$$
\mathcal{E}_{\mathcal{C}_1}(l_1, z^{\alpha n}) = \left\{ N_{\mathcal{C}_1}(l_1, z^{\alpha n}) \geq 2^{n((D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n/2)+1} \right\}. \qquad (4.131)
$$

It represents the case when $N_{\mathcal{C}_1}(l_1, z^{\alpha n})$ exceeds the upper bound on its expected value. The probability of this process is

$$
\begin{aligned}
&\Pr\{\mathcal{E}_{\mathcal{C}_1}(l_1, z^{\alpha n})\} \\
&\quad= \Pr\left\{ N_{\mathcal{C}_1}(l_1, z^{\alpha n}) \geq 2^{n((D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n/2)+1} \right\} \\
&\quad= \Pr\left\{ N_{\mathcal{C}_1}(l_1, z^{\alpha n}) \geq 2\times 2^{n((D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n/2)} \right\} \\
&\quad\overset{(a)}{\leq} \Pr\left\{ N_{\mathcal{C}_1}(l_1, z^{\alpha n}) \geq \mathrm{E}_{\mathcal{C}_1,L_1,Z^{\alpha n}}\left[N_{\mathcal{C}_1}(l_1, z^{\alpha n})\right] + 2^{n((D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n/2)} \right\} \\
&\quad\leq \Pr\left\{ \left| N_{\mathcal{C}_1}(l_1, z^{\alpha n}) - \mathrm{E}_{\mathcal{C}_1,L_1,Z^{\alpha n}}\left[N_{\mathcal{C}_1}(l_1, z^{\alpha n})\right] \right| \geq 2^{n((D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n/2)} \right\} \\
&\quad\overset{(b)}{\leq} \frac{\mathrm{Var}(N_{\mathcal{C}_1}(l_1, z^{\alpha n}))}{2^{2n((D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n/2)}}. \qquad (4.132)
\end{aligned}
$$

where (a) holds because of (4.126); and (b) holds because of the Chebyshev inequality $\Pr(|V-\mu| \geq k\sigma) \leq \frac{1}{k^2}$ where V is a random variable of mean μ and variance σ^2 and k is a constant. Hence, from (4.127) and (4.132), we conclude that, for every message $w_{d_1}^{(0,2)}$, if $(D-1)R^{(1)} + R' - \alpha I(X_1;Z) \geq 0$ then $\Pr\{\mathcal{E}_{\mathcal{C}_1}(l_1, z^{\alpha n})\} \to 0$ as $n \to \infty$.

Furthermore, for a given codebook $\mathcal{C}_1$ and for every message $w_{d_1}^{(0,2)}$, we define the set:

$$
N_{\mathcal{C}_1} := \left| \left\{ \tilde{l}_1 \neq L_1 \colon \left(x_1^{(\alpha n)}(\tilde{l}_1|w_{d_1}^{(0,2)}), Z^{\alpha n}\right) \in \mathcal{T}_\epsilon^{(\alpha n)}(p_X p_{Z|X}) \right\} \right|, \qquad (4.133)
$$

and the random process

$$\mathcal{E}_{\mathcal{C}_1} = \{N_{\mathcal{C}_1} \geq 2^{n((D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n/2)+1}\}. \tag{4.134}$$

In addition, define the indicator random variable $E(w_{d_1}^{(0,2)}) = 1$ if $\left(X_1^{(\alpha n)}(\tilde{l}_1|w_{d_1}^{(0,2)}), Z^{\alpha n}\right) \notin \mathcal{T}_\epsilon^{(n)}(p_X p_{Z|X})$ or the event $\mathcal{E}_{\mathcal{C}_1}$ occurs, and $E(w_{d_1}^{(0,2)}) = 0$ otherwise. In other words, $E(w_{d_1}^{(0,2)}) = 1$ describes the case when the eavesdropper cannot decode the message $x_{d_1}^{(0,2)}$. It occurs when the received $Z^{\alpha n}$ is not jointly typical with the sent $X_1^{(\alpha n)}$ or the number of $\tilde{l}_1$ in the subcodebook $\mathcal{C}_1(w_{d_1}^{(0,2)})$ that give a $X_1^{(\alpha n)}(\tilde{l}_1|w_{d_1}^{(0,2)})$ jointly typical with $Z^{\alpha n}$ is equivalent to their number in other subcodebooks $\mathcal{C}_1(W_{d_1}^{(0,2)} \neq w_{d_1}^{(0,2)})$. The union of events bound gives

$$\Pr\left\{E(w_{d_1}^{(0,2)}) = 1\right\} \leq \Pr\left\{(X_1^{(\alpha n)}(L_1|w_{d_1}^{(0,2)}), Z^{\alpha n}) \notin \mathcal{T}_\epsilon^{(\alpha n)}(p_X p_{Z|X})\right\} + \Pr\left\{\mathcal{E}_{\mathcal{C}_1}\right\}. \tag{4.135}$$

We bound the term $\Pr\{\mathcal{E}_{\mathcal{C}_1}\}$,

$$\begin{aligned}
\Pr\{\mathcal{E}_{\mathcal{C}_1}\} &\leq \sum_{z^{\alpha n} \in \mathcal{T}_\epsilon^{(\alpha n)}(p_Z)} p(z^{\alpha n}) \Pr\{\mathcal{E}_{\mathcal{C}_1}|Z^{\alpha n} = z^{\alpha n}\} \\
&= \sum_{z^{\alpha n} \in \mathcal{T}_\epsilon^{(\alpha n)}(p_Z)} \sum_{l_1} p(z^{\alpha n}) p(l_1|z^{\alpha n}) \Pr\{\mathcal{E}_{\mathcal{C}_1}|Z^{\alpha n} = z^{\alpha n}, L_1 = l_1\} \\
&= \sum_{z^{n} \in \mathcal{T}_\epsilon^{(\alpha n)}(p_Z)} \sum_{l_1} p(z^{\alpha n}) p(l_1|z^{\alpha n}) \Pr\{\mathcal{E}_{\mathcal{C}_1}(l_1, z^{\alpha n})\}.
\end{aligned} \tag{4.136}$$

If $(D-1)R^{(1)} + R' - \alpha I(X_1;Z) \geq 0$, $\Pr\{\mathcal{E}_{\mathcal{C}_1}(l_1, z^{\alpha n})\} \to 0$ as $n \to \infty$, and consequently $\Pr\{\mathcal{E}_{\mathcal{C}_1}\} \to 0$ as $n \to \infty$. In addition, by the law of large numbers, $\Pr\{(X_1^{\alpha n}(L_1|w_{d_1}^{(0,2)}), Z^{\alpha n}) \notin \mathcal{T}_\epsilon^{(\alpha n)}(p_X p_{Z|X})\} \to 0$ as $n \to \infty$. We can conclude that, $\Pr\{E(w_{d_1}^{(0,2)}) = 1\} \to 0$ as $n \to \infty$.

Therefore, the upper bound on the equivocation term becomes

$$\begin{aligned}
&H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)} = w_{d_1}^{(0,2)}, \mathcal{C}_1) \\
&\quad \leq 1 + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)} = w_{d_1}^{(0,2)}, E(w_{d_1}^{(0,2)}) = 1, \mathcal{C}_1) \Pr\{E(w_{d_1}^{(0,2)}) = 1\} \\
&\qquad\qquad + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)} = w_{d_1}^{(0,2)}, E(w_{d_1}^{(0,2)}) = 0, \mathcal{C}_1) \Pr\{E(w_{d_1}^{(0,2)}) = 0\} \\
&\quad \leq 1 + n((D-1)R^{(1)} + R') \Pr\{E(w_{d_1}^{(0,2)}) = 1\} \\
&\qquad\qquad + H(L_1|Z^{\alpha n}, W_{d_1}^{(0,2)} = w_{d_1}^{(0,2)}, E(w_{d_1}^{(0,2)}) = 0, \mathcal{C}_1) \\
&\quad \overset{(a)}{\leq} 1 + n((D-1)R^{(1)} + R') \Pr\{E(w_{d_1}^{(0,2)}) = 1\}
\end{aligned}$$

$$+ \log_2(2^{n((D-1)R^{(1)}+R'-\alpha I(X_1;Z)+\delta(\epsilon)-\epsilon_n/2)+1}), \tag{4.137}$$

where (a) holds because $H(X) \leq \log_2(|X|)$.

Finally, since $\Pr\{E(w_{d_1}^{(0,2)}) = 1\} \to 0$ as $n \to \infty$ if $(D-1)R^{(1)} + R' - \alpha I(X_1; Z) \geq 0$, then

$$\begin{aligned}
&\limsup_{n\to\infty} \frac{1}{n} H(L_1 | Z^{\alpha n}, W_{d_1}^{(0,2)} = w_{d_1}^{(0,2)}, \mathcal{C}_1) \\
&\quad \leq \frac{1}{n} + (D-1)R^{(1)} + R' - \alpha I(X_1; Z) + \delta(\epsilon) - \frac{\epsilon_n}{2} + \frac{1}{n} \\
&\quad \leq (D-1)R^{(1)} + R' - \alpha I(X_1; Z) + \delta'(\epsilon).
\end{aligned} \tag{4.138}$$

Bibliography

[1] O. Goldreich, S. Goldwasser, and S. Halevi. Public-key cryptosystems from lattice reduction problems. In *Proceedings of the International Cryptology Conference on Advances in Cryptology (CRYPTO)*, pages 112–131, 1997.

[2] P. Q. Nguyen. Cryptanalysis of the Goldreich-Goldwasser-Halevi cryptosystem from Crypto'97. In *Proceedings of the International Cryptology Conference on Advance in Cryptology (CRYPTO)*, pages 288–304, Santa Barbara, CA, USA, Aug. 1999.

[3] R. Hooshmand and M. Reza Aref. Public key cryptosystem based on low density lattice codes. *Wireless Personal Communications*, Aug. 2016.

[4] G. Poltyrev. On coding without restrictions for the AWGN channel. *IEEE Transactions on Information Theory*, 40(2):409–417, Mar. 1994.

[5] J.J. Boutros, N. di Pietro, and N. Basha. Generalized low-density (GLD) lattices. In *Proceedings of the IEEE Information Theory Workshop (ITW)*, pages 15–19, Hobart, Tasmania, Australia, Nov. 2014.

[6] L. Babai. On Lovaśz' lattice reduction and the nearest lattice point problem. *Combinatorica*, 6(1):1–13, Mar. 1986.

[7] D. Micciancio. Improving lattice based cryptosystems using the Hermite normal form. In *Proceedings of the Revised Papers from the International Conference on Cryptography and Lattices (CaLC)*, pages 126–145, Providence, Rhode Island, USA, Mar. 2001.

[8] C. Ludwig. The Security and Efficiency of Micciancio's Cryptosystem. *Technical Report. Available at http://www.cdc.informatik.tu-darmstadt.de/reports/TR/TI-02-07.MiccPaper.pdf*, 2004.

[9] S. Saeedi Bidokhti, R. Timo, and M. Wigger. Noisy broadcast networks with receiver caching. 2016. [Online]: https://arxiv.org/abs/1605.02317.

[10] M.A Maddah-Ali and U. Niesen. Fundamental limits of caching. *IEEE Transactions on Information Theory*, 60(5):2856–2867, May 2014.

[11] A. El Gamal and Y. H. Kim. *Network information theory*. Cambridge University Press, 2011.

[12] *ICT Facts and Figures 2016*, 2016 (accessed December 27, 2016). `http://www.itu.int/en/ITU-D/Statistics/Pages/facts/default.aspx`.

[13] *White paper: Cisco VNI Forecast and Methodology, 2015-2020*, 2016 (accessed December 27, 2016). `http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html`.

[14] *ITRC Breach Stats*, 2016 (accessed December 27, 2016). `http://www.idtheftcenter.org/2016databreaches.html`.

[15] *Over Half a Billion Personal Information Records Stolen or Lost in 2015*, 2015 (accessed December 27, 2016). `https://www.symantec.com/content/dam/symantec/docs/infographics/istr-reporting-breaches-or-not-en.pdf`.

[16] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings of the Symposium on Foundations of Computer Science*, pages 124–134, Santa Fe, New Mexico, USA, Nov. 1994.

[17] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Annual ACM symposium on Theory of computing (STOC)*, pages 212–219, Philadelphia, Pennsylvania, USA, May 1996.

[18] R. J. McEliece. A public-key cryptosystem based on algebraic coding theory. *Jet Propulsion Laboratory DSN Progress Report*, pages 42–44, 1978.

[19] H. Niederreiter. Knapsack-type cryptosystems and algebraic coding theory. *Problems of Control and Information Theory*, 15(2):159–166, 1986.

[20] V. M. Sidelnikov and S. O. Shestakov. On insecurity of cryptosystems based on generalized Reed-Solomon codes. *Discrete Mathematics and Applications*, 1(4):439–444, Jan. 1992.

[21] Y.X. Li, R.H. Deng, and X.M. Wang. On the equivalence of McEliece's and Niederreiter's public-key cryptosystems. *IEEE Transactions on Information Theory*, 40(1):271–273, Jan. 1994.

[22] C. Monico, J. Rosenthal, and A. Shokrollahi. Using low density parity check codes in the McEliece cryptosystem. In *Proceedings of the IEEE International Symposium on Information Theory Proceedings (ISIT)*, page 215, Sorrento, Italy, Jun. 2000.

[23] M. Baldi. *QC-LDPC code-based cryptography*. Springer, 2014.

[24] V. Sidelnikov. A public-key cryptosystem based on binary Reed-Muller codes. *Discrete Mathematics and Applications*, 4(3):191—207, Jan. 1994.

[25] E. Gabidulin, A. Paramonov, and O. Tretjakov. Ideals over a non-commutative ring and their applications to cryptography. In *Proceedings of the Annual International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT)*, pages 482–489, Brighton, UK, Apr. 1991.

[26] P. Gaborit. Shorter keys for code based cryptography. In *Proceedings of the International Workshop on Coding and Cryptography (WCC)*, pages 81–91, Bergen, Norway, Mar. 2005.

[27] L. Minder and A. Shokrollahi. Cryptanalysis of the sidelnikov cryptosystem. In *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pages 347–360, Barcelona, Spain, May 2007.

[28] D.J. Bernstein, T. Lange, and C. Peters. Wild McEliece. In *Proceedings of the International Workshop on Selected Areas in Cryptography (SAC)*, pages 143–158, Waterloo, Ontario, Canada, Aug. 2010.

[29] D.J. Bernstein, T. Lange, and C. Peters. Wild McEliece incognito. In *Proceedings of the International Workshop on Post-Quantum Cryptography (PQCrypto)*, pages 244–254, Taipei, Taiwan, Nov. 2011.

[30] J.C. Faugère, L. Perret, and F. de Portzamparc. Algebraic attack against variants of McEliece with Goppa polynomial of a special form advances. In *Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT)*, pages 21–41, Kaoshiung, Taiwan, Dec. 2014.

[31] T. A. Berson. Failure of the McEliece public-key cryptosystem under message-resend and related-message attack. In *Proceedings of the International Cryptology Conference on Advance in Cryptology (CRYPTO)*, pages 213–220, Santa Barbara, California, USA, Aug. 1997.

[32] E. Verheul and J. Doumen and H. Van Tilborg. Sloppy Alice attacks! Adaptive chosen ciphertext attacks on the McEliece public-key cryptosystem. *Information, Coding and Mathematics: Proceedings of Workshop Honoring Prof. Bob McEliece on his 60th Birthday*, pages 99–119, 2002.

[33] K. Kobara and H. Imai. Semantically secure McEliece public-Key cryptosystems conversions for McEliece PKC. In *Proceedings of the International Workshop on Practice and Theory in Public Key Cryptography (PKC)*, pages 19–35, Cheju Island, Korea, Feb. 2001.

[34] R. Dowsley, J. Muller-Quade, and A. C. A. Nascimento. A CCA2 secure public key encryption scheme based on the McEliece assumptions in the standard model. In *Proceedings of the Cryptographers' Track at the RSA Conference (CT-RSA)*, pages 240–251, San Francisco, CA, USA, Apr. 2009.

[35] N. Dottling, R. Dowsley, J. M. Quade, and A. C. A. Nascimento. A CCA2 secure variant of the McEliece cryptosystem. *IEEE Transactions on Information Theory*, 58(10):6672–6680, Jun. 2012.

[36] R. Nojima, H. Imai, K. Kobara, and K. Morozov. Semantic security for the McEliece cryptosystem without random oracles. *Designs, Codes and Cryptography*, 49(1):289–305, Dec. 2008.

[37] C.P. Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theoretical Computer Science*, 53(2):201–224, Aug. 1987.

[38] M. Ajtai. Generating hard instances of lattice problems. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, pages 99–108, Philadelphia PA, USA, 1996.

[39] M. Ajtai and C. Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, pages 284–293, Texas, USA, 1997.

[40] J. Hoffstein, J. Pipher, and J.H. Silverman. NTRU: a ring based public key cryptosystem. In *Proceedings of the International Symposium on Algorithmic Number Theory (ANTS III)*, pages 267–288, 1998.

[41] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, pages 84–93, Baltimore, Maryland, USA, May 2005.

[42] A. Kawachi, K. Tanaka, and K. Xagawa. Multi-bit cryptosystems based on lattice problems. In *Proceedings of the International Conference on Practice and Theory in Public-Key Cryptography*, pages 315–329, Beijing, P.R.China, Apr. 2007.

[43] C. Peikert, V. Vaikuntanathan, and B. Waters. A framework for efficient and composable oblivious transfer. In *Proceedings of the International Cryptology Conference on Advance in Cryptology (CRYPTO)*, pages 554–571, Santa Barbara, CA, USA, Aug. 2008.

[44] C. E. Shannon. Communication theory of secrecy systems. *The Bell System Technical Journal*, 28(4):656–715, Oct. 1949.

[45] A. D. Wyner. The Wire-tap channel. *The Bell System Technical Journal*, 54(8):1355–1387, Oct. 1975.

[46] I. Csiszár. Almost independence and secrecy capacity. *Problems of Information Transmission*, 32(1):48–57, 1996.

[47] A. Mukherjee, S. A. A. Fakoorian, J. Huang, and A. L. Swindlehurst. Principles of physical layer security in multiuser wireless networks: A survey. *IEEE Communications Surveys and Tutorials*, 16(3):1550–1573, Third Quarter 2014.

[48] S. Leung-Yan-Cheong and M. Hellman. The Gaussian wire-tap channel. *IEEE Transactions on Information Theory*, 24(4):451–456, Jul. 1978.

[49] C. Mitrpant, A.J.H. Vinck, and Y. Luo. An achievable region for the Gaussian wiretap channel with side information. *IEEE Transactions on Information Theory*, 52(5):2181–2190, May 2006.

[50] X. Tang, R. Liu, P. Spasojevic, and H. V. Poor. The Gaussian wiretap channel with a helping interferer. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 389–393, Toronto, Ontario, Canada, Jul. 2008.

[51] S. Shafiee and S. Ulukus. Achievable rates in Gaussian MISO channels with secrecy constraints. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 2466–2470, Nice, France, Jun. 2007.

[52] Z. Li, R. Yates, and W. Trappe. Secret communication with a fading eavesdropper channel. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 1296–1300, Nice, France, Jun. 2007.

[53] Y. Liang, H. V. Poor, and S. Shamai. Secure communication over fading channels. *IEEE Transactions on Information Theory*, 54(6):2470–2492, Jun. 2008.

[54] P. K. Gopala, L. Lai, and H. El Gamal. On the secrecy capacity of fading channels. *IEEE Transactions on Information Theory*, 54(10):4687–4698, Oct. 2008.

[55] E. Tekin and A. Yener. The general Gaussian multiple-access and two-way wiretap channels: Achievable rates and cooperative jamming. *IEEE Transactions on Information Theory*, 54(6):2735–2751, Jun. 2008.

[56] Y. Liang and H. V. Poor. Multiple-access channels with confidential messages. *IEEE Transactions on Information Theory*, 54(3):976–1002, Mar. 2008.

[57] E. Tekin and A. Yener. The Gaussian multiple access wire-tap channel. *IEEE Transactions on Information Theory*, 54(12):5747–5755, Dec. 2008.

[58] I. Csiszar and J. Korner. Broadcast channels with confidential messages. *IEEE Transactions on Information Theory*, 24(3):339–348, May 1978.

[59] R. Liu, I. Maric, P. Spasojevic, and R. D. Yates. Discrete memoryless interference and broadcast channels with confidential messages: Secrecy rate regions. *IEEE Transactions on Information Theory*, 54(6):2493–2507, Jun. 2008.

[60] A. Khisti, A. Tchamkerten, and G. W. Wornell. Secure broadcasting over fading channels. *IEEE Transactions on Information Theory*, 54(6):2453–2469, Jun. 2008.

[61] R. Liu and H. V. Poor. Secrecy capacity region of a multiple-antenna Gaussian broadcast channel with confidential messages. *IEEE Transactions on Information Theory*, 55(3):1235–1249, Mar. 2009.

[62] L. Lai and H. El Gamal. The relay–eavesdropper channel: Cooperation for secrecy. *IEEE Transactions on Information Theory*, 54(9):4005–4019, Sept. 2008.

[63] Y. Oohama. Coding for relay channels with confidential messages. In *Proceedings of the IEEE Information Theory Workshop (ITW)*, pages 87–89, Cairns, Australia, Sept. 2001.

[64] E. Ekrem and S. Ulukus. Secrecy in Cooperative Relay Broadcast Channels. *IEEE Transactions on Information Theory*, 57(1):137–155, Jan. 2011.

[65] D.J. Bernstein, J. Buchmann, and E Dahmen. *Post-quantum cryptography*. Springer, 2009.

[66] T. Richardson and R. Urbanke. *Modern coding theory*. Cambridge Univ. Press, 2008.

[67] R. G. Gallager. *Low-density parity-check codes*. Cambridge, MA, MIT Press, 1963.

[68] D. Klinc, J. Ha, S. McLaughlin, J. Barros, and B.-J. Kwak. LDPC codes for the Gaussian wiretap channel. *IEEE Transactions Information Forensics and Security*, 6(3):532–540, Sept. 2011.

[69] X. He and A. Yener. Providing secrecy with structured codes: Tools and applications to Gaussian two-user channels. *IEEE Transactions on Information Theory*, 60(4):2121–2138, Apr. 2014.

[70] C. Ling, Laura Luzzi, Jean-Claude Belfiore, and Damien Stehlé. Semantically secure lattice codes for the Gaussian wiretap channel. *IEEE Transactions on Information Theory*, 60(10):6399–6416, Oct. 2014.

[71] F. Oggier, P. Solé, and J.-C. Belfiore. Lattice codes for the wiretap Gaussian channel: Construction and analysis. *IEEE Transactions on Information Theory*, 62(10):5690–5708, Oct. 2016.

[72] M. Andersson, V. Rathi, R. Thobaben, J. Kliewer, and M. Skoglund. Nested polar codes for wiretap and relay channels. *IEEE Communications Letters*, 14(8):752–754, Aug. 2010.

[73] H. Mahdavifar and A. Vardy. Achieving the secrecy capacity of wiretap channels using polar codes. *IEEE Transactions on Information Theory*, 57(10):6428–6443, Oct. 2011.

[74] E. Şaşoğlu and A. Vardy. A new polar coding scheme for strong security on wiretap channels. In *Proceedings of the IEEE International Symposium on Information Theory Proceedings (ISIT)*, pages 1117–1121, Istanbul, Turkey, Jul. 2013.

[75] N. Cai and R. Yeung. Secure network coding on a wiretap network. *IEEE Transactions on Information Theory*, 57(1):424–435, Jan. 2011.

[76] A.O. Hero. Secure space-time communication. *IEEE Transactions on Information Theory*, 49(12):3235–3249, Dec. 2003.

[77] P. Parada and R. Blahut. Secrecy capacity of SIMO and slow fading channels. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 2152–2155, Adelaide, Australia, Sept. 2005.

[78] Z. Li, W. Trappe, and R. Yates. Secret communication via multi-antenna transmission. In *Proceedings of the Annual Conference on Information Sciences and Systems (CISS)*, pages 905–910, Baltimore, Maryland, USA, Mar. 2007.

[79] A. Khisti, G. Wornell, A. Wiesel, and Y. Eldar. On the Gaussian MIMO wiretap channel. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 2471–2475, Nice, France, Jun. 2007.

[80] A. Khisti and G. W. Wornell. Secure transmission with multiple antennas I: The MISOME wiretap channel. *IEEE Transactions on Information Theory*, 56(7):3088–3104, Jul. 2010.

[81] S. Shafiee, N. Liu, and S. Ulukus. Towards the secrecy capacity of the Gaussian MIMO wire-tap channel: The 2-2-1 channel. *IEEE Transactions on Information Theory*, 55(9):4033–4039, Sept. 2009.

[82] E. Ekrem and S. Ulukus. The secrecy capacity region of the Gaussian MIMO multi-receiver wiretap channel. *IEEE Transactions on Information Theory*, 57(4):2083–2114, Apr. 2011.

[83] A. Khisti and G. W. Wornell. Secure transmission with multiple antennas–part II: The MIMOME wiretap channel. *IEEE Transactions on Information Theory*, 56(11):5515–5532, Nov. 2010.

[84] F. Oggier and B. Hassibi. The secrecy capacity of the MIMO wiretap channel. *IEEE Transactions on Information Theory*, 57(8):4961–4972, Aug. 2011.

[85] Y. Chen, O. O. Koyluoglu, and A. Sezgin. On the individual secrecy for Gaussian broadcast channels with receiver side information. In *Proceedings of the IEEE International Conference on Communication Workshop (ICCW)*, pages 503–508, London, UK, Jun. 2015.

[86] Y. Chen, O. O. Koyluoglu, and A. Sezgin. Individual secrecy for broadcast channels with receiver side information. 2015. [Online]: https://arxiv.org/abs/1501.07547.

[87] A. S. Mansour, R. F. Schaefer, and H. Boche. Secrecy measures for broadcast channels with receiver side information: Joint vs individual. In *Proceedings of the IEEE Information Theory Workshop (ITW)*, pages 426–430, Hobart, Tasmania, Australia, Nov. 2014.

[88] R. F. Wyrembelski, A. Sezgin, and H. Boche. Secrecy in broadcast channels with receiver side information. In *Proceedings of the Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, pages 290–294, Pacific Grove, California, USA, Nov. 2011.

[89] A. Sengupta, R. Tandon, and T. C. Clancy. Fundamental limits of caching with secure delivery. *IEEE Transactions on Information Forensics and Security*, 10(2):355–370, Feb. 2015.

[90] V. Ravindrakumar, P. Panda, N. Karamchandani, and V. Prabhakarany. Fundamental limits of secretive coded caching. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, Barcelona, Spain, Jul. 2016.

[91] Zohaib Hassan Awan and Aydin Sezgin. Fundamental limits of caching in D2D networks with secure delivery. In *Proceedings of the IEEE International Conference on Communication Workshop (ICCW)*, pages 464–469, London, UK, Jun. 2015.

[92] L. W. Dowdy and D. V. Foster. Comparative models of the file assignment problem. *ACM Computing Surveys (CSUR)*, 14(2):287–313, Jun. 1982.

[93] I. Baev, R. Rajaraman, , and C. Swamy. Approximation algorithms for data placement problems. *SIAM Journal on Computing*, 38(4):1411–1429, Jul. 2008.

[94] S. Borst, V. Gupta, , and A. Walid. Distributed caching algorithms for content distribution networks. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, pages 1478–1486, San Diego, California, USA, Mar. 2010.

[95] Y. Birk and T. Kol. Coding on demand by an informed source (ISCOD) for efficient broadcast of different supplemental data to caching clients. *IEEE Transactions on Information Theory*, 52(6):2825–2830, Jun. 2006.

[96] Z. Bar-Yossef, Y. Birk, T. S. Jayram, and T. Kol. Index coding with side information. *IEEE Transactions on Information Theory*, 57(3):1479–1494, Mar. 2011.

[97] R. Timo and M. Wigger. Joint cache-channel coding over erasure broadcast channels. In *Proceedings of the International Symposium on Wireless Communication Systems (ISWCS)*, Bruxelles, Belgium, Aug. 2015.

[98] V. Bioglio, F. Gabry, and I. Land. Optimizing MDS codes for caching at the edge. In *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, San Diego, California, USA, Dec. 2015.

[99] Frédéric Gabry, Valerio Bioglio, and Ingmar Land. On edge caching with secrecy constraints. In *Proceedings of the IEEE International Conference on Communications (ICC)*, Kuala Lumpur, Malaysia, May 2016.

[100] M. Yoshino and N. Kunihiro. Improving GGH cryptosystem for large error vector. In *Proceedings of the International Symposium on Information Theory and its Applications (ISITA)*, pages 416–420, Honolulu, Hawaii, USA, Oct. 2012.

[101] C. F. de Barros and L. MenascheŚchechter. GGH may not be dead after all. *XXXV Braziliaan National Congress in Applied and Computational Mathematics (CNMAC)*, 2014.

[102] J. M. M. Barguil and R. Y. Lino and P. S. L. M. Barreto. Efficient variants of the GGH-YK-M cryptosystem. *XIV Braziliaan Symposium on Information and Computational system security (SBSeg)*, 2014.

[103] S. Paeng, B.E. Jung, and K. Ha. A lattice based public key cryptosystem using polynomial representations. In *Proceedings of the International Workshop on Practice and Theory in Public Key Cryptography (PKC)*, pages 292–308, Miami, FL, USA, Jan. 2003.

[104] D. Han, M.H. Kim, and Y. Yeom. Cryptanalysis of the Paeng-Jung-Ha cryptosystem from PKC 2003. In *Proceedings of the International Workshop on Practice and Theory in Public Key Cryptography (PKC)*, pages 107–117, Beijing, China, Apr. 2007.

[105] R. Misoczki, J.-P. Tillich, N. Sendrier, and P. S. L. M. Barreto. MDPC-McEliece: New McEliece variants from moderate density parity-check codes. In *Proceedings of the IEEE International Symposium on Information Theory Proceedings (ISIT)*, pages 2069–2073, Istanbul, Turkey, Jul. 2013.

[106] C.-P. Jeannerod, C. Pernet, and A. Storjohann. Rank-profile revealing Gaussian elimination and the CUP matrix decomposition. *Journal of Symbolic Computation*, 56:46–68, Sept. 2013.

[107] D. Micciancio and B. Warinschi. A linear space algorithm for computing the Hermite normal form. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation (ISSAC)*, pages 231–236, Ontario, Canada, Jul. 2001.

[108] N. Basha N. di Pietro and J. J. Boutros. Non-binary GLD codes and their lattices. In *Proceedings of the IEEE Information Theory Workshop (ITW)*, pages 1–5, Jerusalem, Apr. 2015.

[109] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv. Optimal decoding for linear codes for minimizing symbol error rate. *IEEE Transactions on Information Theory*, 20(2):284–287, Mar. 1974.

[110] N. Sommer, M. Feder, and O. Shalvi. Low-density lattice codes. *IEEE Transactions on Information Theory*, 54(4):1561–1585, Apr. 2008.

[111] B. Kurkoski and J. Dauwels. Message-passing decoding of lattices using Gaussian mixtures. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 2489–2493, Toronto, Canada, Jul. 2008.

[112] Y. Yona and M. Feder. Efficient parametric decoder of low density lattice codes. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 744–748, Seoul, Korea, Jun. 2009.

[113] H. Yamamoto. Rate-distortion theory for the Shannon cipher system. *IEEE Transactions on Information Theory*, 43(3):827–835, May 1997.

[114] E. Ekrem and S. Ulukus. Multi-receiver wiretap channel with public and confidential messages. *IEEE Transactions on Information Theory*, 59(4):2165–2177, Apr. 2013.

Curriculum Vitae

Sarah Kamel
Institut Mines-Télécom, Télécom-ParisTech
46 rue Barrault, 75013, Paris, France
Email: sarah.kamel@telecom-paristech.fr

Education

2013 – 2017	Ph.D. in Communications and Electronics CEA LIST and Télécom-ParisTech, France
2012 – 2013	M.Sc. in Digital Communication Systems Télécom-ParisTech, France
2008 – 2013	Engineering Dipl. in Telecommunications and Computer Science Lebanese University, Faculty of Engineering II, Lebanon

Professional Experience

2015 – 2016	Teaching assistant for ComElec Department Télécom-ParisTech, France
2015 – 2016	Teaching assistant for Electronic and Computer Science Department ESIPE, MLV, France
2013	Master graduation internship Télécom-ParisTech, France

Publications

Journal Papers

1. **S. Kamel**, M. Sarkiss and M. Wigger, "Achieving individual and joint secrecy with cache-channel coding over erasure broadcast channels", *in preparation for submission to the IEEE Transactions on Information Theory.*

Conferences

1. **S. Kamel**, M. Sarkiss and G. Rekaya-Ben Othman, "Generalized low density lattices for GGH cryptosystem", *in Proceedings of the International Conference on Frontiers of Signal Processing (ICFSP)*, Warsaw, Poland, Oct. 2016.

2. **S. Kamel**, M. Sarkiss and G. Rekaya-Ben Othman, "Improving GGH cryptosystem using generalized low density lattices", *to appear in the Proceedings of the International Conference on Advanced Communication Systems and Information Security (ACOSIS)*, Marrakesh, Morocco, Oct. 2016.

3. **S. Kamel**, M. Sarkiss and M. Wigger, "Secure joint cache-channel coding over erasure broadcast channels", *to appear in the Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, San Francisco, CA, USA, Mar. 2017.

4. **S. Kamel**, M. Sarkiss and M. Wigger, "Achieving joint secrecy with cache-channel coding over erasure broadcast channels", *submitted to the IEEE International Conference on Communications (ICC)*, 2017.