Réseaux de points flexibles pour modulations et codages à hautes efficacités spectrales

Carole Al Bechlawi

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

pour obtenir le grade de Docteur de Télécom Bretagne
En accréditation conjointe avec l'Ecole Doctorale Sicma
Mention : Sciences et Technologies de l'Information et de la Communication

présentée par
Carole Al Bechlawi
préparée dans le département Signal et Communications
Laboratoire Labsticc

Réseaux de points flexibles pour modulations et codages à hautes efficacités spectrales

Thèse soutenue le 15 janvier 2016
Devant le jury composé de :

Emmanuel Boutillon
Professeur, Université de Bretagne-Sud / président

Jean-François Hélard
Professeur, Insa - Rennes / rapporteur

Charly Poulliat
Professeur, INP-ENSEEIHT – Toulouse / rapporteur

Jean-Claude Belfiore
Professeur, Télécom ParisTech / examinateur

Frédéric Guilloud
Maître de conférences, Télécom Bretagne / examinateur

Ramesh Pyndiah
Professeur, Télécom Bretagne / directeur de thèse

Numéro d'ordre: 2016telb0371

Télécom Bretagne

Ecole Doctorale - SICMA

Le Codage des Réseaux de Points pour les Communications Point-à-point sur le Canal Gaussien et les Canaux à Évanouissements de Rayleigh

---

Thèse de Doctorat

Mention:
Sciences et Technologies de l'Information et de la Communication
Présentée par **Carole AL BECHLAWI**
Département: Signal et Communication
Laboratoire: Lab-STICC-CACS UMR 3192

Directeur de thèse: Ramesh Pyndiah
Soutenue: le 15 janvier 2016

Jury:

M. Charly POULLIAT, Professeur à l'INP-ENSEEIHT, Toulouse	Rapporteur
M. Jean-François HÉLARD, Professeur à L'INSA de Rennes	Rapporteur
M. Jean-Claude BELFIORE, Professeur à Télécom ParisTech	Examinateur
M. Emmanuel BOUTILLON, Professeur à l'UBS	Examinateur
M. Ramesh PYNDIAH, Professeur à Télécom Bretagne	Directeur de thèse
M. Frédéric GUILLOUD, Maître de conférences à Télécom Bretagne	Encadrant

"It always seems impossible until it's done."

Nelson Mandela

Acknowledgements

This PhD thesis is the result of three years of work, that was made more valuable and rewarding thanks to the numerous individuals who contributed to it.

First of all, I would like to express my gratitude to the committee members who have honoured me with their presence. I would like to thank the two reviewers, Professors Jean-François Hélard and Charly Poulliat, for taking some time out of their busy schedule and reading my PhD manuscript in detail. Their priceless comments and suggestions have been of a great help in improving the quality of my work. Moreover, I would like to thank the examiner, Professor Emmanuel Boutillon, for also accepting to carry out the task of the committee president. My gratitude also goes to the second examiner, Professor Jean-Claude Belfiore, who unfortunately had to be absent for personal constraints.

I would like to thank my supervisor, Professor Ramesh Pyndiah, for his numerous advice and great support during this work. I really appreciate our honest and fruitful discussions that helped me overcome the inevitable obstacles, and be able to produce an efficient work.

I would like to express my deepest appreciation to my advisor Dr. Frédéric Guilloud, who has been a tremendous mentor for me. Thank you so much for your patience, motivation and continuous support. The countless hours you spent guiding me in my work throughout those three years are something I can't thank you enough for.

This work would not have been possible without the amazing help of Professor Jean-Claude Belfiore, with whom I had the chance and honour to work during my two visits to Télécom ParisTech. Words cannot fairly express my gratitude for your availability, kindness and patience in explaining the complicated mathematical notions. Your help was crucial for me, and my collaboration with you was a turning point in my battle with "lattices".

The long days at Télécom Bretagne would not have been as fun and pleasant if it weren't for my amazing labmates. I choose to refrain from naming you individually, fearing to forget someone. Thank you for the nice moments, the little chit-chats during breaks and the delicious calories that some of you were thoughtful enough to bring to the lab. Most important of all, thank you so much for your great support during the last few months of my PhD. I am so thankful for your encouraging words. I would also like to thank everyone in the Signal and Communications department of Télécom Bretagne. I am grateful for your help, advice and unforgettable moments.

My life in Brest was the best I could ask for thanks to the friends, who later became more like family, that I have met there. Three years were enough to encounter many beautiful and interesting persons, it was really painful to watch some of them leave. Thank you for the precious moments, the laughter, the everlasting memories. Thank you for your support, it did a great job boosting my self-confidence and helping me take this to the very end.

I won't also forget my friends in Lebanon, and other parts of the world, who didn't allow distance to prevent them from being always close to my heart. You guys are awesome, you made me realize that some friendships are truly unbreakable.

And last but not least, I am forever indebted to my parents. I can never, not in a million years, thank you enough for your unconditional love and unlimited sacrifice. You, my brothers and sister are my world and your love was no doubt the key to my success.

Abstract

In this work, we address the application of emergent structures, known as lattices, for the digital communications problem. Endowed with interesting properties such as periodicity and linearity, lattices have recently gained considerable attention as they solved the decades-old problem of achieving full capacity on the AWGN channel. Motivated by these promising results, this work is dedicated to the analysis of the lattice coding performance for point-to-point communications under different scenarios: the studies are carried out over both AWGN and Rayleigh fading channels. After introducing infinite lattices, along with their constructions and sphere decoding algorithm, we tackle the lattice shaping problem that selects a finite number of lattice points. A lattice code decoder that takes into account the shaping region is proposed, and lattices of 8 and 16 dimensions are compared to short packet LTE: lattice coding is proven to achieve better performance in terms of Frame Error Rate for high spectral efficiencies, provided that near-optimal sphere decoding is performed. Due to the high complexity of the sphere decoding algorithm for higher dimensions, lattices based on multilevel constructions are studied. More precisely, the multilevel lattice construction using a set of nested binary linear Reed-Muller codes is proposed for both Gaussian and Rayleigh block fading channels. On the AWGN channel, the construction is carried out over standard binary partition chains of dimensions 1, 2 and 4. For each dimension, we show how to obtain the sufficient number of levels, together with the component codes' rates assigned to each level. We also show how increasing the dimension affects the global lattice performance. For the Rayleigh block fading channel, we resort to algebraic number theory as it was proven to provide the maximum possible diversity. Once the algebraic lattice is built, a rotation and base reduction operations allow us to obtain a rotated version of the integer lattice, and thus carry out the construction by mimicking the AWGN case.

Résumé

Nous vivons dans un monde qui devient de plus en plus connecté. En fait, selon le projet METIS [1], d'ici l'année 2020, les systèmes de communication futures se doivent de répondre à des défis importants, par exemple:

- Un volume de données 100 fois plus grand dans chaque région
- 10 à 100 fois plus de dispositifs connectés
- 10 à 100 fois plus de données consommées par utilisateur

Toutefois, la réponse à ces exigences considérables ne vient pas sans être confrontée à certaines limitations.

Les communications numériques

En 1948, Claude Shannon avait démontré que cette limitation est la capacité du canal [78], définie comme étant la quantité d'information maximale qu'on peut envoyer en se servant d'un système de communication numérique. La valeur exacte de la capacité C a été donnée sur un canal gaussien discret pour une valeur determinée du rapport signal sur bruit (SNR) par la formule suivante:

$$C = \log_2(1 + \text{SNR}) \qquad \text{bits/2 dim} \tag{1}$$

La capacité du canal signifie la chose suivante: si le taux de transmission des données est inférieur à la capacité, il est théoriquement possible de transmettre l'information avec une probabilité d'erreur négligeable en utilisant les bons codes correcteurs d'erreurs. Mais si le taux de tranmssion est supérieur à C, aucun code ne peut garantir une transmission fiable.

Depuis 1948, on a vu l'apparition de plusieurs codes correcteurs d'erreurs tels que les codes de Hamming, les codes algébriques, les codes LDPC et plus récemment les

turbo codes et les codes polaires. Jusqu'à présent, seule la capacité du cas particulier d'un canal gaussien contraint par une modulation binaire est atteinte en utilisant les codes polaires.

D'autre part, il existe en fait des strucutres linéaires et périodiques, nommées réseaux de points ou *lattices* en anglais, qui ont été démontrées d'atteindre la capacité sur un canal gaussien [85, 30]. De ce fait, les lattices font depuis quelques années, l'objet de plusieurs études liées à leur utilisation pour la transmission numérique.

Les réseaux de points

Un réseau de points est un ensemble de centre de sphères régulièrement répartis dans l'espace euclidien à n dimensions $\mathbb{R}^n$. Utilisés en mathématiques pour des travaux liés à la théorie des nombres, les formes quadratiques et la géométrie des nombres, les réseaux de points servent aussi en chimie où les cristallographes étudient les lattices à trois dimensions afin de les lier aux propriétés physiques de certains cristaux. En cryptographie, les lattices sont utilisés pour construire certains des algorithmes les plus forts. En communications numériques, ils servent à construire des modulations sur le canal gaussien et les canaux à évanouissements.

A l'origine, les mathématiciens se sont intéréssés aux lattices pour résoudre un problème assez ancien: l'empilement de sphères. Ce problème consiste à trouver la manière optimale qui nous permet d'entasser le plus grand nombre de sphères identiques dans un espace donné. Sur un canal gaussien, ce problème est équivalent à trouver, pour un code linéaire donné, le nombre maximal de mots de code ayant une certaine distance minimale. Pour un espace à deux dimensions, l'empilement de sphères optimal est fourni par ce qu'on appelle le *lattice hexagonal*, alors que le lattice nommé *fcc* (face-centered cubic) nous donne le meilleur empilement de sphères dans un espace à trois dimensions. Le problème d'empilement de sphères peut être généralisé à un espace de dimension n.

En communications numériques, les réseaux de points ont suscité beaucoup d'intérêt depuis que les modulations codées ont été développées en 1980. En fait, il a été démontré que les modulations multi-dimenssionnelles obtenues à travers les réseaux de points permettent de tranmsettre à des débits plus élevés avec une réduction considérable de la probabilité d'erreur.

Motivé par la reconnaissance des lattices comme outils prometteurs pour les transmissions numériques, ce travail de thèse est dédié à l'analyse des performances des réseaux de points sous différents scénarios: les études sont menées aussi bien sur le

canal gaussien que sur les canaux à évanouissements de Rayleigh, en se limitant au cas des systèmes de communication point-à-point.

L'encodage des réseaux de points est effectué soit par l'encodage direct d'un vecteur d'entiers en utilisant la matrice génératrice du réseau, soit en ayant recours à un schéma de modulation codée. Dans cette thèse, nous nous intéressons au premier cas dans les chapitres 1 et 2, qui par suite forment la première partie, et la deuxième méthode d'encodage est étudiée dans les chapitres 3 et 4 qui forment la seconde partie du manuscrit.

Structure du manuscrit

Ce manuscrit est constitué de quatre chapitres et trois annexes. Les chapitres sont organisés de la façon suivante.

Le chapitre 1 est une introduction aux réseaux de points. On commence par une revue historique des différentes études menées sur les lattices depuis les années 1970. En fait, le premier travail remarquable sur l'utilisation des lattices pour atteindre la capacité sur un canal gaussien est dû à de Buda et date de 1975 [27]. Plus précisément, de Buda a prouvé qu'il existe des constellations sphériques pouvant atteindre une capacité égale à $\log(\frac{P}{\sigma^2})$ bits par 2 dimensions sur le canal gaussien (ce qui est bien proche de la valeur maximale) en se servant de ce qu'on appelle le *lattice decoding*. Ce travail était à l'origine de nombreuses études consacrées aux différents aspects du problème de l'utilisation des réseaux de points pour la transmission numérique, par exemple:

- Les travaux de Poltyrev en 1994 [69] qui ont produit la notion de la *capacité généralisée*, atteinte en se servant de constellations infinies.
- Urbanke et Rimoldi ont montré en 1998 [85] que les réseaux de points finis peuvent atteindre la capacité maximale sur un canal gaussien en utilisant le *lattice code decoding* (décodage optimal).
- En 2004, Erez et Zamir [30] ont prouvé que les constellations finies de lattices peuvent atteindre la capacité sur un canal gaussien avec un décodage sous-optimal, le lattice decoding.

Cette revue historique est suivie d'une énumération des principaux paramètres utiles pour une meilleure compréhension des lattices et de leurs caractéristiques. Ainsi, un lattice Λ peut être généré par sa matrice génératrice, dont les colonnes sont les vecteurs de base de Λ. Le lattice sera donc l'ensemble des points résultant de la multiplication

de cette matrice par un vecteur d'entiers, noté **b**. Un paramètre très important est la *région de Voronoi* d'un point **x** de Λ, qui en fait consiste en tous les points réels qui sont plus proches de **x** que de n'importe quel autre point de Λ. En connaissant la région de Voronoi d'un lattice, on peut déterminer son volume. De plus, une translation de Λ d'un certain vecteur **a** donne ce qu'on appelle un *coset* de Λ.

La liste des paramètres d'un lattice nous permet ensuite de passer à l'explication des principales constructions d'un réseau de points à partir des codes correcteurs d'erreurs binaires linéaires [26]. Selon le nombre de codes employés, on distingue plusieurs types de constructions: la construction A employant un seul code et utilisée pour obtenir les lattices les plus denses pour une dimension inférieure ou égale à 8, avec un gain de codage maximal égal à 4. La construction D utilisée en ayant recours à plusieurs codes correcteurs d'erreurs qui doivent obligatoirement être imbriqués afin d'obtenir un lattice. Cette construction, permet d'obtenir des lattices de grandes dimensions ayant un gain de codage plus élevé. La construction B est un cas particulier de la construction D, où le nombre de codes employés est égal à 2. La construction D' est pareil à la construction D, sauf qu'elle consiste á utiliser les matrices de parité au lieu des matrices génératrices et permet ainsi d'obtenir les parity-check lattices à partir des codes LDPC imbriqués.

On s'intéresse en particulier à la construction D, et le principe de construction des réseaux de type Barnes-Wall (BW) à partir des codes de Reed-Muller ($\mathcal{RM}$) imbriqués est éxpliqué. On utilise ici le fait que les codes de Reed-Muller sont bien connus pour leur construction récursive, reposant sur la construction de "grands" codes $\mathcal{RM}$ à partir de codes plus petits.

Concernant le décodage des réseaux de points, le dédocage à maximum de vraisemblance d'un réseau est effectué en cherchant parmi tous les points du réseau, celui qui est le plus proche, en terme de distance euclidienne, du point reçu. L'algorithme de décodage par sphères permet de limiter la recherche aux points du réseau se trouvant à l'intérieur d'une sphère de rayon R centrée au point reçu. Bien entendu, le choix du rayon est un point crucial de l'algorithme: une valeur trop élevée de R entraîne un grand nombre de points à l'intérieur de la sphère, ce qui diminue la vitesse de recherche, tandis qu'avec une valeur trop faible de R, on risque de ne trouver aucun point du réseau à l'intérieur de la sphère. C'est pourquoi, afin d'être sûrs de toujours trouver un point du réseau à l'intérieur de la sphère, il faut prendre R égal au rayon de recouvrement du réseau. En pratique, on peut adapter R à la variance du bruit, notée σ^2: pour de faibles rapports signal sur bruit (SNR), on a besoin d'un grand rayon, pour des SNRs plus élevés, un petit rayon suffit puisque le point reçu est normalement très proche du point transmis. Cet algorithme de décodage peut être utilisé d'une façon

très efficace, aussi bien sur le canal gaussien que sur les canaux à évanouissement de Rayleigh, et sa complexité est indépendante de la taille de la constellation utilisée. Néanmoins, sa complexité limite son utilisation à des dimensions du réseau inférieures ou égales à 32.

Le décodage par sphères est enfin employé pour montrer les performances des réseaux de points les plus connus pour des dimensions égales à 1, 2 ,4, 8 et 16, représentées par les réseaux A_1, A_2, D_4, E_8 et BW_{16} respectivement. Les performances sont illustrées en taux d'erreur par mot par 2 dimensions, utlisé pour comparer des constellations de dimensions différentes, en fonction du rapport volume sur bruit. On montre aussi le taux d'erreur correpodant aux réseaux entiers de mêmes dimensions. Ainsi, il est facile de remarquer l'amélioration des performances avec l'augmentation de la dimension.

Jusqu'à ce point, les lattices ont été décrits en tant que constellations infinies. Toutefois, nous savons qu'en pratique, la transmission des données nécessite la détermination d'un ensemble fini de points. C'est pourquoi, dans le deuxième chapitre on s'intéresse à une opération essentielle liée aux lattices: le *shaping* ou la mise en forme.

Dans le chapitre 2, on traite donc le problème du shaping, qui consiste à définir un ensemble déterminé des points du lattice pour former un *lattice code*. Ce dernier est le résultat de l'intersection du réseau infini avec une zone de shaping, notée $\mathcal{B}$. Selon la forme de $\mathcal{B}$, on peut distinguer plusieurs mécanismes de shaping. La forme la plus simple est celle d'un hypercube, et on effectue dans ce cas un shaping hypercube. Le gain de shaping est la réduction en SNR nécessaire pour atteindre une certaine probabilité d'erreur par rapport à l'utilisation d'un shaping hypercube. Ce gain est limité par 1.53 dB, une valeur atteinte en utilisant un shaping hypersphère. Toutefois, ce dernier est complexe à implémenter. Le but du shaping est d'éviter la transmission de l'information avec une puissance trop élevée, en s'assurant que seuls les points du réseau appartenant à la zone de shaping $\mathcal{B}$ sont considérés pour la transmission.

Nous avons vu que l'encodage d'un réseau de points peut s'effectuer directement en multipliant la matrice génératrice par un vecteur d'entier $\mathbf{b}$. Or ceci pourrait aboutir à des points du réseau ayant une très grande énergie. Le shaping consiste donc à transformer le vecteur $\mathbf{b}$ en un autre vecteur $\mathbf{b}_s$, de sorte que la multiplication de ce dernier par la matrice génératrice fournit un point du réseau $\mathbf{x}_s$ à l'intérieur de la zone de shaping. Dans ce mauscrit, nous détaillons deux mécanismes de shaping: le shaping hypercube et ce qu'on appelle le *nested shaping* [80]. Ce dernier est le mécanisme qui nous permet d'avoir un gain proche de celui du shaping optimal avec une complexité abordable.

Le nested shaping est un mécanisme impliquant deux lattices: un certain lattice Λ, et un sous-lattice de Λ noté Λ_s. Si $\mathbf{G}$ est la matrice génératrice de Λ, une matrice génératrice de Λ_s sera prise en dilatant $\mathbf{G}$ d'un facteur L. Ce mécanisme consiste à faire le suivant: afin de minimizer l'amplitude de $\mathbf{x}_s$, il faut trouver, dans le réseau Λ_s, le point le plus proche de $\mathbf{x}$. Ceci est possible en appliquant un décodage par sphères au point $\mathbf{x}$ à l'intérieur de réseau Λ_s. Les points du réseaux transmis seront donc uniformément distribués à l'intérieur de la région de Voronoi de Λ_s.

Pour montrer les performances de cette méthode de shaping, ainsi que de celles d'un simple shaping hypercube, on les applique au réseau E_8, le réseau le plus dense dans 8 dimensions, et on les compare à une constellation QAM non codée pour des efficacités spectrale de 1, 2, 3 et 4 bits/dimension. Au début, on remarque que sans aucune opération de shaping, la QAM est plus performante pour toutes les efficacités spectrales, ce qui est normal car les pointss du réseaux E_8 transmis pourraientt avoir une très grande énergie. En appliquant un shaping hypercube, on obtient des courbes très proches des performances de la QAM, ce qui est normal car la QAM n'est autre qu'un lattice entier $\mathbb{Z}^2$ auquel on a appliqué un shaping cubique. L'amélioration des performances qu'on remarque pour des SNR élevés sont donc dûs au gain de codage de E_8 par rapport au lattice entier, puisque E_8 est plus dense. Ensuite, en appliquant un nested shaping, on remarque que les performances s'amérliorent encore plus et le gain augmente avec l'efficacité spectrale. Par contre, pour 1 bit/dim, on voit que la QAM est toujours plus performante. Ceci est dû à la partie décodage.

En fait, du côté récepteur, on a deux façons de décoder: soit on prend en considértion la zone de shaping et on effectue dans ce cas-là du lattice code decoding, soit on ne prend pas en compte cette région et on effectue un simple lattice decoding. La différence réside dans le fait que seul le premier permet de s'assurer que les mots décodés tombent à l'intérieur de $\mathcal{B}$. Nous avons vu que le décodage des lattices est effectué en utilisant l'algorithme de décodage par sphères. Or avec ce dernier, on ne peut pas préciser les bornes de la zone de shaping car on ne connaît pas les vecteurs $\mathbf{b}_s$ et on ne peut pas surveiller les coordonnées pour voir si les points décodés appartiennet à la region de shaping ou pas. Par conséquent, les vecteurs estimés par le décodage par sphères peuvent correspondre à des points de lattice n'appartenant pas à la zone de shaping. C'est pourquoi, nous avons proposé une modification de l'algorithme de décodage afin de vérifier que tous les mots décodés appartiennet bien à la constellation voulue.

L'algorithme consiste en fait à remplacer le sphère décodeur par un list sphère décodeur, qui prend en paramètres le point reçu, la matrice génératrice du lattice et la taille désirée de la liste. Les vecteurs à la sortie du décodeur sont rangés du plus

proche au plus loin en termes de distance euclidienne par rapport au point reçu. On commence par le premier vecteur de la liste et on lui applique une opération de shaping inverse pour en déduire le vecter entier initial. Ensuite, on prend le vecteur obtenu et on effectue une deuxième opération de shaping. Si ce qu'on obtient est égal au vecteur pris à la sortie du list sphère décodeur, donc ce vecteur correspond bien à un point du lattice à l'intérieur de la zone de shaping, sinon on prend le deuxième vecteur de la liste et ainsi de suite.

En regardant les performances du nouveau décodeur, on remarque une amélioration par rapport au décodeur initial, sutout pour les petites efficacités spectales. Par exemple, pour 1 bit/dim, le décodeur proposé permet de gagner environ 1 dB pour un taux derreur de 10^{-4}. Ce gain diminue en augmentant l'efficacité spectrale, et devient négligeable pour 4 bits/dim. Donc en augmentant l'efficacité spectrale, on diminue la probabilité d'avoir des mots décodés à l'extérieur de la zone de shaping avec un simple lattice decoding.

Jusqu' à maintenant, les performances du lattice ont été uniquement comparées à des constellations QAM non codées, on procède donc à une comparaison avec un schéma de modulation codée existant qui est celui de la LTE à courtes trames. Notre choix de courtes trames est basé sur le fait que leur utilisation possède certains avantages, comme par exemple diminuer la latence et augmenter la fiabilité de transmission. En LTE, le plus petit nombre de bits d'information qu'on peut utiliser est 40 bits. Le code utilisé est un turbo code possédant un rendement $\frac{1}{3}$. Ce rendement peut être varié grâce à un simple rate matching. Les modulations utlisées sont la 4-QAM, la 16-QAM et la 64-QAM. Concernant les lattices, deux cas on été utilisés dans la comparaison: le premier est celui du Gosset lattice E_8, et le second est un lattice de 16 dimensions, le BW_{16}. Pour E_8 on a 8 composantes réelles par mot de code, ce qui correspond à 4 composantes complexes. Ainsi, si on mesure la taille de trame par nombre de composantes complexes, on aura une taille de 4 pour E_8 et 8 pour BW_{16}.

Le but est donc de comparer le schéma LTE et lattice codes ayant des proches efficacités spectrales, ainsi qu'un nombres de bits transmis identique. Par exemple, pour 1 bit/dim, la LTE comprend une 16 QAM et un turbo code de rendement $\frac{1}{2}$. Dans ce cas, on a une trame de 24 symboles complexes. Pour cela, il faut transmettre 6 trames E_8 ou 3 trames BW_{16}. De manière similaire, on choisit les rendements de code et les modulations qui nous permettent d'obtenir des efficacités spectrales de 1,5 ,2 et 2,4 bits/dim, ainsi que les tailles de trames LTE correspondantes.

Les simulations présentant le taux d'erreur par trame en function du SNR montrent que pour la premier cas de 1bit/dim, la LTE est la plus performante. Cependant, il ne faut pas oublier qu'entre la taille de trame LTE et celle de E_8 et BW_{16} il y a un

facteur de 6 et 3 respectivement, ce qui fait perdre en performance pour les lattices. En passant à une efficacité spectrale plus grande, le lattice à 16 dimensions commence à devenir plus performant, pareil pour 2 bits/dim. En arrivant à une efficacité spectrale plus élevée, ce qui correspond à une seule trame BW_{16} et 2 trames E_8, on voit que la LTE est moins performante que les deux lattices. Il en ressort que sur le canal gaussien pour les hautes efficacités spectrales, les schémas reposant sur des réseaux de points de type E_8 ou BW_{16} offrent de meilleures performances que les constellations QAM associées à un turbo code. Ceci constitue un résultat intéressant puisque la tendance actuellement c'est d'aller vers les hautes efficacités spectrales.

Une autre remarque qui ressort de ces comparaisons réside dans le fait qu'une augmentation de la dimension du lattice permet de diminuer le taux d'erreur. Ainsi, pour mieux exploiter les lattices, on a intérêt à augmenter leur dimension. Toutefois, avec une grande dimension, la complexité de l'algorithme de décodage par sphères devient très grande et il est donc impossible de l'utiliser. C'est pourquoi, il faut avoir recours à d'autres méthodes d'encodage des réseaux de points.

En fait, au lieu d'être directement obtenu par sa matrice génératrice, un lattice peut aussi être obtenu par une modulation codée, donc en employant un ou plusieurs codes correcteurs d'erreurs. Comme indiqué précédemment, la constrction A est une méthode qui permet de générer un lattice Λ à partir d'un seul code linéaire $\mathcal{C}$. En fait, d'une manière générale, ce code est défini sur un corps fini $\mathbb{F}_p$ (avec p premier), et le réseau Λ est l'ensemble des points dont le modulo-p appartient au code $\mathcal{C}$. Pour les lattices à petites dimensions, on utilise des codes binaires ($p = 2$). Par exemple, le lattice E_8 est obtenu en utilisant le code de Hamming étendu $\mathcal{H}(8, 4, 4)$. Toutefois, dans ce cas-là, la distance euclidienne minimale qu'on peut obtenir est 2 maximum. Pour construire des bons lattices de plus grandes dimensions, il faut que la valeur de p soit plus grande, ce qui ajoute beaucoup de complexité au décodage. Pour cette raison, afin d'utiliser toujours des codes binaires, on fait appel à une construction multi-niveaux qui implique non pas un seul, mais plusieurs codes correcteurs d'erreurs binaires pour construire des lattice de plus grandes dimensions. Ceci n'est autre que la construction D.

Le chapitre 3 traite de la construction multi-niveaux des réseaux de points sur le canal gaussien. On commence par une explication des parties intégrales de la construction multi-niveaux:

- La partition des lattices
- Le choix des codes

- Le décodage

La partition des lattices est expliquée par l'exemple suivant: Prenons le réseau des entiers à une dimension $\mathbb{Z}$. $2\mathbb{Z}$, l'ensemble des entiers pairs, est un sous-lattice de $\mathbb{Z}$. Par défintion, la partition $\mathbb{Z}/2\mathbb{Z}$ est le nombre de décalages qu'il faut appliquer à $2\mathbb{Z}$ pour obtenir $\mathbb{Z}$. Comme $\mathbb{Z}$ est l'union des entiers pairs et des entiers impairs, c'est donc l'union de $2\mathbb{Z}$ et $2\mathbb{Z}+1$. Il s'ensuit que la partition $\mathbb{Z}/2\mathbb{Z}$ est égale à l'ensemble formé de deux éléments $\{0,1\}$, c'est donc une partition binaire. Pareil pour le réseau des entiers à deux dimensions $\mathbb{Z}^2$, qui est l'union du lattice $R\mathbb{Z}^2$ et du décalage de ce dernier par le vecteur $(1,1)$, avec $R = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ un paramètre de rotation. $\mathbb{Z}^2/R\mathbb{Z}^2$ est donc aussi une partition binaire.

En généralisant à une chaîne de lattices imbriqués $\mathbb{Z}, 2\mathbb{Z}, 4\mathbb{Z}$ jusqu'à $2^r\mathbb{Z}$, on obient une chaîne de lattices partitions à une dimension. Comme pour $\mathbb{Z}/2\mathbb{Z}$, on peut vérifier que les autres partitions sont aussi binaires. D'une manière générale, on prend une chaîne de lattices partitions binaires de dimension n, telle que chaque partition est formée de deux vecteurs, $\mathbf{0}$ et et $\mathbf{a}_i$. Les codes utilisés possèdent un alphabet égal à l'ordre des partitions. Donc dans le cas des partitions binaires, les codes sont eux-mêmes binaires, et chaque bit détermine le sous-groupe qui sera envoyé. Le lattice L résultant de cette construction multi-niveaux menée sur une chaîne de partitions $\Lambda_1/\Lambda_2/\cdots/\Lambda_r$ où chaque partition Λ_i/Λ_{i+1} est associée au code $\mathcal{C}_i(N,k_i)$, peut être décrit par la formule suivante:

$$L = \mathbf{a}_1\mathcal{C}_1 + \cdots + \mathbf{a}_{r-1}\mathcal{C}_{r-1} + \mathbf{z} \qquad \text{avec } \mathbf{z} \in (\Lambda_r)^N \tag{2}$$

Si n est la dimension des partitions et N est la longueur des codes utilisés, la dimension de L sera égale à nN.

À la réception, le décodeur est aussi un décodeur multi-niveaux où chacun des codes est decodé séparément, et le décodage sur un niveau dépend des niveaux précédents. Ce type de décodage conduit à une réduction considérable de la complexité par rapport à un décodage de maximum de vraisemblance. Les performances de ce schéma de codage/décodage multi-niveaux dépend fortement du choix des codes. Dans nos constructions, nous avons choisi d'employer les codes de Reed-Muller binaires vu qu'ils sont imbriqués par nature. Comme on a vu précédemment, les codes de Reed-Muller sont à l'origine utilisés pour former les fameuses lattices de Barnes-Wall. Dans ce type de lattices, le choix des codes est basé sur la règle de la *distance équilibrée*. Cette façon de construire a permis d'obtenir des bons lattices pour des dimensions allant jusqu'à 32, mais au-delà ce n'est plus le cas.

Comme nous nous intéressons à un décodeur multi-niveaux où chaque code est décodé séparément, nous suivons une autre règle qui consiste à adapter le rendement de chaque code au niveau auquel il est associé. C'est *la regle de capacité* qui consite à choisir les codes de sorte que leurs rendements s'approchent de la capacité du niveau correpondant.

Une fois on a décrit les principes de la construction multi-niveau, on procède par l'explication de la construction des réseaux utilisant des codes de Reed-Muller binaires avec des partitions de dimensions $n = 1, 2$ puis 4. Pour chaque valeur de n on détermine le nombre de niveaux nécessaires et leurs capacités respectives. Les simulations montrant les performances de réseaux de dimension 1024 montrent que augmenter n, et donc le nombre de niveaux, permet d'améliorer les performances en termes de taux d'erreur par mot, et ceci est d'autant plus vrai que la taille du réseau L est élevée.

Enfin, ce chapitre se termine par la description d'une méthode d'estimation des LLR en se servant de la distribution de *von Mises*. Le principal intérêt de cette méthode est qu'elle permet de remplacer la somme infinie nécessaire au calcul exact de LLR, par une fonction cosinus, et par suite diminuer la complexité. Les résultats montrent que les performances avec la distribution de von Mises sont pratiquement identiques à celles du calcul exact des LLR dans le cas d'un décodeur à deux niveaux sur le canal gaussien.

Dans le chapitre 4, on s'intéresse au codage multi-niveaux des réseaux de points, mais cette fois-ci sur le canal à évanouissements de Rayleigh par bloc. Sur ce type de canal, les meilleurs lattices sont ceux qui fournissent la diversité maximale, et par diversité on veut dire le nombre de composantes différentes entre deux signaux distincts de la constellation. Un outil efficace pour la construction de tels réseaux est la théorie des nombres algébriques. Pour obtenir un lattice algébrique, noté Λ_g, il faut donc se familiariser avec certaines notions algébriques, la première étant celle des corps de nombres. En fait, à partir d'un corps de nombre $K = \mathbb{Q}(\theta)$, on peut déduire l'anneau des entiers et la base correspondante. Le réseau sera obtenu en effectuant un plongement canonique de cette base dans l'espace des nombres réels ou complexes. Afin d'atteindre la diversité maximale, il faut se servir de corps de nombres complètement réels, c.à.d. où tous les plongements canoniques possèdent des images réelles.

Toutefois, le lattice algébrique obtenu de cette façon ne possède pas de forme déterminée. La forme la plus simple qu'on puisse lui donner est la forme cubique. Ceci revient à transformer Λ_g en une version du lattice entier tourné. Pour se ramener à ce cas, des opérations de rotation et de réduction de base doivent être effectuées.

Une fois ce réseau obtenu, on procède avec la construction multi-niveaux en employant des codes de Reed-Muller binaires. Comme dans le chapitre précédent, le rendement des codes appliqués à chaque niveau est choisi en appliquant la règle de capacité. Les résultats obtenus en termes de taux d'erreur par mot montrent que augmenter la dimension du réseau en utilisant la même chaîne de partitions, c.à.d. en augmentant la longueur N des codes employés, avait pour conséquence de dégrader les performances, et ce aussi bien sur les chaînes de partitions à deux dimensions que celles à quatre dimensions. Par ailleurs, ne pas appliquer la règle de capacité et donc utiliser des codes dont le rendement dépasse la capacité du niveau auquel ils sont associés entraîne une dégradation des performances, en particulier lorsque la dimension du réseau augmente. Enfin, comme pour le cas gaussien, augmenter n permet aussi d'améliorer les performances en termes de taux d'erreur par mot.

Conclusions

En conclusion, cette thèse nous permet de tirer les résultats suivants:

- Pour les lattices à petites dimensions:
  - Le shaping est une opération indispensable afin de réduire la puissance de transmission.
  - Le codage par réseaux de points est un bon choix pour la transmission des courtes trames, surtout en grandes efficacités spectrales, comparé à un schéma de QAM codée par Turbo codes.
- Pour la construction multi-niveaux:
  - Augmenter le nombre de niveaux permet d'améliorer les performances.
  - En appliquant la règle de capacité avec un décodeur multi-niveaux, on obtient des taux d'erreur par mot qui restent pratiquement invariants avec l'augmentation de la dimension des lattices.

Perspectives

Sachant que la construction multi-niveaux a été réalisée sans aucune contrainte de puissance, nous considérons, en premier lieu, l'application d'une mise en forme hypercube afin d'obtenir des constellations finies qui satisfont une certaine contrainte de

puissance. Le résultat sera ensuite comparé aux constellations QAM pour différentes efficacités spectrales.

Tout au long de cette thèse, les lattices multi-niveaux sont le résultat de la Construction D employant des codes linéaires binaires imbriqués. C'est pourquoi, nous envisageons d'éliminer cette contrainte de codes imbriqués, et employer ce qui a été récemment proposé dans [46] en tant que Construction π_A. L'élimination de cette contrainte permet de simplifier l'affectation des rendements, et par suite la construction des réseaux. En plus, la Construction π_A entraîne plus de flexibilité puisqu'elle permet d'utiliser des codes définis sur des corps différents. Cette construction a été montrée d'atteindre la capacité sur un canal gaussien avec le décodeur multi-niveaux sous-optimal.

Une autre perspective serait aussi l'implémentation des lattices pour des applications multi-utilisateurs, comme ça a déjà été fait dans le schéma Compute-and-forward [66].

Contents

List of abbreviations **xxix**

List of symbols **xxxi**

Introduction **1**

1 Lattices **7**

1.1 Introduction 7

1.2 Lattices in digital communications: A Brief History 9

1.3 Definitions and Parameters 11

1.4 Lattice Construction 17

1.4.1 Construction A 17

1.4.2 Construction B 18

1.4.3 Construction D 19

1.4.4 Construction D' 20

1.4.5 Barnes-Wall lattices and Reed-Muller codes 21

1.5 The Sphere Decoder algorithm 23

1.5.1 The sphere decoding algorithm 23

1.5.2 The sphere decoder with fading 26

1.6 Infinite lattices on the AWGN channel 27

1.7 Conclusion 30

2 Lattice Shaping **33**

2.1 Introduction . . . 33

2.2 Shaping mechanisms . . . 34

2.2.1 Hypercube shaping . . . 37

2.2.2 Nested shaping . . . 38

2.2.3 Comparison . . . 40

2.3 Proposed receiver algorithm . . . 41

2.3.1 Proposed algorithm . . . 42

2.3.2 Simulation results . . . 43

2.3.3 Influence of the list size . . . 44

2.3.4 Shaping on the Rayleigh fading channel . . . 45

2.3.5 Comparison with short packets LTE . . . 46

2.4 Conclusion . . . 48

3 Multilevel Lattice Coding **51**

3.1 Introduction . . . 51

3.2 Lattice partitions and construction D . . . 53

3.3 Appropriate code choice: Capacity Rule . . . 54

3.4 Multilevel Lattice construction using Reed-Muller codes . . . 57

3.4.1 One-dimensional lattice partitions . . . 58

3.4.2 Two-dimensional lattice partitions . . . 62

3.4.3 Four-dimensional lattice partitions . . . 65

3.5 LLR estimation using the von Mises distribution . . . 71

3.5.1 The von Mises distribution . . . 72

3.5.2 Simulation results . . . 74

3.6 Conclusion . . . 75

4 Multilevel Code Design for Rayleigh Block Fading Channels **79**

4.1 Introduction 79
4.2 Modulation diversity and product distance 80
4.3 Algebraic Number Theory 81
4.3.1 Algebraic number fields 82
4.3.2 Integral basis and Canonical Embedding 85
4.3.3 Totally real algebraic number fields 88
4.3.4 Ideal lattices 91
4.3.5 Construction of $\mathbb{Z}^n$ lattices 93
4.4 Multilevel Construction using binary Reed-Muller codes 94
4.4.1 Two-dimensional lattice partition chain 97
4.4.2 Four-dimensional lattice partition chain 100
4.5 Conclusion 102

Conclusions and perspectives **107**

A The $|\mathbf{u}|\mathbf{u}+\mathbf{v}|$ construction **111**

B Soft-input decoding for Reed-Muller codes **113**

C Commands in Sage **117**

Bibliography **121**

List of Figures **129**

List of Tables **133**

List of Publications **135**

List of abbreviations

AWGN	Additive White Gaussian Noise
BMS	Binary Memoryless Symmetric
FEC	Forward Error Correcting
i.i.d.	independent and identically distributed
LTE	Long Term Evolution
ML	Maximum Likelihood
MCL	Multilevel Coding
MSD	Multistage Decoding
MMSE	Minimum Mean Square Error
MTC	Machine Type Communications
NWER	Normalized Word Error Rate
QAM	Quadrature Amplitude Modulation
SD	Sphere Decoder
SNR	Symbol-to-Noise Ratio
VNR	Volume-to-Noise Ratio
WER	Word Error Rate
fcc	face centered cubic
b/2D	bit per 2 dimensions

List of symbols

General

x, y	Scalars
$\mathbf{x}, \mathbf{y}$	Vectors
$\mathbf{x}^T$	$\mathbf{x}$ transpose
$\mathcal{R}_e(x)$	Real part of x
$\mathcal{I}_m(x)$	Imaginary part of x
V_n	Volume of an n-dimensional sphere of radius 1
$\mathbb{R}^n$	The real field of dimension n
$\mathbb{Z}^n$	The ring of integers of dimension n
$\mathbb{Q}$	The field of rational numbers

Lattices

Λ	Lattice
$\mathbf{G}$	Generator matrix
$\mathbf{G_r}$	Gram matrix
$\mathbf{H} = \mathbf{G}^{-1}$	Parity-check matrix
$\mathbf{g_1}, \cdots, \mathbf{g_n}$	Lattice basis vectors
modΛ	modulo lattice operation
$\mathbf{b}$	Integer vector
$\Lambda_\mathbf{a}$	coset of Λ
$\mathcal{R}(\Lambda)$	Fundamental region of a lattice point λ
$\mathcal{V}(\lambda)$	Voronoi region of Λ
Λ'	Sublattice of Λ
det(Λ)	Determinant of Λ
$K(\Lambda)$	Kissing number of Λ
V(Λ)	Volume of Λ

$\mathrm{V}(\Lambda)^{2/n}$	Normalized Volume of Λ
$\gamma(\Lambda)$	Nominal coding gain of Λ
$d^2_{min}(\Lambda)$	Minimum Euclidean distance between lattice points of Λ
$\rho(\Lambda)$	Covering radius
$r(\Lambda)$	Packing radius
Δ	Lattice density
$\mathcal{C}(N,k,d)$	Code of length N, dimension k and minimum Hamming distance d
R	Code rate
B_d	Number of codewords in $\mathcal{C}$ with minimum Hamming distance d
$\mathcal{RM}(r,m)$	Reed-Muller code of order r and length $N = 2^m$
$\mathbf{I}_n$	Identity matrix of dimension n
C	Channel capacity
A_2	Hexagonal lattice
BW_n	Barnes-Wall lattice of dimension n
P	signal power
σ^2	Noise variance per dimension
$\mathcal{B}$	Shaping region in $\mathbb{R}^n$
η	Spectral efficiency in bits per real dimension

Multilevel lattice coding and Algebraic Number Theory

L	Final lattice resulting from the multilevel construction
F	Diversity order
$d_{p,min}$	Minimal product distance
K	Number field
$\mathcal{O}_K$	Ring of integers of the number field K
$N(x)$	Norm of the algebraic number x
$Tr(x)$	Trace of the algebraic number x
d_K	Discriminant of K
σ_i	Canonical Embedding of K into $\mathbb{C}$

Introduction

Communication represents an essential and fundamental human need. People have always longed to stay connected in order to survive, build relationships and simply get more out of life. Before technology took over, the human communication evolved from primitive means such as smoke signals, homing pigeons and signaling flags, to more developed methods that made communication much easier: that's when telephones, radios and televisions were invented. Nowadays, technology has significantly changed, and the human communication has expanded at a scale never before imagined.

The generations of mobile technology have evolved from analogue to LTE within less than two decades, each generation being motivated by the need to address the gaps of its predecessor. With the demands for wireless data services growing exponentially, LTE needs to be enhanced, and the 5^{th} generation of mobile technology (5G), that is expected to be operational around 2020, needs to hold the potential of fulfilling the society's ever-growing requirements.

Communication systems

The basic block diagram of a communication system, used to reliably transmit information from source to sink over a noisy channel, is shown in Figure 1. In this thesis, we are interested in the channel encoding and modulation blocks, with obviously the corresponding channel decoder and demodulator at the receiver side. Generally speaking, channel encoding consists in introducing some redundancy in the information sequence of length k, and thus convert it into a codeword $\mathbf{c} = (c_1, \cdots, c_n)$ of length $n > k$. Modulation is the operation thanks to which the codeword $\mathbf{c}$ is converted into a point $\mathbf{x} \in \mathbb{R}^n$ that can be physically transmitted over the communication link, and result in the received vector $\mathbf{y}$ at the channel output.

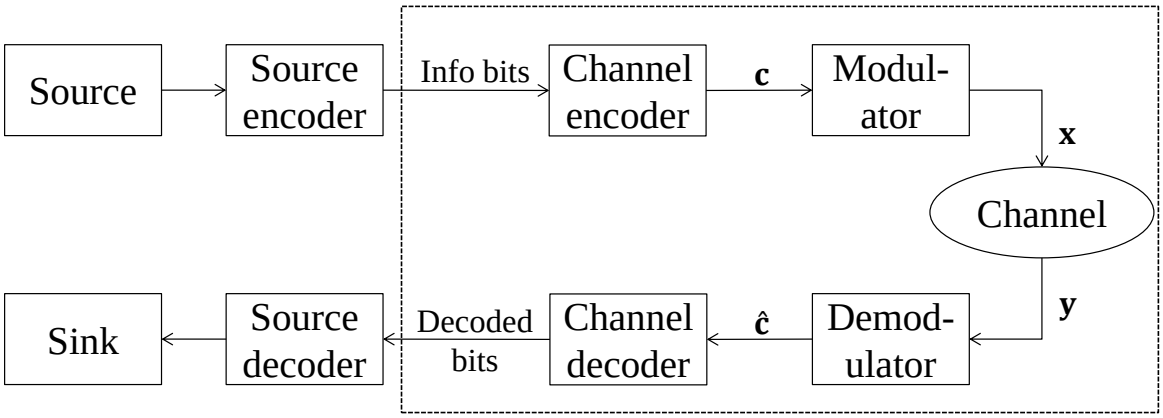


Figure 1 – Basic block diagram of a digital communications system.

For any given communication channel, the maximal rate at which information can be sent with vanishing error probability is known as the *channel capacity*, and is denoted by C. On the discrete Additive White Gaussian Noise (AWGN) channel model, the received vector is: $\mathbf{y} = \mathbf{x} + \mathbf{w}$, where $\mathbf{w}$ is a vector of n independent and identically distributed (i.i.d.) Gaussian noise variables. Shannon proved that achieving the quantity C on the AWGN channel is possible by using a set of independent, random and i.i.d. distributed codewords with $n \to \infty$. This result marked the beginning of a revolution in channel coding, as it stimulated the design of various codes: algebraic codes, convolutional codes, trellis codes, LDPC codes, turbo codes and polar codes. Polar codes are the first provably capacity-achieving codes for the class of binary memoryless symmetric channels (BMS) [8].

Lattices

On the other hand, there exist some interesting entities, known as *lattices*, endowed with a periodic and linear structure, that began to be applied to the communication problem in the early 1970's. Defined as a discrete additive subgroup of $\mathbb{R}^n$, lattices are the Euclidean space counterpart of linear codes in the Hamming space. Just like linear codes, lattices can be expressed by a generator matrix that encodes integer vectors to $\mathbf{x} \in \mathbb{R}^n$. Thus, lattices combine channel encoding and modulation into one single operation. One of the reasons why lattices are expected to play a prominent role in future communication systems is the fact that they solved the decades-old problem of achieving full capacity on the AWGN channel.

Motivated by the recognition of lattice coding as a promising approach for digital transmissions, this work is dedicated to the analysis of lattice coding performance under

different scenarios: the studies are carried out over both AWGN and Rayleigh fading channels, with restriction to the point-to-point communication type.

Lattice encoding is performed either by straightforward encoding of an integer vector into a lattice point using the lattice generator matrix, or by resorting to a coded modulation scheme. In the latter, the lattice is obtained by lifting one or more linear codes into the Euclidean space. In this thesis, we focus on the former lattice encoding method in Chapters 1 and 2, which hence form the first part of the thesis, and the latter encoding operation is studied in Chapters 3 and 4 that belong to the second part.

In the first part, one of the challenges met while employing lattices for digital transmissions is addressed. This challenge consists in finding a finite set of lattice points to be used while avoiding signals with high transmission power. This operation is known as lattice shaping. The shaping operation is studied, with a focus on the decoding process where the receiver is bound to check whether the estimated lattice points belong to the initially considered set or not.

The second part is motivated by the need for building lattices of higher dimensions in order to increase the lattice coding gain [26]. Therefore, we resort for multilevel lattice coding that provides the ability to build such lattices with the ease of working with binary codes. For this type of lattice construction, many problems should be tackled depending on the considered channel conditions. This work includes constructions over both AWGN and Rayleigh block fading channels, where a set of nested binary linear error-correcting codes is assigned to the different levels. The component codes choice, as well as the implemented decoding algorithm are a crucial point in the system's performance.

Thesis outline and contributions

The manuscript is divided into four chapters. The contents and contributions of each one of them are summarized in the following.

Chapter 1 serves as an introduction to lattices. We first provide a historical overview, showing how lattices have been investigated in the digital communication world since the early 1970's. The main lattice parameters are then described, providing the reader with some notions that will be of interest for the remainder of the thesis. We then recall the lattice constructions using binary error-correcting codes, known as Constructions A, B, D and D', and show how Construction D is applied to build Barnes-Wall lattices using nested binary linear Reed-Muller codes. We also re-

call the sphere decoding algorithm that can be efficiently employed, on both Gaussian and Rayleigh fading channels, to decode lattices of dimensions up to 32. Finally, the normalized word error rate performance over the AWGN channel is studied for the best-known lattices.

Chapter 2 introduces a central operation that forms an effective means of guaranteeing minimal transmission power by turning the infinite lattice into a lattice constellation with finite number of lattice points. This operation is referred to as lattice shaping. We first describe the lattice shaping operation and detail two shaping mechanisms, hypercube and nested shaping, that are applied to the 8-dimensional Gosset lattice E_8 and compared to the uncoded QAM constellation for different spectral efficiencies. Then, we move to the decoding part of the transmission system and propose a modified version of the sphere decoding algorithm that takes into account the inevitable shaping operation at the transmitter. The novel results in this chapter were published in [7], and they include:

- A lattice code decoder based on re-shaping that helps reduce the error rate.
- Performance improvement at high spectral efficiencies, in terms of Frame Error rate, compared to LTE baseline of short frame length, using lattices of small dimensions.

In Chapter 3, the focus is switched towards building lattices by means of multiple levels of nested binary error-correcting codes, i.e., construction D. We first recall the notions that lie at the core of our multilevel construction: lattice partitions and the capacity rule. Then, following the construction guidelines provided by Forney *et al.* in [42], we explain the steps for building a lattice using binary Reed-Muller codes over lattice partitions of dimensions $n = 1, 2$ and 4. For each value of n, several issues are addressed, namely the sufficient number of levels and their respective capacities (thus the upper-bound on the code rates to be used). We provide the capacity curves, system model and simulation results for each case. At the end of this chapter, results related to an LLR estimation method using the von Mises distribution are explained. The method aims at a reduction in complexity by replacing the infinite sums in the exact LLR calculation formula by a cosine function. These results were the object of two publications [6] and [5].

While the work in Chapter 3 is carried out over the AWGN channel, Chapter 4 is dedicated to multilevel lattice coding on the Rayleigh block fading channel. The lattice construction in this case is based on algebraic number theory, which is a key enabler for providing maximum diversity. We therefore begin by listing and explaining some number-theoretical concepts, paving the way to their implementation in the

algebraic lattice construction. The multilevel design is then developed for two and four-dimensional lattice partitions. The novelty in this chapter, submitted to [4], includes:

- Applying the multilevel lattice coding scheme to algebraic lattice partition chains built from totally real algebraic number fields.
- Lattices built using multilevel construction show a word error rate performance that barely degrades when increasing the lattice dimension.

Finally, conclusions on the different ideas discussed in the manuscript are derived in a final chapter, along with some further research perspectives on the applied studies.

The manuscript contains three appendices. Some methods are developed in details and placed in appendix for further reference. Also included are Sage commands for readers interested in implementing the examples listed in Chapter 4.

CHAPTER 1

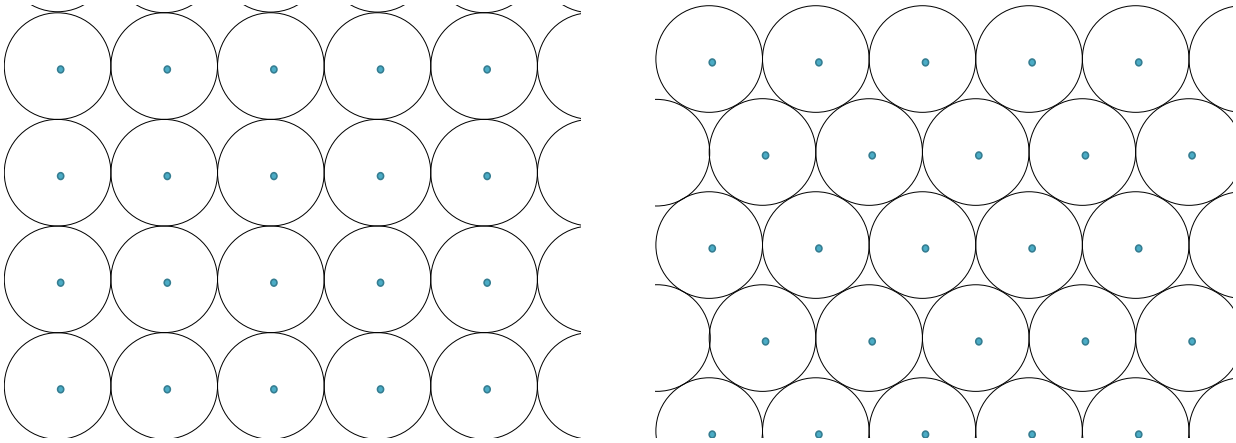
Lattices

1.1 Introduction

Generally speaking, a lattice is an arrangement of sphere centers having a periodic structure in the n-dimensional Euclidean space. Used in mathematics for works related to number theory, quadratic forms and the geometry of numbers, lattices also have connections with chemistry, since crystallographers study three-dimensional lattices and relate them to the physical properties of common crystals. In cryptography, lattices are the cornerstone of some of the strongest key algorithms. In digital communications, they serve as a tool for building modulations over both the Additive White Gaussian Noise (AWGN) and the fading channels.

The study of lattices began with the sphere packing problem, which has long been a topic of interest for mathematicians. On the AWGN channel, this problem is equivalent to maximizing the codewords for a given minimum distance of a linear code. The sphere packing problem consists in finding out how densely a large number of identical spheres can be packed together in a given space, in other words, it consists in finding the best way to pack oranges in a box, or even a huge warehouse. In two dimensions for example, there are several, efficient and non-efficient, ways to do that. Figures 1.1a and 1.1b show a rectangular and hexagonal two-dimensional arrangements respectively. It is clear that the hexagonal packing is much more efficient, since we have less space wasted between the spheres. Moving to three dimensions, we have the orange packing shown in Figure 1.2, where the centers of the oranges fall on the so-called face centered cubic *fcc* lattice. The question does not stop here, and we can generalize the sphere packing problem to n dimensions.

A sphere packing is called a lattice if the centers of the spheres form a discrete group closed under addition in $\mathbb{R}^n$. In the twentieth century, mathematicians have developed dense sphere packings based on lattices for higher dimensions (for example, the Barnes-Wall lattice [10] and the Leech lattice [59]). The book of Conway and Sloane



(a) Rectangular arrangement (b) Hexagonal arrangement

Figure 1.1 – Two-dimensional packings.

Figure 1.2 – Orange packing in 3 dimensions.

[26] is a good reference for what is known about this topic, it also serves as a bridge between the communications world and the mathematically oriented literature about lattices. In digital communications, lattices have attracted a great deal of interest ever since coded modulation schemes were developed in 1980. In fact, it was shown that high-dimensional modulations carved out of lattices can be used for transmissions at high rates with a significant improvement in error performance.

After a brief history on lattices and their evolution throughout time in Section 1.2, we introduce, in Section 1.3, the main notions and parameters that will help us get a better understanding about lattices and their characteristics. In Section 1.4 we

explain the lattice constructions using error-correcting codes, followed by a description of the sphere decoding algorithm in Section 1.5. We then give some simulation results showing the performance on the AWGN channel, in terms of Normalized Word Error Rate, of the most-known lattices in Section 1.6. Finally, the chapter is concluded in Section 1.7.

Note that throughout the thesis, the term "log" denotes the binary logarithm $\log_2$.

1.2 Lattices in digital communications: A Brief History

In 1948, Shannon's seminal work *A Mathematical Theory of Communication* [78] inaugurated the long road to channel capacity, i.e., the maximal rate at which information can be reliably transmitted over a communications channel. In his paper, Shannon showed that for any transmission rate R smaller than the channel capacity C, small error probability can be arbitrarily approached using intelligent coding techniques. Conversely, if R is greater than C, no coding technique can achieve reliable transmission. Shannon proved that the capacity of a discrete AWGN channel with noise variance σ^2 per dimension and power constraint P per dimension is given by the explicit formula:

$$C = \log(1 + \frac{P}{\sigma^2}) \qquad \text{bits per 2 dimensions (b/2D).} \tag{1.1}$$

Unfortunately, Shannon's theorem is not a constructive proof: it merely proves that such codes exist, but does not give a method to construct them, or show how complex it might be to implement them.

Ever since the paper was published, information theorists have been challenged to find structured codes that can achieve Shannon's capacity at affordable complexity. The following years, the 1970's and 1980's in particular, were therefore marked by the blossoming of information theory. Lattices were already known at that time. In fact, the interest in lattice theory arose in the second half of the nineteenth century, while its contribution was mainly restricted to mathematical purposes. Particularly, significant examples include the works on the geometry of numbers and quadric forms of Minkowski [65], Zolotarev [53] and Voronoi [88]. This theory, however, had to be confronted with some hostility and the notions that were developed at the time were tucked away until they regained consideration around the 1930s.

Even though Shannon's work had no mention of lattices, it indicated that there exist sphere packings in high dimensions with sufficiently high density that can approach

the channel capacity. Lattices have then emerged as a powerful tool for the design of structured codes for the AWGN channel. The application of lattices to this type of channels originated in the work of de Buda [27], where it was asserted that lattices can approach the capacity of an AWGN channel. In fact, based on the Minkowski-Hlawka theorem, de Buda showed the existence of spherical lattices that can achieve a rate of $\log(\frac{P}{\sigma^2})$ b/2D, which is quite close to the maximal rate. Loeliger [63] reproved this result by deriving a version of the Minkowski-Hlawka theorem based on standard averaging for linear codes applied to lattices (Construction A). Note that these two results were proved under *lattice decoding*, i.e., decoding to the nearest lattice point in the infinite lattice (without taking into account the shaping boundaries of the lattice).

In another paper [28], de Buda proved that lattice codes can actually achieve the full rate of $\log(1 + \frac{P}{\sigma^2})$, and for that he considered a "thick-shell" shaping, rather than a spherical shaping, along with a lattice code decoder. Later, a mistake in [28] was pointed out by Linder *et al.* in [61], and the thick shells were replaced with thin shells. However, because of this thin bounding region, the corresponding codes lose some of their structure, and we'd be talking more about random spherical codes rather than lattice codes.

In [69], Poltyrev brought forth the notion of AWGN-good lattices while considering a channel without restrictions. However, in this case, talking about transmission rate is meaningless, because with no restrictions the rate can be increased without any limit. That's why, Poltyrev introduced the notion of generalized capacity, defined in [69] as the maximal codeword density that can be reliably recovered. He also proved the achievability of $\log(\frac{P}{\sigma^2})$ b/2D.

For the power-constrained AWGN channel, Urbanke and Rimoldi [85] proved that maximal capacity can be achieved through lattice codes provided that the decoder uses *lattice code decoding* (takes into consideration the shaping region). This was also proved by Forney in [39]. Thus, they left open the question of whether lattice decoding can achieve the channel capacity on an AWGN channel. The answer was given later by Erez and Zamir in [30] where they proved that lattice coding and decoding can indeed achieve the capacity of the Gaussian channel. In fact, with two nested lattices (one used for coding and the other for shaping), an MMSE factor at the receiver, a random dither at the transmitter and a mod-Λ decoding of the scaled received vector, the SNR can be enhanced by "one" to achieve the full Gaussian capacity $\log(1 + \frac{P}{\sigma^2})$ b/2D. A few years later, Ling and Belfiore proposed a scheme in [62] using a discrete Gaussian distribution, that requires neither shaping nor dithering, and which, under lattice decoding, achieves the capacity of the Gaussian channel.

Forney published a series of work through the 1980s and 1990s where he addressed several lattice-related problems. He provided a long survey about coset codes and discussed the relation between lattices and binary codes in [36] and [37]. He extended the idea of Voronoi-shaped codebooks introduced by Conway and Sloane in [25] to the high SNR regime and multidimensional constellations [38], [41]. He introduced Trellis shaping in [35]. In [42], he proposed the modulo-lattice channel and showed that multilevel lattices based on binary lattice partition chains can achieve the generalized capacity [69] or the sphere bound if the codes on each level are carefully chosen.

More recently, versions of lattices inspired by analogies to constructions of binary linear error-correcting codes started to appear. For instance, a form of lattice codes, inspired from LDPC codes, was proposed in [79]: Low-density lattice codes, the corresponding decoder was described in [93]. Low-density parity-check lattices were introduced in [73]. Built from nested LDPC codes, these lattices are endowed with a sparse parity-check matrix. Convolutional lattice codes, also known as signal codes, were introduced in [77]. Construction D was used on Turbo codes to build efficient turbo lattices [74], [73], and polar lattices using a multilevel structure were introduced in [91], [92].

Lattices have been extensively used in different applications such as the design of codes for the Rayleigh channels [22], multilevel flash memories [57], the Gaussian or Rayleigh wiretap channel [17] [24], relaying communications [67, 31, 81], wireless networks suffering from interference [66] and the multiple access relay channel [58].

Motivated by the interesting results of studies carried out on lattice coding theory, this thesis serves as a tool for harnessing lattices. More specifically, efficient lattice designs are described and employed over both AWGN and Rayleigh fading channels.

After this overview on lattices and the different related studies, we now consider their most important features and characteristics.

1.3 Definitions and Parameters

In this section, we introduce the main lattice parameters that we judge important in order to understand what is a lattice, its characteristics and what it represents. For more details, we refer the reader to [26]. Note that we will adopt the column vector convention.
Let us first begin by giving the definition of a lattice:

Definition 1.3.1. *Lattice*
An n-dimensional lattice Λ is a discrete subgroup of the real Euclidean p-space $\mathbb{R}^p$, where $n \leq p$. For example, the set of integers $\mathbb{Z}$ is a discrete subgroup of $\mathbb{R}$, so $\mathbb{Z}$ is a one-dimensional lattice.
Λ can also be defined as the set of all integer linear combinations of n independent vectors in $\mathbb{R}^p$: $\mathbf{g_1}, \mathbf{g_2}, \cdots, \mathbf{g_n}$. Every point $\mathbf{x} \in \Lambda$ can be written as:

$$\mathbf{x} = \mathbf{g_1}b_1 + \mathbf{g_2}b_2 + ... + \mathbf{g_n}b_n; b_i \in \mathbb{Z}.$$

This is referred to as a $\mathbb{Z}$-module generated by the vectors $\{\mathbf{g_i}\}_{i=1}^{n}$.

Being an additive group, Λ contains the origin $\mathbf{0}$, and since the topology of a group is invariant with the translation of any of its elements, we can study the properties of the origin point $\mathbf{0}$, and generalize to any point $\mathbf{x} \in \Lambda$.

Definition 1.3.2. *Coset*
The coset of a lattice is the set of points obtained after a specific vector is added to each lattice point:

$$\Lambda_{\mathbf{a}} = \mathbf{a} + \Lambda = \{\mathbf{a} + \mathbf{x} : \mathbf{x} \in \Lambda, \mathbf{a} \in \mathbb{R}^n\}.$$

Note that the coset itself is not a lattice, since it is not closed under addition and reflection, and it does not contain the origin.

Definition 1.3.3. *Lattice constellation*
An n-dimensional lattice constellation is the finite set $(\Lambda + \mathbf{a}) \cap \mathcal{R}$ of points in a translate $\Lambda + \mathbf{a}$ of an infinite n-dimensional lattice Λ that lies in a certain bounding region $\mathcal{R} \in \mathbb{R}^p$.

Definition 1.3.4. *Sublattice*
A subset Λ' of Λ that is also a lattice is called sublattice. The quotient group Λ/Λ' has then a finite order $M = |\Lambda/\Lambda'|$, and Λ is the disjoint union of M cosets of Λ'

$$\Lambda = \cup_{\mathbf{a} \in A} \Lambda' + \mathbf{a}$$

where A is the set of M coset representatives for the cosets of Λ' in Λ. The notion of sublattices will be seen in more details in Chapter 3.

Definition 1.3.5. *Lattice basis*
The set of vectors $\mathbf{g_1}, \mathbf{g_2}, \cdots, \mathbf{g_n}$ forms the basis of the lattice Λ.

Definition 1.3.6. *Generator matrix*
The vectors $\mathbf{g_1}, \mathbf{g_2}, \cdots, \mathbf{g_n}$ form the columns of the $p \times n$ lattice generator matrix $\mathbf{G}$,

which can be then written as:

$$\mathbf{G} = \begin{pmatrix} \mathbf{g_1} & \cdots & \mathbf{g_n} \end{pmatrix} = \begin{pmatrix} g_{11} & \cdots & g_{n1} \\ \vdots & & \vdots \\ g_{1p} & \cdots & g_{np} \end{pmatrix} \tag{1.2}$$

Hence, any lattice point $\mathbf{x}$ is a column vector that equals: $\mathbf{x} = \mathbf{Gb}$, with $\mathbf{b} = (b_1, \cdots, b_n)^T$ a point in $\mathbb{Z}^n$.

When $\mathbf{G}$ is the identity matrix, we obtain the integer lattice $\Lambda = \mathbb{Z}^n$, also called the *cubic lattice* or $\mathbb{Z}$ lattice. Any lattice can be viewed as a linear transformation, by the generator matrix, of the integer lattice:

$$\Lambda = \mathbf{G}\mathbb{Z}^n. \tag{1.3}$$

Definition 1.3.7. *Gram matrix*
The Gram matrix $\mathbf{G_r}$ is defined by:

$$\mathbf{G_r} = \mathbf{G}^T\mathbf{G}. \tag{1.4}$$

The elements of $\mathbf{G_r}$ correspond to all the possible inner products $(\mathbf{g_i}, \mathbf{g_j})$ between all generating vectors. The Gram matrix is a symmetric, positive-definite matrix, since the elements of the diagonal represent the square norm of the basis vectors elements.

Proposition 1. *Change of basis*
There are different ways of choosing a basis for a given lattice as shown in Figure 1.3, where the represented two-dimensional lattice can have for example $\{\mathbf{g_1}, \mathbf{g_2}\}$ or $\{\mathbf{g_1}, \mathbf{g'_2}\}$ as a basis.
Two matrices $\mathbf{G}$ and $\mathbf{G'}$ generate the same lattice, if and only if:

$$\mathbf{G'} = \mathbf{GT}$$

where $\mathbf{T}$ is a unimodular matrix, i.e., an integer matrix with $\det(\mathbf{T}) = \pm 1$.

Since a lattice basis is not unique, a question that may occur to our minds is : *What is the best lattice basis.* The answer is that, in general, the best basis for a given lattice is the basis that has the following properties:

- The basis vectors $\mathbf{g_1}, \mathbf{g_2}, \cdots, \mathbf{g_n}$ have the shortest Euclidean norm.
- the basis vectors are nearly orthogonal.

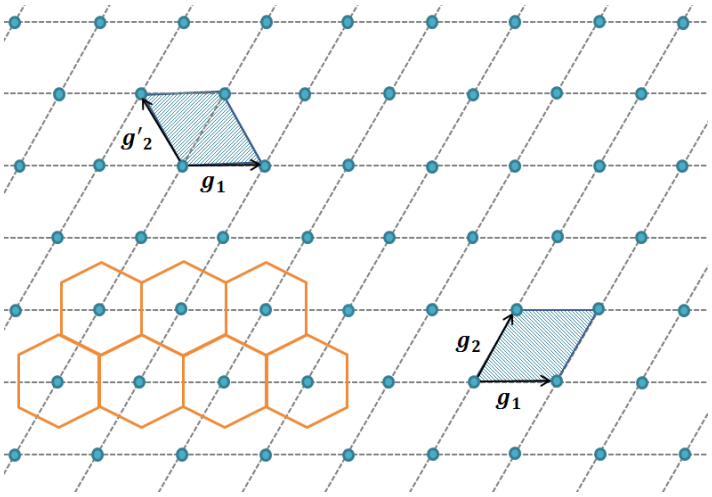


Figure 1.3 – Hexagonal lattice.

Definition 1.3.8. *Fundamental Region*
The parallelotope consisting of the points

$$r_1\mathbf{g_1} + \cdots + r_n\mathbf{g_n} \qquad (0 \leq r_i < 1) \tag{1.5}$$

is a fundamental parallelotope. Figure 1.3 shows the fundamental parallelotope determined by the two vectors $\mathbf{g_1}$ and $\mathbf{g_2}$ of a two-dimensional lattice. A fundamental parallelotope is an example of a *fundamental region* for the lattice, that is, a building block which when repeated many times covers the whole space with only one lattice point in each copy.

The fundamental parallelotope of a lattice is not unique, since it is determined by the choice of the basis. However, the volume of the fundamental region is uniquely determined by Λ. If $\mathbf{G}$ is a full rank matrix ($n = p$), the fundamental volume is equal to the absolute value of the *determinant* of $\mathbf{G}$:

$$\mathrm{V}(\Lambda) = |\det(\mathbf{G})|. \tag{1.6}$$

Note that the *Normalized volume* of an n-dimensional lattice Λ is defined as $\mathrm{V}(\Lambda)^{\frac{2}{n}}$ and may be regarded as the volume of Λ per 2 dimensions.

Definition 1.3.9. *Determinant of a lattice*
The determinant of Λ is defined to be the determinant of the Gram matrix:

$$\det(\Lambda) = \det(\mathbf{G_r}). \tag{1.7}$$

Hence, according to the previous notions, it is necessary to remember that there are different ways to represent the same lattice. Therefore, knowing the Gram or the

generator matrix does not allow us to determine the corresponding lattice. A lattice's invariants, such as the volume or the dimension can help, but two equal determinants do not necessarily correspond to similar lattices.

Definition 1.3.10. *Similarity*
Two lattices are similar (or equivalent) if one of them is obtained from another by a rotation, reflection or a change of scale. The corresponding generator matrices $\mathbf{G}$ and $\mathbf{G}'$ are related by the following formula:

$$\mathbf{G}' = c\mathbf{UGB}$$

where c is a nonzero constant, $\mathbf{U}$ is a matrix with integer entries and determinant ± 1, and $\mathbf{B}$ is a real orthogonal matrix (with $\mathbf{BB}^T = \mathbf{I}_n$).

Definition 1.3.11. *Voronoi region*
The *Voronoi region* $\mathcal{V}(\mathbf{x})$ of a lattice point $\mathbf{x}$ is the set of points closer to $\mathbf{x}$ than to any other lattice point:

$$\mathcal{V}(\mathbf{x}) = \{\mathbf{z} \in \mathbb{R}^p, ||\mathbf{z} - \mathbf{x}|| \leq ||\mathbf{z} - \mathbf{x}_2|| \ \forall \mathbf{x}_2 \in \Lambda\}. \tag{1.8}$$

The Voronoi regions thus span the Euclidean space entirely, filling the space between the packing spheres. The periodic aspect of a lattice makes all the Voronoi cells a shifted version of the fundamental Voronoi cell ($\mathcal{V}_0$), that is the Voronoi cell associated to the origin ($\mathbf{x} = 0$). The Voronoi region of the hexagonal lattice is shown in Figure 1.3.

Definition 1.3.12. Minimum distance
The notation $d_{min}(\Lambda)$ is used to denote the length of the shortest nonzero vector of the lattice Λ. It also refers to the minimum Euclidean distance between lattice points.

Definition 1.3.13. Coding gain
The nominal coding gain $\gamma(\Lambda)$ of the lattice Λ is defined by:

$$\gamma(\Lambda) := \frac{d_{min}^2}{\mathrm{V}(\Lambda)^{\frac{2}{n}}}. \tag{1.9}$$

Also known as the Hermite constant [26], the coding gain depends only on the invariants of a lattice. Notice that $\gamma(\mathbb{Z}^n) = 1$, which means that $\gamma(\Lambda)$ compares the coding gain of a given lattice Λ to that of the integer lattice having the same dimension. It is a good indicator for the performance of Λ over an AWGN channel: the higher the coding gain, the lower the error probability.

Definition 1.3.14. *Covering radius*
The covering radius of a lattice Λ in Euclidean space, denoted by $\rho(\Lambda)$, is defined as the smallest radius ρ such that the closed spheres of radius ρ centered at all lattice points cover the entire space, which is also the outer radius of the Voronoi region, as shown in Figure 1.4.

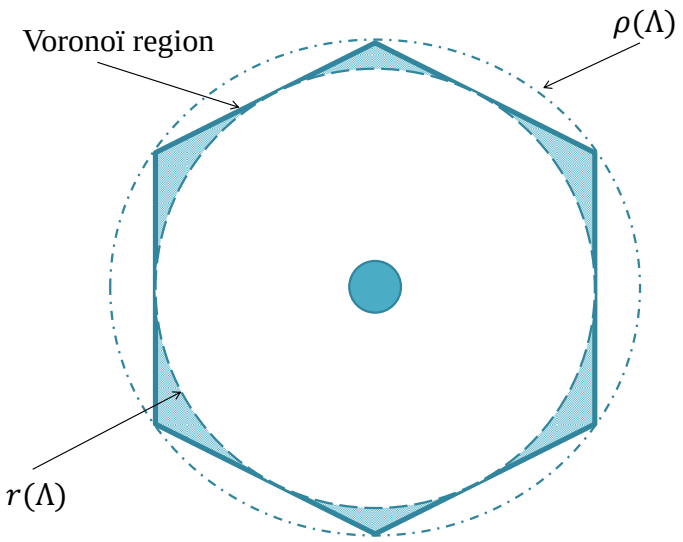


Figure 1.4 – The covering and packing radii with respect to the Voronoi region.

Definition 1.3.15. *Packing radius*
The packing radius of a lattice Λ, denoted by $r(\Lambda)$, is defined as the largest radius r such that spheres of radius r placed at lattice points intersect only at their boundaries. As Figure 1.4 shows, $r(\Lambda)$ is the inner radius of the Voronoi region.
If d^2_{min} is the minimal squared distance between distinct lattice points, the packing radius of Λ is given by:

$$r = \frac{1}{2}\sqrt{d^2_{min}}. \tag{1.10}$$

Definition 1.3.16. *Kissing number*
The kissing number of a lattice $K(\Lambda)$ is defined as the number of nearest neighbours to any lattice point $\mathbf{x}$. In the sphere packing problem, $K(\Lambda)$ would be the number of spheres touching a specific sphere.

Definition 1.3.17. *Lattice density*
Lattice density Δ is the fraction of the space that is occupied by the packing spheres, it is therefore the ratio of the volume of one sphere of radius r over the fundamental volume:

$$\Delta = \frac{\text{volume of one sphere of radius r}}{\text{fundamental volume}} = \frac{V_n \times \rho^n}{\mathrm{V}(\Lambda)}$$

where V_n is the volume of an n-dimensional sphere of radius 1

$$V_n = \frac{\pi^{n/2}}{\Gamma(\frac{n}{2}+1)} = \begin{cases} \dfrac{\pi^{n/2}}{\Gamma(\frac{n}{2}+1)} & n \text{ is even} \\ \dfrac{2^n \pi^{\frac{n-1}{2}} (\frac{n-1}{2})!}{n!} & n \text{ is odd} \end{cases}$$

where $\Gamma(x)$ is Euler's gamma function: $\Gamma(x) = \int_0^\infty u^{x-1} e^{-u} du$.

In the remainder of the work, we will deal with full rank lattices, i.e., $p = n$. In this case, $\mathbf{G}$ is a square matrix, and we have $\det(\Lambda) = \det(\mathbf{G})^2$.

1.4 Lattice Construction

Constructing lattices from error-correction codes is a classically studied topic [26]. Lattices are obtained by "lifting" one or more q-ary codes into the Euclidean space, which allows them to inherit some of the underlying codes' properties. Depending on the structure of these codes, lattice constructions can be categorized into different types. In this section, we describe the most popular lattice constructions known as Constructions A, B, D and D', used with binary ($q = 2$) codes of length n to construct lattices in $\mathbb{R}^n$. Note that another construction called Construction C is not to be considered here since it results in a sphere packing that is generally not a lattice.

1.4.1 Construction A

Let $\mathcal{C}$ be an (n, k, d) binary code that maps k information bits into binary codewords of length n. The parameter d is the minimum distance of the code, i.e., the Hamming minimum weight (number of ones) over all non-zero codewords. Construction A is a method of generating a lattice by "lifting" a linear binary code $\mathcal{C}$ to the Euclidean space [94]. $\mathbf{x} = \{x_1, \cdots, x_n\}$ is an integer lattice point if and only if its modulo-2 reduction belongs to $\mathcal{C}$. We can thus write:

$$\Lambda = \{\mathbf{x} \in \mathbb{Z}^n : \mathbf{x} \bmod 2 \in \mathcal{C}\}. \tag{1.11}$$

This is equivalent to writing:

$$\Lambda = \mathcal{C} + 2\mathbb{Z}^n. \tag{1.12}$$

When the generator matrix of the code $\mathcal{C}$ has a systematic form $[I_k|P]$, the lattice Λ obtained from $\mathcal{C}$ via Construction A will have a generator matrix of the form:

$$\mathbf{G} = \begin{pmatrix} \mathbf{I}_k & 0 \\ \mathbf{P} & 2\mathbf{I}_{n-k} \end{pmatrix} \tag{1.13}$$

a minimum distance of:

$$d_{min}(\Lambda) = \min\{2, \sqrt{d}\}$$

Its coding gain is:

$$\gamma(\Lambda) = \begin{cases} 4^{\frac{k}{n}} & \text{if } d \geq 4 \\ \frac{(d_{min}(\Lambda))^2}{4} 4^{\frac{k}{n}} & \text{if } d < 4 \end{cases}$$

and the kissing number:

$$K(\Lambda) = \begin{cases} 2^d B_d & \text{if } d < 4 \\ 2n + 16B_4 & \text{if } d = 4 \\ 2n & \text{if } d > 4 \end{cases}$$

where B_d denotes the number of codewords in $\mathcal{C}$ with minimum weight d.

Checkerboard Lattices that are the densest lattices known in dimensions $n = 3, 4$ and 5, denoted by D_n, can be constructed with single parity check codes $\text{SPC}(n, n-1, 2)$ via Construction A. Moreover, the densest lattices in dimensions $n = 6, 7$ and 8 denoted by E_6, E_7 and E_8 are also obtained using Construction A.

Example 1.4.1. Obtaining the Gosset lattice E_8 from Hamming code $\mathcal{H}(8, 4, 4)$
The Hamming code $\mathcal{H}(8, 4, 4)$ has the systematic generator matrix:

$$\mathbf{G}_{\mathcal{H}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ \underbrace{\qquad\qquad}_{\mathbf{I}_4} & & & & \underbrace{\qquad\qquad}_{\mathbf{P}} & & & \end{pmatrix}$$

So the generator matrix of E_8 is, according to (1.13):

$$\mathbf{G}_{E_8} = \begin{pmatrix} \mathbf{I}_4 & 0 \\ \mathbf{P} & 2\mathbf{I}_4 \end{pmatrix}$$

1.4.2 Construction B

Let $\mathcal{C}_1(n, k, d)$ and $\mathcal{C}_2(n, n-1, 2)$ be two linear codes that satisfy the condition $\mathcal{C}_1 \subseteq \mathcal{C}_2$. A lattice Λ is constructed by taking all the vectors $\mathbf{x} = (x_1, \cdots, x_n)$ that have their modulo 2 reduction in $\mathcal{C}_1$, and the sum $\sum_{i=1}^{i=n} x_i$ divisible by 4. We can thus write:

$$\Lambda = \mathcal{C}_1 + 2\mathcal{C}_2 + 4\mathbb{Z}^n. \tag{1.14}$$

Example 1.4.2. The Barnes-Wall lattice BW_{16} is constructed using the first order Reed-Muller code $\mathcal{C}_1 = (16, 5, 8)$ and the SPC code $\mathcal{C}_2 = (16, 15, 2)$.

Example 1.4.3. Another example is the densest lattice known in dimension 24, the Leech lattice Λ_{24} obtained using the $(24, 12, 8)$ Golay code and the SPC code $(24, 23, 2)$.

1.4.3 Construction D

Construction D is useful for constructing lattices of high coding gains. In fact, Construction A produces lattices with a coding gain less or equal to 4, and Construction B is limited to the class of codes with $d_{min} = 8$.

First explained in [9], Construction D generalizes what was proposed by Barnes and Wall in [10]. It uses a family of nested binary linear codes to produce a lattice packing in $\mathbb{R}^n$.

Let $\mathcal{C}_1 \subseteq \cdots \subseteq \mathcal{C}_a$ be a set of nested binary linear codes, where each code $\mathcal{C}_\ell$ has parameters (n, k_ℓ, d_ℓ) for $1 \leq \ell \leq a$. We denote by $\mathbf{c}_1, \cdots, \mathbf{c}_{k_\ell}$ the k_ℓ vectors that generate the ℓ^{th} code. A lattice Λ obtained using Construction D is the set of vectors of the form:

$$\mathbf{x} = \sum_{\ell=1}^{a} 2^{\ell-1} \sum_{j=1}^{k_\ell} u_j^{(\ell)} \mathbf{c}_j + \mathbf{z} \tag{1.15}$$

where $\mathbf{z} \in 2^a\mathbb{Z}^n$ and $u_j^{(\ell)} \in \{0, 1\}$.
Λ can also be described using the code formula [42]:

$$\Lambda = \mathcal{C}_1 + 2\mathcal{C}_2 + \cdots + 2^{a-1}\mathcal{C}_a + 2^a\mathbb{Z}^n. \tag{1.16}$$

Its determinant is:

$$\det(\Lambda) = 2^{an - \sum_{i=1}^{i=a} k_i} \tag{1.17}$$

and the minimal distance is written as:

$$d_{min}(\Lambda) = \min\left\{2, \frac{\sqrt{d_\ell}}{2^{\ell-1}}\right\}$$

If $d_\ell \geq \frac{4^\ell}{\beta}$, for $1 < \ell < a$ and $\beta = 1$ or 2, then the squared minimum distance of Λ is at least $\frac{4}{\beta}$, and its coding gain satisfies:

$$\gamma(\Lambda) \geq \beta^{-1} 4^{\Sigma_{\ell=1}^{\alpha} \frac{k_\ell}{n}}.$$

It is clear that when $a = 1$, Construction D reduces to the case of a Construction A. A typical example is that of the Gosset lattice E_8 seen earlier. When $a = 2$, $\mathcal{C}_2$ is

the SPC code $(n, n-1, 2)$ and Construction D reduces to the case of a Construction B.

This construction is used to build performing lattices of relatively high dimensions, namely the *Barnes-Wall* lattices obtained by applying Construction D to a family of Reed-Muller codes, as explained later in Section 1.4.5.

1.4.4 Construction D'

Similarly to Construction D, Construction D' can produce lattices with high coding gains, but while dealing with sets of parity checks rather than generator sets of codes. In fact, Construction D' converts a set of parity-checks defining a family of codes into congruences for a lattice [26].

Let $\mathcal{C}_1 \supseteq \cdots \supseteq \mathcal{C}_a$ be a set of nested binary linear codes. Let $\mathbf{h}_1, \cdots, \mathbf{h}_n$ be a basis in $\mathbb{F}_2^n$, such that each code $\mathcal{C}_\ell$ for $1 \leq \ell \leq a$ is defined by $r_\ell = n - k_\ell$ parity-check vectors $\mathbf{h}_1, \cdots, \mathbf{h}_{r_\ell}$. We define the lattice Λ as the set of vectors $\mathbf{x} \in \mathbb{Z}^n$ that satisfy the congruences:

$$\mathbf{h}_j.\mathbf{x} \equiv 0 \ (\text{mod} 2^{\ell+1}) \tag{1.18}$$

for all $\ell = 1, \cdots, a$ and $r_{a-\ell-1} + 1 \leq j \leq r_{a-\ell}$.

The parity check matrix $\mathbf{H}$ of the lattice Λ is:

$$\mathbf{H} = [\mathbf{h}_1, \cdots, \mathbf{h}_{r_0}, 2\mathbf{h}_{r_0+1}, \cdots, 2\mathbf{h}_{r_1}, \cdots, 2^a\mathbf{h}_{r_{a-1}+1}, \cdots, 2^a\mathbf{h}_{r_a}]^T. \tag{1.19}$$

It is known that Λ has a determinant [19]:

$$\det(\Lambda) = 2^{\sum_{\ell=1}^{\ell=a} r_\ell} \tag{1.20}$$

Example 1.4.4. Let $a = 3$ and $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_3$ be three nested codes whose duals are generated by $\{1001\}, \{1001, 1101\}$ and $\{1001, 1101, 1010\}$ respectively. In that case, we have $\mathbf{h}_1 = \{1001\}, \mathbf{h}_2 = \{1101\}$ and $\mathbf{h}_3 = \{1010\}$. The parity check matrix of the lattice Λ obtained via Construction D' is:

$$\mathbf{H} = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 2 & 2 & 0 & 2 \\ 4 & 0 & 4 & 0 \end{pmatrix}$$

A vector $\mathbf{x}$ is in Λ if and only if $\mathbf{H}\mathbf{x}^T \equiv 0 \ (\text{mod} 2^3)$.

Construcion D' is applied to a nested set of LDPC codes to obtain what is known as: *low density parity check lattices* [73].

1.4.5 Barnes-Wall lattices and Reed-Muller codes

We now explain the Construction D of the famous Barnes-Wall lattices using a set of nested binary linear Reed-Muller codes. Reed-Muller codes are among the oldest known codes and have found widespread applications. They were discovered by Muller and provided with a decoding algorithm by Reed in 1954 [71]. These codes were initially given as binary codes, but modern generalizations to q-ary codes exist. We will restrict our investigation to the binary case.

Reed-Muller codes $\mathcal{RM}(N, k, d)$ are a class of linear block codes over GF(2), where N is the length of the codeword, k is the length of the information block and d is the minimum Hamming distance. Conventionally, Reed-Muller codes are denoted by $\mathcal{RM}(r, m)$ with $0 \leq r \leq m$, and the following relations between N, k and d:

- $N = 2^m$
- $k = \binom{m}{0} + \binom{m}{1} + \cdots + \binom{m}{r}$
- $d = 2^{m-r}$
  The 0^{th} order Reed-Muller code $\mathcal{RM}(0, m)$ is defined to be the repetition code $\{0, 1\}$ of length 2^m.

Reed-Muller codes are famous for their recursive construction, which means that large Reed-Muller codes can be constructed from smaller ones. In fact, $\mathcal{RM}$ codes can be generated using the Plotkin ($|u|u + v|$) structure explained in more details in Appendix A. Thus, for any $r \geq 2$, the r^{th} order Reed-Muller code $\mathcal{RM}(r, m)$ is defined recursively by:

$$\mathcal{RM}(r, m) = \begin{cases} \mathbb{Z}_2^{2r} & r = m \\ (\mathbf{u}, \mathbf{u} + \mathbf{v}), \mathbf{u} \in \mathcal{RM}(r, m-1) \text{ and } \mathbf{v} \in \mathcal{RM}(r-1, m-1) & r \neq m \end{cases} \tag{1.21}$$

Example 1.4.5. Considering the even weight code $\mathcal{C}_1 = \mathcal{RM}(1, 2) = (4, 3, 2)$ and the repetition code $\mathcal{C}_2 = \mathcal{RM}(0, 2) = (4, 1, 4)$, we can construct the extended Hamming code $\mathcal{C} = \mathcal{RM}(1, 3) = (8, 4, 4)$. This code, along with the repetition code $\mathcal{RM}(0, 3) = (8, 1, 8)$, give us the first-order Reed-Muller code $\mathcal{RM}(1, 4) = (16, 5, 8)$, and so on.

Figure 1.5 illustrates the graphical representation of the decomposition process for the code $\mathcal{RM}(3, 6)$. The latter is repeatedly split into two codes $\mathcal{RM}(r, m-1)$ and

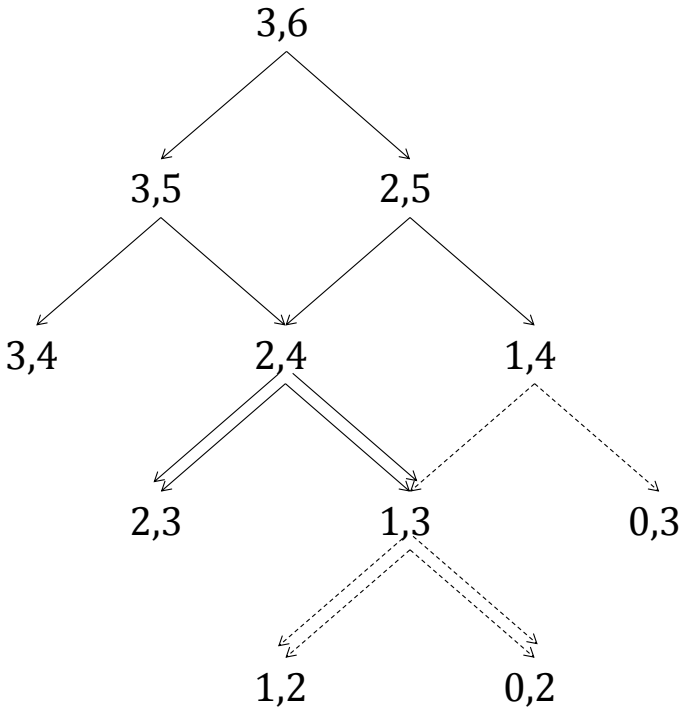


Figure 1.5 – Decomposition of code $\mathcal{RM}(3,6)$.

$\mathcal{RM}(r-1,m-1)$, which correspond to $\mathbf{u}$ and $\mathbf{v}$ respectively; once a terminal node is reached, the code is not decomposed any further. We can choose to end the decomposition at either $\mathcal{RM}(0,m)$ or $\mathcal{RM}(1,m)$ codes. We show in Appendix B that ending this process at a first-order $\mathcal{RM}$ code significantly improves the decoding algorithm.

The connection between Reed-Muller $\mathcal{RM}$ codes and lattices dates back to 1983 [9], where $\mathcal{RM}$ codes employed in a multilevel scheme resulted in interesting structures commonly known as Barnes-Wall lattices. This relation was also addressed by Forney in [37], where he explained the construction of BW lattices, their sublattices and their code formulas in real and complex forms.

Barnes-Wall lattices are N-dimensional complex, or $2N$-dimensional real lattices, that exist when $N = 2^m$. Using the lattice Construction D, they are built from nested Reed-Muller codes according to the following code formula:

$$BW(m) = \sum_{\substack{1\leq r'\leq m \\ m-r' \text{ odd}}} \mathcal{RM}(r', m+1)2^{\frac{r'-1}{2}} + 2^{m/2}\mathbb{Z}^{2N} \qquad \text{for } m \text{ even}$$

$$BW(m) = \sum_{\substack{1\leq r'\leq m \\ m-r' \text{ even}}} \mathcal{RM}(r', m+1)2^{\frac{r'-1}{2}} + 2^{\frac{m+1}{2}}\mathbb{Z}^{2N} \qquad \text{for } m \text{ odd}$$

For example:

$$BW_{32} = BW(4) = \mathcal{RM}(1,5) + 2\mathcal{RM}(3,5) + 4\mathbb{Z}^{32}$$

and

$$BW_{64} = BW(5) = \mathcal{RM}(1,6) + 2\mathcal{RM}(3,6) + 4\mathcal{RM}(5,6) + 8\mathbb{Z}^{64}.$$

Having at our disposal methods to construct efficient multidimensional lattices, consideration will now be given to the universal lattice decoding algorithm: the *sphere decoder*.

1.5 The Sphere Decoder algorithm

The sphere decoder is an algorithm that aims at finding the closest lattice point (in the sense of minimum Euclidean distance) of a given point in $\mathbb{R}^n$. Based on the Finke-Pohst enumeration [32] which searches for the closest lattice point within a sphere of radius R around the received vector $\mathbf{y}$, the sphere decoding algorithm was later improved by Boutros and Viterbo in [87] by dynamically updating the search radius. If $\mathbf{x}$ is the transmitted vector, the sphere decoder is an ML decoder when trying to decode $\mathbf{x}$ from observation $\mathbf{y}$. This universal algorithm is suitable for both AWGN and Rayleigh fading channels.

The strength of the sphere decoder stems from the fact that it eliminates the exhaustive search among all the points of the lattice constellation. Even though the complexity of the algorithm is independent of the size of the constellation used for transmission, it is limited by the dimension of the lattice (affordable complexity up to dimension equal to 32).

1.5.1 The sphere decoding algorithm

In the Gaussian channel case, the ML decoding of an n-dimensional lattice Λ requires searching for the lattice point that is the closest to the received vector, i.e., the lattice point that minimizes the following metric:

$$m(\mathbf{y}|\mathbf{x}) = ||\mathbf{y} - \mathbf{x}||^2 = \sum_{i=1}^{n} |y_i - x_i|^2 \tag{1.22}$$

where $\mathbf{x}$ is the transmitted lattice point, $\mathbf{y} = \mathbf{x} + \mathbf{w}$ is the received vector and $\mathbf{w} = (w_1, \cdots w_n)$ is the noise vector with real zero-mean independent Gaussian random variables with variance σ^2. As aforementioned, the lattice points can be written

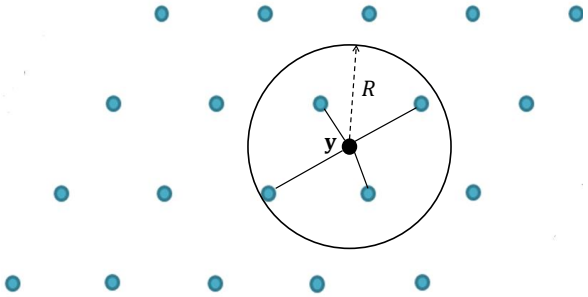


Figure 1.6 – The geometrical representation of the Sphere Decoding algorithm.

as: $\{\mathbf{x} = \mathbf{Gb}\}$, with $\mathbf{G}$ the lattice generator matrix, and $\mathbf{b} = (b_1, \cdots, b_n)^T$ is the integer component vector.

The sphere decoding algorithm consists in finding the closest lattice point without actually searching all the lattice points: it reduces the search to lattice points that lie within a search radius R, as shown in Figure 1.6 . Note that a lattice point $\mathbf{Gb}$ lies inside a sphere of radius R centered at $\mathbf{y}$ if and only if:

$$R^2 \geq ||\mathbf{y} - \mathbf{Gb}||^2. \tag{1.23}$$

The search for the closest point consists in computing minimum and maximum bounds for each component of the integer vector, and checking all lattice points inside the sphere of radius R by calculating increasingly each vector component interval. The algorithm begins with a QR decomposition of the lattice generator matrix:

$$\mathbf{G} = \mathbf{QR} \tag{1.24}$$

where $\mathbf{R}$ is an $n \times n$ upper triangular matrix and $\mathbf{Q}$ is an $n \times n$ orthogonal matrix. The condition (1.23) can therefore be written as:

$$R^2 \geq ||\mathbf{y} - \mathbf{QRb}||^2 = ||\mathbf{Q}^*\mathbf{y} - \mathbf{Rb}||^2$$

where $(.)^*$ denotes the Hermitian matrix transposition.

Writing $\mathbf{z} = \mathbf{Q}^*\mathbf{y}$ yields:

$$d^2 \geq \sum_{i=1}^{n} \left(z_i - \sum_{j=i}^{n} r_{i,j} b_j \right)^2 \tag{1.25}$$

where $r_{i,j}$ is an (i, j) entry of $\mathbf{R}$. Expanding the above inequality gives:

$$R^2 \geq (z_n - r_{n,n} b_n)^2 + (z_{n-1} - r_{n-1,n-1} b_{n-1} - r_{n-1,n} b_n)^2 + \cdots . \tag{1.26}$$

It is clear from (1.26) that the first term depends only on b_n, the second depends on $\{b_{n-1}, b_n\}$, and so on. The first necessary condition for the lattice point to lie inside the sphere of radius R is $R^2 \geq (z_n - r_{n,n}b_n)$. This condition gives the interval of b_n:

$$\left\lceil \frac{-R + z_n}{r_{n,n}} \right\rceil \leq b_n \leq \left\lfloor \frac{R + z_n}{r_{n,n}} \right\rfloor \tag{1.27}$$

where $\lceil x \rceil$ denotes the nearest integer greater than x, and $\lfloor x \rfloor$ denotes the nearest integer smaller than x.

After choosing a possible value for b_n in the interval (1.27), we move to the $(n-1)^{th}$ term b_{n-1}. We first set $R_{n-1}^2 = R^2 - (z_n - r_{n,n}b_n)^2$, and the second condition $R_{n-1}^2 \geq (z_{n-1} - r_{n-1,n-1}b_{n-1} - r_{n-1,n}b_n)^2$ leads to b_{n-1} belonging to the interval:

$$\left\lceil \frac{-R_{n-1} + z_{n-1} - r_{n-1,n}b_n}{r_{n-1,n-1}} \right\rceil \leq b_{n-1} \leq \left\lfloor \frac{R_{n-1} + z_{n-1} - r_{n-1,n}b_n}{r_{n-1,n-1}} \right\rfloor. \tag{1.28}$$

In a similar way, after having chosen some possible values for $(b_{i+1}, \cdots, b_n)$, we can obtain the general intervals of the component b_i based on the condition $R_i^2 \geq (z_i - \sum_{j=i}^{n} r_{i,j}b_j)^2$:

$$\left\lceil \frac{-R_i + z_i - \sum_{j=i+1}^{n} r_{i,j}b_j}{r_{i,i}} \right\rceil \leq b_i \leq \left\lfloor \frac{R_i + z_i - \sum_{j=i+1}^{n} r_{i,j}b_j}{r_{i,i}} \right\rfloor. \tag{1.29}$$

When a point inside the sphere is found, its distance from $\mathbf{y}$ is saved:

$$R_{new}^2 = R^2 - R_1^2 + (y_1 - r_{1,1}s_1)^2.$$

Note that the possible choices for $(b_{i+1}, \cdots, b_n)$ might result in a void interval for b_i. In this case, another choice for b_{i+1} should be done, up to b_n until a valid interval is obtained. The algorithm is formalized as in Algorithm 1.

The choice of the search radius is a crucial step of the algorithm: if R is too large, we obtain too many points and the search speed decreases, whereas if R is too small, we don't obtain any point at all. In order to make sure to find at least one lattice point inside the sphere, we should take R equal to the covering radius of the lattice (see definition 1.3.14). In practice, we adjust the radius to the noise variance σ^2: for small SNRs, a large radius is needed, whereas for high SNRs, a small radius is sufficient since the received point is normally close to the decoded lattice point. This point was addressed in [44], where the radius was calculated using the formula:

$$R^2 = 2n\sigma^2. \tag{1.30}$$

Algorithm 1 Search for the closest lattice point to $\mathbf{y}$ inside the sphere of radius R

Input: $\mathbf{y}, \mathbf{Q}, \mathbf{R}, \mathbf{z} = \mathbf{Q}^*\mathbf{y}, R$

Output: $\hat{\mathbf{b}}$

1: $k = n, R'_n = R, z'_n = z_n$
2: Set bounds for $\hat{b}_k$: $UB(\hat{b}_k) = \left\lfloor \frac{R'_k + z_k}{r_{k,k}} \right\rfloor, \hat{b}_k = \left\lceil \frac{-R'_k + z_k}{r_{k,k}} \right\rceil - 1$
3: Increase $\hat{b}_k : \hat{b}_k = \hat{b}_k + 1$
4: **if** $\hat{b}_k \leq UB(\hat{b}_k)$ **then**
5: Go to 9
6: **else**
7: Go to 14
8: **end if**
9: **if** $k = 1$ **then**
10: Go to 20
11: **else**
12: $k = k - 1, z'_k = z_k - \sum_{i=k+1}^{n} r_{k,i} s_i, R'_k = \sqrt{R'^2_{k+1} - (z'_{k+1} - r_{k+1,k+1}\hat{b}_{k+1})^2}$ and go to 2
13: **end if**
14: $k = k + 1$
15: **if** $k = n + 1$ **then**
16: Terminate algorithm
17: **else**
18: Go to 3
19: **end if**
20: Save $\hat{\mathbf{b}}$ and its distance from $\mathbf{y}$: $R'^2_n - R'^2_1 + (z'_1 - r_{1,1}\hat{b}_1)^2$ and go to 3.

1.5.2 The sphere decoder with fading

Let us now consider the transmission over an independent Rayleigh flat fading channel. With perfect channel state information (CSI) given at the receiver and no inter-symbol interference, the received signal can be written as:

$$\mathbf{y} = \alpha \odot \mathbf{x} + \mathbf{w} \tag{1.31}$$

where $\odot$ denotes the component-wise vector multiplication, $\mathbf{x}$ is the sent lattice point, and the noise vector $\mathbf{w}$ has real, independent, Gaussian distributed random variables with zero-mean and variance σ^2. The fading coefficients $\alpha = \{\alpha_1, \cdots, \alpha_n\}$ have unit second moment and are assumed to be independent from one symbol to the next. In

this case, ML decoding requires minimizing the following metric:

$$m(\mathbf{y}|\mathbf{x},\alpha) = ||\mathbf{y} - \alpha \odot \mathbf{x}||^2 = \sum_{i=1}^{n} |y_i - \alpha_i x_i|^2 \tag{1.32}$$

If $\mathbf{G}$ if the generator matrix of the lattice Λ, then we can consider a new lattice Λ_c with generator matrix:

$$\mathbf{G_c} = \text{diag}(\alpha_1, \cdots, \alpha_n) \times \mathbf{G}. \tag{1.33}$$

This new lattice Λ_c can be seen as a lattice in which each point was compressed or enlarged by a factor α_i. Let $\mathbf{x^{(c)}}$ be a point $\in \Lambda_c$, $\mathbf{x^{(c)}}$ can then be written as:

$$\mathbf{x^{(c)}} = (x_1^{(c)}, \cdots, x_n^{(c)}) = (\alpha_1 x_1, \cdots, \alpha_n x_n)$$

and the metric to minimize becomes:

$$m(\mathbf{y}|\mathbf{x},\alpha) = \sum_{i=1}^{n} |y_i - x_i^{(c)}|^2.$$

The received point $\mathbf{y}$ can thus be decoded by applying the same lattice decoding algorithm presented previously to the new lattice Λ_c. The decoded lattice point $\hat{\mathbf{x}}^{(\mathbf{c})}$ has the same integer vector $\mathbf{b}$ as $\hat{\mathbf{x}} \in \Lambda$.

The additional complexity encountered when dealing with the Rayleigh fading channel case comes from the fact that to each received signal $\mathbf{y}$ corresponds a different lattice Λ_c, thus a new $\mathbf{QR}$ decomposition.

The choice of the search radius is more critical here. In fact, when the transmission takes place in the presence of a deep fading, too many points may fall inside the sphere, which slows the decoding process down. To offset this problem, it is important to adapt R to the fading coefficients α_i.

1.6 Infinite lattices on the AWGN channel

As already seen in the previous section, the AWGN channel is given by the relation:

$$\mathbf{y} = \mathbf{x} + \mathbf{w}$$

When dealing with infinite lattices, the common notion of signal-to-noise ratio (SNR) is not relevant. It is therefore replaced by the so-called *Volume-to-Noise Ratio* defined as [42]

$$\text{VNR} = \frac{\text{V}(\Lambda)^{2/n}}{2\pi e\sigma^2} \tag{1.34}$$

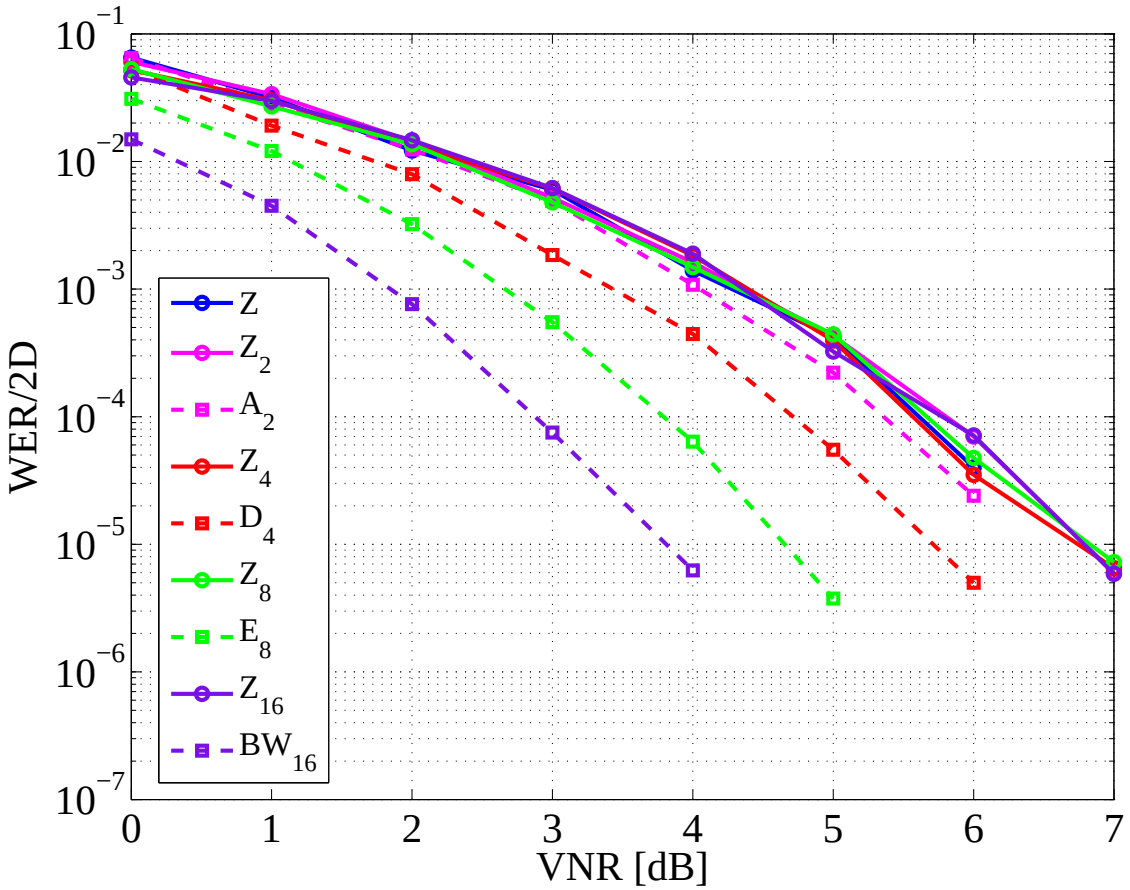


Figure 1.7 – Performance of some of the most popular lattices on the AWGN channel in terms of Normalized Word Error Rate.

Table 1.1 provides some features of the most-known lattices for dimensions up to 256. Among the main ones, we note the hexagonal lattice A_2, the Schlafli lattice D_4, the Gosset lattice E_8 and the Barnes-Wall lattice BW_{16}. This list of lattices is used for transmission over the AWGN channel, and the resulting performances are shown in Figure 1.7. In this figure, the Normalized Word Error Rate (NWER) is plotted as a function of the VNR, the NWER being the Word Error Rate per 2 dimensions (WER/2D) [82], i.e., WER $\times \frac{2}{n}$, suitable for comparing constellations having different dimensions. A word is erroneous when at least one symbol is erroneous. The lattices are also compared to the corresponding integer lattices $\mathbb{Z}^n$ having the same dimension.

It is easy to notice the improvement achieved when the lattice dimension increases. This is coherent with the fact that the coding gain, i.e., the lattice performance in comparison to that of the corresponding integer lattice $\mathbb{Z}^n$ improves with the dimension n, as seen in Table 1.1. Note that unlike dense lattice packings, integer lattices $\mathbb{Z}^n$

Table 1.1 – Features of the most-known lattices for dimensions up to 256.

n	Λ	$K(\Lambda)$	Δ	$\gamma(\Lambda)_{dB}$
1	$\Lambda_1 = A_1$	2	1	0
2	$\Lambda_2 = A_2$	6	0.90690	0.62
3	$\Lambda_3 = D_3$	12	0.74048	1
4	$\Lambda_4 = D_4$	24	0.61685	1.51
5	$\Lambda_5 = D_5$	40	0.46526	1.81
6	$\Lambda_6 = E_6$	72	0.37295	2.22
7	$\Lambda_7 = E_7$	126	0.29530	2.58
8	$\Lambda_8 = E_8$	240	0.25367	3.01
9	Λ_9	272	0.14577	3.01
10	Λ_{10}	336	0.09202	3.14
11	K_{11}	432	0.06043	3.3
12	Λ_{12}	648	0.04173	3.51
12	K_{12}	756	0.04945	3.64
13	K_{13}	918	0.02921	3.72
14	Λ_{14}	1422	0.02162	3.96
15	Λ_{15}	2340	0.01686	4.21
16	$\Lambda_{16} = BW_{16}$	4320	0.01471	4.52
17	Λ_{17}	5346	0.008811	4.6
18	Λ_{18}	7398	0.005928	4.75
19	Λ_{19}	10668	0.004121	4.91
20	Λ_{20}	17400	0.003226	5.12
24	Λ_{24}	196560	0.001930	6.02
32	Λ_{32}	208320	–	6.02
32	BW_{32}	146880	–	6.02
32	Q_{32}	261120	–	6.28
36	Λ_{36}	234456	–	6.19
48	Λ_{48}	–	–	7.53
64	BW_{64}	9694080	–	7.53
64	Q_{64}	2611200	–	7.78
64	P_{64c}	–	–	8.09
128	BW_{128}	1260230400	–	9.03
128	P_{128b}	–	–	10.02
128	η_{E_8}	–	–	10.16
256	BW_{256}	325139443200	–	10.54

maintain the same volume, thus equal performance $\forall n$. This proves that the sphere packing density is a key point for finding good lattices for transmission over the AWGN channel.

1.7 Conclusion

After recalling some fundamentals concerning lattice-related studies, we listed some basic concepts that enabled a better understanding of lattice theory. We explained afterwards the various lattice constructions using error-correcting codes. The ML lattice decoding algorithm denoted the sphere decoder was presented. Simulations were finally shown for unbounded lattices on the Gaussian channel, with respect to the VNR, which therefore does not reflect the performance of a lattice as a codebook.

Until now, the lattices were described in their infinite version. However, we know that in practice, data transmission requires sending a finite number of points. Therefore, consideration will hereafter be given to an essential operation related to lattices: *lattice shaping*. The next chapter will be dedicated to the description of the modification brought to both encoding and decoding parts.

CHAPTER 2 Lattice Shaping

2.1 Introduction

As introduced in Chapter 1, lattices are infinite discrete subsets of $\mathbb{R}^n$. However, in practice, the amount of information to be sent per channel use is finite, and so should be the number of lattice points considered for transmission. Therefore, the use of a lattice for digital transmissions requires 2 important tasks:

- Encoding: Associating the information bits to the symbols of the lattice code.
- Shaping: Defining a finite set of lattice points to create what is known as a *lattice code* or a *lattice constellation*. A lattice code, denoted by $\Lambda_{\mathcal{B}}$, can be defined as the intersection between a lattice Λ and a compact bounding region of $\mathbb{R}^n$ called the shaping region and denoted by $\mathcal{B}$.
  Note that in this Chapter, probabilistic shaping, as proposed for lattices in [62], will not be considered.

Depending on the shape of $\mathcal{B}$, the error rate performance may vary. A simple choice for $\mathcal{B}$ is to give it the form of a hypercube, and thus perform hypercube shaping. The gain related to the reduction in required signal-to-noise ratio to achieve a certain error probability with respect to using hypercube shaping and same spectral efficiency is called the *shaping gain* and is upper bounded by 1.53 dB [35], which is reached when $\mathcal{B}$ is a hypersphere. Unfortunately, hypersphere shaping is too complex to implement. If $\mathcal{B}$ is a hypercube, the shaping gain is obviously 0 dB. Note that the shaping gain can even be negative, if the shape of $\mathcal{B}$ is worse than a hypercube. By shaping a lattice, we aim at preventing the transmitted power from being too large by making sure that only lattice points close to the origin are transmitted. A survey of popular shaping techniques was proposed by Sommer, Feder and Shalvi in [80] and applied to low-density lattice codes to improve the shaping gain. Among them, the best shaping gain is achieved by the so-called nested shaping.

The encoding task (mapping the information bits to the lattice symbols lying in $\mathcal{B}$) is also an issue since the cardinal of the lattice code increases exponentially with the dimension of the lattice and with the spectral efficiency. A simple look-up table mapping is not possible and possibly new encoding schemes have to be considered. Note that the bounding region and the encoding scheme can be entangled.

At the receiver side, we distinguish two types of decoders for lattice codes: the *lattice decoder* and the *lattice code decoder*. A lattice decoder is simply a decoder in the infinite lattice: there is no checking whether the estimated transmitted symbol lies inside the shaping region or not. This decoder will be denoted as naive lattice decoder. A lattice code decoder, on the contrary, takes into consideration the shaping region and thus ensures that the decoded lattice point falls inside the boundaries of $\mathcal{B}$.

This chapter is organized as follows: in Section 2.2 we describe the different shaping mechanisms of a lattice on the AWGN channel, namely the hypercube shaping and the nested shaping. In Section 2.3, we focus on the decoding part and propose a modification of the lattice code decoder algorithm that achieves an improved error rate performance even for low spectral efficiencies and/or low lattice dimensions. Simulations are computed on both AWGN and Rayleigh fading channels. At the end of this section the decoder is compared to the LTE baseline. The chapter is concluded in Section 2.4.

2.2 Shaping mechanisms

A basic block diagram of a system employing lattice constellations is shown in Figure 2.1. In this case, the lattice is sent in its infinite version, there is no shaping boundary that determines the region to which the sent lattice points must belong. Nevertheless, the vector of integers $\mathbf{b}$ generally has its components drawn from a certain interval. If the number of information bits per dimension is a power of 2, the mapping between these bits and the information integers is straightforward. Otherwise, it is possible to apply a non-uniform mapping like for example in [54, 68]. We assume hereafter that the information integers of vector $\mathbf{b}$ are uniformly drawn from the interval $(0, \cdots, L-1)$, L being a positive integer. The spectral efficiency is thus defined as $\eta = \log_2(L)$ bits per real dimension.

To study the performance of a lattice sent over the AWGN channel, we compare the Word Error Rate per 2 dimensions, as a function of the channel signal-to-noise ratio (SNR), of the Gosset lattice E_8 seen in Section 1.4.1 and the uncoded QAM constellation having the same spectral efficiency.

Lattice E_8 is the best known lattice in 8 dimensions, in the sense of having the highest packing density (see Section 1.6). It consists of the vectors $\mathbf{x} = (x_1, x_2, \cdots, x_8)^T$, where x_i are all integers, or all halves of integers, and $x_1 + \cdots + x_8$ is even. A generator matrix of lattice E_8 is [26]:

$$\begin{pmatrix} 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0.5 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0.5 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0.5 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0.5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0.5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 \end{pmatrix}.$$

If $\mathbf{x}$ is the sent lattice point, the received vector $\mathbf{y}$ is written as: $\mathbf{y} = \mathbf{x} + \mathbf{w}$, where $\mathbf{w}$ is a vector of $n = 8$ independent samples drawn from a centered Gaussian distribution with variance σ^2. The SNR is defined as:

$$\text{SNR} = \mathbb{E}\left(\frac{(\mathbf{x}-\mathbf{t})^T(\mathbf{x}-\mathbf{t})}{n\sigma^2}\right). \tag{2.1}$$

where the translation vector $\mathbf{t}$ shall be equal to the codewords expectation: $\mathbf{t} = \mathbb{E}(\mathbf{x})$. Hence, the expectation of the sent symbols is zero. Note that the shifting operation is not mentioned in Figure 2.1, since it does not affect the error rate performance on an AWGN channel, provided that the SNR is calculated as in (2.1).

The simulation results are depicted in Figure 2.2 for a spectral efficiency $\eta = 1, 2, 3$ and 4 bits/dim, corresponding to integer vectors $\mathbf{b}$ drawn from the intervals $[0, 1], [0, 3], [0, 7]$ and $[0, 15]$ respectively. The search for the closest lattice point was computed using the sphere decoder algorithm described in Section 1.5. The use of E_8 is clearly not an interesting choice in this case, the QAM constellation outperforms the 8-dimensional lattice for all values of η. This can be explained by the fact that too many E_8 lattice points with high energy are being used for transmission, thus the performance of the lattice is degraded. To improve the error rate performance, a shaping has to be processed in order to lower the number of high energy symbols.

The shaping operation is explained in Figure 2.3. Suppose we have a two-dimensional lattice with integer components $\{b_1, b_2\}$ that may take any value in the set $\{0, 1, 2, 3\}$. The number of total lattice points is then $4^2 = 16$ points. In the no-shaping case, the lattice codewords are any of the 16 lattice points belonging to a parallelogram as illustrated in Figure 2.3 (In fact, this is not necessarily a parallelogram, the shape is defined by the lattice generator matrix). If a shaping mechanism is applied, the 16

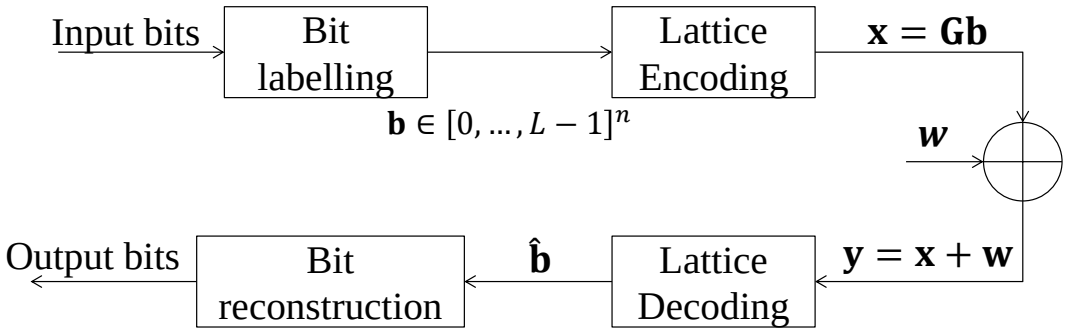


Figure 2.1 – Basic block diagram of a system employing lattice constellations.

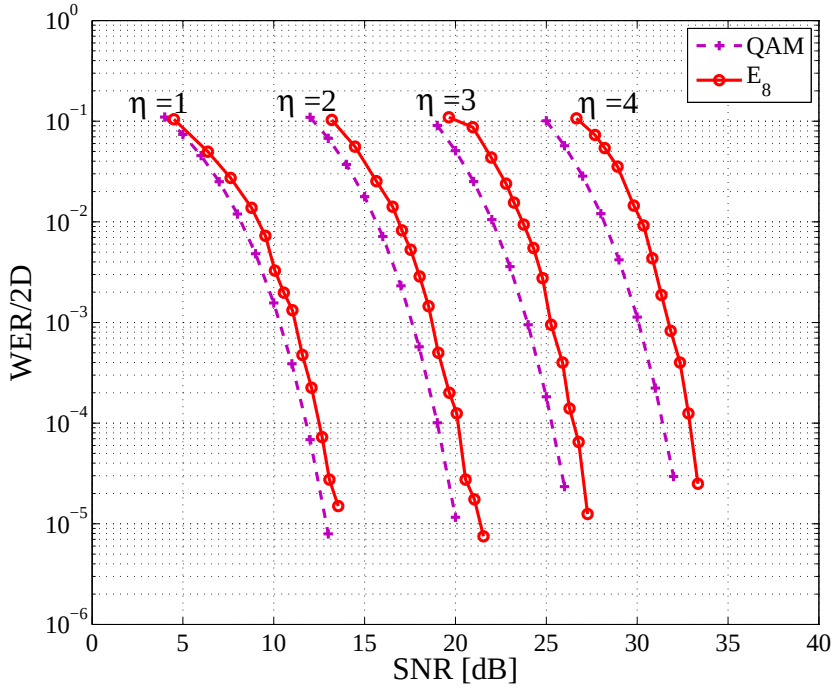


Figure 2.2 – Gosset lattice E_8 performance without shaping at the encoder.

lattice points are those belonging to a specific shaping domain, say a sphere, resulting in an average power that is lower than the no-shaping case. This gain in average power increases when the boundaries of the integer components b_i increase, since the lattice points inside the parallelogram may have average energy that is much higher than those inside the sphere. This is consistent with the simulations of Figure 2.2 where the gap between E_8 and the uncoded QAM increases with η.

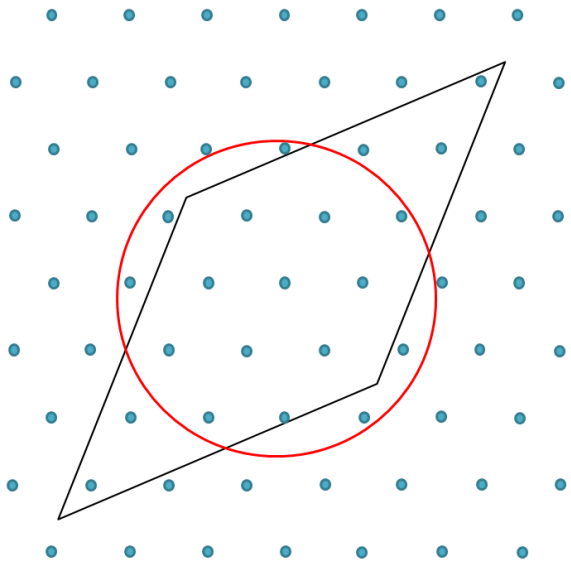


Figure 2.3 – Sent lattice points with and without shaping.

Hence, the key idea in the shaping operation is to choose a shaping region $\mathcal{B}$, and send any lattice point belonging to that one and only region. The straightforward linear encoding of an integer information vector $\mathbf{b}$ by the lattice point $\mathbf{x} = \mathbf{G}\mathbf{b}$ might not fall within $\mathcal{B}$ associated to the lattice code $\Lambda_{\mathcal{B}}$. Consequently, the shaping operation consists in finding another vector of integers, denoted by $\mathbf{b}_s$, such that its linear encoding $\mathbf{x}_s = \mathbf{G}\mathbf{b}_s$ is guaranteed to fall inside the shaping region. The new transmission scheme on the AWGN channel is depicted in Figure 2.4. At the receiver side, the received vector $\mathbf{y}$ is decoded using the MMSE lattice decoding proposed by Erez and Zamir in [30], where it was proven that the addition of an MMSE factor is one of the requirements for achieving the full Gaussian channel capacity with lattice decoding. The MMSE coefficient is equal to:

$$\text{MMSE} = \frac{\text{SNR}}{1 + \text{SNR}}. \tag{2.2}$$

We now describe the shaping methods presented by Sommer et *al* in [80]. These methods assume that $\mathbf{H} = \mathbf{G}^{-1}$ is a lower triangular matrix with ones on the diagonal.

2.2.1 Hypercube shaping

First, we consider the hypercube shaping based on Tomlinson-Harashima precoding [83]. This method uses a hypercube shaping domain, and it finds the vectors $\mathbf{b_s}$ such that the components of the shaped codewords $\mathbf{x}_s$ are uniformly distributed. The shaping operation is:

$$b_{s_i} = b_i - L_i k_i \quad i = 1, 2, \cdots, n. \tag{2.3}$$

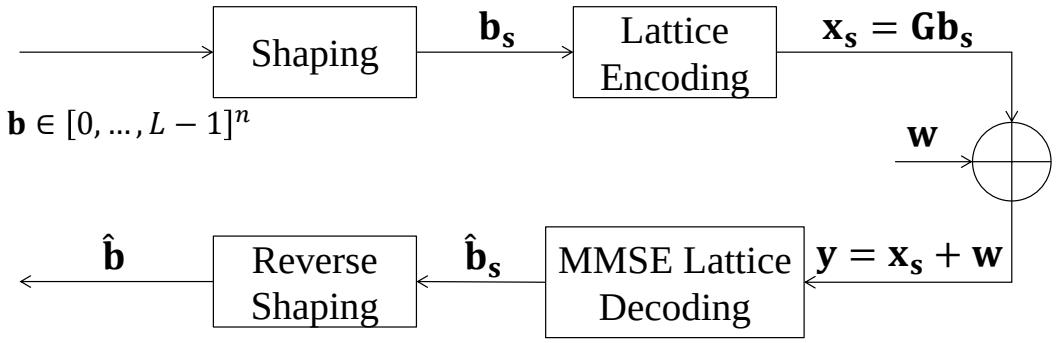


Figure 2.4 – Transmission model employing a shaping operation.

where k_i is an integer chosen in a way to ensure that each element x_{s_i} of the codeword $\mathbf{x_s}$ lies within the interval $[-\frac{L_i}{2}, \frac{L_i}{2})$, which is a hypercube. The value of k_i is given by [80]:

$$k_i = \left\lfloor \frac{1}{L_i} \left(b_i - \sum_{l=1}^{i-1} H_{i,l} x_{s_l} \right) \right\rceil .$$

Then the i^{th} element of the lattice codeword $\mathbf{x_s}$ is given by:

$$x_{s_i} = b_{s_i} - \sum_{j=1}^{i-1} H_{i,j} x_{s_j}.$$

At the decoder, the information integers b_i are recovered from b_{s_i} by a simple modulo operation:

$$b_i = b_{s_i} \mathrm{mod} L_i$$

The assumption of hypercube shaping not only forces a reduced average power, but also has practical advantages, for example, the complexity of the shaping operation is generally low. However, as we have stated earlier, there is no shaping gain under hypercube shaping.

2.2.2 Nested shaping

Nested lattice Let's consider two lattices Λ and Λ_s such that $\mathrm{Vol}(\Lambda_s) \geq \mathrm{Vol}(\Lambda)$. Λ_s is called the *coarse* lattice and Λ is called the *fine* lattice. If Λ and Λ_s are nested lattices, it means that Λ_s is a sublattice of Λ ($\Lambda_s \subset \Lambda$):

$$\forall\ \mathbf{x} \in \Lambda_s,\ \mathbf{x} \in \Lambda.$$

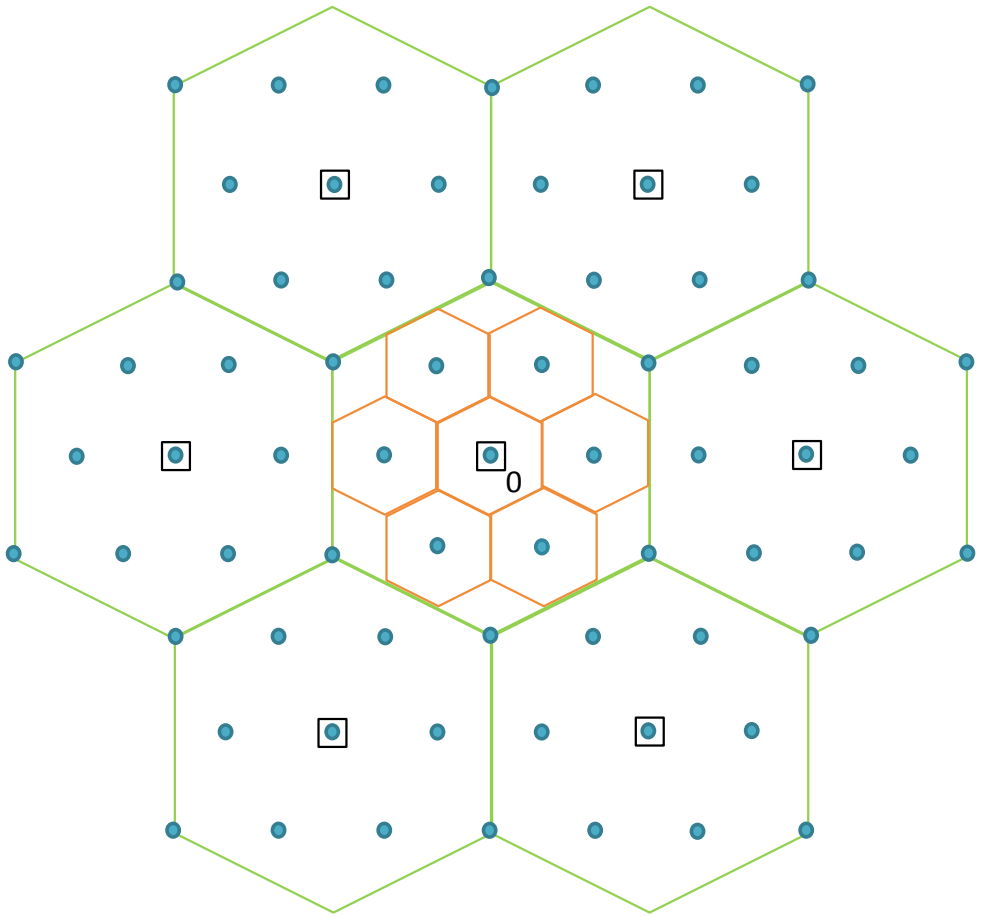


Figure 2.5 – Two nested lattices: the hexagonal lattice A_2 and a scaled version of A_2 of factor 3.

A simple choice for Λ_s is to take a scaled version of the lattice Λ of factor L. An example of two nested lattices is shown in Figure 2.5, where the fine lattice Λ is the hexagonal lattice A_2 (blue dots) and the coarse lattice Λ_s is a scaled version of A_2 of factor three (black squares).

Using vector notations for (2.3), the new vector of integers $\mathbf{b_s}$ is written as:

$$\mathbf{b_s} = \mathbf{b} - L\mathbf{k}. \tag{2.4}$$

Applying linear encoding to Equation (2.4) yields to:

$$\mathbf{x}_s = \mathbf{x} - L\mathbf{G}\mathbf{k}. \tag{2.5}$$

The nested shaping mechanism is explained as follows: The operation consists in minimizing the amplitude of the transmitted symbol $\mathbf{x}_s$, which is equivalent to minimizing the amplitude of $(\mathbf{x} - L\mathbf{G}\mathbf{k})$. In order to do so, we must find the closest

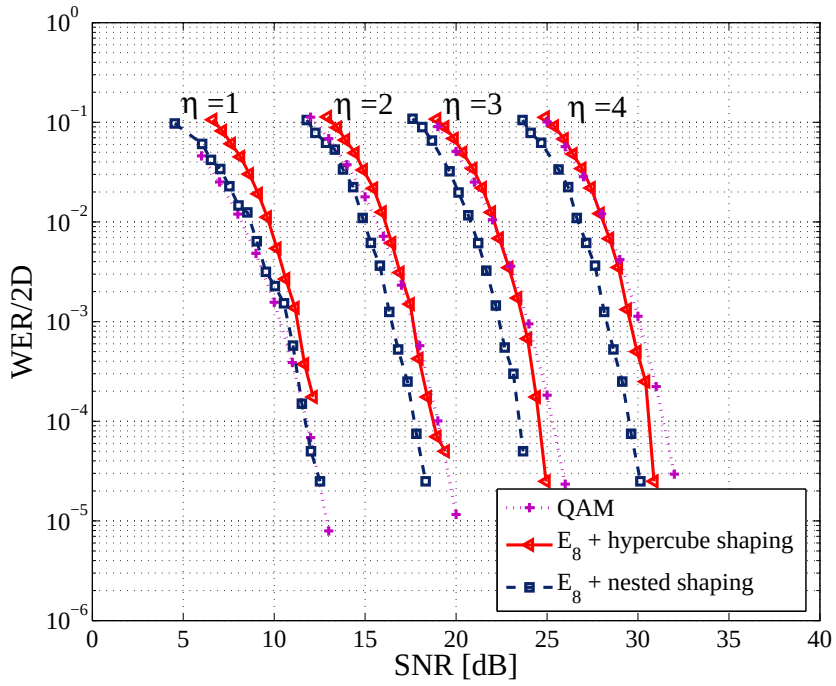


Figure 2.6 – Simulation results for hypercube and nested shaping applied to the Gosset lattice E_8.

coarse lattice point $L\mathbf{Gk}$ to the fine lattice point $\mathbf{x}$, which can be done by applying a sphere decoder on $\mathbf{x}$ in the lattice having the generator matrix $L\mathbf{G}$, which is the coarse lattice Λ_s. As a result, the transmitted lattice points will be uniformly distributed along the Voronoi region of the coarse lattice. In conclusion, the resulting scheme is equivalent to *nested lattice coding* [30], [25], where the shaping domain of a lattice is chosen to be the Voronoi region of the coarse lattice, that is a scale version of the fine lattice.

2.2.3 Comparison

We now compare the two shaping mechanisms described above by applying each of them to the 8-dimensional Gosset lattice E_8. The simulation results are shown in Figure 2.6 for spectral efficiencies of $\eta = 1, 2, 3$ and 4 bits/dimension.

With nested shaping at the transmitter, the E_8 lattice outperforms the QAM constellation except for a low spectral efficiency $\eta = 1$ bit/dim. The shaping gain increases with the spectral efficiency and can reach up to 1.1 dB with $\eta = 4$ bits/dim and a Word Error Rate per 2 dimensions of 10^{-4}. In fact, for $\eta = 1$ bit/dim, the total number of transmitted lattice points is $2^8 = 256$ points, which is a very small number compared

to the 8^8 or 16^8 lattice points that can be transmitted with $\eta = 3$ and $\eta = 4$ bits/dim respectively. This explains why in the former case a shaping mechanism may not provide a good performance.

Concerning the hypercube shaping, it was implemented by taking $L_i = L \ \ \forall i$, i.e., all integer components b_i have the same constellation size L. Simulations show that E_8 with hypercube shaping is also outperformed by the uncoded QAM for $\eta = 1$ bit/dim. Note that the uncoded QAM constellation is the infinite integer lattice with hypercube shaping. For higher values of η, E_8 with hypercube shaping is almost as good as the QAM for low SNRs, but starts to have smaller error rates for high SNR.

We remind that the search for the closest lattice point was implemented by the previously seen sphere decoder. In Section 1.5, the sphere decoder algorithm was described for lattices having an infinite number of elements. However, as explained above, we generally deal with a lattice constellation whose integer vectors take their components inside a given interval: $b_i = [0, \cdots, L-1]^n$, for $i = 1 \cdots, n$. Therefore, we must consider the bounds of this interval in the algorithm. Without implementing a specific shaping technique the minimum and maximum values (0 and $L-1$) of the lattice point coordinates are known. They can be used in the sphere decoder implementation to check for the bounds of $\mathbf{b}$ as proposed in [72].

When applied with a shaping operation, sphere decoding is not ML decoding since the closest lattice point found may lie outside $\mathcal{B}$. Figure 2.6 shows that this is not an issue for high dimensions and high spectral efficiencies, but a loss in error rate is observed otherwise. To get ML performance in the low dimension and low spectral efficiency case, sphere decoding must be performed inside the region $\mathcal{B}$. However, in this case, the coordinates are modified in such a way that it becomes too complex to restrict the solution of the sphere decoder to the shaping region.

In the following, we propose a modification of the lattice code decoder algorithm to overcome this problem and thus achieve improved error rate performance even for low spectral efficiencies and / or low lattice dimensions.

2.3 Proposed receiver algorithm

The proposed receiver is based on re-shaping and does not depend on the shaping technique, provided it is known at the receiver side. However, since nested shaping achieves a quasi-optimal shaping with an affordable complexity, it will be used as an illustrative example throughout the remainder of this section.

2.3.1 Proposed algorithm

We have seen that decoding at the receiver side consists first in finding the closest lattice point to $\mathbf{y}$, then applying a modulo$-L$ operation to its integer coordinates to have an estimation of the initial integer information vector $\mathbf{b}$. As mentioned before, a lattice code decoder gives better performance than a naive lattice decoder but it is complex to implement within a sphere decoding algorithm, as the boundaries of the code are unknown to the receiver when using a shaping operation. So we suggest to re-shape the result of the lattice sphere decoder in order to check whether it belongs to the lattice code or not. Moreover, replacing the sphere decoder by a list sphere decoder (LSD) [45] with a list of size l_s enables the re-shaping of a maximum of l_s points, increasing the chance to get a decoder output within the shaping region.

The proposed decoding algorithm is resumed in Algorithm 2.

Algorithm 2 Search for the closest lattice point to $\mathbf{y}$ inside the Voronoi region of Λ_S

Input: $\mathbf{y}, \mathbf{G}, l_s, L$

Output: $\hat{b}$

1: $\left\{\hat{\mathbf{b}}_s^{(1)}, \cdots, \hat{\mathbf{b}}_s^{(l_s)}\right\} = \text{LSD}(\mathbf{y}, \mathbf{G}, l_s)$
2: **for** $i = 1 : l_s$ **do**
3: $\quad \hat{\mathbf{b}}^{(i)} = \hat{\mathbf{b}}_s^{(i)} \mod L$
4: $\quad$ **if** $\left(\text{LSD}(\hat{\mathbf{b}}^{(i)}, L\mathbf{G}, 1) == \hat{\mathbf{b}}_s^{(i)}\right)$ **then**
5: $\quad\quad$ **return** $\hat{\mathbf{b}} = \hat{\mathbf{b}}^{(i)}$
6: $\quad$ **end if**
7: **end for**

The inputs of the proposed algorithm are the received observation $\mathbf{y}$, the generator matrix $\mathbf{G}$ of the lattice Λ, the list size parameter l_s and the constellation size for each integer L. The notation $\text{LSD}(\mathbf{a}, \mathbf{B}, c)$ denotes the processing of the list sphere decoder on the observation $\mathbf{a}$, in the lattice described by the generator matrix $\mathbf{B}$ and which returns the c coordinate vectors of the lattice points which are the closest to $\mathbf{a}$ and sorted in the ascending order. On line 1, the list sphere decoder $\text{LSD}(\mathbf{y}, \mathbf{G}, l_s)$ returns the list of coordinates of the l_s closest points to $\mathbf{y}$ in the lattice Λ. We begin with the first point in the list, and obtain its initial coordinate estimate through the modulo-L operation on line 3. Then, reshaping the result consists in applying a sphere decoding operation in the shaping lattice Λ_s (line 4). If the result is equal to the coordinates output by the LSD, it means that the candidate belongs to the shaping region. In this case, there is no need to proceed further. If the equality in line 4 is not satisfied, it means that the decoded point lies outside $\mathcal{B}$ and so the next point in the list has to

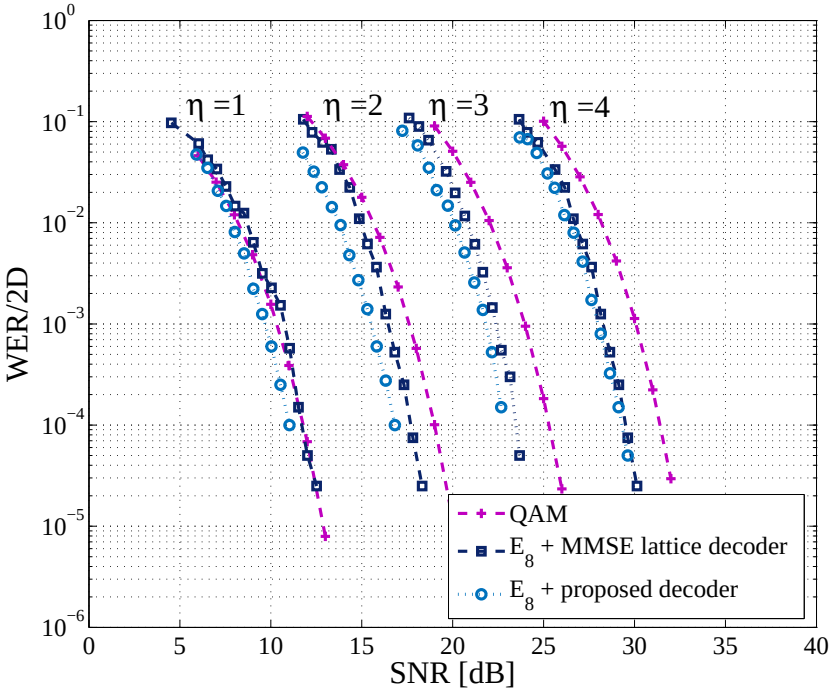


Figure 2.7 – Performance of the proposed modified decoder.

be processed. Of course, the higher the list size l_s, the higher the chance to find the closest point to the observation inside the shaping region, and thus the closer the ML approximation.

2.3.2 Simulation results

Simulation results of the Word Error Rate (WER) per 2 dimensions (WER/2D) are plotted in Figure 2.7 as a function of the SNR defined in Equation (2.1). Encoding in E_8 is performed with a nested shaping. For the decoding, two lattice decoding algorithms are compared: the first one is the MMSE lattice decoding scheme and the second is the proposed algorithm described previously. Note that the MMSE lattice decoding proposed in [30] is a combination of MMSE estimation, dithering and nested lattice codes. The dithering was not taken into account in the simulations of Figure 2.7, as it was judged unnecessary in [94]. When we apply the proposed algorithm with $l_s = 10$, the performance is improved: we now obtain a coding gain for all the simulated spectral efficiencies. For 1 bit/dim, our proposed algorithm can achieve a gain of almost 1 dB at a WER/2D $= 10^{-3}$ over the MMSE lattice decoder. This gain decreases for higher values of η, since in that case, the shaping region $\mathcal{B}$ is larger, and

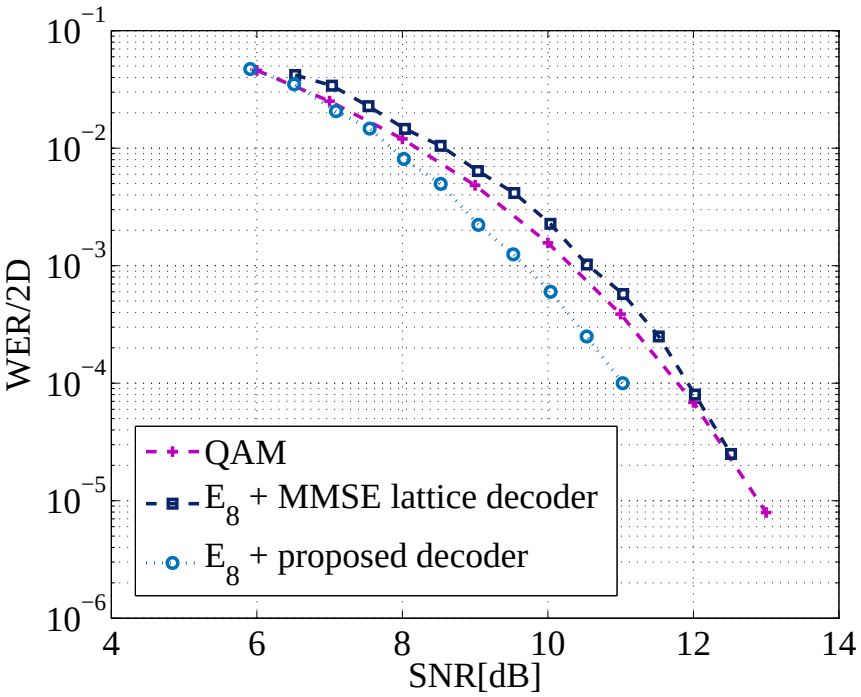


Figure 2.8 – Performance of the proposed modified decoder for $\eta = 1$ bit/dim.

the probability of having estimated lattice points outside $\mathcal{B}$ using a lattice decoder instead of a lattice code decoder decreases.

Figure 2.8 shows more clearly what happens for $\eta = 1$ bit/dim. Comparing both decoding techniques to the uncoded QAM, we notice that the latter is only outperformed by an E_8 lattice decoded with the proposed lattice code decoder. Moreover, the lattice performance is improved by almost 1.6 dB over the MMSE lattice decoding for WER/2D=10^{-4}.

2.3.3 Influence of the list size

The complexity and the performance of the proposed decoder depend on the list size l_s. The influence of the list size is depicted in Figure 2.9, where list sizes of $1, 3, 5, 10$ and 15 are applied to an E_8 lattice with $\eta = 1$ bit/dim. Note that parameter l_s has a major influence for low spectral efficiencies. Note also that $l_s = 1$ corresponds to the naive lattice decoder. It is clear from the simulation results that the higher the list size, the better the error rate. However, as from a certain size (here equal to 10), the performances stop improving, showing that opting for a longer size is unnecessary. Note also that the increase in complexity induced by implementing a LSD instead of a

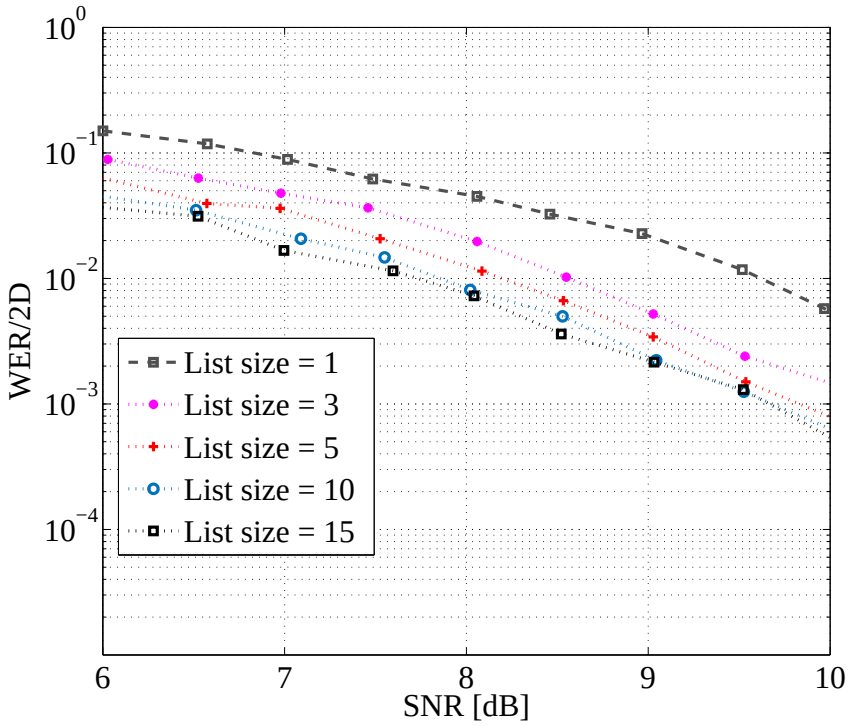


Figure 2.9 – Impact of the list size l_s on the decoder's performance.

single output sphere decoder is not a major issue since in all our simulations, the search radius is such that a much higher number of lattice points than l_s were collected.

2.3.4 Shaping on the Rayleigh fading channel

We now assume that the lattice code is used over a fast Rayleigh fading channel, with perfect channel state information at the receiver. The received vector is then:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w}.$$

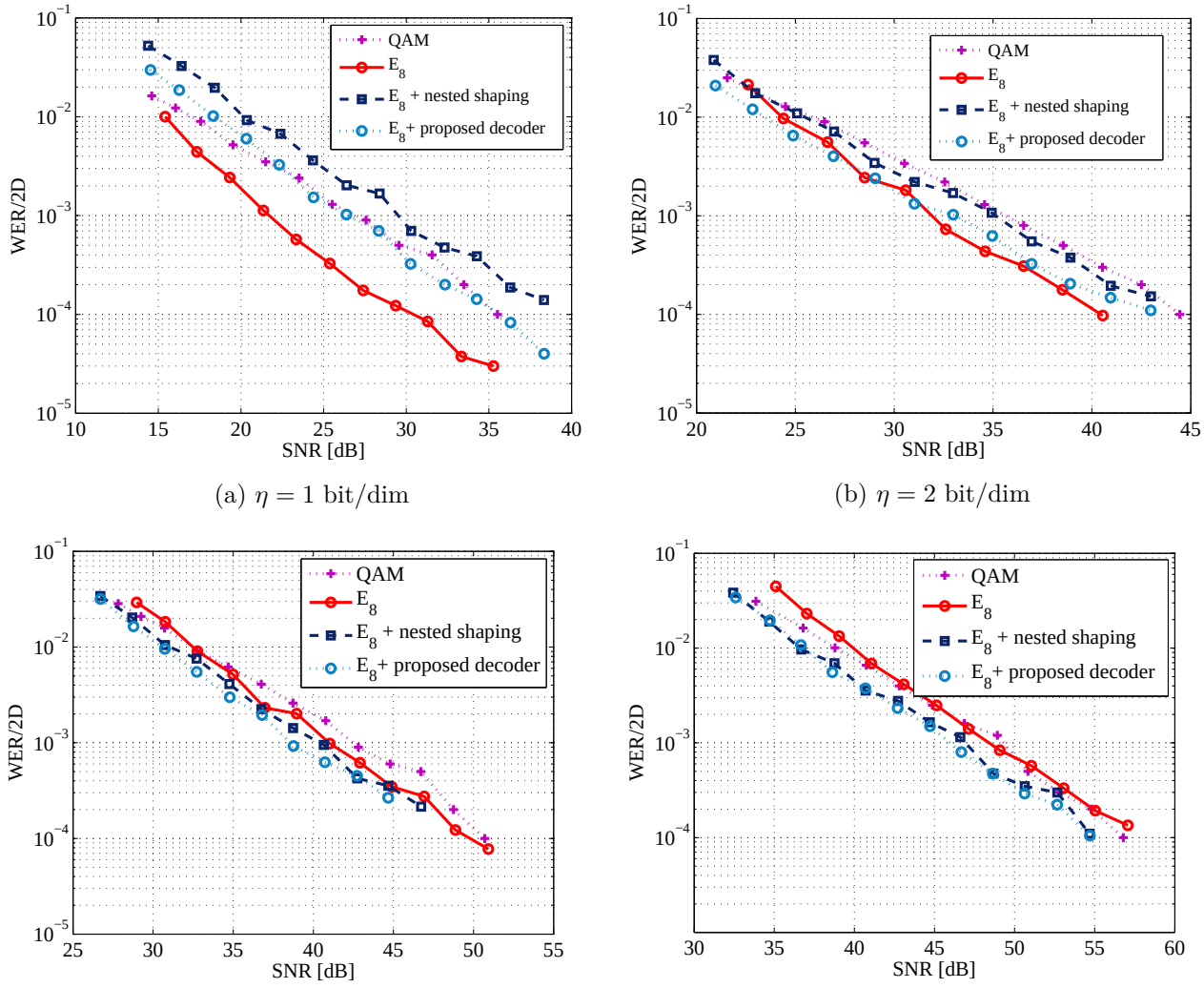
where $\mathbf{H} = \text{diag}(h_i)$ for $i = 1, \cdots, n$ is the matrix of random independent fading components h_i.

As already pointed out in Section 1.5.2, when dealing with lattice codes over fading channels, the decoding process can simply be carried out over the new lattice Λ_c having the generator matrix:

$$\mathbf{G_c} = \mathbf{H}.\mathbf{G} = \text{diag}(h_1, \cdots, h_n) \times \mathbf{G}.$$

Still taking lattice E_8 as an example, the simulation results are shown in Figure 2.10, where the WER per 2 dimensions is plotted as a function of the channel SNR. We notice

a difference in each curve's behaviour between a low ($\eta = 1$) and a high ($\eta = 4$) spectral efficiency. For $\eta = 4$ bits/dim, lattice codes with nested shaping at the encoder begin to outperform the lattice with no shaping, which does not happen for lower spectral efficiencies. This can be due to the fact that for low spectral efficiencies, a fewer number of E_8 lattice points is considered for transmission, i.e., the points are more bounded, in a way that nested shaping does not have a lot to add to the system's performances.



(a) $\eta = 1$ bit/dim

(b) $\eta = 2$ bit/dim

(c) $\eta = 3$ bit/dim

(d) $\eta = 4$ bit/dim

Figure 2.10 – Performance of the proposed lattice code decoder on the Rayleigh fading channel.

2.3.5 Comparison with short packets LTE

The use of short packets is seen as a key enabler to fulfill the needs of future wireless and mobile communication demands [2], namely low-latency and high reliability

transmissions. In fact, emergent Machine-Type Communications (MTC) are featuring an ever-increasing number of devices, whereas the data from each device may be very small.

The use of short packets has a direct impact on the physical layer, especially on the forward-error correction code (FEC) performance. Modern FEC schemes such as Turbo codes or low-density parity check (LDPC) codes can achieve high coding gains close to the theoretical limits provided that the code length is long enough. Using long FECs with short packet transmissions requires spreading the FEC codeword on a relatively high number of packets. Such an approach preserves the coding gain but dramatically affects in turn the latency.

The alternative is to match the code length to the packet size and the modulation order. It is then desirable to implement the best possible FEC scheme for a given small code length. If the design of capacity-approaching (or even achieving) FEC codes is now well understood in the asymptotic (long blocklength) regime, as demonstrated by the discovery of Polar codes [8], recent theoretical work by Polyanskiy [70] has shown that there is a severe back-off from capacity at short blocklengths.

In this section, we compare the performance on the AWGN channel, in terms of Frame Error Rate (FER), of two systems using E_8 and BW_{16} lattice codes respectively, to the LTE baseline using the LTE turbo code [3]: The generator of the Turbo encoder whose generator is given by $\mathbf{G} = [1, \mathbf{g_0}/\mathbf{g_1}]$, where $\mathbf{g_0} = [1011]$ and $\mathbf{g_1} = [1101]$. Knowing that the mother code rate of the LTE encoder is 1/3, a puncturing pattern will be employed as proposed in [60]. The parameters used for the LTE baseline have been chosen to achieve comparable spectral efficiencies in bits per real dimension, and are described in Table 2.1.

Using E_8, the $n = 8$ real coordinates are equivalent to $n/2 = 4$ complex symbols. Thus, the frame length will hereafter be measured as the number of complex symbols to be sent, and will be denoted F using an *ad hoc* subscript. The obtained frame length using lattice E_8 is then $F_{E_8} = n/2 = 4$ complex symbols. Similarly, $F_{BW_{16}} = n/2 = 8$ complex symbols. Concerning the LTE baseline, the frame length depends on the interleaver length, the code rate and the constellation size. For example, for the first row in Table 2.1, we have $K = 48$ uncoded bits, thus $\frac{K}{R} = 48 \times 2 = 96$ coded bits, which corresponds to $F_{\text{LTE}} = \frac{96}{M} = 24$ complex symbols. The value of K is fixed in a way to have comparable frame lengths with E_8 and BW_{16}. Note that $K = 40$ is the smallest frame length in LTE.

Simulation results of the Frame Error Rate (FER) are plotted in Figure 2.11 as a function of the SNR expressed in dB. The performance curves show how the lattice

Table 2.1 – Code and modulation parameters used for lattice coding and the LTE baseline

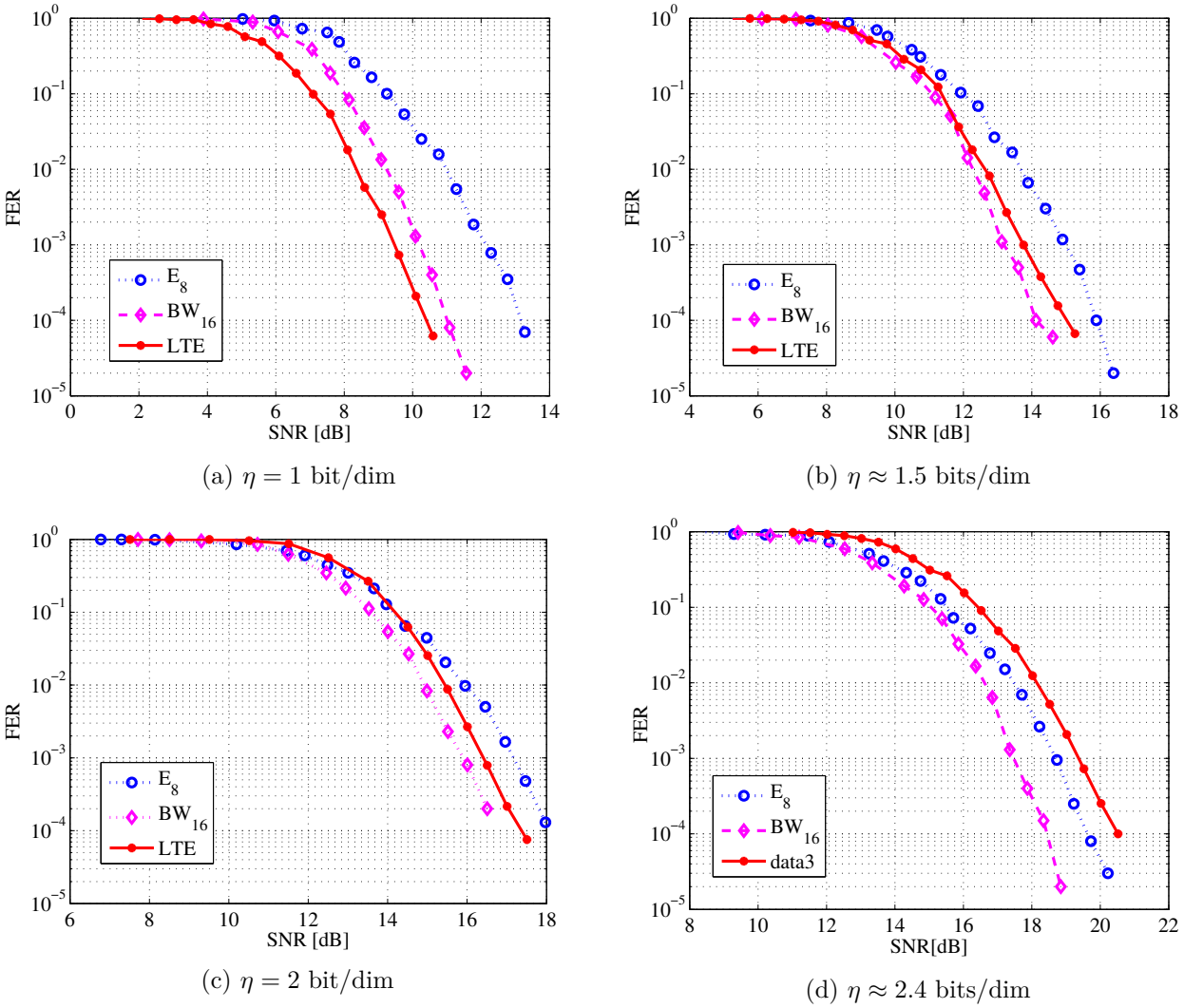
BW_{16}			E_8			LTE baseline				
L	$F_{BW_{16}}$	η	L	F_{E_8}	η	K	R	M-QAM	F_{LTE}	η
2	8	1	2	4	1	48	1/2	16−QAM	24	1
3	8	1.585	3	4	1.585	48	1/2	64−QAM	16	1.5
4	8	2	4	4	2	64	2/3	64−QAM	16	2
5	8	2.32	5	4	2.32	40	4/5	64−QAM	8.3	2.4

coding scheme turns into an interesting approach while increasing the spectral efficiency. The simulations were run for schemes having comparable spectral efficiencies and the same amount of transmitted data bits. Observing the performance at a FER of 10^{-4}, we notice that a lattice coding scheme using BW_{16} provides a gain of 2 dB over the LTE baseline having the smallest frame length $K = 40$ uncoded bits. The gain is a result of the joint near-optimal decoding of the lattice, which is made possible through the sphere decoder thanks to the linearity of the lattice. Moreover, it is clear that increasing the lattice dimension also has a significant impact on the system's performance. Therefore, dealing with higher-dimensional lattices, of dimensions even higher than 16, is a point worth considering.

2.4 Conclusion

This chapter focuses on the lattice shaping operation over both AWGN and Rayleigh fading channels. We have seen that shaping is mandatory when using lattices for transmission over the AWGN channel in order to minimize the average transmission power. More importantly, the chapter proposes a lattice code decoder algorithm based on re-shaping, that outputs a lattice point inside the shaping boundaries, thus resolving the naive lattice decoding issue where the decoded lattice points may fall outside the shaping domain.

When dealing with a Rayleigh fading channel, the lattice points are modified in such a way that the shaping gain exists for higher values of the spectral efficiency. Indeed, simulations applied to the Gosset lattice E_8 have shown that nested shaping noticeably improves the system's performance as for a spectral efficiency equal to $\eta = 4$ bits/dim.



(a) $\eta = 1$ bit/dim

(b) $\eta \approx 1.5$ bits/dim

(c) $\eta = 2$ bit/dim

(d) $\eta \approx 2.4$ bits/dim

Figure 2.11 – Frame error rate comparison between E_8, BW_{16} and short frame LTE Turbo code.

To this point, lattices were obtained by straightforward encoding of an integer vector $\mathbf{b}$ to a lattice point $\mathbf{x} = \mathbf{Gb}$, the lattice decoding being processed using the popular sphere decoder algorithm. This encoding/decoding scheme, applied to lattices of low dimensions can no longer be implemented at affordable complexity for moderate to high-dimensional lattices. Consequently, attention will hereafter be devoted toward methods for constructing lattices of higher dimension n, specifically constructions using error-correcting codes.

CHAPTER 3

Multilevel Lattice Coding

3.1 Introduction

Results of Chapter 2 have shown us that the use of lattices for information transmission must be complemented by a shaping operation in order to reduce the average transmission power. A near-to-optimal shaping using a sphere decoder was implemented and applied to lattices of low dimensions.

In order to improve the lattice performance, we now wish to build lattices of higher dimensions, for which however the sphere decoding algorithm is no longer a practical solution. In this case, instead of straightforward encoding of an integer vector to a lattice point, we will resort to lattice encoding using *Coded Modulation*. In fact, as mentioned in Section 1.4, lattices can be constructed using one or more q-ary error-correcting codes, with q prime, which allows them to inherit the underlying codes' properties. When restricting the construction to one code, i.e. using Construction A, q needs to be large enough in order to have a good lattice [29], which in turn increases the decoding complexity. Therefore, instead of working with one code over $\mathbb{F}_q$, we can use multilevel structures that provide the ability to construct the desired lattices with the ease of working with $\mathbb{F}_2$.

Multilevel Coding (MLC) was first introduced by Imai and Harawaki in [49] as an interesting approach for coded modulation. MLC is a method for constructing long, powerful lattice codes using nested component codes defined over a smaller alphabet. This approach of combining coding and modulation through MLC was proven to provide a power and bandwidth-efficient scheme, which was investigated in an important body of literature [36, 37, 23, 47, 48, 90, 89, 50, 92].

The MLC scheme is based on a combination of several error-correcting codes, with one independent code associated to subsets of a signal constellation at each level. Each level represents a sub-channel induced by a subset partition, and the component codes operate at different rates that can be assigned according to various design rules. This approach is flexible in the choice of codes, given that it allows the use of block, convolutional or concatenated codes.

In retrospect, Constructions B through D' used for building dense lattices (see section 1.4) can be viewed as a multilevel coding approach. For instance, Barnes-Wall lattices are the result of an MLC construction where component codes are Reed-Muller codes and the Leech lattice results from choosing the component codes to be Golay codes. This idea was enlightened by Forney *et al.* in [42], where the authors provided guidelines for multilevel codes design that were found to be useful in practice. Their multilevel construction, which is equivalent to lattice Construction D, was based on binary lattice partition chains and a good choice of binary codes, and was shown to provide an interesting scheme to approach the Poltyrev capacity.

For the receiver side, Imai and Hirakawa also introduced the multistage decoder (MSD). The MSD procedure consists in decoding each component code individually starting from the lowest level (which corresponds to the most powerful code), and proceeding by taking into consideration decisions made in prior stages. While it may seem suboptimal (since decoding errors are passed to higher levels), this process induces a considerable reduction in complexity compared to the maximum-likelihood decoder. Moreover, it was shown that the resulting lattice can still be capacity-achieving using MSD, if and only if the individual component codes are properly chosen [48].

In this chapter, our interest shall be focused on the multilevel lattice coding scheme over the AWGN channel. We first introduce basic multilevel lattice coding concepts related to lattice partitions and construction D in Section 3.2. Then, we explain the appropriate component codes choice using capacity rule in Section 3.3. In Section 3.4, the famous Reed-Muller codes are implemented in our different multilevel designs. For these designs, we choose to employ one, two and four dimensional standard binary partition chains, for which we give the capacity curves, system models and simulation results. In Section 3.5, we propose a method for LLR approximation in an MLC scheme based on the von Mises distribution. The chapter is concluded in Section 3.6.

3.2 Lattice partitions and construction D

Multilevel coding is based on partitioning a signal constellation into subsets. Assume a signal set S_0, the partitioning operation consists in dividing S_0 into M non-overlapping subsets such that the union of these subsets forms S_0. We will only consider the case where all the subsets have equal number of elements. If S_1 is a subset of S_0 then the resulting partition is denoted by S_0/S_1, and the cardinal of S_0/S_1, denoted by $|S_0/S_1|$, is called the partition order and is equal to M .

Analogously, if Λ' is a sublattice of the n-dimensional lattice Λ, then the quotient group Λ/Λ' is called a lattice partition, and Λ is the disjoint union of M cosets of Λ':

$$\Lambda = \cup_{\mathbf{a}\in A}(\Lambda' + \mathbf{a}) \tag{3.1}$$

where A is the set of all the coset representatives for the cosets of Λ' in Λ.

A lattice partition chain $\Lambda_1/, \cdots /\Lambda_r$ is obtained by a repeated partitioning of subsets, i.e., using a chain of nested lattices $\Lambda_r \subseteq \Lambda_{r-1} \subseteq \cdots \subseteq \Lambda_1$ with quotient groups $\Lambda_1/\Lambda_2, \cdots, \Lambda_{r-1}/\Lambda_r$. We will restrict our attention to lattice partition chains in which all subset partitions at any given level have the same order, i.e., $|\Lambda_i/\Lambda_{i+1}| = M \ \forall i$.

If $|\Lambda_i/\Lambda_{i+1}| = M = 2^m$, then m bits can be used to represent the different cosets of Λ_{i+1} in Λ_i. Thus, each of the M coset representatives is a vector, in the set A, labeled with m bits.

For example, the set of integers $\mathbb{Z}$ is a one-dimensional lattice that can be divided into even and odd integers, which correspond to the subsets $2\mathbb{Z}$ and $2\mathbb{Z}+1$ respectively. Therefore, $\mathbb{Z}/2\mathbb{Z}$ is a partition of order 2 and it is called a *binary partition*. $2\mathbb{Z}$ can in turn be divided into two subsets $4\mathbb{Z}$ and $4\mathbb{Z}+2$, and so on until the partition chain $\mathbb{Z}/2\mathbb{Z}/\cdots/2^r\mathbb{Z}$ is obtained. The coset representatives for each partition level $2^{i-1}\mathbb{Z}/2^i\mathbb{Z}$ belong to the set $A = \{0, 2^{i-1}\}$.

In the MLC scheme, each partitioning level Λ_i/Λ_{i+1} is associated to a code $\mathcal{C}_i(N, k_i)$, which selects a sequence of coset representatives for the cosets of Λ_{i+1} in Λ_i. For example, with the one-dimensional integer partition $\mathbb{Z}/2\mathbb{Z}$, the code $\mathcal{C}_1$ takes its elements in the set $\{0, 1\}$, and $\mathcal{C}_1$ is thus a binary code. In the remainder of the chapter, we will restrict ourselves to binary partitions, and therefore to binary component codes.

When dealing with nested binary linear codes $\mathcal{C}_1 \subseteq \mathcal{C}_2 \subseteq \cdots \subseteq \mathcal{C}_{r-1}$, the multilevel construction is tantamount to the previously seen "construction D" (see Section 1.4). The result is a lattice L consisting of all the vectors of the form:

$$\mathbf{a}_1\mathcal{C}_1 + \cdots + \mathbf{a}_{r-1}\mathcal{C}_{r-1} + \mathbf{z} \tag{3.2}$$

where $\mathbf{a}_i$, for $i = 1, \cdots, r-1$, is a coset representative for the partition Λ_i/Λ_{i+1}, and $\mathbf{z} \in (\Lambda_r)^N$.

The lattice L admits a volume:

$$\mathrm{V}(L) = \frac{\mathrm{V}(\Lambda_r)^N}{2^{\sum_{i=1}^{r-1} k_i}} = 2^{-N\sum_{i=1}^{r-1} R_i}\mathrm{V}(\Lambda_r)^N = 2^{-NR_{\mathcal{C}}}\mathrm{V}(\Lambda_r)^N. \tag{3.3}$$

where $R_{\mathcal{C}}$ denotes the sum rate of component codes.

The question we shall now address is how to choose the right binary codes.

3.3 Appropriate code choice: Capacity Rule

The crucial point for the design of a multilevel coding scheme is the assignment of code rates for each coding level. The capacity rule, proposed in [47], [89] and [40], states that the capacity C of a digital modulation scheme can be achieved if the rate R_i, at the individual coding level i, is chosen to approach the capacity C_i of the equivalent channel.

Consequently, computing the capacity of each level is an essential requirement for the considered multilevel approach. The idea is, given a lattice partition chain $\Lambda_1/\cdots/\Lambda_r$, to obtain the capacity curve of each partition level Λ_i/Λ_{i+1}, and thus fix a maximum value for the component codes' rates that are allowed to be used. We hereafter explain how to compute the capacity of the channel associated to any lattice partition Λ/Λ'.

Equation 3.1 shows that any vector $\mathbf{x} \in \Lambda$ can be written as:

$$\mathbf{x} = \mathbf{x}' + \mathbf{a}$$

where $\mathbf{x}' \in \Lambda'$ and $\mathbf{a} \in A$. Hence, we can write [42]:

$$\mathbf{a} = \mathbf{x} \bmod \Lambda'.$$

For example, if $\Lambda/\Lambda' = \mathbb{Z}/2\mathbb{Z}$, we have $\mathbf{a} = \mathbf{x} \bmod 2\mathbb{Z}$, which in this case is equivalent to:

$$\mathbf{a} = \mathbf{x} \bmod 2 = \begin{cases} 0 & \text{if } \mathbf{x} \text{ is even} \\ 1 & \text{if } \mathbf{x} \text{ is odd} \end{cases}.$$

Given a lattice partition Λ/Λ', the Λ/Λ' channel is defined as a mod-Λ' channel, i.e., a channel with a mod-Λ' operation at the receiver front end, and whose input

is restricted to discrete lattice points drawn from the set $(\Lambda + \mathbf{a}) \cap \mathcal{R}(\Lambda')$ for some translate vector $\mathbf{a}$, i.e., the set containing elements of a translate $\Lambda + \mathbf{a}$ of Λ that fall in a fundamental region $\mathcal{R}(\Lambda')$ of Λ'. The capacity of the Λ/Λ' channel for a noise variance σ^2 per dimension is [42]:

$$C(\Lambda/\Lambda', \sigma^2) = C(\Lambda', \sigma^2) - C(\Lambda, \sigma^2). \tag{3.4}$$

$C(\Lambda, \sigma^2)$ is defined as the capacity of a mod-Λ channel, whose input is any point drawn from a fundamental region $\mathcal{R}(\Lambda)$ of Λ, and that has a mod-Λ operation at the receiver front end. Hence, if $\mathbf{y} = \mathbf{x} + \mathbf{w}$ is a vector at the input of the mod-Λ channel, with $\mathbf{x}$ the transmitted vector and $\mathbf{w}$ a white Gaussian noise vector of variance σ^2 per dimension, then $\mathbf{y}$ is first subject to a mod-Λ operation, which results in:

$$\mathbf{y}' = \mathbf{y} \bmod \Lambda = (\mathbf{x} + \mathbf{w}) \bmod \Lambda = \mathbf{x} \bmod \Lambda + \mathbf{w} \bmod \Lambda.$$

We set $\mathbf{w}' = \mathbf{w} \bmod \Lambda$. $\mathbf{w}'$ is referred to as the Λ-aliased white Gaussian noise vector, i.e., the white Gaussian noise after the mod-Λ operation.

For an n-dimensional lattice Λ, the capacity $C(\Lambda, \sigma^2)$ is defined as [42]:

$$C(\Lambda, \sigma^2) = \log \mathrm{V}(\Lambda) - h(\Lambda, \sigma^2). \tag{3.5}$$

where $h(\Lambda, \sigma^2)$ is the differential entropy of the Λ-aliased noise over the fundamental region $\mathcal{R}(\Lambda)$ of Λ.

Let f_{σ^2} be the probability density function (PDF) of the Gaussian noise $\mathbf{w} \in \mathbb{R}^n$ of mean zero and variance σ^2:

$$f_{\sigma^2}(\mathbf{w}) = (2\pi\sigma^2)^{-n/2} \mathrm{e}^{-||\mathbf{w}||^2/2\sigma^2}.$$

The Λ-aliased Gaussian function, i.e., the function that maps $\mathbf{w}$ to $\mathbf{w}'$, with $\mathbf{w}'$ drawn from a fundamental region of Λ is:

$$f_{\Lambda,\sigma^2}(\mathbf{w}') = \sum_{k \in \Lambda} f_{\sigma^2}(\mathbf{w}' + k) = (2\pi\sigma^2)^{-n/2} \sum_{k \in \Lambda} \mathrm{e}^{-||\mathbf{w}'+k||^2/2\sigma^2} \qquad \mathbf{w}' \in \mathcal{R}(\Lambda). \tag{3.6}$$

Thus, we say that the noise element $\mathbf{w}$ is "wrapped" into $\mathbf{w}'$ and $f_{\Lambda,\sigma^2}(\mathbf{w}')$ is equivalent to the Wrapped normal distribution ($\mathcal{WN}$) [64].

Going back to our example, we have $\Lambda = \mathbb{Z}$ and $\mathcal{R}(\mathbb{Z}) = [-1/2, 1/2]$. The $\mathbb{Z}$-aliased probability density function, shown in Figure 3.1, is:

$$f_{\mathbb{Z},\sigma^2}(\mathbf{w}') = \sum_{k=-\infty}^{k=+\infty} f_{\sigma^2}(\mathbf{w}' + k) \qquad k \in \mathbb{Z}. \tag{3.7}$$

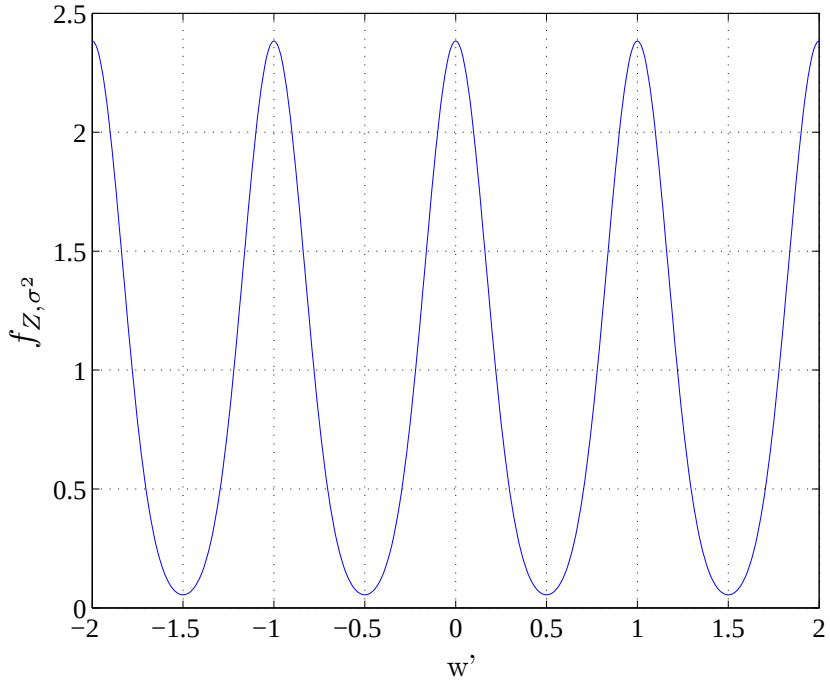


Figure 3.1 – $\mathbb{Z}$-aliased Gaussian density function $f_{\mathbb{Z},\sigma^2}(\mathbf{w}')$.

Clearly, $f_{\Lambda,\sigma^2}(\mathbf{w}')$ is Λ-periodic, in other words:

$$f_{\Lambda,\sigma^2}(\mathbf{w}') = f_{\Lambda,\sigma^2}(\mathbf{w}' + \lambda)$$

for any $\lambda \in \Lambda$.

The differential entropy of the Λ-aliased noise is written as:

$$h(\Lambda, \sigma^2) = -\int_{\mathcal{R}(\Lambda)} f_{\Lambda,\sigma^2}(\mathbf{w}') \log f_{\Lambda,\sigma^2}(\mathbf{w}') d(\mathbf{w}').$$

And the capacity of the Λ/Λ' channel in Equation 3.4 is:

$$C(\Lambda/\Lambda', \sigma^2) = h(\Lambda, \sigma^2) - h(\Lambda', \sigma^2) + \log\left(\frac{V(\Lambda')}{V(\Lambda)}\right).$$

As an extension, a lattice partition chain $\Lambda_1/\Lambda_2/, \cdots /\Lambda_r$ has a capacity computed as follows:

$$\begin{aligned} C(\Lambda_1/\Lambda_r, \sigma^2) &= C(\Lambda_1/\Lambda_2, \sigma^2) + C(\Lambda_2/\Lambda_3, \sigma^2) + \cdots + C(\Lambda_{r-1}/\Lambda_r, \sigma^2) \\ &= C(\Lambda_r, \sigma^2) - C(\Lambda_1, \sigma^2). \end{aligned} \tag{3.8}$$

For example, for the binary lattice partition $\mathbb{Z}/2\mathbb{Z}$ is a mod-$2\mathbb{Z}$ (or simply mod-2) channel with a mod-2 operation at the receiver front end. The capacity of this channel is given by:

$$C(\mathbb{Z}/2\mathbb{Z}, \sigma^2) = C(2\mathbb{Z}, \sigma^2) - C(\mathbb{Z}, \sigma^2) = h(\mathbb{Z}, \sigma^2) - h(2\mathbb{Z}, \sigma^2) + 1.$$

3.4 Multilevel Lattice construction using Reed-Muller codes

Reed-Muller codes are good candidates for creating lattice constellations via MLC, given their nested nature and good performance [56, 55]. The relation between Reed-Muller codes and lattices was already mentioned in Section 1.4.5, where nested $\mathcal{RM}$ codes were employed to obtain the famous Barnes-Wall lattices. In this case, the component codes' rates are chosen using what is known as the *balanced distance rule*. The rule states that at each level i, the minimum Euclidean distance d_i of the signal points and the minimum Hamming distance δ_i of each code $\mathcal{C}_i$ are related by the following formula:

$$d^2 > \min\{d_i^2 \delta_i\}$$

where d^2 is the squared minimum Euclidean distance of multilevel codewords.

Since we wish to maximize the minimum Euclidean distance, the natural solution would be to choose the product $d_i^2\delta_i$ to be equal for all the levels. Since d_i is imposed by the choice of the lattice partition chain, it is the Hamming distance that leads to choosing one code over another.

Even though this strategy has offered the densest known lattices for dimensions up to 32, it may not provide the best performance for lattices of moderate to high dimensions. In this section, we follow the multilevel lattice construction provided by Forney *et al.* in [42], while employing binary Reed-Muller codes of length N as component codes: the code rates at each level are chosen based on the capacity rule. The multilevel design is studied over lattice partition chains of various dimensions n. The issues addressed for each dimension are related to the choice of lattices, the number of code levels and the appropriate code rates.

The multilevel design consists in building a lattice L, of dimension equal to nN, that achieves a target error probability $P_e \to 0$ at a $\mathrm{VNR}(L, \sigma^2)$ as close as possible to 1 (0 dB), where $\mathrm{VNR}(L, \sigma^2)$ is the volume-to-noise ratio defined in Equation (1.34). A lattice L satisfying this condition approaches the Poltyrev capacity [69].

Combining Equation (3.3) in Equation (1.34), the log of VNR can be written as:

$$\begin{aligned}
\log(\mathrm{VNR}(L, \sigma^2)) &= \log\left(\frac{\mathrm{V}(L)^{2/nN}}{2\pi e\sigma^2}\right) \\
&= \log\frac{2^{\frac{-2R_c}{n}}\mathrm{V}(\Lambda_r)^{2/n}}{2\pi e\sigma^2} \\
&= \frac{-2}{n}R_c + \frac{2}{n}\log(\mathrm{V}(\Lambda_r)) - \log(2\pi e\sigma^2). \qquad (3.9)
\end{aligned}$$

As seen in Section 3.2, the multilevel construction is implemented over the lattice partition chain $\Lambda_1/\Lambda_2/\cdots/\Lambda_r$, where each partition Λ_i/Λ_{i+1} is associated to a code $\mathcal{C}_i$ of rate $R_i = \frac{k_i}{N}$. The total error probability for a lattice L described as in (3.2) is upper-bounded by:

$$P_e(L, \sigma^2) \leq \sum_{i=1}^{r} P_e(\mathcal{C}_i, \sigma^2) + P_e(\Lambda_r, \sigma^2). \tag{3.10}$$

Which means that in order to have a very small overall decoding error probability, we should choose Λ_r such that $P_e(\Lambda_r, \sigma^2) \to 0$ and codes $\mathcal{C}_i$ with error probabilities that also tend to zero. According to [42] the desired lattice L is obtained when the following conditions are met:

1. The lattice Λ_1 is such that $\mathrm{VNR}(\Lambda_1, \sigma^2)$ is too small that $C(\Lambda_1, \sigma^2) \approx 0$. More specifically, $\mathrm{VNR}(\Lambda_1, \sigma^2) < 0$ dB.
2. The lattice Λ_r is such that $P_e(\Lambda_r, \sigma^2) \approx 0$.
3. The code $\mathcal{C}_i$ approaches the capacity of the Λ_i/Λ_{i+1} channel.

We hereafter explain in details the choice of Λ_1, Λ_r and the code rates R_i for multilevel constructions over 1, 2 and 4 dimensional binary lattice partition chains.

3.4.1 One-dimensional lattice partitions

We consider the one-dimensional lattice partition chain $\mathbb{Z}/2\mathbb{Z}/\cdots/2^r\mathbb{Z}$. As already mentioned, each partition $2^{i-1}\mathbb{Z}/2^i\mathbb{Z}$ is a binary partition, i.e., $2^{i-1}\mathbb{Z}/2^i\mathbb{Z} = \mathrm{GF}(2)$ for all i, and it has coset representatives chosen from the set $A_i = \{0, 2^{i-1}\}$.

The set of nested binary linear codes $\mathcal{C}_i(N, k_i)$ for $1 \leq i \leq r-1$ are chosen such that $\mathcal{C}_i$ has a rate $R_i = \frac{k_i}{N}$ that is close to the capacity $C(2^{i-1}\mathbb{Z}/2^i\mathbb{Z})$. In order to determine the sufficient number of levels, capacity curves must be available. The curves are plotted in Figure 3.2 which shows that for any value of the noise variance per dimension σ^2, we have two effective levels, i.e., two levels whose capacity is not too close to either 0 or 1, which leaves us with the two-level construction $\mathbb{Z}/2\mathbb{Z}/4\mathbb{Z}$. This is in conformity with [42], where the authors have pointed out that coding over $\mathbb{Z}/2\mathbb{Z}/4\mathbb{Z}$ suffices for practical use.

The next step is to find the value of σ^2 for which $\mathrm{VNR}(L, \sigma^2) \approx 0$ dB. On one hand, condition 1 states that $\mathrm{VNR}(\mathbb{Z}, \sigma^2)$ must be less than 0 dB. On the other hand, even though the point is to have a powerful code at the first level, i.e., a code of relatively high redundancy, we know that achieving a high redundancy comes at the expense of

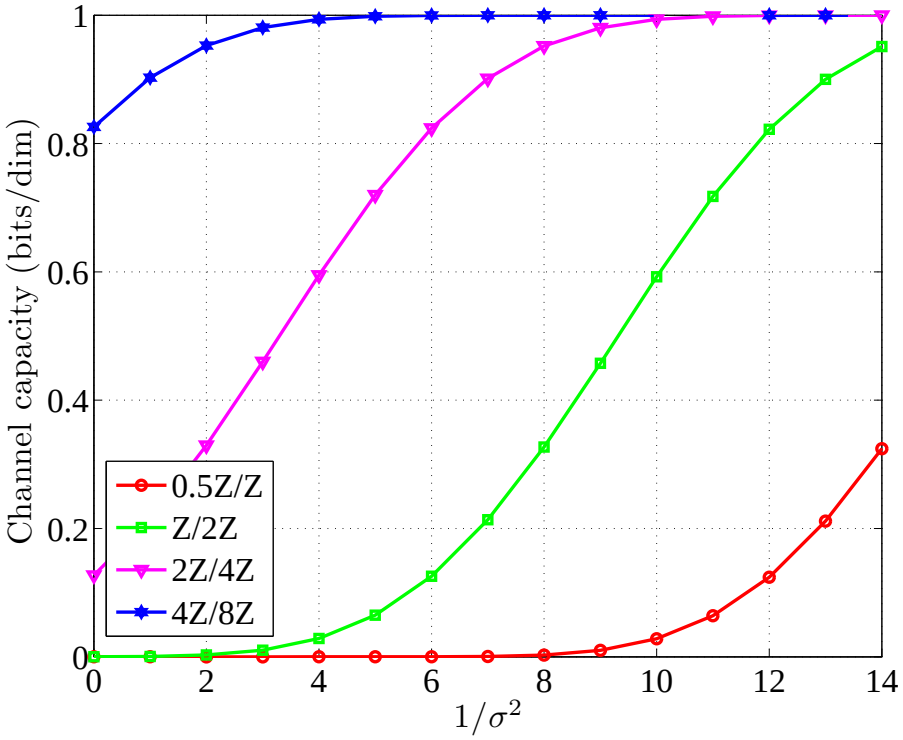


Figure 3.2 – Capacity curves for the one-dimensional lattice partition chain $\mathbb{Z}/2\mathbb{Z}/\cdots/2^r\mathbb{Z}$.

bandwidth expansion. Consequently, the goal is to choose a compromising value of σ^2 that allows us to approach $\mathrm{VNR}(L, \sigma^2) = 0$ dB while achieving a reasonable code redundancy.

The code redundancy $\rho(\mathcal{C})$ in bits per 2 dimensions is defined as [42]:

$$\rho(\mathcal{C}) = \log(\mathrm{VNR}(L, \sigma^2)) - \log(\mathrm{VNR}(\Lambda_1, \sigma^2)). \tag{3.11}$$

Table 3.1 displays, for different possible values of $\mathrm{VNR}(\mathbb{Z}, \sigma^2)$, the corresponding noise variance per dimension σ^2, the capacities C_1 and C_2 associated to the channels $\mathbb{Z}/2\mathbb{Z}$ and $2\mathbb{Z}/4\mathbb{Z}$ respectively, the code redundancy $\rho(\mathcal{C})$ and $\mathrm{VNR}(L, \sigma^2)$. The latter can be deducted from equation (3.9) by taking $n = 1$ and $\log(\mathrm{V}(4\mathbb{Z})) = 2$.

For $\sigma^2 = 0.1041$, the various factors are combined in a way that is well-suited for constructing the lattice L. This value of σ^2 makes it possible to both have a low code redundancy $\rho(\mathcal{C})$, and use a powerful code $\mathcal{C}_1$, with rate $R_1 \leq 0.56$. Using two nested binary linear codes $\mathcal{C}_1 \subseteq \mathcal{C}_2$, we build the lattice L using a multilevel Construction D as follows:

$$L = \mathcal{C}_1 + 2\mathcal{C}_2 + 4\mathbb{Z}^N. \tag{3.12}$$

Table 3.1 – The two-level construction constants for different value of σ^2 (n=1).

VNR$(\mathbb{Z}, \sigma^2)$ (dB)	σ^2	C_1	C_2	$\rho(\mathcal{C})$ (b/2D)	VNR(L, σ^2) (dB)
0	0.0585	0.8477	0.9998	0.3	0.92
-1	0.0737	0.7516	0.9988	0.5	0.5033
-1.5	0.0827	0.6956	0.9976	0.6	0.3473
-2.5	0.1041	0.5687	0.9912	0.88	0.1504
-3	0.1168	0.5012	0.9846	1.02	0.0966

Note that N must be taken sufficiently large in order to find a good range of rates that allow us to approach the different levels' capacity.

The system model of the corresponding multilevel coding/multistage decoding scheme is depicted in Figure 3.3. The transmitted lattice point $\mathbf{x} \in L$, obtained according to Equation (3.12), is sent over the AWGN channel. At the receiver side, the received observation $\mathbf{y}$ is first subject to a mod-2 operation:

$$\mathbf{y} \bmod 2 = (\mathbf{x} + \mathbf{w}) \bmod 2 = (\mathbf{c}_1 + 2\mathbf{c}_2 + \mathbf{z} + \mathbf{w}) \bmod 2 = (\mathbf{c}_1 + \mathbf{w}) \bmod 2.$$

Hence, the first level serves as a decoder for only the first code $\mathcal{C}_1$. The soft information is fed to the Reed-Muller soft-input decoder, which in turn outputs the codeword $\hat{\mathbf{c}}_\mathbf{1}$. The soft information of each of the N components of $\mathbf{c_1}$, solely dependent on the received observation $\mathbf{y}$, is computed as follows:

$$\text{LLR} = \ln\left(\frac{\Pr(y_j|c_{1,j}=0)}{\Pr(y_j|c_{1,j}=1)}\right) = \ln\left(\frac{\sum\limits_{c\in 2\mathbb{Z}} \mathrm{e}^{\frac{-|y_j-c|^2}{2\sigma^2}}}{\sum\limits_{c'\in 2\mathbb{Z}+1} \mathrm{e}^{\frac{-|y_j-c'|^2}{2\sigma^2}}}\right). \tag{3.13}$$

where y_j and $c_{1,j}$, for $j = 1, \cdots N$, denote the j^{th} component of $\mathbf{y}$ and $\mathbf{c_1}$ respectively. The soft-input decoder algorithm for Reed-Muller codes is described in Appendix B.

The decoder proceeds with the second level, whose job is to decode the second code $\mathcal{C}_2$. To this aim, it begins by subtracting the obtained codeword $\hat{\mathbf{c}}_1$, resulting in the vector :

$$\mathbf{y}^{(1)} = \mathbf{y} - \hat{\mathbf{c}}_1$$

which depends on both the received observation and the decoded word of the previous level. For this reason, a good choice of $\mathcal{C}_1$ is very important, since decoding errors in the first level are passed to the second one.

Similarly, $\mathbf{y}^{(1)}$ is this time subject to a mod-4 operation in order to keep what is only associated to $\mathcal{C}_2$, and the decoder outputs the second decoded codeword $\hat{\mathbf{c}}_2$ thanks

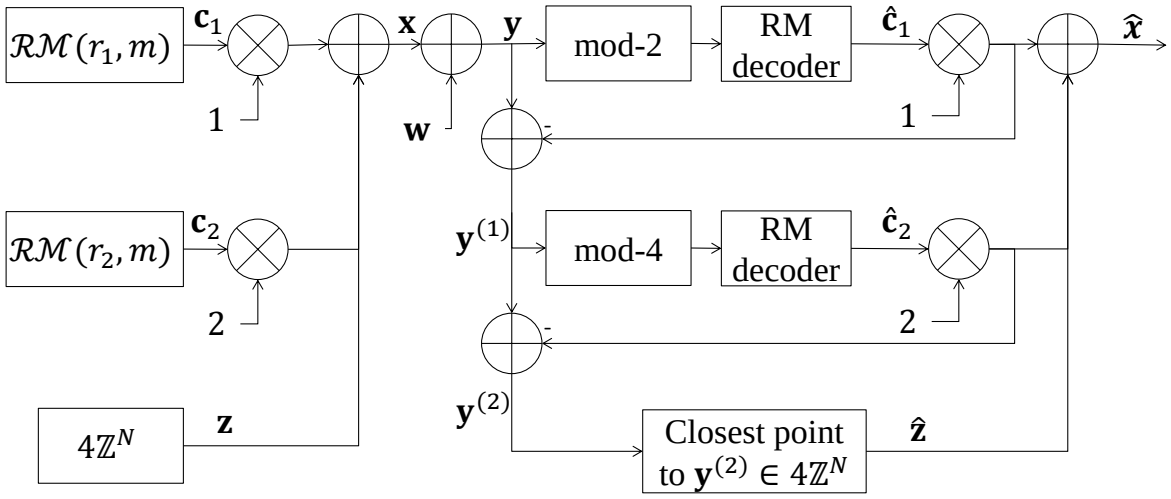


Figure 3.3 – MLC encoding and MSD decoding schemes for a two-level lattice construction (n=1).

to the soft information:

$$\text{LLR} = \ln\left(\frac{\Pr(y_j^{(1)}|c_{2,j}=0)}{\Pr(y_j^{(1)}|c_{2,j}=1)}\right) = \ln\left(\frac{\sum\limits_{c\in 4\mathbb{Z}} \mathrm{e}^{\frac{-|y_j^{(1)}-c|^2}{2\sigma^2}}}{\sum\limits_{c'\in 4\mathbb{Z}+2} \mathrm{e}^{\frac{-|y_j^{(1)}-c'|^2}{2\sigma^2}}}\right). \tag{3.14}$$

The last stage of the decoding process corresponds to the uncoded part of the lattice construction. Once the decoded codeword $\hat{\mathbf{c}}_\mathbf{2}$ is subtracted, we search in $4\mathbb{Z}^N$ for the closest point to the vector:

$$\mathbf{y}^{(2)} = \mathbf{y}^{(1)} - 2 \times \hat{\mathbf{c}}_2.$$

Now that all three parts of the equation (3.12) are available, the deduced decoded lattice point is then:

$$\hat{\mathbf{x}} = \hat{\mathbf{c}}_1 + 2\hat{\mathbf{c}}_2 + \hat{\mathbf{z}} \qquad \text{where} \quad \hat{\mathbf{z}} \in 4\mathbb{Z}^N.$$

Figure 3.4 shows the Word Error Rate plotted as a function of the VNR for a 2-level lattice construction using binary Reed-Muller codes of length $N = 1024$. The rates of the component codes are $R_1 = 0.377$ and $R_2 = 0.9453$, corresponding to $\mathcal{RM}(4, 10)$ and $\mathcal{RM}(7, 10)$ respectively. Also shown is the performance of the Barnes-Wall lattice having the same dimension. For $N = 1024$, the lattice BW_{1024} is produced with a 5-level construction, which obviously violates the capacity rule. Knowing that the same

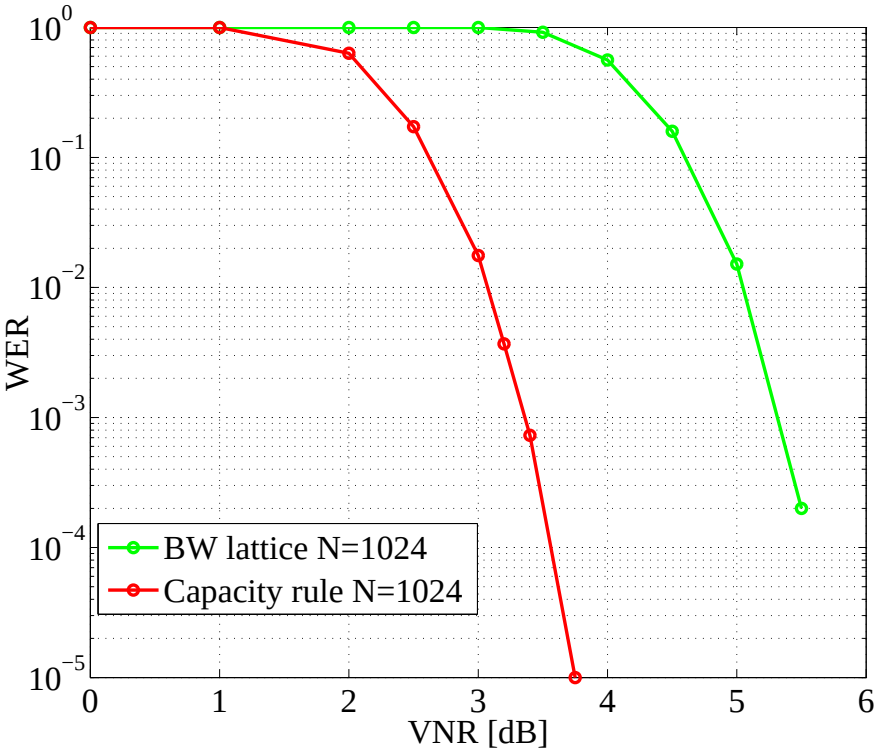


Figure 3.4 – Performance comparison of a Barnes-Wall lattice and a 2-level lattice construction built according to capacity rule for a code length $N = 1024$.

multistage decoding was used for both cases, it is clear that opting for the capacity rule in lieu of Barnes-Wall lattices leads to a considerable performance improvement. If we set $P_e = 10^{-5}$, then the gap to the Poltyrev capacity, is 3.8 dB. This gap is due to the capacity losses of component codes.

3.4.2 Two-dimensional lattice partitions

We now consider the lattice partition chain $\mathbb{Z}^2/D_2/2\mathbb{Z}^2/2D_2/\cdots/2^r\mathbb{Z}^2/2^rD_2$, where D_2 is the checkerboard lattice of dimension 2. In the complex domain, this lattice partition chain is equivalent to $\mathbb{Z}[j]/(1+j)\mathbb{Z}[j]/\cdots/2^r\mathbb{Z}[j]$ (where $j^2 = -1$). The lattice $(1+j)\mathbb{Z}[j]$ can be viewed as a rotated version of $\mathbb{Z}[j]$, with $R = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ the scaled rotation operator. That's why, it is also common to find this lattice partition noted as $\mathbb{Z}^2/R\mathbb{Z}^2/2\mathbb{Z}^2/2R\mathbb{Z}^2/\cdots$.

The capacity curves in this case are depicted in Figure 3.5, which shows that for a two-dimensional lattice partition chain, the number of effective levels increases to 4.

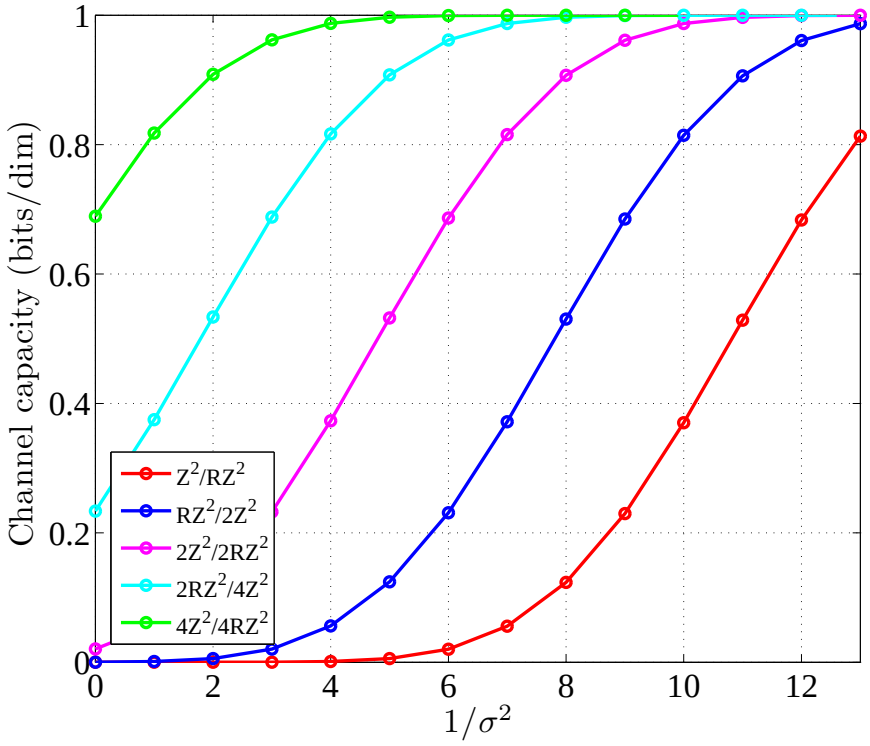


Figure 3.5 – Capacity curves for the two-dimensional lattice partition chain $\mathbb{Z}^2/R\mathbb{Z}^2/\cdots/2^r\mathbb{Z}^2$.

As a result, the multilevel construction will be carried out over the lattice partition chain $\mathbb{Z}^2/R\mathbb{Z}^2/2\mathbb{Z}^2/2R\mathbb{Z}^2/4\mathbb{Z}^2$.

In order to proceed with the lattice construction, we need to determine the coset representatives for each partition level Λ_{i-1}/Λ_i. This can be done with the help of Figure 3.6, where points belonging to the nested lattices $4\mathbb{Z}^2 \subseteq 2R\mathbb{Z}^2 \subseteq 2\mathbb{Z}^2 \subseteq R\mathbb{Z}^2 \subseteq \mathbb{Z}^2$ are visualized in the plane $\mathbb{Z}^2$. Since binary lattice partitions are being used, we know that $\Lambda_{i-1}/\Lambda_i = \{0, \mathbf{a}_i\}$, in other words, the lattice Λ_{i-1} is the union of two cosets of Λ_i: Λ_i itself and $\Lambda_i + \mathbf{a}_i$ where $\mathbf{a}_i$ is a lattice point in Λ_{i-1} but not in Λ_i. The relations between the different subsets are then:

- $\mathbb{Z}^2 = R\mathbb{Z}^2 \cup (R\mathbb{Z}^2 + (1,0))$.
- $R\mathbb{Z}^2 = 2\mathbb{Z}^2 \cup (2\mathbb{Z}^2 + (1,1))$.
- $2\mathbb{Z}^2 = 2R\mathbb{Z}^2 \cup (2R\mathbb{Z}^2 + (2,0))$.
- $2R\mathbb{Z}^2 = 4\mathbb{Z}^2 \cup (4\mathbb{Z}^2 + (2,2))$.

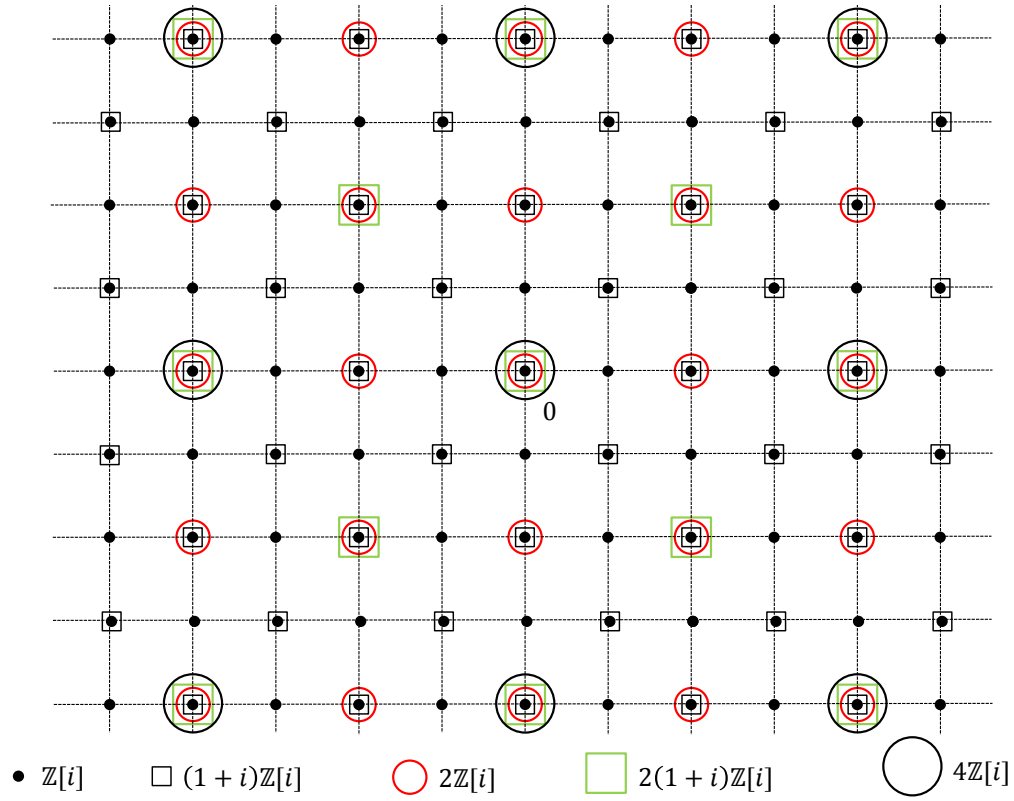


Figure 3.6 – The integer two-dimensional lattice $\mathbb{Z}^2$ and its sublattices.

Thus, the lattice L can be described using the following code formula:

$$L = \mathbf{a}_1\mathcal{C}_1 + \mathbf{a}_2\mathcal{C}_2 + \mathbf{a}_3\mathcal{C}_3 + \mathbf{a}_4\mathcal{C}_4 + (4\mathbb{Z}^2)^N.$$

The system model illustrated in Figure 3.7 shows the multilevel construction over the two-dimensional lattice partition chain, resulting in a lattice L of dimension $2N$. The sent lattice points $\mathbf{x} \in L$ are of the form:

$$\mathbf{x} = (1,0)\mathbf{c_1} + (1,1)\mathbf{c_2} + (2,0)\mathbf{c_3} + (2,2)\mathbf{c_4} + \mathbf{z}.$$

where $\mathbf{z}$ is a point in $(4\mathbb{Z}^2)^N$. At the receiver, each of the four underlying codes is decoded separately using log likelihood ratios that are calculated similarly to the previous case of the one-dimensional partition. At level i associated to the partition Λ_{i-1}/Λ_i, the soft information is:

$$\text{LLR} = \ln\left(\frac{\Pr(\mathbf{y}_j^{(i-1)}|\mathbf{c}_{i,j}=0)}{\Pr(\mathbf{y}_j^{(i-1)}|\mathbf{c}_{i,j}=1)}\right) = \ln\left(\frac{\sum\limits_{\mathbf{c}\in\Lambda_i} \mathrm{e}^{\frac{-||\mathbf{y}_j^{(i-1)}-\mathbf{c}||^2}{2\sigma^2}}}{\sum\limits_{\mathbf{c}'\in\Lambda_i+\mathbf{a_i}} \mathrm{e}^{\frac{-||\mathbf{y}_j^{(i-1)}-\mathbf{c}'||^2}{2\sigma^2}}}\right). \tag{3.15}$$

Note that for the first level $i = 1$, $\mathbf{y}^{(i-1)} = \mathbf{y}$.

Table 3.2 – The four-level construction constants for different value of σ^2 (n=2).

VNR($\mathbb{Z}^2, \sigma^2$) (dB)	σ^2	C_1	C_2	C_3	C_4	$\rho(\mathcal{C})$ (b/2D)	VNR(L, σ^2) (dB)
0	0.0585	0.726	0.9696	0.9996	1	0.3	0.92
-1.5	0.0827	0.5	0.89	0.9955	1	0.6	0.35
-3	0.1168	0.2754	0.727	0.969	0.999	1.02	0.0966
-4	0.1471	0.158	0.583	0.924	0.997	1.3377	0.0269
-5	0.1852	0.077	0.422	0.845	0.991	1.6647	0.01

To determine the value of σ^2 for which VNR$(L, \sigma^2) \approx 0$ dB, we proceed similarly to the previous case, and calculate in Table 3.2 for different possible values of VNR$(\mathbb{Z}^2, \sigma^2)$, the corresponding variance per dimension σ^2, the level capacities C_i for $i = 1, 2, 3, 4$, the code redundancy $\rho(\mathcal{C})$ and VNR(L, σ^2) calculated as in Equation 3.9) with $n = 2$:

$$\log(\text{VNR}(L, \sigma^2)) = -R_{\mathcal{C}} + \log(\text{V}(4\mathbb{Z}^2)) - \log(2\pi e\sigma^2).$$

By looking at Table 3.2, we notice that in order to be able to code over 4 levels, VNR$(\mathbb{Z}^2, \sigma^2)$ must be less than -3 dB, otherwise, $C(2R\mathbb{Z}^2/4\mathbb{Z}^2) \approx 1$. For this reason, we choose to set VNR$(\mathbb{Z}^2, \sigma^2) = -4$ dB, which also allows us to have a good trade-off between the final lattice VNR(L, σ^2) on one hand, and the code redundancy $\rho(\mathcal{C})$ on the other hand.

An example is illustrated in Figure 3.8, where the Word Error Rate is plotted as a function of the VNR for a lattice L constructed using binary Reed-Muller codes of length $N = 512$. The respective code rates are: $R_1 = 0.0898, R_2 = 0.2539, R_3 = 0.7461$ and $R_4 = 0.98$ corresponding to $\mathcal{RM}(2, 9), \mathcal{RM}(3, 9), \mathcal{RM}(5, 9)$ and $\mathcal{RM}(7, 9)$ respectively. Note that the resulting lattice L is of dimensions $n \times N = 1024$. The figure also shows the performance of a lattice of equal dimension 1024, but constructed using a 2-level Construction D over a one-dimensional binary partition chain. It is clear that for lattices of equal dimensions, increasing the number of levels through coding over lattice partition chains of higher dimensions, improves the system's error rate. This gain is about 0.3 dB for a WER equal to 10^{-3}.

3.4.3 Four-dimensional lattice partitions

We keep increasing the lattice partition chain dimension in order to illustrate the impact of n on the global lattice performance. In four dimensions, the chain of binary lattice partitions is $\mathbb{Z}^4/D_4/R\mathbb{Z}^4/RD_4/2\mathbb{Z}^4/2D_4/2R\mathbb{Z}^4/\cdots$, where D_4 is the four-dimensional

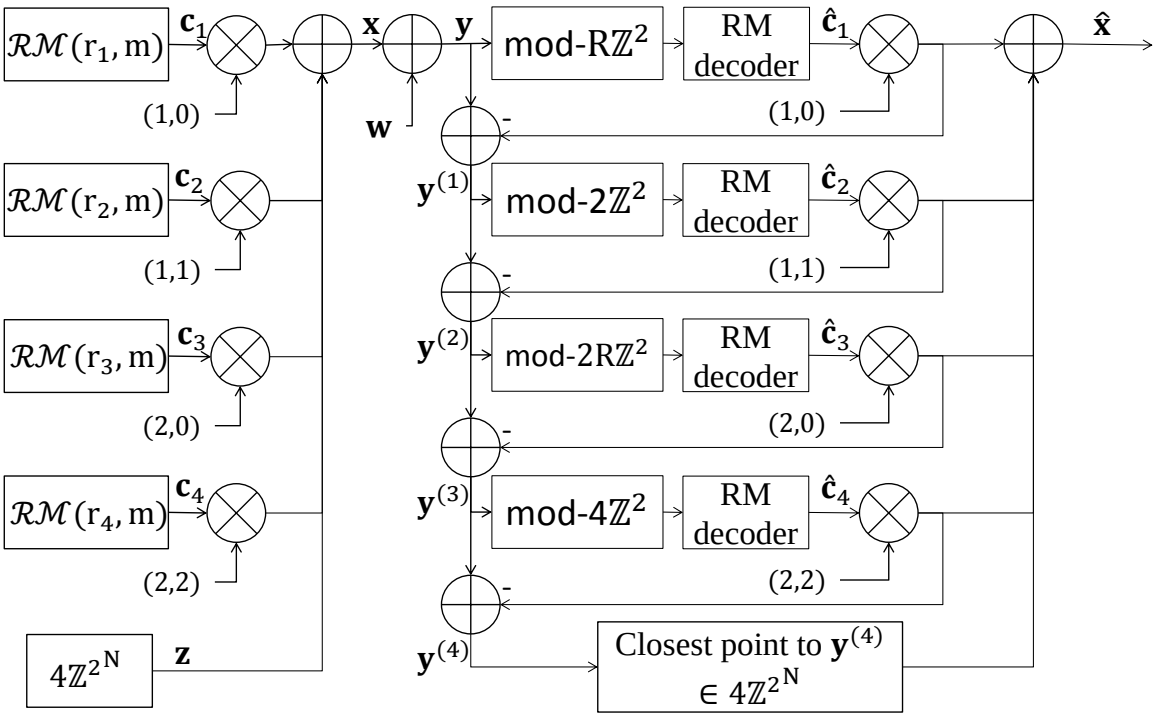


Figure 3.7 – MLC encoding and MSD decoding schemes for a four-level lattice construction (n=2).

checkerboard lattice. $R\mathbb{Z}^4$ and RD_4 are rotated versions of $\mathbb{Z}^4$ and D_4 respectively, and R is the scaled rotation operator that operates on each pair of coordinates as follows:

$$R = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{pmatrix}.$$

The lattices D_4 and RD_4 can be deduced from $\mathbb{Z}^4$ via construction A, using the parity-check code $\mathcal{C}(4,3,2)$ and the repetition code $\mathcal{C}(4,1,4)$ respectively. Hence, they can be written as:

$$D_4 = (4,3,2) + 2\mathbb{Z}^4.$$
$$RD_4 = (4,1,4) + 2\mathbb{Z}^4.$$

Finding the coset representatives for the four-dimensional lattice partition chain is not as simple as for the previous cases. We know that for each binary partition Λ_{i-1}/Λ_i, the coset representatives take their values in the set $A = \{0, \mathbf{a}_i\}$. In order to find $\mathbf{a}_i$, we are going to use Table 3.3 where the minimal algebraic and Euclidean norms are indicated for each lattice. In fact, the coset representative $\mathbf{a}_i$ must meet the following conditions:

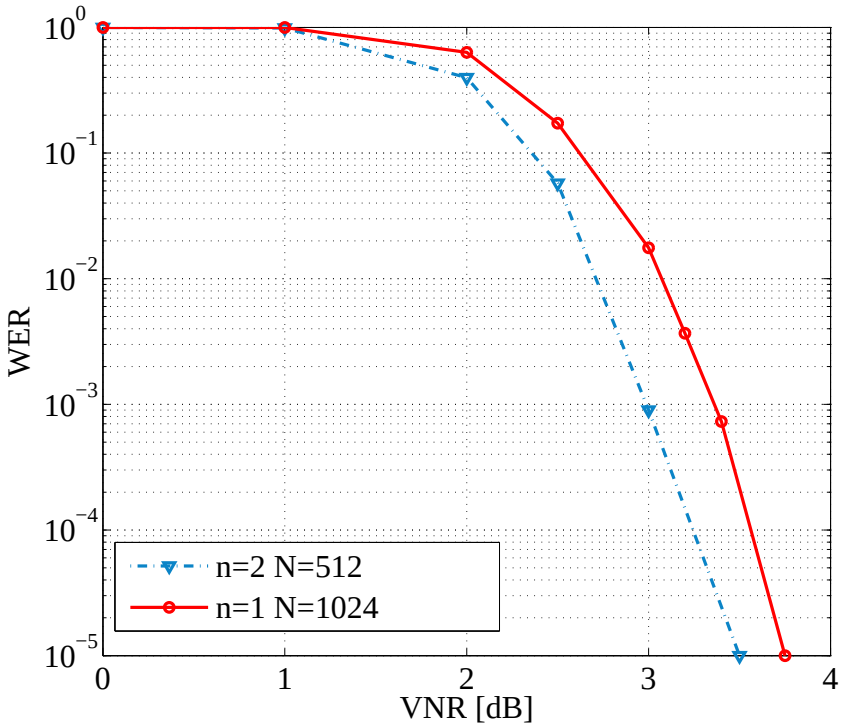


Figure 3.8 – Performance comparison between two lattices of dimension 1024 resulted from multilevel construction over one and two-dimensional lattice partition chains respectively.

Table 3.3 – Minimal algebraic and Euclidean norms of the four-dimensional lattices forming the lattice partition chain $\mathbb{Z}^4/D_4/R\mathbb{Z}^4/RD_4/2\mathbb{Z}^4/2D_4$.

Lattice Λ	Algebraic Norm $N(\Lambda)$	Euclidean Norm $d^2(\Lambda)$
$\mathbb{Z}^4$	1	1
D_4	2	2
$R\mathbb{Z}^4$	4	2
RD_4	8	4
$2\mathbb{Z}^4$	16	4
$2D_4$	32	8

- $\mathbf{a}_i \in \Lambda_{i-1}$ and $\mathbf{a}_i \notin \Lambda_i$.
- $N(\mathbf{a}_i) = N(\Lambda_{i-1})$.
- $d^2(\mathbf{a}_i) = d^2(\Lambda_{i-1})$.

Let's start with the first partition $\mathbb{Z}^4/D_4$. We know that all the vectors in $\mathbb{Z}^4$ having an Euclidean norm equal to 1 are not in D_4 because the minimal Euclidean norm in

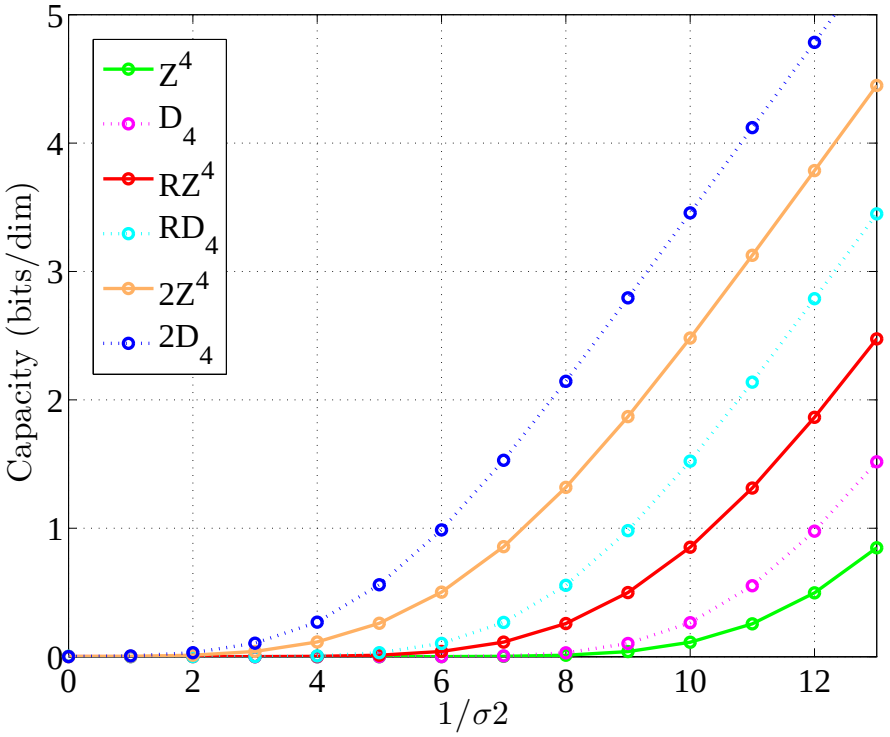


Figure 3.9 – Capacity of four-dimensional lattices versus $1/\sigma^2$.

D_4 is 2. Thus, $\mathbf{a_1}$ is a lattice point in $\mathbb{Z}^4$ having one and only one of its components equal to 1 (or -1). For instance, we can take $\mathbf{a_1} = (1, 0, 0, 0)$.

In a similar way, we conclude the following coset unions:

- $\mathbb{Z}^4 = D_4 \cup (D_4 + (1, 0, 0, 0))$
- $D_4 = R\mathbb{Z}^4 \cup (R\mathbb{Z}^4 + (1, 0, 1, 0))$
- $R\mathbb{Z}^4 = RD_4 \cup (RD_4 + (1, 1, 0, 0))$
- $RD_4 = 2\mathbb{Z}^4 \cup (2\mathbb{Z}^4 + (1, 1, 1, 1))$

Figure 3.9 shows that the capacities $C(\mathbb{Z}^4, \sigma^2)$ and $C(D_4, \sigma^2)$, in bits/dimension, are closer to each other compared to capacity difference between the rest of the lattices. This suggests that two consecutive partitions such as $D_4/R\mathbb{Z}^4$ and $R\mathbb{Z}^4/RD_4$ will have quasi equal capacities. This is confirmed in the capacity curves depicted in Figure 3.10, which also shows that for a multilevel lattice construction over a four-dimensional lattice partition chain, the number of effective levels increases to 5.

Shown in Figure 3.11 is the block diagram of the multilevel encoding/multistage decoding scheme for a five-level lattice construction over the four-dimensional lattice

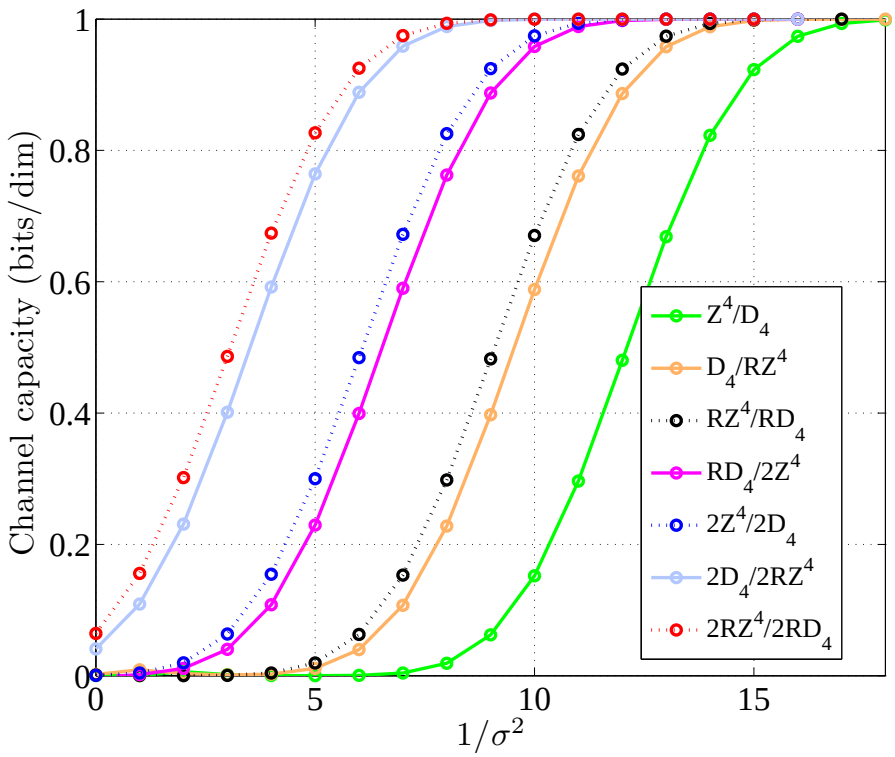


Figure 3.10 – Capacity curves for the four-dimensional lattice partition chain $\mathbb{Z}^4/D_4/R\mathbb{Z}^4/RD_4/2\mathbb{Z}^4/2D_4/2R\mathbb{Z}^4/\cdots$.

partition. The lattice L consists of all the vectors of the form:

$$\mathbf{x} = \mathbf{a}_1\mathcal{C}_1 + \mathbf{a}_2\mathcal{C}_2 + \mathbf{a}_3\mathcal{C}_3 + \mathbf{a}_4\mathcal{C}_4 + \mathbf{a}_5\mathcal{C}_5 + \mathbf{z}.$$

where $\mathbf{z} \in 2D_4$.

By the same reasoning as that applied for $n = 1$ and 2, we use Table 3.4 to choose the value of σ^2 that allows us to build a lattice L having $\mathrm{VNR}(L, \sigma^2) \to 0$, while providing a modest code redundancy. We set $\mathrm{VNR}(\mathbb{Z}^4) = -1$ dB, which corresponds to $\sigma^2 = 0.0737$. Simulations are applied to Reed-Muller codes of length $N = 256$ with codes rates 0.36, 0.63, 0.85, 0.96 and 0.99 associated to levels 1 through 5 respectively. The results are depicted in Figure 3.12 along with lattices obtained with one and two-dimensional lattice partition chains. Another comparison of the three multilevel lattice constructions investigated in this Section is shown in Figure 3.13, where L has a dimension $n \times N = 4096$.

The simulations shed the light on the way the partition dimension n affects the system's global performance. It is clear that increasing n leads to lower decoding error probabilities. Moreover, the higher the dimension of the resulting lattice L, the higher the impact of n. This proves that, although it is generally desirable to choose lattice

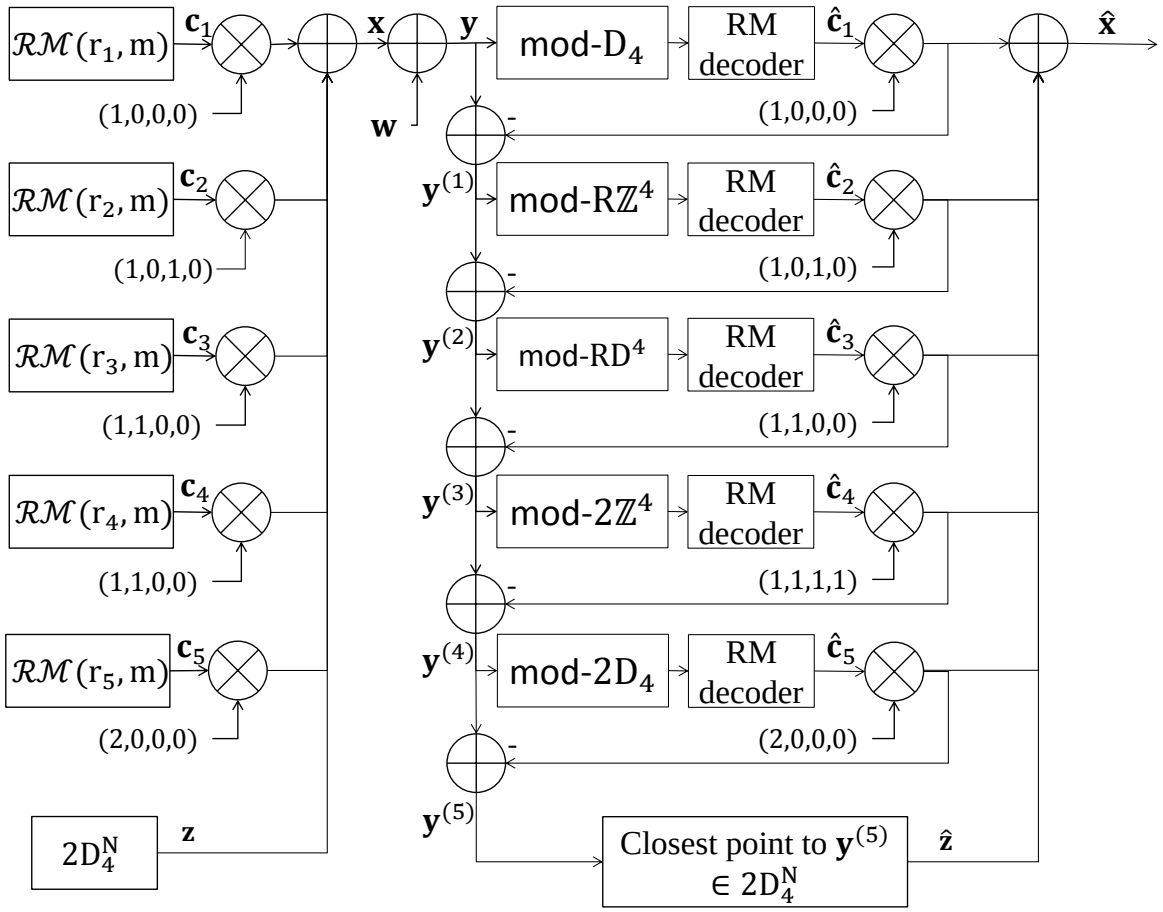


Figure 3.11 – MLC encoding and MSD decoding schemes for a five-level lattice construction (n=4).

Table 3.4 – The five-level construction constants for different value of σ^2 (n=4).

VNR($\mathbb{Z}^4, \sigma^2$) (dB)	σ^2	C_1	C_2	C_3	C_4	C_5	$\rho(\mathcal{C})$ (b/2D)	VNR(L, σ^2) (dB)
0	0.0585	0.541	0.91	0.94	0.998	0.999	0.3	0.92
-1	0.0737	0.356	0.8	0.86	0.991	0.995	0.5	0.5
-2	0.1041	0.198	0.644	0.72	0.968	0.98	0.88	0.15
-3	0.1168	0.091	0.458	0.543	0.91	0.94	1.1	0.0984

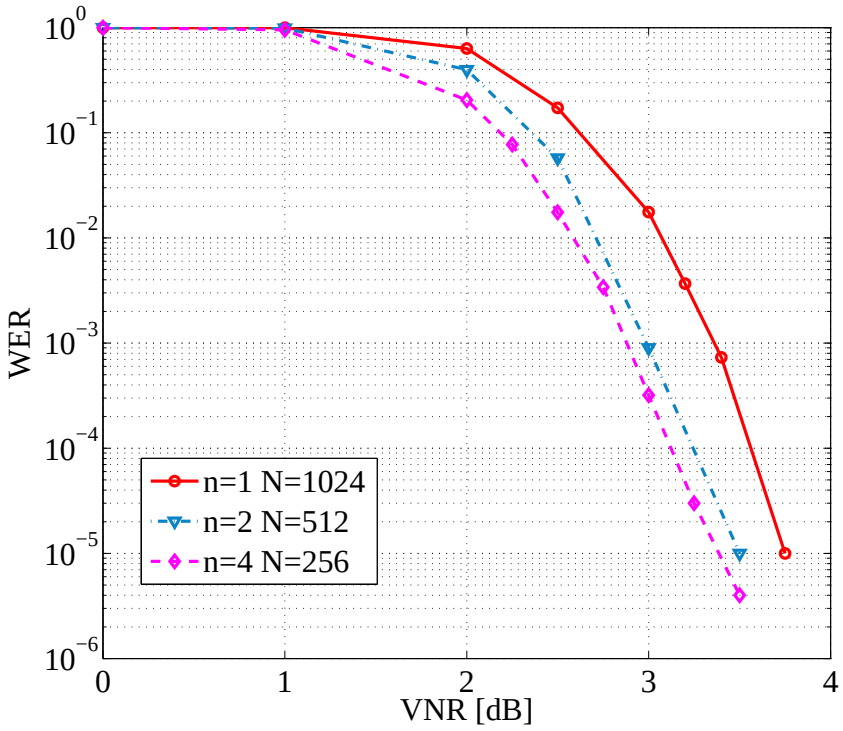


Figure 3.12 – Simulation results for an obtained lattice L of dimension 1024.

partitions of low dimension so as to reduce the number of decoding levels, this factor does actually matter in the described MLC scheme. For instance, for a lattice L of dimension 4096, using two-dimensional lattice partitions achieves a gain of almost 0.6 dB for WER=10^{-3} over using the one-dimensional chain $\mathbb{Z}/2\mathbb{Z}/4\mathbb{Z}$, while building L over lattice partitions of dimension $n = 4$ provides a gain of 0.9 dB over $n = 1$ for the same error rate.

3.5 LLR estimation using the von Mises distribution

In the last section of this chapter, we present results related to the log-likelihood ratio estimation at each level of the multistage decoding scheme for $n = 1$. (The results can be generalized to n=2 and 4). The exact LLR calculation, computed as in Equations (3.13) and (3.14), induces infinite sums. We therefore propose an LLR approximation based on the von Mises distribution in order to alleviate this problem. This result was the subject of an article published in the Electronics letters journal [6].

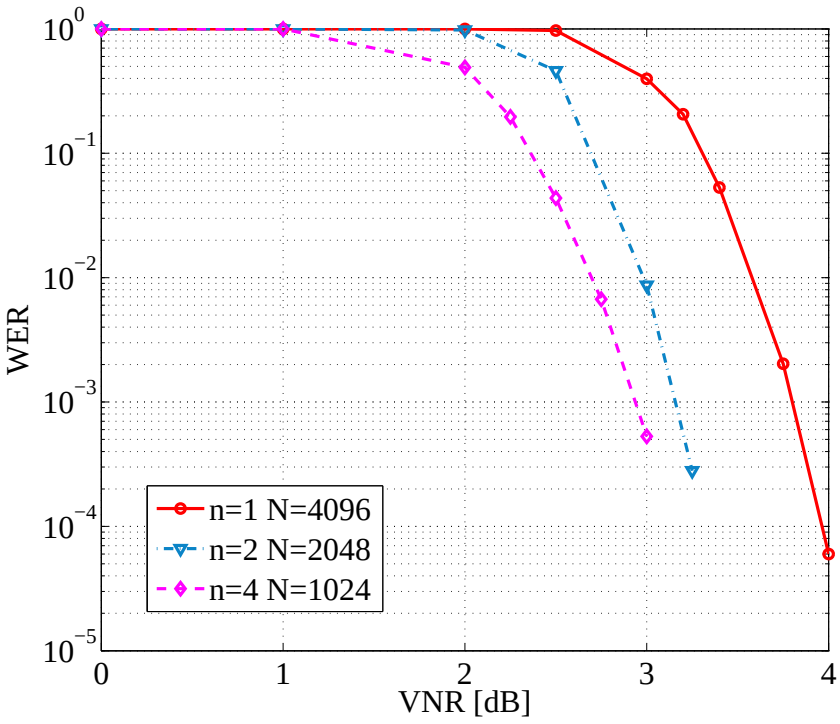


Figure 3.13 – Simulation results for an obtained lattice L of dimension 4096.

3.5.1 The von Mises distribution

The von Mises distribution, also known as the circular normal distribution or Tikhonov distribution, is appropriate for modelling a random variable of circular nature. It can be considered as the circular analogue of the normal distribution.

The von Mises distribution, shown in Figure 3.14, is defined as [34] [86]:

$$f(\theta, \eta, \kappa) = \frac{\mathrm{e}^{\kappa \cos(\theta - \eta)}}{2\pi I_0(\kappa)}, \tag{3.16}$$

where $\theta \in$ any interval of length 2π, η is the mean direction, $\kappa \geq 0$ is the concentration parameter and

$$I_n(\kappa) = \frac{1}{2\pi} \int_0^{2\pi} \mathrm{e}^{\kappa \cos \xi} \cos(n\xi) d\xi$$

is the order n modified Bessel function. The mean direction η is analogous to the mean of the Gaussian distribution, while the concentration parameter κ is analogous to the inverse of the variance in the Gaussian distribution (See figure 3.14).

An approximation of the von Mises and the Wrapped Normal distributions has been proposed in [33], by setting $\mu = \eta$ and $\sigma^2 = -2\ln(Q(\kappa))$, where $Q(\kappa) = \frac{I_1(\kappa)}{I_0(\kappa)}$.
Conversely, the von Mises distribution will have a mean μ and a concentration param-

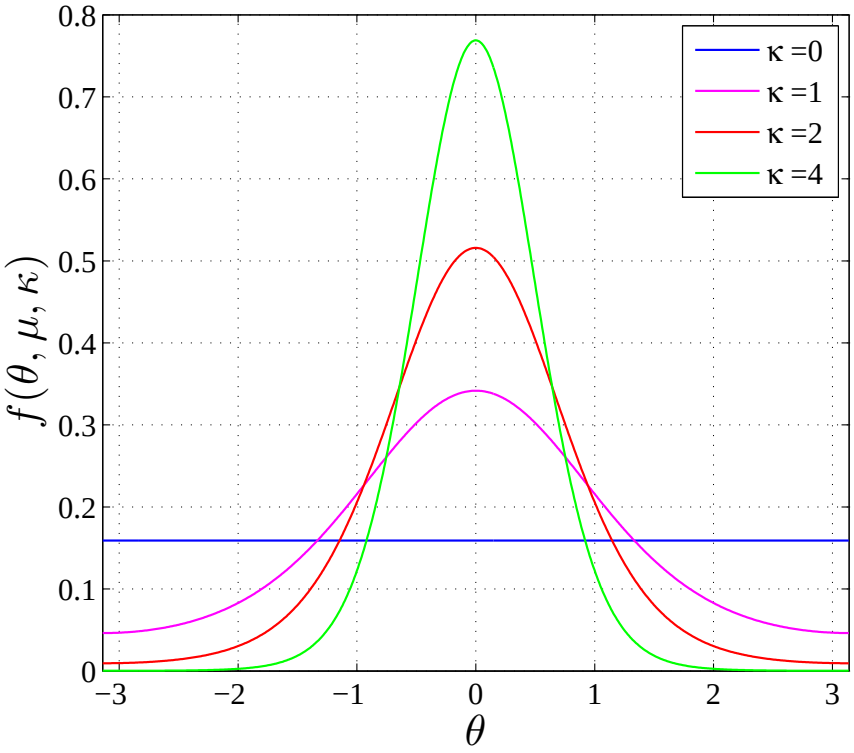


Figure 3.14 – PDF of the von Mises distribution for $\mu = 0$ and different values of κ.

eter:

$$\kappa = Q^{-1}(\exp(\frac{-\sigma^2}{2})) \tag{3.17}$$

It follows that the noise vector $\mathbf{w}'$ of Equation (3.6), which was shown to follow a Wrapped Normal distribution, also behaves according to a von Mises distribution with $\mu = 0$ and κ computed as in (3.17). Therefore, according to (3.16), each element y_j of the received vector has the conditional PDFs:

$$\begin{cases} \Pr(y_j|x_j = 1) = \dfrac{\mathrm{e}^{\kappa\cos(\pi y_j - \pi)}}{2\pi I_0(\kappa)} \\ \Pr(y_j|x_j = 0) = \dfrac{\mathrm{e}^{\kappa\cos(\pi y_j)}}{2\pi I_0(\kappa)} \end{cases}$$

and the LLR estimation is:

$$\begin{aligned} \mathrm{LLR} &= \ln\left(\frac{\Pr(y_j|x_j = 0)}{\Pr(y_j|x_j = 1)}\right) \\ &= \kappa(\cos(\pi y_j) - \cos(\pi y_j - \pi). \end{aligned} \tag{3.18}$$

To illustrate the performance of the proposed LLR approximation method, we hereafter compare it with 2 other methods: the exact LLR calculation using the Wrapped

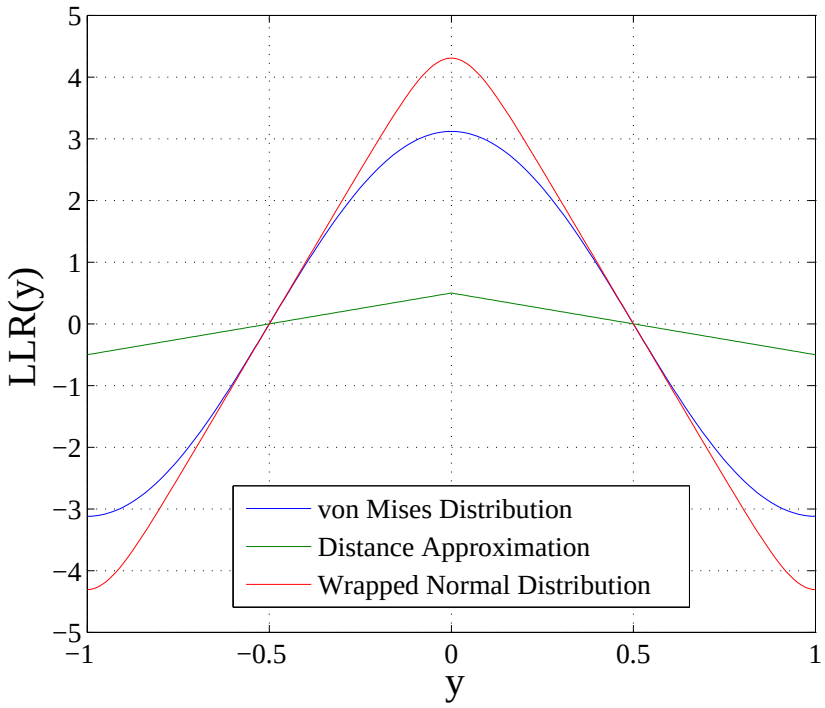


Figure 3.15 – LLR curves for 3 different methods of LLR estimation.

Normal distribution seen in Equation (3.6)and a method that we refer to as *distance approximation* explained in what follows.

For the case of $n = 1$ (one-dimensional lattice partition chain), we have seen that the received vector $\mathbf{y}$ was first subject to a mod-2 operation, meaning that it can be defined over an interval of length 2, say $[-1, 1]$. The distance approximation method consists in defining two threshold (-0.5 and 0.5 in this case) and computing the Euclidean distance that separates each component y_j of $\mathbf{y}$ from each of the thresholds. The minimum distance is the value of the corresponding LLR. The LLR signs are fixed according to Figure 3.15.

3.5.2 Simulation results

We apply the proposed LLR estimation to a soft-decision multistage decoder of two levels. The underlying binary linear codes are the Reed-Muller codes $\mathcal{RM}(3, 9)$ and $\mathcal{RM}(6, 9)$ of length 512, and rates 0.36 and 0.96 respectively. At the receiver, the computed LLRs are fed to a soft-input decoder and the Word Error Rate is calculated. No power constraint was considered in the simulations shown in Figure 3.16: The WER

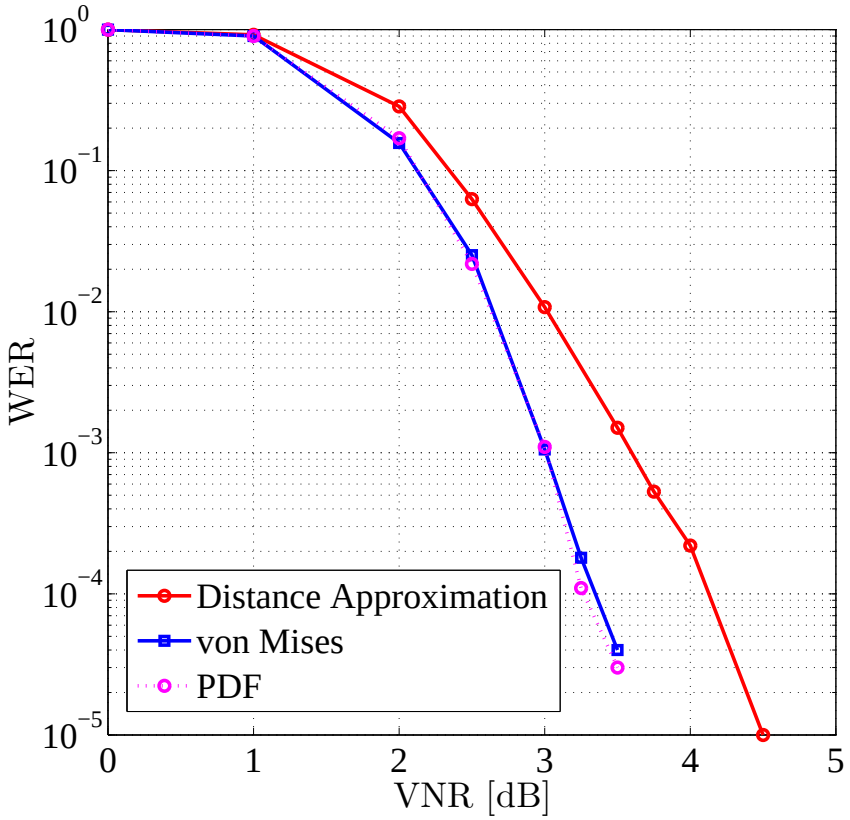


Figure 3.16 – Word Error Rate Vs VNR for multiple methods of LLR estimation.

is plotted as a function of the VNR. For each VNR value, a different concentration parameter κ is calculated as in Equation (3.17) by means of a look-up table.

Figure 3.16 shows the impact of the LLR estimation method on the decoder's performance. It is clear that opting for the simple distance approximation method is not the best option as it is likely to degrade these performances. As for the proposed method, simulation results show that no loss in error rate can be observed compared to the exact log-likelihood ratio computation. Although both methods may provide close decoding performances, the von Mises distribution has the advantage to be easily implementable since the infinite sums involved in the log-likelihood ratio expression are replaced by a cosine function and a lookup table.

3.6 Conclusion

This chapter presents an encoding/decoding scheme suitable for building lattice codes of moderate to high dimensions over the AWGN channel. The approach is based on

combining coding and modulation through multilevel coding, as opposed to the simple straightforward mapping of the integer vector to the lattice point using the lattice generator matrix.

After recalling the main multilevel lattice construction concepts, such as lattice partitions, construction D and channel capacities, we implemented the construction inspired form [42] using Reed-Muller codes. We have explained how to choose the sufficient number of levels and how to select the appropriate binary codes for standard binary lattice partitions of dimensions $n = 1, 2$ and 4. Simulation results have shown the gain in performance obtained while increasing n.

Additionally, seeing that the LLR calculation fed to the soft-input decoders involved infinite sums, we proposed a new LLR calculation method based on the von Mises distribution. Simulations showed that this method provides close decoding performance compared to the exact LLR calculation, with the advantage of being easily implementable since the infinite sums were replaced by a simple cosine function.

We shall now proceed by examining the multilevel lattice construction over the Rayleigh block fading channel. On this type of channels, the focus is shifted towards lattices of high diversity. For that, algebraic number theory will be of particular interest, since it was shown to be a key enabler for constructing lattices over fading channels.

CHAPTER 4

Multilevel Code Design for Rayleigh Block Fading Channels

4.1 Introduction

Having described the multilevel lattice construction over the AWGN channel, we now resort to more precise and complex channel models, and thus consider the use of lattice constellations for transmissions over communication links that suffer from time-varying channel conditions, i.e., fading channels. This calls for a new code design capable of improving the loss occurred by this type of channels.

On the Gaussian channel, the goodness of a lattice constellation is judged by relying on the sphere packing problem [26], i.e., how to pack, as densely as possible, a large number of identical spheres within a containing space (see Section 1.1). On this type of channels, sphere packings with high density are known to provide the best and most popular lattices (for example, lattices $D_4, E_8, \Lambda_{16}, \Lambda_{24}$). When dealing with fading channels, the crucial point lies in providing the maximum diversity [20], where diversity is the number of different component values of any two distinct elements in the constellation. According to the Chernoff bound on the pairwise error probability, the performance of lattice constellations on fading channels also depends on another parameter: the *minimum product distance*. Consequently, providing both the maximal diversity and the minimum product distance needed in the transmission system helps mitigate the fading effect that is likely to disrupt the communication. Lattices used over the Gaussian channel are not a good choice for fading channels, since they have a very small diversity.

Algebraic number theory was shown to be an effective mathematical tool that makes it possible to design good lattices for the fading channels. Extended work has been

done on single antenna fading channels to study algebraic lattice codes defined over algebraic number fields [11],[12],[13],[15]. In [43], Giraud and Belfiore showed that totally real algebraic number fields are a powerful tool for constructing lattices over the Rayleigh fading channel as they provide the maximum possible diversity. In [22], the authors have introduced lattices that are good for both Gaussian and Rayleigh fading channels by finding a trade-off between the sphere packing density of a lattice and its degree of diversity. Knowing that the initial search for lattices providing full diversity was performed with no restrictions on the shape of the lattice, resulting in a loss in average energy, the authors in [14] have presented families of rotated $\mathbb{Z}^n$ lattices using the theory of ideal lattices.

The purpose of this chapter is twofold. We first present an overview on algebraic number theory by introducing the relevant concepts and results that will be of use for our algebraic lattice construction. The exposition will be clarified by means of simple examples that illustrate the different notions. The second part consists in describing the implementation of the resulted lattices in a multilevel coding scheme. On the Rayleigh fading channel, this operation needs to be accompanied by the appropriate procedures: the lattice rotation and base reduction.

The chapter begins by introducing the two main parameters for constructing lattices on fading channels in Section 4.2. Then we provide an overview on algebraic number theory in Section 4.3, which describes the algebraic lattice design for Rayleigh fading channels and introduces the notion of ideals among which the special category of principal ideals will be of interest in order to set the ground for our MLC design. The latter is detailed in Section 4.4 and applied to two-dimensional and four-dimensional binary lattice partition chains. The chapter is concluded in Section 4.5.

4.2 Modulation diversity and product distance

The symbol error probability is upper-bounded by:

$$P_e \leq \frac{1}{M} \sum_{i \neq j} P(\mathbf{x_i} \rightarrow \mathbf{x_j}) \tag{4.1}$$

where M is the total number of symbold of length N forming the constellation, and $P(\mathbf{x_i} \rightarrow \mathbf{x_j})$ is the pairwise error probability, i.e., the probability of detecting $\mathbf{x_j}$ when $\mathbf{x_i}$ has been sent ($i \neq j$).

According to the Chernoff bound, the pairwise error probability is upper-bounded by [84]:

$$P(\mathbf{x_i} \to \mathbf{x_j}) \leq \prod_{n=1}^{N} \frac{1}{1 + \frac{\mathrm{SNR}|x_{in} - x_{jn}|^2}{4}} \tag{4.2}$$

where $x_{i,n}$ is the n^{th} component of the symbol $\mathbf{x_i}$.

We denote by $f_{i,j}$ the number of components for which $\mathbf{x_i}$ is different than $\mathbf{x_j}$. For high SNRs, the dominant terms in the sum of Equation (4.1) are those corresponding to $F = \min(f_{i,j})$, in other words, the couples $(\mathbf{x_i}, \mathbf{x_j})$ having the minimum number of distinct components. F is called the diversity order of the signal constellation, i.e., the minimum number of distinct components between any two different constellation points. Among the terms having the same diversity order F, the overall error probability is dominated by those having the minimum product distance, denoted by $d_{p,min}$, where the product distance is:

$$d_p^{(f_{i,j})}(\mathbf{x_i}, \mathbf{x_j}) = \prod_{n=1}^{f_{i,j}} |x_{in} - x_{jn}|.$$

Hence, good signal constellations have high order diversity F and minimum product distance $d_{p,min}$.

The increase in modulation diversity can be achieved thanks to a certain rotation of the signal constellation [16], as shown in Figure 4.1 for the 4-QAM constellation. The figure shows that the rotation is performed in such a way that for any value of channel fading that hits one of the symbols, the latter does not collapse with any of the other points of the constellation, as it is the case for the unrotated QAM.

It's easy to see that even a small rotation is sufficient to achieve maximal diversity, however, in order to minimize the error probability, it is important to also maximize the minimum product distance between any two points. In this example, a rotation of 13 degrees maximizes $d_{p,min}$. For a 16 QAM, the angle is $\frac{\pi}{8}$. The optimum rotation angle for a four-dimensional lattice can be found in [52]. However, as we go higher in dimensions, determining the best rotation becomes a rather complicated task. An efficient tool to construct multidimensional constellations with maximal diversity is *algebraic number theory.*

4.3 Algebraic Number Theory

Generally speaking, algebraic number theory is the study of the arithmetic of algebraic number fields and related objects, which involves using techniques from algebra and

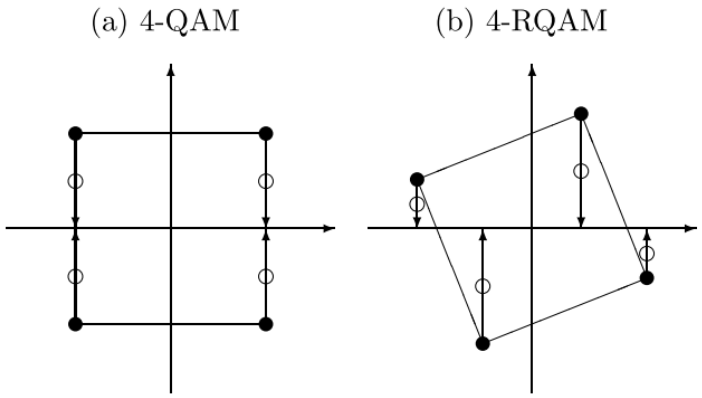


Figure 4.1 – Example of increasing modulation diversity (a) F =1, (b) F=2.

finite number theory. In order to explain how a lattice can be constructed using number theory, we need to provide some basic mathematical concepts. Therefore, we dedicate this section to listing the essential notions and definitions, while giving simple illustrative examples. For more details, we let the reader refer to [18] and [75].

Let $\mathbb{Z}$ be the ring of rational integers, and let $\mathbb{Q}$ be the field of rational numbers. The scope of this section is, starting from these two sets, to become familiar with some number-theoretical concepts, mainly:

- A number field K and its ring of integers $\mathcal{O}_K$.
- The integral basis of K.
- Canonical embeddings of an algebraic number field into the set of complex numbers $\mathbb{C}$.
- Constructing algebraic lattices from totally real number fields.
- Ideal lattices.
- Rotated $\mathbb{Z}^n$ lattices.

4.3.1 Algebraic number fields

Before going into the details, let us remind the definitions of a ring and a field.

Definition 4.3.1. *A ring*
Let S be a set together with two operations $+$ and $\times$:

$$\begin{aligned} +: S, S &\to S \\ a, b &\to a + b \end{aligned} \qquad \begin{aligned} \times: S, S &\to S \\ a, b &\to a \times b \end{aligned}$$

S is said to be ring if it satisfies the following conditions:

1. Additive associativity: $\forall a, b, c \in S, (a + b) + c = a + (b + c)$.
2. Additive commutativity: $\forall a, b \in S, a + b = b + a$.
3. There exists a neutral element 0, such that: $\forall a \in S, a + 0 = a$.
4. $\forall a \in S$, there exists an inverse $-a$ such that: $-a + a = 0$.
5. Multiplicative associativity: $\forall a, b, c \in S, (a \times b) \times c = a \times (b \times c)$.
6. The operation $\times$ is distributive over $+$:

   $\forall a, b, c \in S, a \times (b + c) = (a \times b) + (a \times c)$ and $(b + c) \times a = (b \times a) + (c \times a)$.

   A ring may also satisfy some other optional conditions:
7. Multiplicative commutativity: $\forall a, b \in S, a \times b = b \times a$.
8. There exists an element $1 \in S$ such that $1.a = a,\ \forall a \in S$.
9. $\forall a \neq 0 \in S$, there exists an element $a^{(-1)} \in S$ such that:

$$\forall a \neq 0 \in S, a \times a^{(-1)} = a^{(-1)} \times a = 1.$$

Definition 4.3.2. *A Field*
A ring satisfying all additional conditions 7-9 is called a field, whereas a ring satisfying conditions 8 and 9 is called a skew field (or a division ring).

One can easily verify that $\mathbb{Z}$ and $\mathbb{Q}$ are respectively a ring and a field. As a first step, we define some basic algebraic structures until we get to grasp the notion of what is known as a *number field*.

Definition 4.3.3. *Field extension*
Let K and K' be two fields. If $K' \subseteq K$, we say that K is a field extension of K'.

Starting from $\mathbb{Q}$, we can build a field extension K by taking an element not in $\mathbb{Q}$, say $\sqrt{2}$, and adding all its multiples and powers to $\mathbb{Q}$. That way, we obtain a field that contains both $\mathbb{Q}$ and $\sqrt{2}$, and we denote it by $\mathbb{Q}(\sqrt{2})$. We say that $K = \mathbb{Q}(\sqrt{2})$ is a field extension of $\mathbb{Q}$.

Definition 4.3.4. *Finite extension*
The dimension of K as a vector space over $\mathbb{Q}$ is called the *degree* of K over $\mathbb{Q}$ and is denoted by $[K|\mathbb{Q}]$. If $[K|\mathbb{Q}]$ is finite, we say that K is a finite extension of $\mathbb{Q}$.

Definition 4.3.5. *Number field*
A number field is a field extension of $\mathbb{Q}$ of finite degree.

Still with $K = \mathbb{Q}(\sqrt{2})$, K has a structure of a vector space over $\mathbb{Q}$. Any point $\mathbf{x} \in K$ can be written as $\mathbf{x} = a + b\sqrt{2}$ with $a, b \in \mathbb{Q}$. Thus, $\mathbb{Q}(\sqrt{2})$ is a vector space over $\mathbb{Q}$ of dimension 2, so it is a number field of degree 2.

Definition 4.3.6. *Algebraic number*
Let x be an element of K, x is called an algebraic number if it is a root of a monic polynomial with coefficients in $\mathbb{Q}$. The monic polynomial of lowest degree whose root is x is called the minimum polynomial of x, and is denoted M_x.

In our example, the minimal polynomial of $\sqrt{2}$ over $\mathbb{Q}$ is $M_{\sqrt{2}} = X^2 - 2$. $\sqrt{2}$ is the solution of a polynomial with integer coefficients. $\sqrt{2}$ is therefore said to be an algebraic number.

Definition 4.3.7. *Algebraic extension*
If all the elements of K are algebraic, we say that K is an algebraic extension of $\mathbb{Q}$.

It is clear to see that any $x \in K = \mathbb{Q}(\sqrt{2})$ is a root of the polynomial $M_x(X) = X^2 - 2aX + a^2 - 2b^2$ with $(a, b) \in \mathbb{Q})$. Thus, $\mathbb{Q}(\sqrt{2})$ is an algebraic extension of $\mathbb{Q}$.

Theorem 4.3.1. *If K is a vector space over $\mathbb{Q}$, there is an algebraic number $\theta \in K$ such that K is generated by the powers of θ, θ is called the primitive element and we write $K = \mathbb{Q}(\theta)$. If degree(K)= n, then $(1, \theta, \cdots, \theta^{n-1})$ is a basis of K, and the degree of the minimal polynomial of θ is n.*

Definition 4.3.8. *Ring of integers*
We define the ring of integers of a number field K, denoted by $\mathcal{O}_K$, as the set of all algebraic integers of K, where an algebraic integer is a root of a monic polynomial with coefficients in $\mathbb{Z}$.

In this example, it is clear that the algebraic integers are the set $\mathcal{O}_K = \mathbb{Z}[\sqrt{2}] = \{a + b\sqrt{2},\ a, b \in \mathbb{Z}\}$. Note that $\mathcal{O}_K$ is a ring since it is closed under all operations except for the inversion. For example, $(2 + 2\sqrt{2})^{-1} = \frac{\sqrt{2}-1}{2}$ does not belong to $\mathbb{Z}[\sqrt{2}]$.

4.3.2 Integral basis and Canonical Embedding

We will now take a look at the structure of the ring of integers $\mathcal{O}_K$ and define its integral basis. We will also become familiar with the embedding of a number field K into $\mathbb{C}$, the field of complex numbers, which will allow us to define two invariants of K: the discriminant and the signature.

For the number field $K = \mathbb{Q}(\sqrt{2})$, we have seen that the corresponding ring of integers is $\mathcal{O}_K = \mathbb{Z}[\sqrt{2}] = \{a+b\sqrt{2},\ a, b \in \mathbb{Z}\}$, which means that $\mathcal{O}_K$ has a basis given by $\{1, \sqrt{2}\}$ over $\mathbb{Z}$. $\mathcal{O}_K$ is referred to as a $\mathbb{Z}$-module. As a generalization, if K is a number field of degree n, the ring of integers $\mathcal{O}_K$ constitutes a $\mathbb{Z}$-module of rank n, i.e., it has a basis of n vectors over $\mathbb{Z}$.

Let $\{\omega_1, \omega_2, \cdots, \omega_n\}$ be a basis of K. If $\{\omega_i\}$ is also a generator of the $\mathbb{Z}$-module $\mathcal{O}_K$, then it is called an integral basis of K. In that case, any element x in $\mathcal{O}_K$ can be written as $x = \sum_{i=1}^{n} a_i\omega_i$, with $a_i \in \mathbb{Z}$.

We will now see how K can be represented, or embedded into $\mathbb{C}$.

Definition 4.3.9. *Embedding*
Let K and K' be two algebraic extensions of $\mathbb{Q}$. We introduce the map $\phi : K \to K'$ as a $\mathbb{Q}$-homomorphism if for each $a \in \mathbb{Q}$ we have $\phi(a) = a$. If $K' = \mathbb{C}$, the $\mathbb{Q}$-homomorphism $\phi : K \to \mathbb{C}$ is called an embedding of K into $\mathbb{C}$. The embedding is an injective map which, for each $a, b \in K$, satisfies:

- $\phi(a + b) = \phi(a) + \phi(b)$
- $\phi(a.b) = \phi(a).\phi(b)$
- $\phi(1) = 1$

Theorem 4.3.2. *Let $K = \mathbb{Q}(\theta)$ be a number field of degree n over $\mathbb{Q}$. There are exactly n distinct embeddings $\sigma_i : K \to \mathbb{C}$ $(i = 1, \cdots, n)$. Each embedding is defined by $\sigma_i(\theta) = \theta_i$, where θ_i are the distinct zeros in $\mathbb{C}$ of the minimum polynomial of the primitive θ over $\mathbb{Q}$.*

Note that $\sigma_1(\theta) = \theta_1 = \theta$, which means that σ_1 is the identity map: $\sigma_1(K) = K$. By taking any element $x \in K$, we have $x = \sum_{j=1}^{n} a_j\theta^j$ with $a_j \in \mathbb{Q}$, and applying the embedding σ_i we get:

$$\sigma_i(x) = \sigma_i(\sum_{j=1}^{n} a_j\theta^j) = \sum_{j=1}^{n} \sigma_i(a_j)\sigma_i(\theta^j) = \sum_{j=1}^{n} a_j\theta_i^j$$

which shows that the image of any element x by σ_i is given entirely by θ_i.

By means of these embeddings, we define two quantities that will be relevant in the following algebraic lattice construction: the norm and the trace.

Definition 4.3.10. *Norm* and *Trace*
Let $x \in K$. The elements $\sigma_i(x)$ for $i = 1, \cdots, n$ are called the conjugates of x, and

$$N(x) = \prod_{i=1}^{n} \sigma_i(x) \ , \ \mathrm{Tr}(x) = \sum_{i=1}^{n} \sigma_i(x)$$

are respectively the *norm* and *trace* of x.

Theorem 4.3.3. *For any $x \in K$, we have $N(x)$ and $Tr(x) \in \mathbb{Q}$. If $x \in \mathcal{O}_K$, then $N(x)$ and $Tr(x) \in \mathbb{Z}$.*

Going back to our example with $K = \mathbb{Q}(\sqrt{2})$, we have seen that the minimal polynomial is $X^2 - 2$, which roots are $\theta_1 = \sqrt{2}$ and $\theta_2 = -\sqrt{2}$. Thus, we have $\sigma_1(\theta) = \sqrt{2}$ and $\sigma_2(\theta) = -\sqrt{2}$. For $x \in K, x = a + b\sqrt{2}$, we have:

$$\sigma_1(x) = a + b\sigma_1(\sqrt{2}) = a + b\sqrt{2} \qquad \sigma_2(x) = a + b\sigma_2(\sqrt{2}) = a - b\sqrt{2}.$$

The norm and trace of x are respectively $N(x) = \sigma_1(x)\sigma_2(x) = a^2 - 2b^2$ and $\mathrm{Tr}(x) = \sigma_1(x) + \sigma_2(x) = 2a$.

Definition 4.3.11. *Discriminant*
Still with $K = \mathbb{Q}(\theta)$ of degree n, let $\{\omega_1, \cdots, \omega_n\}$ be the basis of K. We define the discriminant of this basis to be:

$$d_K = \det^2[\sigma_j(\omega_i)_{i,j=1}^n]$$

Theorem 4.3.4. *The discriminant of any basis for $K = \mathbb{Q}(\theta)$ is non-zero. If all the conjugates of θ are real, then the discriminant of any basis is positive.*

Example: Let us compute the discriminant of $\mathbb{Q}(\sqrt{2})$:

$$d_K = \det^2 \begin{pmatrix} \sigma_1(1) & \sigma_1(\sqrt{2}) \\ \sigma_2(1) & \sigma_2(\sqrt{2}) \end{pmatrix} = \det^2 \begin{pmatrix} 1 & \sqrt{2} \\ 1 & -\sqrt{2} \end{pmatrix} = 8.$$

Let's now take another example of an algebraic number field and apply the different notions seen so far.

We have $K = \mathbb{Q}(\sqrt{5})$. An integral basis for K is not $\{1, \sqrt{5}\}$ as one may conclude from the previous example. In order to determine the integral basis of $\mathcal{O}_K$, we refer to

[51], where it was stated that, in general, for $K = \mathbb{Q}(\sqrt{d})$, with d a squarefree integer, we have:

$$\mathcal{O}_K = \begin{cases} \mathbb{Z}[\sqrt{d}] & \text{if } d \equiv 2,3 \bmod 4 \\ \mathbb{Z}[\frac{1+\sqrt{d}}{2}] & \text{if } d \equiv 1 \bmod 4 \end{cases}$$

It follows that, any point in $x \in \mathcal{O}_K$ can be written as:

$$x = \begin{cases} a + b\sqrt{d} \quad \text{for } a,b \in \mathbb{Z}, b \neq 0 & \text{if } d \equiv 2,3 \bmod 4 \\ a + b(\frac{1+\sqrt{d}}{2}) \quad \text{for } a,b \in \mathbb{Z}, b \neq 0 & \text{if } d \equiv 1 \bmod 4 \end{cases}$$

Thus, a basis for $\mathcal{O}_K$ is $\{1, \frac{1+\sqrt{5}}{2}\}$. The minimal polynomial of $\frac{1+\sqrt{5}}{2}$ is $X^2 - X - 1$, whose roots are $\theta_1 = \frac{1+\sqrt{5}}{2}$ and $\theta_2 = \frac{1-\sqrt{5}}{2}$. Thus, we have $\sigma_1(\theta) = \frac{1+\sqrt{5}}{2}$ and $\sigma_2(\theta) = \frac{1-\sqrt{5}}{2}$, and the discriminant of K is:

$$d_K = \det^2 \begin{pmatrix} \sigma_1(1) & \sigma_1(\frac{1+\sqrt{5}}{2}) \\ \sigma_2(1) & \sigma_2(\frac{1+\sqrt{5}}{2}) \end{pmatrix} = \det^2 \begin{pmatrix} 1 & \frac{1+\sqrt{5}}{2} \\ 1 & \frac{1-\sqrt{5}}{2} \end{pmatrix} = 5$$

Definition 4.3.12. *Signature*
Let $\sigma_1, \cdots, \sigma_n$ be the n embeddings of K into $\mathbb{C}$. We suppose that r_1 is the number of embeddings with image in $\mathbb{R}$, and $2r_2$ the number of embeddings with signature in $\mathbb{C}$. We then have:

$$n = r_1 + 2r_2.$$

The pair (r_1, r_2) is called the *signature* of K. If $r_2 = 0$ we have a totally real algebraic number field. If $r_1 = 0$ we have a totally complex algebraic number field. In all other cases, we speak about complex algebraic number fields. Note that $K = \mathbb{Q}(\sqrt{2})$ is a totally real number field with $n = r_1 = 2$.

Let's now consider the algebraic number field $K = \mathbb{Q}(\sqrt{-3})$. $\sqrt{-3}$ has a minimal polynomial equal to $X^2 + 3$, which has two complex roots. The signature of K is therefore (0,1). $K = \mathbb{Q}(\sqrt{-3})$ is an example of a totally complex algebraic number field.

Definition 4.3.13. *Canonical Embedding*
Definition: Let us arrange the embeddings σ_i in a way that $\sigma_i(x) \in \mathbb{R}$ for $1 \leq i \leq r_1$, and $\sigma_{j+r_2}(x)$ is the complex conjugate of $\sigma_j(x)$ for $r_1 + 1 \leq j \leq r_1 + r_2$. We call canonical embedding $\sigma : K \to \mathbb{R}^{r_1} \times \mathbb{C}^{r_2}$ the homomorphism defined by

$$\sigma(x) = (\sigma_1(x), \cdots, \sigma_{r_1}(x), \sigma_{r_1+1}(x), \cdots, \sigma_{r_1+r_2}(x)) \in \mathbb{R}^{r_1} \times \mathbb{C}^{r_2}.$$

If we identify $\mathbb{R}^{r_1} \times \mathbb{C}^{r_2}$ with $\mathbb{R}^n$, the canonical embedding can be rewritten as $\sigma : K \to \mathbb{R}^n$

$$\sigma(x) = (\sigma_1(x), \cdots, \sigma_{r_1}(x), \mathcal{R}_e\sigma_{r_1+1}(x), \mathcal{I}_m\sigma_{r_1+1}(x), \cdots, \mathcal{R}_e\sigma_{r_1+r_2}(x), \mathcal{I}_m\sigma_{r_1+r_2}(x)).$$

where $\mathcal{R}_e$ and $\mathcal{I}_m$ indicate the real and imaginary part, respectively.

The canonical embedding is the means by which we will obtain the desired algebraic lattices. Looking closely at canonical embeddings, we see that they consist in mapping an element in an algebraic number field K of degree n to a vector in the n-dimensional Euclidean space. We now finally get to the final step of the algebraic construction of lattice with the following theorem:

Theorem 4.3.5. *Let K be an algebraic number field of degree n, with basis $\{1, \omega_1, \cdots, \omega_n\}$ and discriminant d_K. The n vectors $\sigma(\omega_i), i = 1, \cdots n$ are linearly independent, and thus they define a full rank lattice Λ whose generator matrix is given by:*

$$\mathbf{G} = \begin{pmatrix} \sigma_1(\omega_1) & \sigma_1(\omega_2) & \cdots & \sigma_1(\omega_n) \\ \vdots & \vdots & & \vdots \\ \sigma_{r_1}(\omega_1) & \sigma_{r_1}(\omega_2) & \cdots & \sigma_{r_1}(\omega_n) \\ \mathcal{R}_e\sigma_{r_1+1}(\omega_1) & \mathcal{R}_e\sigma_{r_1+1}(\omega_2) & \cdots & \mathcal{R}_e\sigma_{r_1+1}(\omega_n) \\ \mathcal{I}_m\sigma_{r_1+1}(\omega_1) & \mathcal{I}_m\sigma_{r_1+1}(\omega_2) & \cdots & \mathcal{I}_m\sigma_{r_1+1}(\omega_n) \\ \vdots & \vdots & & \vdots \\ \mathcal{R}_e\sigma_{r_1+r_2}(\omega_1) & \mathcal{R}_e\sigma_{r_1+r_2}(\omega_2) & \cdots & \mathcal{R}_e\sigma_{r_1+r_2}(\omega_n) \\ \mathcal{I}_m\sigma_{r_1+r_2}(\omega_1) & \mathcal{I}_m\sigma_{r_1+r_2}(\omega_2) & \cdots & \mathcal{I}_m\sigma_{r_1+r_2}(\omega_n) \end{pmatrix}.$$

The columns of $\mathbf{G}$ form the basis of Λ, whose volume is given by:

$$\mathrm{V}(\Lambda) = |\det(\mathbf{G})| = 2^{-r_2}\sqrt{|d_K|}.$$

Consequently,

$$\det(\Lambda) = 2^{-2r_2}|d_K|.$$

4.3.3 Totally real algebraic number fields

Let's first take a look at the relation between a lattice point $\mathbf{x} \in \Lambda$ and an element $x = \sum_{i=1}^{n} b_i\omega_i$, with $b_i \in \mathbb{Z}$, in the ring of integers $\mathcal{O}_K$. Any non-zero lattice point $\mathbf{x} = (x_1, x_2, \cdots, x_{r_1}, x_{r_1+1}, \cdots, x_{r_1+r_2})^T$ can be written as:

$$\mathbf{x} = \sum_{i=1}^{n} \mathbf{v_i} b_i$$

with $\mathbf{v_i} = \sigma(\omega_i)$ the basis vectors of Λ.
Each component of $\mathbf{x}$ can be written as: $x_j = \sum_{i=1}^{n} b_i \sigma_j(\omega_i)$ thus:

$$\begin{aligned}
\mathbf{x} &= (\sum_{i=1}^{n} b_i\sigma_1(\omega_i), \cdots, \sum_{i=1}^{n} b_i\mathcal{R}_e\sigma_{r_1+1}(\omega_i), \sum_{i=1}^{n} b_i\mathcal{I}_m\sigma_{r_1+1}(\omega_i), \cdots, \sum_{i=1}^{n} b_i\mathcal{I}_m\sigma_{r_1+r_2}(\omega_i) \\
&= (\sigma_1(\sum_{i=1}^{n} b_i\omega_i), \cdots, \sigma_{r_1}(\sum_{i=1}^{n} b_i\omega_i), \mathcal{R}_e\sigma_{r_1+1}(\sum_{i=1}^{n} b_i\omega_i), \mathcal{I}_m\sigma_{r_1+1}(\sum_{i=1}^{n} b_i\omega_i), \cdots, \mathcal{I}_m\sigma_{r_1+r_2}(\sum_{i=1}^{n} b_i\omega_i)) \\
&= (\sigma_1(x), \cdots, \mathcal{R}_e\sigma_{r_1+1}(x), \mathcal{I}_m\sigma_{r_1+1}(x), \cdots, \mathcal{I}_m\sigma_{r_1+r_2}(x)) \\
&= \sigma(x).
\end{aligned}$$

This correspondence between a lattice point $\mathbf{x}$ and an algebraic integer x in $\mathcal{O}_K$ will help us compute the diversity of algebraic lattices. If we take a look at the above equation, we see that the first r_1 components of $\mathbf{x}$ are nonzero, since $\mathbf{x} \neq 0$, which means that $b_i \neq 0 (\forall i = 1, \cdots, n)$, and thus $\sum_{i=1}^{n} b_i\omega_i \neq 0$. The minimum number of nonzero elements in the remaining $2r_2$ components is r_2 since the real and imaginary part of any component cannot be null together. This implies that the diversity of the algebraic lattice is $F \geq r_1 + r_2$. Now if we take the special case of the element $x = 1 \in \mathcal{O}_K$, we find exactly $r_1 + r_2$ nonzero components ($\sigma_j(1) = 1$ for any j). Hence, we can confirm the following theorem:

Theorem 4.3.6. *An algebraic lattice exhibits a diversity [22]*

$$F = r_1 + r_2.$$

In the case of a totally real algebraic number field ($r_2 = 0$), the maximum degree diversity is attained $F = r_1 = n$ and the lattice generator matrix becomes:

$$\mathbf{G} = \begin{pmatrix} \sigma_1(\omega_1) & \sigma_1(\omega_2) & \cdots & \sigma_1(\omega_n) \\ \sigma_2(\omega_1) & \sigma_2(\omega_2) & \cdots & \sigma_2(\omega_n) \\ & \vdots & & \\ \sigma_n(\omega_1) & \sigma_n(\omega_2) & \cdots & \sigma_n(\omega_n) \end{pmatrix}.$$

Since our main concern is to achieve the maximal diversity, lattices constructed from totally real algebraic number fields will be of prime interest for us in the remainder of the chapter. In this case, the product distance of an arbitrary lattice point $\mathbf{x}$ from $\mathbf{0}$ is:

$$d_p^{(n)}(\mathbf{0}, \mathbf{x}) = \prod_{x_j \neq 1} |x_j| = \prod_{x_j \neq 0} |\sigma_j(x)| = |N(x)|.$$

Since $x = \sum_{i=1}^{n} b_i\omega_i$ is a non-zero element of $\mathcal{O}_K$, $N(x) \neq 0$ and thus $d_p^{(n)}(\mathbf{0}, \mathbf{x}) \geq 0$.
The minimal product distance of the algebraic lattice is $d_{p,min} = 1$, resulted from the elements of $\mathcal{O}_K$ whose norm is equal to 1, called the *units* of $\mathcal{O}_K$.

An example of totally real algebraic number fields are $\mathbb{Q}(\sqrt{2})$ and $\mathbb{Q}(\sqrt{5})$ seen previously. Let us now show the algebraic lattice construction of a totally complex number field, and for that we will reconsider the example of $K = \mathbb{Q}(i\sqrt{3})$. We have the integral basis $(1, 1 + i\frac{\sqrt{3}}{2})$, and the two embeddings are:

$$\sigma_1(i\sqrt{3}) = i\sqrt{3} \text{ and } \sigma_2(i\sqrt{3}) = -i\sqrt{3}.$$

The lattice generator matrix is then:

$$\mathbf{G} = \begin{pmatrix} \mathcal{R}_e\sigma_1(1) & \mathcal{R}_e\sigma_1(1 + i\frac{\sqrt{3}}{2}) \\ \mathcal{I}_m\sigma_2(1) & \mathcal{I}_m\sigma_2(1 + i\frac{\sqrt{3}}{2}) \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} \end{pmatrix}.$$

The fundamental volume is $|\det(\mathbf{G})| = \frac{\sqrt{3}}{2}$, and the diversity is $F = r_2 = 1$. Note that this example represents the hexagonal lattice A_2.

To summarize, we have seen that the key ingredient to build a lattice from an algebraic number field $K = \mathbb{Q}(\theta)$ is the existence of a $\mathbb{Z}$-basis in $\mathcal{O}_K$, the lattice construction is then carried out according to the following steps:

1. Choose a number field $K = \mathbb{Q}(\theta)$ of degree n and find its integral basis, which identifies its ring of integers $\mathcal{O}_K$.
2. Compute the n embeddings $\sigma_1, \cdots, \sigma_n$ from the roots of the minimal polynomial M_θ.
3. Compute the lattice generator matrix using the canonical embeddings.

In the remainder of the chapter, we will restrict ourselves to totally real algebraic number fields in order to ensure maximum diversity.

As seen previously, a multilevel construction is based on a lattice partition chain $\Lambda_0/\Lambda_1/\cdots/\Lambda_r$ where each partition Λ_i/Λ_{i+1} is induced by the sublattice Λ_{i+1} of Λ_i. We have just learned how, on a Rayleigh fading channel, we can proceed to build the first algebraic lattice of the lattice partition chain using the ring of integers $\mathcal{O}_K$. In order to go ahead in our multilevel construction, we need to take a subset of $\mathcal{O}_K$ that is also a free $\mathbb{Z}$-module of rank n and use it to build the corresponding algebraic lattice (which naturally will be a sublattice of the former lattice). This subset is what is known as the *ideal* of $\mathcal{O}_K$, and the corresponding algebraic lattice is called *ideal lattice*.

Hence, the MLC scheme on a Rayleigh fading channel requires going a little bit further in algebraic number theory so as to get a better understanding of ideals and their related characteristics. This will constitute the objective of the next paragraph.

4.3.4 Ideal lattices

In the sequel, K is a number field of rank n, and $\mathcal{O}_K$ is its ring of integers.

Definition 4.3.14. *Ideal*
An ideal $\mathcal{I}$ of $\mathcal{O}_K$ is a subset of $\mathcal{O}_K$ closed under addition, such that for any $x \in \mathcal{O}_K$, $x\mathcal{I} \subseteq \mathcal{I}$. We say that $\mathcal{I}$ is stable under multiplication.

Among the ideals of a ring, some can be generated by one algebraic integer $\alpha \in \mathcal{O}_K$, and we can write $\mathcal{I} = \alpha\mathcal{O}_K$. An ideal obtained this way is denoted by $\mathcal{I} = (\alpha)$ and it is called a *principal ideal*. This category of ideals will be of particular importance for the remainder of the chapter.

In order to build the algebraic lattice Λ' from $\mathcal{I}$, we need to have the integral basis of $\mathcal{I}$. We have ω_i for $i = 1, \cdots, n$ the integral basis of $\mathcal{O}_K$. If x is a non-zero element of $\mathcal{I}$, we can write $x\mathcal{O}_K \subset \mathcal{I} \subset \mathcal{O}_K$, which means that $\mathcal{I}$ is included in a set of rank n, and at the same time includes a set of rank n. Thus, $\mathcal{I}$ is a set of rank n, and therefore endowed with a $\mathbb{Z}$-basis γ_i for $i = 1, \cdots, n$. We can write $\mathcal{I} = \gamma_1\mathbb{Z} + \gamma_2\mathbb{Z} + \cdots + \gamma_n\mathbb{Z}$.

In Theorem 4.3.5, we can replace the integral basis of $\mathcal{O}_K$ by that of $\mathcal{I}$ so we can obtain the algebraic lattice Λ' after applying the canonical embedding σ to $\mathcal{I}$. $\Lambda' = \sigma(\mathcal{I})$ is a sublattice of the algebraic lattice $\Lambda = \sigma(\mathcal{O}_K)$, and its generator matrix is written as:

$$\mathbf{G}' = \begin{pmatrix} \sigma_1(\gamma_1) & \sigma_1(\gamma_2) & \cdots & \sigma_1(\gamma_n) \\ \vdots & \vdots & & \vdots \\ \sigma_{r_1}(\gamma_1) & \sigma_{r_1}(\gamma_2) & \cdots & \sigma_{r_1}(\gamma_n) \\ \mathcal{R}_e\sigma_{r_1+1}(\gamma_1) & \mathcal{R}_e\sigma_{r_1+1}(\gamma_2) & \cdots & \mathcal{R}_e\sigma_{r_1+1}(\gamma_n) \\ \mathcal{I}_m\sigma_{r_1+1}(\gamma_1) & \mathcal{I}_m\sigma_{r_1+1}(\gamma_2) & \cdots & \mathcal{I}_m\sigma_{r_1+1}(\gamma_n) \\ \vdots & \vdots & & \vdots \\ \mathcal{R}_e\sigma_{r_1+r_2}(\gamma_1) & \mathcal{R}_e\sigma_{r_1+r_2}(\gamma_2) & \cdots & \mathcal{R}_e\sigma_{r_1+r_2}(\gamma_n) \\ \mathcal{I}_m\sigma_{r_1+r_2}(\gamma_1) & \mathcal{I}_m\sigma_{r_1+r_2}(\gamma_2) & \cdots & \mathcal{I}_m\sigma_{r_1+r_2}(\gamma_n) \end{pmatrix}.$$

The generator matrix $\mathbf{G}'$ of Λ' can certainly be obtained from the generator matrix $\mathbf{G}$ of Λ. In the case where $\mathcal{I}$ is a principal ideal, we have $\gamma_i = \{\alpha\omega_i\}$ for $i = 1, \cdots, n$.

The norm and determinant of $\mathcal{I} = \alpha\mathcal{O}_K$ can be easily calculated. In this case, the norm of $\mathcal{I}$ is equal to the absolute value of the algebraic norm of its generating element: $N(\mathcal{I}) = |N(\alpha)|$ and the volume is:

$$\text{vol}(\Lambda') = |\det(\mathbf{G}')| = N(\mathcal{I})\text{vol}(\Lambda) = 2^{-r_2}N(\mathcal{I})\sqrt{|d_K|}.$$

So far, we have offered an extensive study on the construction of algebraic lattices. However, a problem that we haven't considered yet is the very important lattice shap-

ing. In fact, the lattices were built without imposing any shaping conditions, which leads to a loss in the system's performance as seen in Chapter 2. We already know that opting for a hypersphere shaping is not an easy task, that's why a good and satisfying trade-off would be a cubic shaping which provides a good shaping gain along with an affordable complexity. The idea is therefore to stretch the algebraic lattice into another, a $\mathbb{Z}^n$ lattice for instance. This construction, explained in [21] [14], will be described in what follows using the notion of *ideal lattices*.

An ideal lattice is a lattice whose generator matrix is written as: $\mathbf{M} = \mathbf{AG'}$, where $\mathbf{A}$ is a diagonal matrix used to turn the lattice Λ into a rotated $\mathbb{Z}^n$ lattice. The elements of this matrix are functions of a totally positive algebraic number α, i.e., $\alpha \in K$ such that α and its conjugates $\sigma_i(\alpha)$ are all positive. In order to obtain $\mathbf{M}$, we introduce the notion of a *twisted canonical embedding* $\sigma_\alpha : K \to \mathbb{R}^n$ as:

$$\sigma_\alpha(x) = (\sqrt{\alpha_1}\sigma_1(x), \sqrt{\alpha_2}\sigma_2(x), \cdots, \sqrt{\alpha_n}\sigma_n(x)).$$

where $\alpha_i = \sigma_i(\alpha)$ for $i = 1, \cdots, n$. The generator matrix $\mathbf{M}$ of the lattice $\Lambda_\mathcal{I} = \sigma_\alpha(\mathcal{I})$ is written as:

$$\begin{aligned}\mathbf{M} &= \begin{pmatrix} \sqrt{\alpha_1}\sigma_1(\gamma_1) & \cdots & \sqrt{\alpha_1}\sigma_1(\gamma_n) \\ \sqrt{\alpha_2}\sigma_2(\gamma_1) & \cdots & \sqrt{\alpha_2}\sigma_2(\gamma_1) \\ \vdots & \vdots & \vdots \\ \sqrt{\alpha_n}\sigma_n(\gamma_1) & \cdots & \sqrt{\alpha_n}\sigma_n(\gamma_n) \end{pmatrix} \\ &= \underbrace{\begin{pmatrix} \sqrt{\alpha_1} & & 0 \\ & \ddots & \\ 0 & & \sqrt{\alpha_n} \end{pmatrix}}_{\mathbf{A}} \mathbf{G'}.\end{aligned}$$

and the determinant of the lattice is:

$$\det(\Lambda_\mathcal{I}) = N(\alpha)N(\mathcal{I})^2|d_K| \tag{4.3}$$

The elements of the corresponding Gram matrix $\mathbf{G_r} = \mathbf{M}^T\mathbf{M}$ can be easily computed:

$$\begin{aligned} g_{rij} &= \sum_{k=1}^{n} \sqrt{\alpha_k}\sigma_k(\gamma_i)\sqrt{\alpha_k}\sigma_k(\gamma_j) \\ &= \sum_{k=1}^{n} \alpha_k\sigma_k(\gamma_i\gamma_j) \\ &= \mathrm{Tr}(\alpha\gamma_i\gamma_j). \end{aligned}$$

An ideal lattice is uniquely identified by this trace form of the Gram matrix .

The objective now is to build an ideal lattice $\Lambda_\mathcal{I}$ that is equivalent to a rotated $\mathbb{Z}^n$ lattice, for $n \geq 2$. The corresponding Gram matrix is therefore the identity matrix I_n,

and the generator matrix $\mathbf{M}$ is such that $\mathbf{M}^T\mathbf{M} = \mathbf{I}_n$, i.e., $\mathbf{M}$ is an orthogonal matrix. We know that the determinant of $\mathbb{Z}^n$ is 1, and thus if the Gram matrix is a scaled version of $\mathbb{Z}^n$, its determinant will be equal to $\det(\mathbf{G_r}) = c^n$, with c an integer. This allows us to write, using Equation (4.3.4):

$$N(\alpha)N(\mathcal{I})^2|d_K| = c^n. \tag{4.4}$$

The problem consists in doing the following: for a number field K of degree n and an ideal $\mathcal{I} \subseteq \mathcal{O}_K$, choose the value of α that satisfies condition (4.4). The next section will explain how to do that.

4.3.5 Construction of $\mathbb{Z}^n$ lattices

Let us take the simple case of building a 2-dimensional $\mathbb{Z}$ lattice, and for that we take a totally real algebraic number field $K = \mathbb{Q}(\sqrt{d})$ with d a square-free positive number. For $\mathcal{I} = \mathcal{O}_K$, Equation (4.4) becomes:

$$N(\alpha)|d_K| = c^2.$$

The problem is therefore to look for a totally positive number α such that $N(\alpha) = \frac{c^2}{|d_K|}$. This is solved using the following lemma.

Lemma 1. Let m be an algebraic norm in K. If we can find a unit u such that $N(u) = -1$, then we can also find an algebraic number α such that $N(\alpha) = m$ and $\sigma_i(\alpha) > 0 \ \forall \ i$.

Proof. Let β be an algebraic number of given norm m. Four different scenarios might occur:
If $\sigma_1(\beta) > 0$ and $\sigma_2(\beta) > 0$, all we have to do is take $\alpha = \beta$.
If $\sigma_1(\beta) > 0$ and $\sigma_2(\beta) < 0$, we take $\alpha = \beta.u$.
If $\sigma_1(\beta) < 0$ and $\sigma_2(\beta) > 0$, we take $\alpha = -\beta.u$.
And finally, If $\sigma_1(\beta) < 0$ and $\sigma_2(\beta) < 0$, we take $\alpha = -\beta$.

Example 4.3.1. Let us consider the field $K = \mathbb{Q}(\sqrt{5})$, whose discriminant is $d_K = 5$. In order to obtain $\mathbb{Z}^2$, we should find the element α that satisfies

$$N(\alpha)d_K = N(\alpha).5 = c^2 \qquad \text{with} \qquad c \in \mathbb{Z}.$$

The natural solution is to look for α such that $\mathbb{N}(\alpha) = 5$. We know that $\beta = \sqrt{5}$ has norm of 5, but it is not a totally algebraic number since $\sigma_2(\sqrt{5}) < 0$. On the other hand, we have $u = -1 + \frac{1+\sqrt{5}}{2}$ with $N(u) = -1$. So according to Lemma 1, we can take $\alpha = \sqrt{5}.u = 3 - \frac{1+\sqrt{5}}{2}$, and easily check that:

$$\sigma_1(\alpha) = 3 - \frac{1+\sqrt{5}}{2} > 0 \qquad \sigma_2(\alpha) = 3 - \frac{1-\sqrt{5}}{2} > 0$$

$$N(\alpha) = \sigma_1(\alpha)\sigma_2(\alpha) = 5 \qquad \det(\mathbf{G_r}) = \det(\mathbf{M^T M}) = 5^2.$$

With the overview on algebraic number theory provided in this section, the various visited notions can now be collected to begin the description of the MLC/MSD scheme on the Rayleigh block fading channel.

4.4 Multilevel Construction using binary Reed-Muller codes

We assume a block fading channel, i.e., a channel whose gain remains constant over a block of length N and changes for different block lengths based on a Rayleigh distribution. We also assume perfect channel state information (CSI) at the receiver. The received observation $\mathbf{y}$ is given by:

$$\mathbf{y} = \mathbf{Hx} + \mathbf{w}. \tag{4.5}$$

where $\mathbf{H} = \text{diag}(h_i)$ for $i = 1, \cdots, n$ is the diagonal matrix of n independent positive real-valued fading coefficients h_i following a Rayleigh distribution, $\mathbf{x} \in L$ is the transmitted lattice point and $\mathbf{w}$ is the zero-mean noise vector of variance σ^2 per dimension. The matrix form of Equation 4.5 is then:

$$\begin{pmatrix} y_{11} & \cdots & y_{1N} \\ & \vdots & \\ y_{n1} & \cdots & y_{nN} \end{pmatrix} = \begin{pmatrix} h_1 & & 0 \\ & \ddots & \\ 0 & & h_n \end{pmatrix} \times \begin{pmatrix} x_{11} & \cdots & x_{1N} \\ & \vdots & \\ x_{n1} & \cdots & x_{nN} \end{pmatrix} + \begin{pmatrix} w_{11} & \cdots & w_{1N} \\ & \vdots & \\ w_{n1} & \cdots & w_{nN} \end{pmatrix}.$$

The lattice L is built via multilevel construction D, which, as explained in Chapter 3, is induced by a lattice partition chain $\Lambda_1/\cdots/\Lambda_r$. On the Gaussian channel, the top lattice Λ_1 for partition chains of different dimensions n was chosen to be a version of $\mathbb{Z}^n$. On the Rayleigh fading channel, the multilevel construction must be carried out on algebraic lattices that provide maximum diversity n. For this reason, the top lattice is the algebraic lattice built from a totally real algebraic number field $K = \mathbb{Q}(\theta)$ of dimension n.

Once K is defined, the top lattice will be generated by the ring of integers $\mathcal{O}_K = \mathbb{Z}[\theta]$, the generator matrix being noted by $\mathbf{G}_n$. $\mathcal{O}_K$ is later partitioned into

subsets, and the result is a principal ideal $\mathcal{I}^1 = \theta\mathcal{O}_K$, which is also partitioned into subsets in order to give the ideal $\mathcal{I}^2 = \theta\mathcal{I}^1 = \theta^2\mathcal{O}_K$, and so on until reaching the last lattice of the partition chain. The latter is then written as:

$$\mathcal{O}_K/\mathcal{I}^1/\mathcal{I}^2/\cdots/\mathcal{I}^r.$$

The choice of K, and thus $\mathcal{O}_K$, is a key point in the multilevel construction on fading channels. However, building the algebraic lattice generated by $\mathcal{O}_K$ (and the principal ideals forming the partition chain) does not suffice to build efficient transmission schemes on this type of channels, since problems of shaping and finding the orthogonal basis must also be addressed. To this end, the generator matrix $\mathbf{G}_n$ is carefully adjusted until the desired lattice is obtained, where the desired lattice is a rotated version of $\mathbb{Z}^n$. Adjusting $\mathbf{G}_n$ includes multiplying by the appropriate rotation and base reduction matrices.

In summary, the final lattice L is obtained by the following step-by-step construction:

1. Choose the number field $K = \mathbb{Q}(\theta)$.
2. Write the lattice generator matrix $\mathbf{G}_n$.
3. Find an element α such that the matrix $\mathbf{M} = \mathbf{A} \times \mathbf{G}_n$ corresponds to a lattice admitting a cubic shaping. The matrix $\mathbf{A}$ is the diagonal matrix whose elements are $\sqrt{\sigma_i(\alpha)}$ for $i = 1, \cdots, n$.
4. Find the base reduction matrix $\mathbf{U}_n$ that leads to a lattice generated by an orthogonal basis. The new Gram matrix must be of the form: $(\mathbf{AG}_n\mathbf{U}_n)^T\mathbf{AG}_n\mathbf{U}_n = c\mathbf{I}_n$.
5. The final determinant is normalized to 1, and the rotated $\mathbb{Z}^n$ generator matrix is:

$$\mathbf{O}_n = \frac{1}{\sqrt{c}}\mathbf{AG}_n\mathbf{U}_n.$$

Finding the matrix $\mathbf{O}_n$ sets the ground for a multilevel construction computed by mimicking the Gaussian case of Chapter 3. In fact, on the Rayleigh block fading channel, the construction is carried out on the lattice partition chain $\mathbf{O}_n\Lambda_1/\mathbf{O}_n\Lambda_2/\cdots/\mathbf{O}_n\Lambda_r$, where $\Lambda_1/\cdots/\Lambda_r$ is the same binary partition chain employed for the AWGN channel. Hence, the capacity curves depicted in Section 3.4 are still valid, and the component code rates can be taken equal to those used on the Gaussian channel.

At the receiver side, a list sphere decoder is performed on the received vector $\mathbf{y}$ inside the lattice generated by $\mathbf{HO}_n\mathbb{Z}^n$, so as to output the list of closest lattice points,

in terms of Euclidean distance, to the transmitted vector $\mathbf{x}$. This list is then employed in the LLR calculation as in Equation 3.15. At each level i the soft-information of each component of the received vector $\mathbf{y}^{i-1}$ is written as:

$$\mathrm{LLR} = \ln\left(\frac{\Pr(y^{(i-1)}|c_i=0)}{\Pr(y^{(i-1)}|c_i=1)}\right) = \ln\left(\frac{\sum\limits_{\mathbf{c}\in\mathbf{HO_n}\Lambda_{i+1}} \mathrm{e}^{\frac{-||\mathbf{y^{(i-1)}}-\mathbf{c}||^2}{2\sigma^2}}}{\sum\limits_{\mathbf{c'}\in\mathbf{HO_n}\Lambda_{i+1}+\mathbf{a_i}} \mathrm{e}^{\frac{-||\mathbf{y^{(i-1)}}-\mathbf{c'}||^2}{2\sigma^2}}}\right).$$

Outage probability For slow-varying fading channels, we define the notion of channel *outage capacity* where outage is the event that occurs when the channel is poor due to deep fades. In that case, no scheme can communicate reliably above a certain rate, leading to a loss in transmitted data. A new parameter is introduced called *outage probability* and denoted by P_{out}, which is the probability that the system can be in outage. Let us derive the formula giving P_{out} for a system of diversity $F = n$.

In order to define the outage probability, we can follow the sphere-bound rule employed by Forney *et al.* in [42] for the AWGN channel, where he states that in order for the error probability to be small, the lattice VNR must satisfy the condition:

$$\mathrm{VNR} > 1 \Rightarrow \sigma^2 < \frac{V(L)^{2/nN}}{2\pi\mathrm{e}}. \tag{4.6}$$

The error probability is large otherwise.

On the Rayleigh fading channel, the total lattice volume is altered by the fading coefficients $\{h_i\}_{i=1}^n$, resulting in $\mathrm{V}(L)_{fading}$ written as:

$$\mathrm{V}(L)_{fading} = (\prod_{i=1}^{n} h_i^N) \times \mathrm{V}(L).$$

The VNR for a noise variance per dimension σ^2 becomes:

$$\begin{aligned}\mathrm{VNR}_{fading} &= \frac{[\mathrm{V}(L)_{fading}]^{2/nN}}{2\pi\mathrm{e}\sigma^2} \\ &= \prod_{i=1}^{n} h_i^{2/n} \times \mathrm{VNR}.\end{aligned}$$

Following the same condition in (4.6), the outage probability can be defined by:

$$P_{out} = \Pr(\mathrm{VNR}_{fading} < 1) = \Pr(\prod_{i=1}^{n} h_i^{2/n} < \frac{1}{\mathrm{VNR}}). \tag{4.7}$$

In this section, we detail the different steps of multilevel lattice construction over two and four-dimensional lattice partition chains. Simulation results for each case are compared to the channel outage probability.

4.4.1 Two-dimensional lattice partition chain

All totally real algebraic extensions provide maximum diversity, but for $n = 2$, $K = \mathbb{Q}(\sqrt{5})$ also has the minimal absolute discriminant [22], in other words, it gives the rotated constellation having the minimal energy. However, $K = \mathbb{Q}(\sqrt{5})$ does not lead to binary partitions [75], thus this number field requires the use of non-binary codes. Since we wish to restrict our construction to binary Reed-Muller codes, we choose the number field having the second minimal absolute discriminant for $n = 2$, which is $K = \mathbb{Q}(\sqrt{2})$.

Once we choose the number field and its corresponding ring of integers, the next step towards the construction of the algebraic lattice is to find the integral basis. In Section 4.3.2, it was shown that $\mathcal{O}_K = \mathbb{Z}[\sqrt{2}] = \{a + b\sqrt{2},\ a, b \in \mathbb{Z}\}$ and therefore $\mathbb{Z}[\sqrt{2}]$ has a basis over $\mathbb{Z}$ equal to $B = \{1, \sqrt{2}\}$. Applying the canonical embeddings σ_1, σ_2 to B, we get the lattice generator matrix:

$$\mathbf{G}_2 = \begin{pmatrix} 1 & \sqrt{2} \\ 1 & -\sqrt{2} \end{pmatrix}$$

whose discriminant is $d_K = \det(\mathbf{G}_2^T \mathbf{G}_2) = 8 \neq c^2$, with c an integer. Thus, the lattice Λ_g whose generator matrix is $\mathbf{G}_2$ does not correspond to a scaled version of $\mathbb{Z}^2$.

Going back to Equation (4.4), the necessary condition for obtaining a scaled version of $\mathbb{Z}^2$ is to find an element α such that:

$$N(\alpha).d_K = N(\alpha).2^3 = c^2.$$

This leads the search for a totally real algebraic number α whose norm is equal to 2. Following lemma 1, we have the unit $u = 1 + \sqrt{2} \in K$ with

$$N(u) = \sigma_1(u)\sigma_2(u) = (1 + \sqrt{2})(1 - \sqrt{2}) = -1.$$

We also know that $\beta = \sqrt{2}$ is such that $N(\beta) = 2$ and $\sigma_1(\beta) = \sqrt{2} > 0, \sigma_2(\beta) = -\sqrt{2} < 0$. Thus, the value we are seeking is: $\alpha = \beta.u = \sqrt{2} + 2$. The new lattice is generated by:

$$\begin{aligned} \mathbf{M} = \mathbf{A}\mathbf{G}_2 &= \begin{pmatrix} \sqrt{\sigma_1(\alpha)} & 0 \\ 0 & \sqrt{\sigma_2(\alpha)} \end{pmatrix} \begin{pmatrix} 1 & \sqrt{2} \\ 1 & -\sqrt{2} \end{pmatrix} \\ &= \begin{pmatrix} \sqrt{2+\sqrt{2}} & 0 \\ 0 & \sqrt{2-\sqrt{2}} \end{pmatrix} \begin{pmatrix} 1 & \sqrt{2} \\ 1 & -\sqrt{2} \end{pmatrix}. \end{aligned}$$

And the new Gram matrix is:

$$\mathbf{G_r} = \mathbf{M}^T\mathbf{M} = \begin{pmatrix} 4 & 4 \\ 4 & 8 \end{pmatrix}$$

which is still not the Gram matrix of a scaled version of $\mathbb{Z}^2$. In order to have the appropriate Gram matrix, a base reduction must be performed. This operation is computed using a mathematics software including number theoretical features. Appendix C contains the commands in "Sage" that lead to the reduction matrix $\mathbf{U}_2$:

$$\mathbf{U}_2 = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}.$$

It is easy to verify that:

$$(\mathbf{AG}_2\mathbf{U}_2)^T(\mathbf{AGU}) = \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix} = \mathbf{G_r}(2\mathbb{Z}^2).$$

Finally, the determinant is normalized to 1, so as to be equivalent to that of a rotated $\mathbb{Z}^2$ lattice. To this aim, the new matrix is divided by a factor of 2, and the final lattice generator matrix is:

$$\mathbf{O}_2 = 0.5 \times \mathbf{AG}_2\mathbf{U}_2.$$

with

$$\mathbf{O}_2^T\mathbf{O}_2 = \mathbf{I}_2.$$

The final matrix $\mathbf{O}_2$ is the key ingredient for the multilevel construction on the Rayleigh block fading channel. The lattice partition chain $\mathbb{Z}[\sqrt{2}]/\mathcal{I}^1/\mathcal{I}^2/\mathcal{I}^3/\mathcal{I}^4$ is now transformed into $\mathbf{O}_2\mathbb{Z}^2/\mathbf{O}_2R\mathbb{Z}^2/\mathbf{O}_2 2\mathbb{Z}^2/\mathbf{O}_2 2R\mathbb{Z}^2/\mathbf{O}_2 4\mathbb{Z}^2$. The coset representatives for each partition are those of the binary lattice partition chain $\mathbb{Z}^2/R\mathbb{Z}^2/2\mathbb{Z}^2 2/R\mathbb{Z}^2/4\mathbb{Z}^2$ (see Section 3.4.2) multiplied by the matrix $\mathbf{O}_2$.

One can easily verify that:

$$(\mathbf{O}_2R)^T(\mathbf{O}_2R) = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} = \mathbf{G_r}(R\mathbb{Z}^2)$$

Similarly, $(\mathbf{O}_2 2)^T(\mathbf{O}_2 2) = \mathbf{G_r}(2\mathbb{Z}^2)$, $(\mathbf{O}_2 2R)^T(\mathbf{O}_2 2R) = \mathbf{G_r}(2R\mathbb{Z}^2)$, and so on.

Simulation results are shown in Figure 4.2, where the word error rate is plotted as a function of the VNR for lattices using codes of lengths $N = 256, 512$ and 1024. The WER are also compared to the channel outage probability defined as:

$$P_{out} = \Pr(h_1 h_2 < \frac{1}{\text{VNR}}).$$

The component codes are binary Reed-Muller codes, where each code's rate was chosen according to the same capacity rule explained in Section 3.4. The lattice L

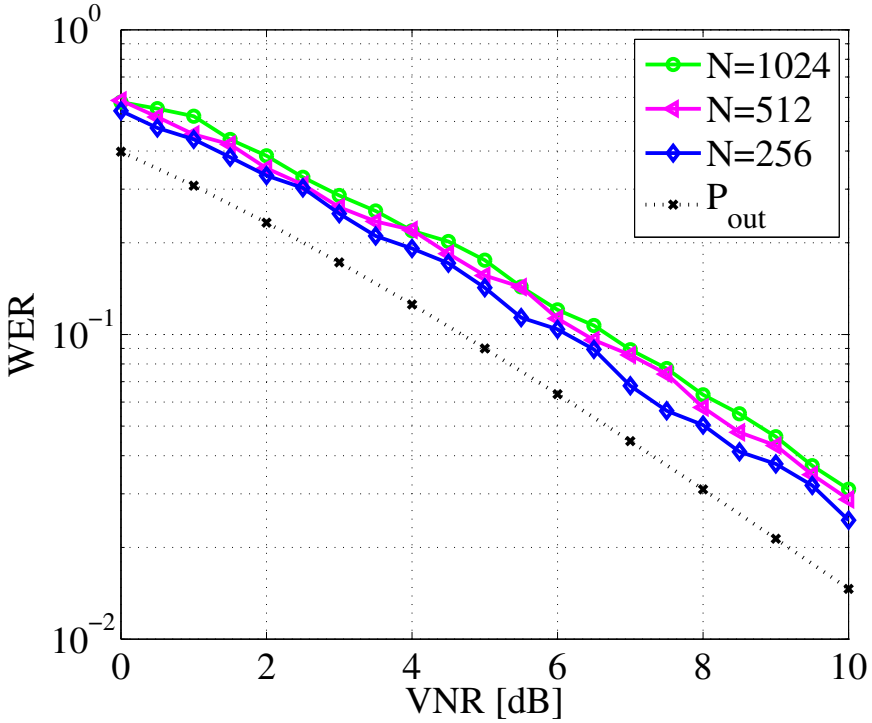


Figure 4.2 – The multilevel design performance over the Rayleigh block fading channel for $n = 2$.

is the result of a four-level construction. Simulation results show that the WER for the different word lengths is almost the same, in other words, increasing the lattice dimension does not degrade the error rate. For WER=10^{-1}, the gap to the channel outage probability is about 0.6 dB.

In Figure 4.3, we show the performance of also three different lattices in which codes of lengths $N = 256, 512$ and 1024 are employed. For $N = 256$, the lattice is the same as in Figure 4.2. However, for $N = 512$ and 1024, the component codes assigned to each of the four levels do not have rates that respect the capacity rule. Figure 4.3 first shows that violating the capacity rule degrades the global lattice performance. This can clearly be noticed by making a comparison with the curves corresponding to $N = 512$ and 1024 in Figure 4.2. Moreover, the different word error rates are now clearly spaced; the error rate increases with the code length N. For WER $= 10^{-1}$, 2.4 dB separate a lattice L obtained with $N = 256$ from that obtained with codes of length $N = 1024$, against 0.3 dB in Figure 4.2.

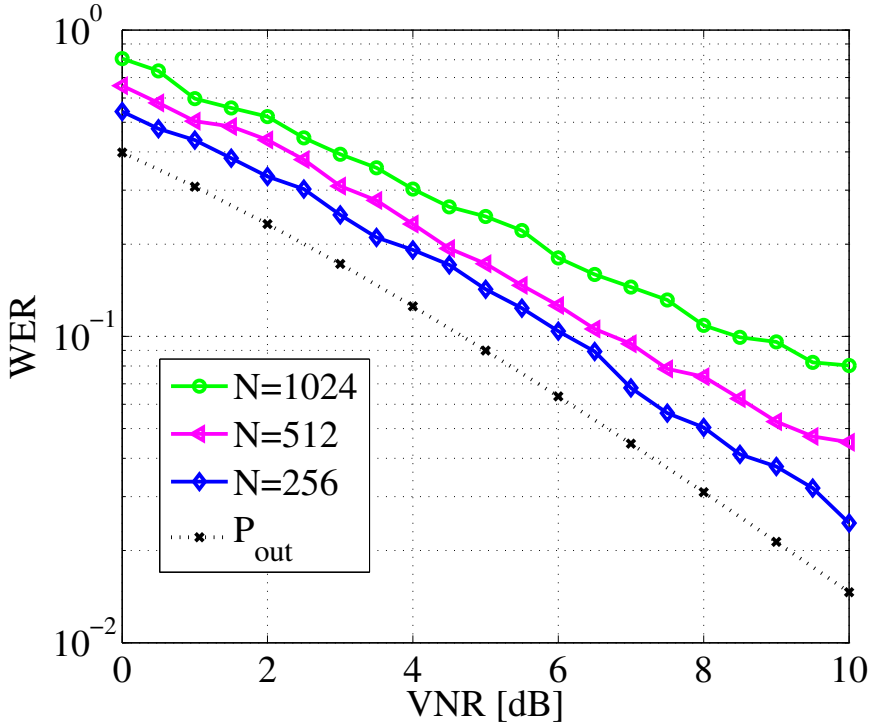


Figure 4.3 – The multilevel design performance over the Rayleigh block fading channel for $n = 2$ without respecting the capacity rule.

4.4.2 Four-dimensional lattice partition chain

For the same reason explained in the two-dimensional case, we choose to work in the algebraic number field $K = \mathbb{Q}(\theta)$ with $\theta = \sqrt{2+\sqrt{2}}$. In that case, the basis of K is $B = \{1, \theta, \theta^2, \theta^3\}$, and the minimal polynomial of θ of degree n is:

$$M_\theta = X^4 - 4X^2 + 2.$$

The conjugates of θ, which are the roots of M_θ, are also the canonical embeddings $\sigma_i(\theta)$ for $i = 1, \cdots, 4$. Thus we have:

$$\sigma_1(\theta) = \sqrt{2+\sqrt{2}} = \theta \qquad \sigma_2(\theta) = \sqrt{2-\sqrt{2}} = \theta'$$

$$\sigma_3(\theta) = -\sqrt{2+\sqrt{2}} = -\theta \qquad \sigma_4(\theta) = -\sqrt{2-\sqrt{2}} = -\theta'.$$

The algebraic lattice is generated by:

$$\mathbf{G}_4 = \begin{pmatrix} 1 & \sigma_1(\theta) & \sigma_1(\theta^2) & \sigma_1(\theta^3) \\ 1 & \sigma_2(\theta) & \sigma_2(\theta^2) & \sigma_2(\theta^3) \\ 1 & \sigma_3(\theta) & \sigma_3(\theta^2) & \sigma_3(\theta^3) \\ 1 & \sigma_4(\theta) & \sigma_4(\theta^2) & \sigma_4(\theta^3) \end{pmatrix}.$$

where $\sigma(\theta^i) = (\sigma(\theta))^i$. $\mathbf{G}_4$ is then equal to:

$$\mathbf{G}_4 = \begin{pmatrix} 1 & \theta & \theta^2 & \theta^3 \\ 1 & \theta' & \theta'^2 & \theta'^3 \\ 1 & -\theta & (-\theta)^2 & (-\theta)^3 \\ 1 & -\theta' & (-\theta')^2 & (-\theta')^3 \end{pmatrix}.$$

The discriminant of K is $d_K = \det(\mathbf{G}_4^T \mathbf{G}_4) = 2^{11} \neq c^4$, with c an integer. The next step is therefore to find the totally real algebraic number α such that $N(\alpha) = 2$, so as to obtain $d_K = 2^{12} = 8^4$. However, for $n = 4$, finding α does not come as naturally as for only 2 dimensions. That's where the software Sage comes in handy and gives the exact value of α. The command lines in Appendix C explain how the software can pick the right combination of units $\in K$, and at the end output α that can be written in function of θ as follows:

$$\alpha = 2\theta^3 + 4\theta^2 - \theta - 2.$$

The new lattice generator matrix is:

$$\mathbf{M} = \mathbf{A}\mathbf{G}_4 = \begin{pmatrix} \sqrt{\sigma_1(\alpha)} & 0 & 0 & 0 \\ 0 & \sqrt{\sigma_2(\alpha)} & 0 & 0 \\ 0 & 0 & \sqrt{\sigma_3(\alpha)} & 0 \\ 0 & 0 & 0 & \sqrt{\sigma_4(\alpha)} \end{pmatrix} \begin{pmatrix} 1 & \theta & \theta^2 & \theta^3 \\ 1 & \theta' & \theta'^2 & \theta'^3 \\ 1 & -\theta & \theta^2 & -\theta^3 \\ 1 & -\theta' & \theta'^2 & -\theta'^3 \end{pmatrix}.$$

and the Gram matrix:

$$\mathbf{G_r} = \mathbf{M}^T\mathbf{M} = \begin{pmatrix} 24 & 40 & 80 & 136 \\ 40 & 80 & 136 & 272 \\ 80 & 136 & 272 & 464 \\ 136 & 272 & 464 & 928 \end{pmatrix}.$$

shows that a base reduction must be performed. According to Sage (Appendix C, in order for the lattice to be generated by an orthogonal basis, $\mathbf{M}$ must be multiplied by the matrix $\mathbf{U}_4$ equal to:

$$\mathbf{U}_4 = \begin{pmatrix} -3 & 1 & -1 & -3 \\ 0 & -4 & -3 & 3 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & -1 \end{pmatrix}.$$

We can now verify that:

$$(\mathbf{A}\mathbf{G}_4\mathbf{U}_4)^T(\mathbf{A}\mathbf{G}_4\mathbf{U}_4) = 8\mathbf{I}_4 = \mathbf{G_r}(2\sqrt{2}\mathbb{Z}^4).$$

Normalizing by $\frac{1}{2\sqrt{2}}$:

$$\mathbf{O}_4 = \frac{1}{2\sqrt{2}}\mathbf{A}\mathbf{G}_4\mathbf{U}_4.$$

With the matrix $\mathbf{O}_4$ computed, the final lattice is indeed a rotated version of $\mathbb{Z}^4$. The multilevel construction can hereafter be performed over the binary lattice partition chain $\mathbf{O}_4\mathbb{Z}^4/\mathbf{O}_4D_4/\mathbf{O}_4R\mathbb{Z}^4/\mathbf{O}_4RD_4/\mathbf{O}_42\mathbb{Z}^4/\mathbf{O}_42D_4$. The coset representatives are equal to those of $\mathbb{Z}^4/D_4/R\mathbb{Z}^4/RD_4/2\mathbb{Z}^4/2D_4$ multiplied by $\mathbf{O}_4$.

Using the capacity curves shown in Section 3.4.3, we proceed with the 5-level multilevel construction using codes of lengths $N = 256, 512$ and 1024. Simulation results are shown in Figure 4.4 where the WER is also plotted as the function of the VNR and compared to the channel outage probability written as:

$$P_{out} = \Pr((h_1h_2h_3h_4)^{1/2} < \frac{1}{\text{VNR}}).$$

The same observation made earlier applies to the four-dimensional lattice, i.e., increasing the codeword length barely degrades the system's performance. In fact, for WER=10^{-1}, the outage probability is at 1.6 dB, 1.7 dB and 2 dB from the curves corresponding to $N = 256, 512$ and 1024 respectively. On the other hand, Figure 4.5 compares lattices employing component codes of the same lengths, but where the code rates violate the capacity rule for the cases of $N = 512$ and $N = 1024$. The gap between the different error curves is more visible in this case, and for WER=10^{-1}, P_{out} is at 1.1 dB and 2.8 dB from $N = 512$ and $N = 1024$ respectively. The codes choice is thus a crucial point in the resulting lattice performance.

Moreover, increasing n does have the impact of improving the final lattice L performance. This is illustrated in the simulation results of Figure 4.6 where two lattices of same dimension 2048, constructed over two and four-dimensional lattice partition chains, are compared in terms of WER. The gain achieved with $n = 4$ is equal to 3 dB for a word error rate of 3.10^{-2}.

4.5 Conclusion

This chapter describes a multilevel lattice coding scheme on the Rayleigh block fading channel. Coding on this type of channels calls for a new code design capable of mitigating the loss in performance due to the fading effect. The chapter offers an overview on the algebraic number theoretical notions, paving the way to their implementation in the multilevel code design.

The design was applied to the construction of algebraic lattices from 2 and 4-dimensional algebraic number fields. For each dimension n, the algebraic lattices underwent a rotation, base reduction and normalization operations so as to become equivalent to a rotated $\mathbb{Z}^n$ lattice. This allowed for a construction similar to the one

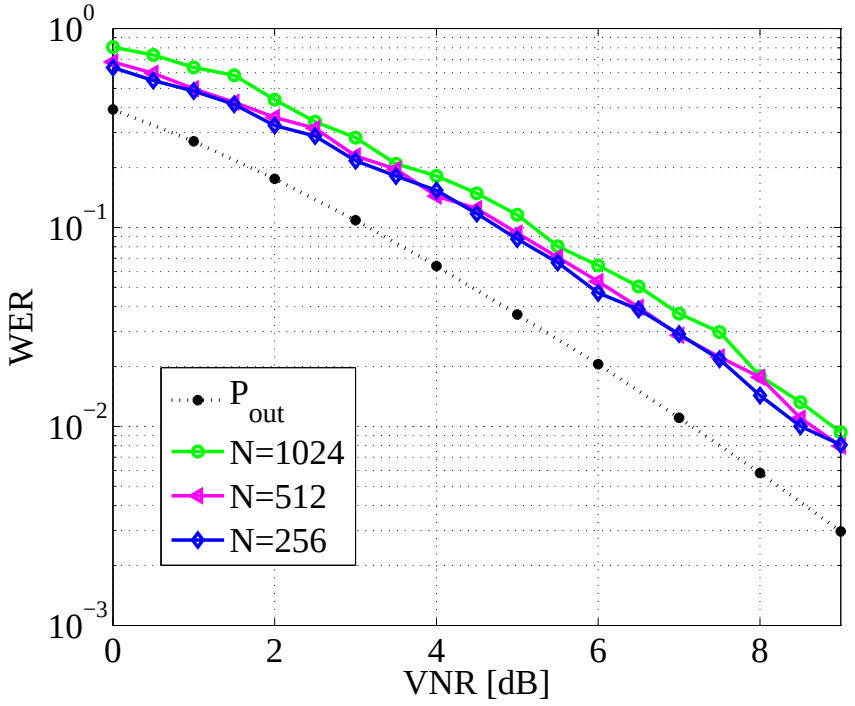


Figure 4.4 – The multilevel design performance over the Rayleigh block fading channel for $n = 4$.

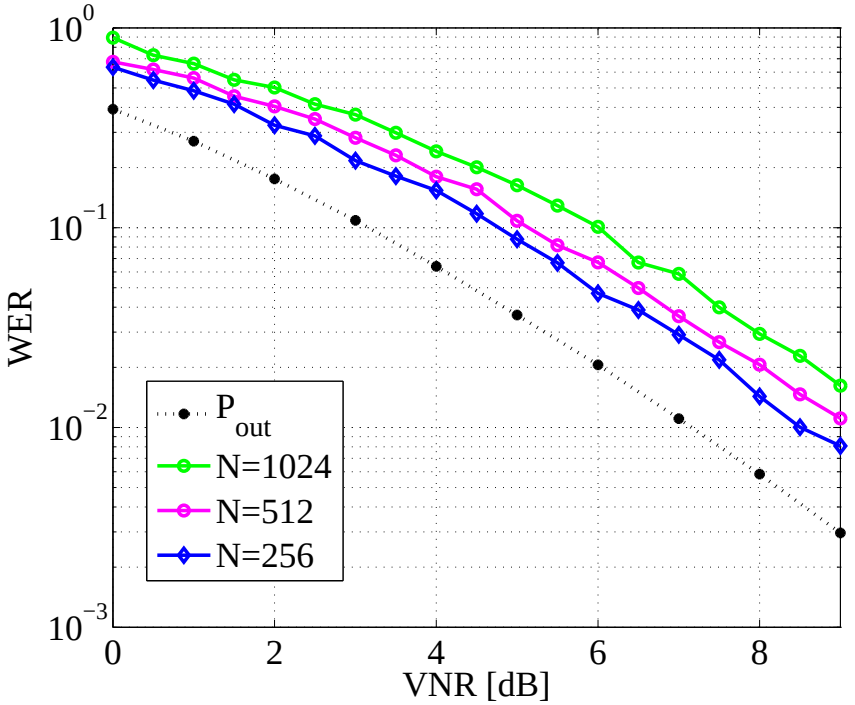


Figure 4.5 – The multilevel design performance over the Rayleigh block fading channel for $n = 4$ without respecting the capacity rule.

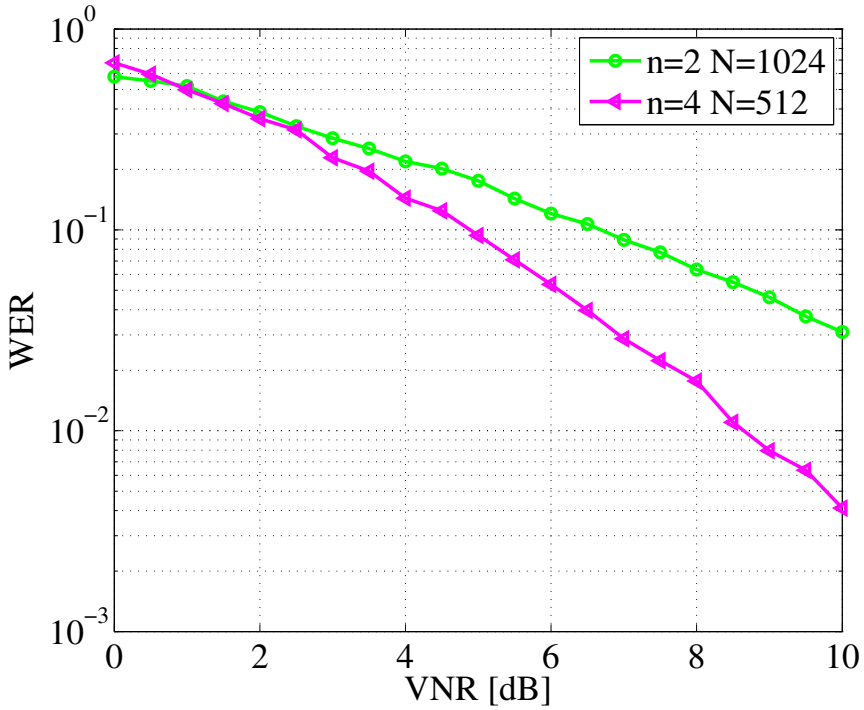


Figure 4.6 – Two lattices of dimension 2048 obtained with two and four-dimensional lattice partition chains.

applied in the Gaussian case. In fact, the same capacity curves, thus the same number of effective levels and maximum component code rates seen in Chapter 3 remain valid for this case.

Simulation results have shown that even though the gap of the WER from the channel outage probability was not reduced when increasing n, the performance of a lattice L of dimension 2048 was improved by 2 dB between a two and a four-dimensional lattice partition chain. This gain in performance however comes at the expense of increasing the system's complexity, as a result of an increased number of levels.

Conclusions and perspectives

Conclusions

The tremendous growth of wireless communication systems calls for new transmission designs capable of guaranteeing reliable data transmissions over limited resources. Lattice codes are structured signal constellations proven to achieve the capacity of the AWGN channel [30]. Motivated by these attractive properties, this thesis serves as a tool for harnessing lattices. More specifically, efficient lattice designs are described and employed over both AWGN and Rayleigh fading channels.

The first chapter of the thesis serves as an introduction to lattices: We provide a brief history on the development of lattice coding theory in digital communications and list the most important lattice parameters. Then, we explain Constructions A, B, D and D' that produce lattices using error-correction codes, followed with a description of the universal sphere decoder algorithm used to efficiently decode lattices of small dimensions.

The second chapter is focused on lattices produced by straightforward encoding an integer vector to a lattice point using the lattice generator matrix. Once obtained, the infinite lattice is subject to a shaping operation, and the resulting lattice code (or lattice constellation) performance is determined by the shaping domain. The decoding process can either take into account the shaping region or not. In the former case, we talk about lattice code decoding, and in the latter we talk about (naive) lattice decoding. In this chapter, we opt for the nested shaping mechanism, which employs two nested lattices: the fine lattice and the coarse lattice, and the lattice code is the set containing the fine lattice points that are inside the Voronoi region of the coarse lattice. This shaping mechanism helps achieve a good shaping gain with affordable complexity. A lattice code decoder was proposed, which performs a reshaping operation on a list of decoded lattice points, to check whether they belong to the shaping boundaries or

not. Simulation results applied to the 8-dimensional Gosset lattice E_8 have shown that the proposed lattice code decoder increases the shaping gain, an increase that however becomes less significant as we go higher with the spectral efficiency. We also compare the lattice code scheme, using both lattices E_8 and BW_{16}, to the LTE baseline where the frame length is determined by the modulation, code rate and the size of uncoded bits. Simulation results show that for high spectral efficiencies, the lattice code can achieve a better performance in terms of frame error rate.

In Chapter 3, we were interested in increasing the lattice dimension in order to achieve better coding gains, and for that we had to resort to coded modulation, more specifically, for multilevel coding. This chapter aims at explaining the construction of efficient, capacity-approaching (Poltyrev capacity) lattices on the AWGN channel using nested binary Reed-Muller codes, where the component codes' rates are chosen based on the capacity rule. The construction is performed on standard binary partitions of dimensions $n = 1, 2$ and 4. Simulation results show that for lattices having the same dimension, increasing n (thus, the number of levels) improves the overall performance in terms of word error rate.

In the last chapter, the multilevel lattice coding is extended to the Rayleigh block fading channel. This type of channels requires lattices that achieve maximum diversity, and therefore an overview on algebraic number theory was necessary, since it is known that algebraic number theory is an effective tool for designing good lattices for the fading channels. After introducing number-theoretical concepts that are relevant for the algebraic lattice construction, the latter was subject to a rotation and base reduction operations that allowed us to obtain rotated versions of the binary lattice partition chains used in Chapter 3. Therefore, the construction on the Rayleigh block fading channel is carried out by mimicking the Gaussian case. Computing the word error rate for lattices obtained using the followed multilevel construction shows that increasing the lattice dimension (i.e. increasing the component codes' length) barely degrades the global performances. Indeed, using component code rates that exceed the different levels' capacities, and thus violate the capacity rule which forms the cornerstone of our lattice construction, shows a clear increase in the WER with the lattice dimension.

Perspectives

Knowing that the multilevel lattice coding was performed without any power constraint, we consider, as a perspective, the application of a hypercube shaping in order to design finite multilevel lattice constellations that satisfy a certain power constraint.

This results in truncated versions of the constructed lattices, that can later be compared to QAM constellations for different spectral efficiencies.

In this thesis, multilevel lattices are the result of a Construction D in which the different component codes form a set of nested binary linear codes employed on the different levels of a lattice partition chain. Therefore, as another perspective, we consider to remove the requirement of nested codes, and perform what was recently proposed in [46] as Construction π_A (and its generalization to Construction π_D). The removal of such requirement makes the rate allocation and hence the lattice construction easier. In addition, Construction π_A makes the construction also more flexible by allowing the codes to be over different fields. This construction was shown to achieve the AWGN capacity under the suboptimal multistage decoding. A possible future work is therefore to perform Construction π_A for building lattices on the Rayleigh block fading channel.

In addition to their linear structure and potential to achieve the AWGN channel capacity, lattices are known to have salient structural properties well-suited for multiple access channels. This was shown through the promising Compute-and-Forward scheme [66] that allows us to harness the multiple access interference through the use lattice coding. This framework is applicable to any relaying channel. That said, we propose for future work, the implementation of the multilevel lattice scheme of Chapter 4 in multi-user applications.

APPENDIX A

The $|\mathbf{u}|\mathbf{u}+\mathbf{v}|$ construction

Let $\mathcal{C}_1(N, k_1, d_1)$ and $\mathcal{C}_2(N, k_2, d_2)$ be two binary codes, and let $\mathbf{u}$ and $\mathbf{v}$ be two codewords in $\mathcal{C}_1$ and $\mathcal{C}_2$ respectively. We form the following linear code:

$$\mathcal{C} = |\mathcal{C}_1|\mathcal{C}_1 + \mathcal{C}_2| = \{|\mathbf{u}|\mathbf{u}+\mathbf{v}| : \mathbf{u} \in \mathcal{C}_1, \mathbf{v} \in \mathcal{C}_2\}$$

which encoder can be represented as in Figure A.1.

Theorem A.0.1. *$|\mathbf{u}|\mathbf{u}+\mathbf{v}|$ is a $(2N, k_1 + k_2, min\{2d_1, d_2\})$ binary code.*

Proof. Length$(\mathcal{C}) = 2N$: The length of the codewords is $2N$ by construction.
Dimension$(\mathcal{C}) = k_1 + k_2$: If $|\mathbf{u}|\mathbf{u}+\mathbf{v}| = |\mathbf{u}'|\mathbf{u}'+\mathbf{v}'|$ then $\mathbf{u} = \mathbf{u}'$ and $\mathbf{v} = \mathbf{v}'$, so the assignment $(\mathbf{u}, \mathbf{v}) \mapsto |\mathbf{u}|\mathbf{u}+\mathbf{v}|$ gives a bijection between $\mathcal{C}_1 \times \mathcal{C}_2$ and $|\mathcal{C}_1|\mathcal{C}_1+\mathcal{C}_2|$. Therefore, the size of $\mathcal{C}$ is the same as $\mathcal{C}_1 \times \mathcal{C}_2$, which is $2^{k_1}2^{k_2} = 2^{k_1+k_2}$. Thus the number of codewords is $k_1 + k_2$.
Minimum distance$(\mathcal{C}) = \min\{2d_1, d_2\}$:
If $\mathbf{x} = |\mathbf{u}|\mathbf{u}+\mathbf{v}|$ and $\mathbf{y} = |\mathbf{u}'|\mathbf{u}'+\mathbf{v}'|$ are distinct codewords, then we have

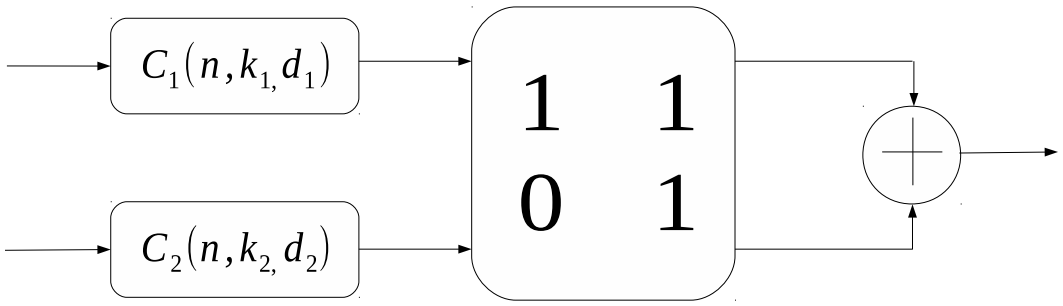


Figure A.1 – Encoder for $\mathcal{C}(N, k, d)$ generated by $|\mathbf{u}|\mathbf{u}+\mathbf{v}|$ construction.

$d(\mathbf{x},\mathbf{y}) = d(\mathbf{u},\mathbf{u}') + d(\mathbf{u}+\mathbf{v},\mathbf{u}'+\mathbf{v}')$, by the definition of distance.
If $\mathbf{v} = \mathbf{v}'$ then $d(\mathbf{u}+\mathbf{v},\mathbf{u}'+\mathbf{v}') = d(\mathbf{u},\mathbf{u}')$, so $d(\mathbf{x},\mathbf{y}) = 2d(\mathbf{u},\mathbf{u}') \geq 2d_1$.
If $\mathbf{v} \neq \mathbf{v}'$ then

$$\begin{aligned} d(\mathbf{x},\mathbf{y}) &= w(\mathbf{u}-\mathbf{u}') + w(\mathbf{u}+\mathbf{v}-\mathbf{u}'-\mathbf{v}') \\ &= w(\mathbf{u}'-\mathbf{u}) + w(\mathbf{u}+\mathbf{v}-\mathbf{u}'-\mathbf{u}) \end{aligned}$$

According to the triangle inequality: $w(\mathbf{x}+\mathbf{y}) \leq w(\mathbf{x}) + w(\mathbf{y})$. So

$$\begin{aligned} d(\mathbf{x},\mathbf{y}) &\geq w(\mathbf{u}'-\mathbf{u}+\mathbf{u}+\mathbf{v}-\mathbf{u}'-\mathbf{v}') \\ &= w(\mathbf{v}-\mathbf{v}') \\ &\geq d_2 \end{aligned}$$

This shows that $d(|\mathcal{C}_1|\mathcal{C}_1+\mathcal{C}_2|) \geq \min\{2d_1, d_2\}$.
If $d(\mathbf{u},\mathbf{u}') = d_1$ then $d(|\mathbf{u}|\mathbf{u}+\mathbf{v}|, |\mathbf{u}'|\mathbf{u}'+\mathbf{v}|) = 2d(\mathbf{u},\mathbf{u}') = 2d_1$.
If $d(\mathbf{v},\mathbf{v}') = d_2$ then $d(|\mathbf{u}|\mathbf{u}+\mathbf{v}'|, |\mathbf{u}|\mathbf{u}+\mathbf{v}'|) = d(\mathbf{v},\mathbf{v}') = d_2$.
So we have equality.

APPENDIX B

Soft-input decoding for Reed-Muller codes

Different decoding algorithms were designed for the Reed-Muller codes, the first was the majority algorithm developed by Reed in [71]. Later on, the recursive construction of Reed-Muller codes has allowed the employment of a decoding technique that splits the original $\mathcal{RM}(r,m)$ code, over and over, into two different codes of shorter length until reaching a previously chosen set of terminal nodes. In [76], the authors have described a soft-decision decoding procedure for $\mathcal{RM}$ codes, in which the code decomposition stops at repetition $\mathcal{RM}(0,m)$ and even-weight $\mathcal{RM}(m-1,m)$ codes.

We assume that the transmitted bits b_i for $i=1,\cdots,N$ are mapped to symbols $c_i=(-1)^{b_i}$. Any $\mathcal{RM}$ codeword $\mathbf{c}$ belongs then to $\{-1,+1\}^N$, and is a concatenation of the codes $\{\mathbf{u},\mathbf{u}\odot\mathbf{v}\}$. The transmission is carried out over an AWGN channel, the received vector $\mathbf{y}$ consists of two halves $\mathbf{y}'$ and $\mathbf{y}''$, which correspond to the corrupted versions of $\mathbf{u}$ and $\mathbf{u}\odot\mathbf{v}$, respectively. At the receiver, the soft-input information for each element of $\mathbf{y}$ is given by:

$$\rho_i=\ln(q(c_i))=\ln\frac{\Pr(y_i|c_i=+1)}{\Pr(y_i|c_i=-1)}$$

The decoder begins by decoding $\mathbf{v}=\mathcal{RM}(r-1,m-1)$, and then proceeds with $\mathbf{u}=\mathcal{RM}(r,m-1)$.

For decoding $\mathbf{v}$, both parts of the received vector $\mathbf{y}'$ and $\mathbf{y}''$ are needed ($\mathbf{v}=\mathbf{u}.(\mathbf{u}.\mathbf{v})$). The soft-decision metric is computed as follows:

$$\rho_i^v=\ln(q(v_i))=\ln\frac{\Pr(y_i|v_i=+1)}{\Pr(y_i|v_i=-1)} \tag{B.1}$$

It is obvious that $v_i=+1$ if both u_i and u_iv_i have the same sign, and $v_i=-1$ otherwise. Consequently, equation (B.1) can be written as:

$$\rho_i^v=\ln\frac{1+q(c_i).q(c_{i+N/2})}{q(c_i)+q(c_{i+N/2})}$$

If $\hat{\mathbf{v}}$ is the hard decision of $\rho(v)$, then $\mathbf{u}$ has two LLRs: one derived from $\mathbf{y}'$, and the second from $\hat{\mathbf{v}}.\mathbf{y}''$. The soft-decision metric is:

$$\rho_i^u = \ln \frac{\Pr(y_i|\hat{v}_i, u_i = +1)}{\Pr(y_i|\hat{v}_i, u_i = -1)}$$

which can be approximated by:

$$\rho_i^u = \frac{1}{2}(\rho_i + \hat{v}_i.\rho_{i+N/2})$$

The decoding algorithm is presented below. This algorithm takes as input the received vector $\mathbf{y}$, the order r and dimension m of the considered $\mathcal{RM}$ code, and an integer $\upsilon = \{0, 1\}$ which determines the order of the codes that are not decomposed further. The decoder is based on soft-decision ML decoding of parity-check and either repetition of biorthogonal codes. If the one of the terminating notes is not yet reached, the $\mathcal{RM}$ code is decomposed and recursively decoded using the decoder itself. In the final step, the decoder outputs the solution. For example, in figure B.1, we present simulation

Algorithm 3 Recursive Decoding of $\mathcal{RM}(r, m)$

Input: $\mathbf{y}, r, m, \upsilon$

Output: $\hat{c}$

1: **if** $r = \upsilon$ or $r = m - 1$ **then**
2: ML decoding of $\mathcal{RM}(r, m)$
3: Go to 10
4: **else**
5: Calculate $\rho_i^v = \ln \frac{1+q(c_i).q(c_{i+N/2})}{q(c_i)+q(c_{i+N/2})}$ for $i = 1, \cdots, \frac{N}{2}$
6: $\hat{\mathbf{v}} \leftarrow$ decode $\mathbf{y}'$ according to $\mathcal{RM}(r-1, m-1)$.
7: Calculate $\rho_i^u = \frac{1}{2}.(\rho_i + \hat{v}_i\rho_{i+N/2})$ for $i = 1, \cdots, \frac{N}{2}$
8: $\hat{\mathbf{u}} \leftarrow$ decode $\mathbf{y}''$ according to $\mathcal{RM}(r, m-1)$.
9: **end if**
10: Solution: $\hat{\mathbf{c}} = (\hat{\mathbf{u}}, \hat{\mathbf{u}}\hat{\mathbf{v}})$.

results for $\mathcal{RM}(4, 9)$, where the Word Error Rate is plotted as function of $Eb/N0$, Eb being the energy per transmitted information bit. The figure shows that numerical performance improve by taking $\upsilon = 1$ instead of $\upsilon = 0$, i.e., if we perform ML decoding on first-order $\mathcal{RM}$ codes, instead of repetition codes.

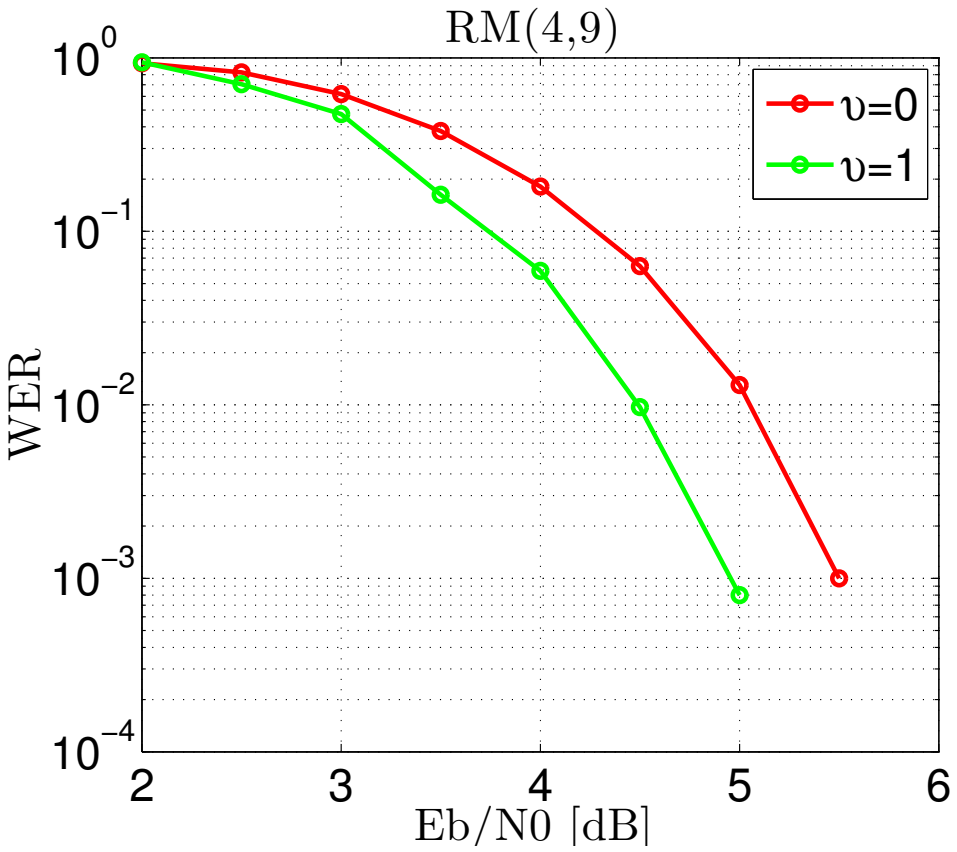


Figure B.1 – Recursive decoding of $\mathcal{RM}(4,9)$ for $\upsilon = 0$ and 1.

APPENDIX C

Commands in Sage

Dimension 2

```
\\Define the minimal polynomial

K.<w>=NumberField(x^2 - 48*x + 544);
theta=K.ideal(2).prime_factors()[0].gens_reduced()[0];
theta.minpoly()


x^2 - 2

\\Find the fundamental unit and its conjugates

K.<w>=NumberField(x^2-2);
g2=(K.ideal(2).prime_factors()[0]).gens_reduced()[0];
Gu=K.unit_group();
su=Gu.gens_values();uf=su[1];
print g2.complex_embeddings();
print uf.complex_embeddings();


-1.41421356237310, 1.41421356237310]
-0.414213562373095, 2.41421356237309

\\Find alpha

g2pos=uf*g2;alpha=g2pos;
g2vec=g2pos.complex_embeddings();
print g2vec
```

```
trlat=[sqrt(g2vec[0]),sqrt(g2vec[1])];
did=Matrix(RR,diagonal_matrix(trlat));


[0.585786437626905, 3.41421356237309]

\\w=sqrt(2)

g2pos

 w+2

\\Find U and the final Gram matrix

idb=K.integral_basis();
gen=did*Matrix(RR,K.inkowski_embedding(idb));
gr=gen.transpose()*gen;
gram=Matrix([[gr[i,j].round() for i in [0..1]] for j in [0..1]],sparse=False);
U=gram.LLL_gram();
gram_red=U.transpose()*gram*U;
gram_red


[16  0]
[0  16]


U

[ 1  -1]
[ 0   1]
```

Dimension 4

```
\\Find the fundamental units


Gu=K.unit_group();
su=Gu.gens_values();uf1=su[1];uf2=su[2];uf3=su[3];
```

```
print g2.complex_embeddings()
print uf1.complex_embeddings()
print uf2.complex_embeddings()
print uf3.complex_embeddings()
```

```
[-1.84775906502257, -0.765366864730180, 0.765366864730180, 1.84775906502257]
[2.41421356237309, -0.414213562373095, -0.414213562373095, 2.41421356237309]
[-0.847759065022573, 0.234633135269820, 1.76536686473018, 2.84775906502257]
[0.566454497350521, -1.17958042710327, 0.351153302357085, 4.26197262739567]
```

```
\\Find alpha and all its conjugates

alpha=g2*uf2*uf3;
print alpha.complex_embeddings()
```

```
[0.887325194919154, 0.211829556901869, 0.474461944113370, 22.4263833040656]
```

```
\\w= sqrt(2+sqrt(2))

alpha

2*w^3 + 4*w^2 - w - 2
```

```
\\The diagonal matrix M

alphavec=alpha.complex_embeddings()
trlat=[sqrt(alphavec[i]) for i in [0..3]]
did=Matrix(RR,diagonal_matrix(trlat));did
```

```
[0.941979402598143 0.000000000000000 0.000000000000000 0.000000000000000]
[0.000000000000000 0.460249450735000 0.000000000000000 0.000000000000000]
[0.000000000000000 0.000000000000000 0.688811980233627 0.000000000000000]
[0.000000000000000 0.000000000000000 0.000000000000000 4.73565025145076]
```

```
U

[-3 1 -1 -3]
[ 0 -4 -3 3]
[ 1  0 0  1]
[ 0 1  1 -1]

\\The matrix O

O=gen*U;

O

 0.137949689641472  0.693519922661074 -0.587937801209680 0.392847479193551]
[-0.392847479193551  0.587937801209679  0.137949689641472 -0.693519922661074]
[-0.587937801209679 -0.392847479193551 -0.693519922661074 -0.137949689641471]
[ 0.693519922661074 -0.137949689641472 -0.392847479193550 -0.587937801209680]

\\The Gram matrix

OO.transpose()*OO

[  1.00000000000000 -1.38777878078145e-16  4.99600361081320e-16 -6.10622663543836e-16
[-1.38777878078145e-16      1.00000000000000 -2.22044604925031e-16
5.27355936696949e-16]
[ 4.99600361081320e-16 -2.22044604925031e-16      1.00000000000000
-5.82867087928207e-16]
[-6.10622663543836e-16  5.27355936696949e-16 -5.82867087928207e-16
1.00000000000000]
```

Bibliography

[1] Scenarios, requirements and kpis for 5g mobile and wireless system. Deliverable D1.1 of the METIS project, April 2013. available online on https://www.metis2020.com. xi

[2] 3GPP TS 22.368 service requirements for machine-type communications, V13.0.0, June 2014. 46

[3] 3GPP TS 36.212 multiplexing and channel coding, V12.2.0, Sept. 2014. 47

[4] C. Al Bechlawi, J. C. Belfiore, and F. Guilloud. Reed-Muller lattice coding for the Rayleigh block fading channel. In *IEEE Wireless Communications and Networking Conference (WCNC), Submitted*, March 2016. 5

[5] C. Al Bechlawi and F. Guilloud. Approximation du LLR pour un décodage multi-niveaux de réseaux de points imbriqués. In *25ème édition du colloque Gretsi, Lyon, France*, Septembre 2015. 4

[6] C. Al Bechlawi and F. Guilloud. Efficient LLR estimation for multistage decoding. *Electronics Letters*, 51(14):1076–1078, 2015. 4, 71

[7] C. Al Bechlawi and F. Guilloud. Frame length reduction for massive-machine communications. In *IEEE 81st Vehicular Technology Conference (VTC Spring)*, May 2015. 4

[8] E. Arikan. Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Transactions on Information Theory,*, 55(7):3051–3073, July 2009. 2, 47

[9] E. S. Barnes and N. J. A. Sloane. New lattice packings of spheres. *Canad. J. Math*, pages 117–130, 1983. 19, 22

[10] E. S. Barnes and G. E. Wall. Some extreme forms defined in terms of Abelian groups. *Journal of the Australian Mathematical Society*, 1(01):47–63, August 1959. 7, 19

[11] E. Bayer-Fluckiger. Definite unimodular lattices having an automorphism of given characteristic polynomial. *Commentarii Mathematici Helvetici*, 59(1):509–538, 1984. 80

[12] E. Bayer-Fluckiger. Lattices and number fields. *Contemporary Mathematics*, 241:69–84, 1999. 80

[13] E. Bayer-Fluckiger. Modular lattices over cyclotomic fields. *Journal of Number Theory*, 114:394–411, 2005. 80

[14] E. Bayer-Fluckiger, F. Oggier, and E. Viterbo. New algebraic constructions of rotated $\mathbb{Z}^n$-lattice constellations for the Rayleigh fading channel. *IEEE Transactions on Information Theory*, 50(4):702–714, April 2004. 80, 92

[15] E. Bayer-Fluckiger and I. Suarez. Ideal lattices over totally real number fields and Euclidian minima. *Archiv Math.*, 86:217–225, 2006. 80

[16] J. C. Belfiore and K. Boulle. Modulation schemes designed for the Rayleigh channel. *Proc. CISS, Princeton*, pages 288–293, 1992. 81

[17] J.-C. Belfiore and F. Oggier. Lattice code design for the Rayleigh fading wiretap channel. In *IEEE International Conference on Communications Workshops (ICC)*, June 2011. 11

[18] Z. I. Borevich and I. R. Shafarevich. *Number Theory*. New York, Academic Press, 1966. 82

[19] A. Bos, J. Conway, and N. J. A. Sloane. Further lattice packings in high dimensions. *Mathematika*, 29:171–180, 1982. 20

[20] J. Boutros and E. Viterbo. High diversity lattices for fading channels. In *Proceedings of the IEEE International Symposium on Information Theory*, Sep 1995. 79

[21] J. Boutros and E. Viterbo. Signal space diversity: a power and bandwidth-efficient diversity technique for the Rayleigh fading channel. *IEEE Transactions on Information Theory*, 44(4):1453–1467, Jul 1998. 92

[22] J. Boutros, E. Viterbo, C. Rastello, and J.-C. Belfiore. Good lattice constellations for both Rayleigh fading and Gaussian channels. *IEEE Transactions on Information Theory*, 42(2):502–518, Mar 1996. 11, 80, 89, 97

[23] A.R. Calderbank. Multilevel codes and multistage decoding. *IEEE Transactions on Communications*, 37(3):222–229, Mar 1989. 51

[24] Li-Chia Choo, Cong Ling, and Kai-Kit Wong. Achievable rates for lattice coded Gaussian wiretap channels. In *IEEE International Conference on Communications Workshops (ICC)*, June 2011. 11

[25] J. Conway and N. Sloane. A fast encoding method for lattice codes and quantizers. *IEEE Transactions on Information Theory*, 29(6):820–824, Nov 1983. 11, 40

[26] J. Conway and N. J. A. Sloane. *Sphere Packings, Lattices and Groups*. Springer New York, 2010. xiv, 3, 8, 11, 15, 17, 20, 35, 79

[27] R. de Buda. The upper error bound of a new near-optimal code. *IEEE Transactions on Information Theory*, 21(4):441–445, Jul 1975. xiii, 10

[28] R. de Buda. Some optimal codes have structure. *IEEE Journal on Selected Areas in Communications*, 7(6):893–899, Aug 1989. 10

[29] U. Erez, S. Litsyn, and R. Zamir. Lattices which are good for (almost) everything. *IEEE Transactions on Information Theory*, 51(10):3401–3416, Oct 2005. 51

[30] U. Erez and R. Zamir. Achieving $\frac{1}{2}$log (1+SNR) on the AWGN Channel with Lattice Encoding and Decoding. *IEEE Transactions on Information Theory*, 50(10):2293–2314, 2004. xii, xiii, 10, 37, 40, 43, 107

[31] Chen Feng, D. Silva, and F.R. Kschischang. Design criteria for lattice network coding. In *45th Annual Conference on Information Sciences and Systems (CISS)*, March 2011. 11

[32] U. Finke and M. Pohst. Improved methods for calculating vectors of short length in a lattice, including a complexity analysis. *Mathematics of Computation*, 44(170):463–471, 1985. 23

[33] N. I. Fisher. Problem with the current definition of the standard deviation of wind direction. *Journal of Climate and Applied Meteorology*, 26:1522–1529, 1987. 72

[34] N. I. Fisher. *Statistical Analysis of Circular Data*. Cambridge University Press, 1996. 72

[35] G. D. Jr. Forney. Trellis shaping. *IEEE Transactions on Information Theory*, 38(2):281–300, 1992. 11, 33

[36] G.D. Jr. Forney. Coset codes. I. Introduction and geometrical classification. *IEEE Transactions onInformation Theory*, 34(5):1123–1151, Sep 1988. 11, 51

[37] G.D. Jr. Forney. Coset codes. II. Binary lattices and related codes. *IEEE Transactions on Information Theory*, 34(5):1152–1187, Sep 1988. 11, 22, 51

[38] G.D. Jr. Forney. Multidimensional constellations. II. Voronoi constellations. *IEEE Journal on Selected Areas in Communications*, 7(6):941–958, Aug 1989. 11

[39] G.D. Jr. Forney. Approaching the capacity of the AWGN channel with coset codes and multilevel coset codes. In *Proceedings of the IEEE Journal on Information Theory*, pages 164–, Jun 1997. 10

[40] G.D. Jr. Forney. Approaching the capacity of the AWGN channel with coset codes and multilevel coset codes. In *Proceedings of the IEEE International Symposium on Information Theory*, pages 164–, Jun 1997. 54

[41] G.D. Jr. Forney and L.-F. Wei. Multidimensional constellations. I. Introduction, figures of merit, and generalized cross constellations. *IEEE Journal on Selected Areas in Communications*, 7(6):877–892, Aug 1989. 11

[42] Jr. Forney, G.D., M.D. Trott, and Sae-Young Chung. Sphere-bound-achieving coset codes and multilevel coset codes. *IEEE Transactions on Information Theory*, 46(3):820–850, May 2000. 4, 11, 19, 27, 52, 54, 55, 57, 58, 59, 76, 96

[43] X. Girand and J.C. Belfiore. Constellations matched to the Rayleigh fading channel. *IEEE Transactions on Information Theory*, 42(1):106–115, Jan 1996. 80

[44] B. Hassibi and H. Vikalo. On the sphere-decoding algorithm I. Expected complexity. *IEEE Transactions on Signal Processing*, 53(8):2806–2818, Aug 2005. 25

[45] B.M. Hochwald and S. Ten Brink. Achieving near-capacity on a multiple-antenna channel. *IEEE Transactions on Communications*, 51(3):389–399, March 2003. 42

[46] Y.-C. Huang and K. R. Narayanan. Construction π_A and π_D Lattices: Construction, Goodness, and Decoding Algorithms. *ArXiv e-prints*, June 2015. xxii, 109

[47] J. Huber. Multilevel Codes: Distance Profiles and Channel Capacity. In *ITG-Fachbericht 130*, pages 305–319, Oct 1994. 51, 54

[48] J. Huber and U. Wachsmann. Capacities of equivalent channels in multilevel coding schemes. *Electronics Letters*, 30(7):557–558, Mar 1994. 51, 52

[49] H. Imai and S. Hirakawa. A new multilevel coding method using error-correcting codes. *IEEE Transactions on Information Theory*, 23(3):371–377, May 1977. 51

[50] A. Ingber and M. Feder. On the optimality of multilevel coding and multistage decoding. In *IEEE 25th Convention of Electrical and Electronics Engineers in Isreal*, pages 731–735, Dec 2008. 51

[51] K. Ireland and M. Rosen. *A Classical Introduction to Modern Number Theory.* Springer New York, 1998. 87

[52] B.D. Jelicic and S. Roy. Design of trellis coded QAM for flat fading and AWGN channels. *IEEE Transactions on Vehicular Technology*, 44(1):192–201, Feb 1995. 81

[53] A. Korkin and G. Zolotarev. Sur les formes quadratiques. *Math. Ann.*, 6:366–389, 1873. 9

[54] F.R. Kschischang and S. Pasupathy. Optimal nonuniform signaling for Gaussian channels. *IEEE Transactions on Information Theory*, 39(3):913–929, May 1993. 34

[55] S. Kudekar, M. Mondelli, E. Şaşoğlu, and R. Urbanke. Reed-Muller Codes Achieve Capacity on the Binary Erasure Channel under MAP Decoding. *ArXiv e-prints*, May 2015. 57

[56] S. Kumar and H. D. Pfister. Reed-Muller Codes Achieve Capacity on Erasure Channels. *ArXiv e-prints*, May 2015. 57

[57] B.M. Kurkoski. Coded modulation using lattices and Reed-Solomon codes, with applications to flash memories. *IEEE Journal on Selected Areas in Communications*, 32(5):900–908, May 2014. 11

[58] Chung-Pi Lee, Shih-Chun Lin, Hsuan-Jung Su, and H.V. Poor. Multiuser lattice coding for the multiple-access relay channel. *IEEE Transactions on Wireless Communications*, 13(7):3539–3555, July 2014. 11

[59] J. Leech. Notes on sphere packings. *Canad. J. Math*, 19:251–267, 1967. 7

[60] S. Lembo, K. Ruttik, and O. Tirkkonen. Modeling BLER performance of punctured Turbo codes. In *12th International Symposium on Wireless Personal Multimedia Communications WPMC*, 7-10 Sept. 2009. 47

[61] T. Linder, C. Schlegal, and K. Zeger. Corrected proof of de Buda's theorem [lattice channel codes]. *IEEE Transactions on Information Theory*, 39(5):1735–1737, Sep 1993. 10

[62] C. Ling and J.-C. Belfiore. Achieving AWGN channel capacity with lattice Gaussian coding. *ArXiv e-prints*, February 2013. 10, 33

[63] H.-A. Loeliger. Averaging bounds for lattices and linear codes. *IEEE Transactions on Information Theory*, 43(6):1767–1773, Nov 1997. 10

[64] K. Mardia. *Statistics of Directional Data*. Academic Press, 1972. 55

[65] H. Minkowski. *Geometrie der Zahlen*. Leipzig Teubner, 1910. 9

[66] B. Nazer and M. Gastpar. Compute-and-forward: Harnessing interference through structured codes. *IEEE Transactions on Information Theory*, 57(10):6463–6486, Oct 2011. xxii, 11, 109

[67] M. Nokleby and B. Aazhang. Lattice coding over the relay channel. In *IEEE International Conference on Communications (ICC)*, June 2011. 11

[68] N. Palgy and R. Zamir. Dithered probabilistic shaping. In *IEEE 27th Convention of Electrical Electronics Engineers in Israel (IEEEI)*, Nov 2012. 34

[69] G. Poltyrev. On Coding Without Restrictions for the AWGN Channel. *IEEE Transactions on Information Theory*, 40(2):409–417, 1994. xiii, 10, 11, 57

[70] Y. Polyanskiy, H. V. Poor, and S. Verdu. Channel coding rate in the finite block-length regime. *IEEE Transactions on Information Theory*, 56(5):2307–2359, May 2010. 47

[71] I. Reed. A class of multiple-error-correcting codes and the decoding scheme. *Transactions of the IRE Professional Group on Information Theory*, 4(4):38–49, Sept 1954. 21, 113

[72] G. Rekaya-Ben Othman. *Nouvelles constructions algébriques de codes spatio-temporels atteignant le compromis "multiplexage-diversité"*. PhD thesis, TELECOM ParisTech, 2004. 41

[73] M.-R. Sadeghi, A.H. Banihashemi, and D. Panario. Low-density parity-check lattices: Construction and decoding analysis. *48th Annual Allerton Conference on Information Theory*, 52(10):4481–4495, Oct 2006. 11, 20

[74] A. Sakzad, M.-R. Sadeghi, and D. Panario. Construction of turbo lattices. In *48th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 14–21, Sept 2010. 11

[75] P. Samuel. *Algebraic Theory of Numbers*. Paris, France: Hermann, 1971. 82, 97

[76] G. Schnabl and M. Bossert. Soft-decision decoding of Reed-Muller codes as generalized multiple concatenated codes. *IEEE Transactions on Information Theory*, 41(1):304–308, Jan 1995. 113

[77] O. Shalvi, N. Sommer, and M. Feder. Signal codes: Convolutional lattice codes. *IEEE Transactions on Information Theory*, 57(8):5203–5226, Aug 2011. 11

[78] C.E. Shannon. A mathematical theory of communication. *Bell System Technical Journal, The*, 27(4):623–656, Oct 1948. xi, 9

[79] N. Sommer, M. Feder, and O. Shalvi. Low density lattice codes. In *IEEE International Symposium on Information Theory*, pages 88–92, July 2006. 11

[80] N. Sommer, M. Feder, and O. Shalvi. Shaping methods for low-density lattice codes. In *Information Theory Workshop, 2009. ITW 2009. IEEE*, pages 238–242, 2009. xv, 33, 37, 38

[81] Yiwei Song and N. Devroye. Lattice codes for the Gaussian relay channel: Decode-and-forward and compress-and-forward. *IEEE Transactions on Information Theory*, 59(8):4927–4948, Aug 2013. 11

[82] V. Tarokh, A. Vardy, and K. Zeger. Universal bound on the performance of lattice codes. *IEEE Transactions on Information Theory*, 45(2):670–681, Mar 1999. 28

[83] M. Tomlinson. New automatic equaliser employing modulo arithmetic. *Electronics Letters*, 7(5):138–139, March 1971. 37

[84] D. Tse and P. Viswanath. *Fundamentals of Wireless Communication*. Cambridge University Press, 2005. 81

[85] R. Urbanke and B. Rimoldi. Lattice codes can achieve capacity on the [awgn. *IEEE Transactions on Information Theory*. xii, xiii, 10

[86] Mardia K. V and Jupp P. E. *Directional Statistics*. New York: Wiley, 1999. 72

[87] E. Viterbo and J. Boutros. A universal lattice code decoder for fading channels. *IEEE Transactions on Information Theory*, 45(5):1639–1642, Jul 1999. 23

[88] G. F. Voronoi. Nouvelles applications des paramètres continus à la thórie de formes quadratiques. *Journal fur die reine und angewandte Mathematik*, 134:198–287, 1908. 9

[89] U. Wachsmann, R.F.H. Fischer, and J.B. Huber. Multilevel codes: theoretical concepts and practical design rules. *IEEE Transactions on Information Theory*, 45(5):1361–1391, Jul 1999. 51, 54

[90] U. Wachsmann and J. Huber. Power and bandwidth efficient digital communication using Turbo codes in multilevel codes. *EUROP. TRANS. TELECOMMUN. (ETT*, 6:557–567, 1995. 51

[91] Yanfei Yan and Cong Ling. A construction of lattices from polar codes. In *Information Theory Workshop (ITW), 2012 IEEE*, pages 124–128, Sept 2012. 11

[92] Yanfei Yan, Cong Ling, and Xiaofu Wu. Polar lattices: Where Arikan meets Forney. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 1292–1296, July 2013. 11, 51

[93] Y. Yona and M. Feder. Efficient decoding of low density lattice codes. In *IEEE 25th Convention of Electrical and Electronics Engineers in Israel*, pages 484–488, Dec 2008. 11

[94] R. Zamir. *Lattice Coding for Signals and Networks*. Cambridge University press, 2014. 17, 43

List of Figures

1 Basic block diagram of a digital communications system. 2

1.1 Two-dimensional packings. 8

1.2 Orange packing in 3 dimensions. 8

1.3 Hexagonal lattice. 14

1.4 The covering and packing radii with respect to the Voronoi region. . . . 16

1.5 Decomposition of code $\mathcal{RM}(3,6)$. 22

1.6 The geometrical representation of the Sphere Decoding algorithm. . . . 24

1.7 Performance of some of the most popular lattices on the AWGN channel in terms of Normalized Word Error Rate. 28

2.1 Basic block diagram of a system employing lattice constellations. 36

2.2 Gosset lattice E_8 performance without shaping at the encoder. 36

2.3 Sent lattice points with and without shaping. 37

2.4 Transmission model employing a shaping operation. 38

2.5 Two nested lattices: the hexagonal lattice A_2 and a scaled version of A_2 of factor 3. 39

2.6 Simulation results for hypercube and nested shaping applied to the Gosset lattice E_8. 40

2.7 Performance of the proposed modified decoder. 43

2.8 Performance of the proposed modified decoder for $\eta = 1$ bit/dim. . . . 44

2.9 Impact of the list size l_s on the decoder's performance. 45

2.10 Performance of the proposed lattice code decoder on the Rayleigh fading channel. 46

2.11 Frame error rate comparison between E_8, BW_{16} and short frame LTE Turbo code. 49

3.1 $\mathbb{Z}$-aliased Gaussian density function $f_{\mathbb{Z},\sigma^2}(\mathbf{w}')$. 56

3.2 Capacity curves for the one-dimensional lattice partition chain $\mathbb{Z}/2\mathbb{Z}/\cdots/2^r\mathbb{Z}$. 59

3.3 MLC encoding and MSD decoding schemes for a two-level lattice construction (n=1). 61

3.4 Performance comparison of a Barnes-Wall lattice and a 2-level lattice construction built according to capacity rule for a code length $N = 1024$. 62

3.5 Capacity curves for the two-dimensional lattice partition chain $\mathbb{Z}^2/R\mathbb{Z}^2/\cdots/2^r\mathbb{Z}^2$. 63

3.6 The integer two-dimensional lattice $\mathbb{Z}^2$ and its sublattices. 64

3.7 MLC encoding and MSD decoding schemes for a four-level lattice construction (n=2). 66

3.8 Performance comparison between two lattices of dimension 1024 resulted from multilevel construction over one and two-dimensional lattice partition chains respectively. 67

3.9 Capacity of four-dimensional lattices versus $1/\sigma^2$. 68

3.10 Capacity curves for the four-dimensional lattice partition chain $\mathbb{Z}^4/D_4/R\mathbb{Z}^4/RD_4/2\mathbb{Z}^4/2D_4/2R\mathbb{Z}^4/\cdots$. 69

3.11 MLC encoding and MSD decoding schemes for a five-level lattice construction (n=4). 70

3.12 Simulation results for an obtained lattice L of dimension 1024. 71

3.13 Simulation results for an obtained lattice L of dimension 4096. 72

3.14 PDF of the von Mises distribution for $\mu = 0$ and different values of κ. . 73

3.15 LLR curves for 3 different methods of LLR estimation. 74

3.16 Word Error Rate Vs VNR for multiple methods of LLR estimation. . . 75

4.1 Example of increasing modulation diversity (a) F =1, (b) F=2. 82

4.2 The multilevel design performance over the Rayleigh block fading channel for $n = 2$. 99

4.3 The multilevel design performance over the Rayleigh block fading channel for $n = 2$ without respecting the capacity rule. 100

4.4 The multilevel design performance over the Rayleigh block fading channel for $n = 4$. 103

4.5 The multilevel design performance over the Rayleigh block fading channel for $n = 4$ without respecting the capacity rule. 103

4.6 Two lattices of dimension 2048 obtained with two and four-dimensional lattice partition chains. 104

A.1 Encoder for $\mathcal{C}(N, k, d)$ generated by $|\mathbf{u}|\mathbf{u} + \mathbf{v}|$ construction. 111

B.1 Recursive decoding of $\mathcal{RM}(4, 9)$ for $\upsilon = 0$ and 1. 115

List of Tables

1.1 Features of the most-known lattices for dimensions up to 256. 29

2.1 Code and modulation parameters used for lattice coding and the LTE baseline 48

3.1 The two-level construction constants for different value of σ^2 (n=1). . . 60

3.2 The four-level construction constants for different value of σ^2 (n=2). . . 65

3.3 Minimal algebraic and Euclidean norms of the four-dimensional lattices forming the lattice partition chain $\mathbb{Z}^4/D_4/R\mathbb{Z}^4/RD_4/2\mathbb{Z}^4/2D_4$. 67

3.4 The five-level construction constants for different value of σ^2 (n=4). . . 70

List of Publications

1. Al Bechlawi, C.; Guilloud, F., "Frame Length Reduction for Massive-Machine Communications," in *IEEE 81st Vehicular Technology Conference (VTC Spring)*, 11-14 May 2015.

2. Al Bechlawi, C.; Guilloud, F., "Efficient LLR estimation for multistage decoding," in *Electronics Letters* , vol.51, no.14, pp.1076-1078, 7 9 2015.

3. Al Bechlawi C., Guilloud F., Approximation du LLR pour un Décodage Multi-niveaux de Réseaux de Points Imbriqués, *GRETSI 2015 : 25ème édition du colloque GRETSI*, 08-11 septembre 2015, Lyon, France, 2015.

4. Al Bechlawi C., Belfiore J. C., Guilloud, F., "Reed-Muller Lattice Coding for the Rayleigh Fading Channel", in *IEEE Wireless Communications and Networking Conference (WCNC)*, 3-6 April 2016, *Submitted*.

Résumé

Dans cette thèse, nous étudions l'utilisation des réseaux de points dans le contexte des communications numériques. Les réseaux de points sont des ensembles de points discrets dans l'espace euclidien de dimension n. Ce travail de thèse est motivé par le fait que ces réseaux de points permettent d'atteindre la capacité du canal à bruit additif blanc gaussien. Les réseaux de points étant des groupes additifs, la propriété de linéarité est ainsi présente dans l'espace des signaux envoyés. Ce travail est ainsi dédié à l'analyse du codage par réseaux de points pour les communications point-à- point étudiées sous différents scénarios: l'étude est menée sur le canal gaussien et le canal avec évanouissements. Après l'introduction des réseaux de points infinis avec leurs constructions et algorithme de décodage par sphère, nous attaquons le problème de la mise en forme des réseaux de points qui consiste à sélectionner un nombre fini de points du réseau. Un décodeur permettant de prendre en compte cette mise en forme est proposé, et des comparaisons sont faites entre l'utilisation des réseaux de points de dimensions 8 et 16 et le schéma LTE à trames courtes: les résultats montrent que le codage par réseaux de points atteint des meilleures performances pour des grandes efficacités spectrales, à condition qu'un décodeur quasi-optimal soit mis en oeuvre. Vue la complexité élevée du décodeur par sphère pour des dimensions plus grandes, on étudie ensuite les constructions dites multi-niveaux de réseaux de points. Plus précisément, une construction multi-niveaux utilisant un ensemble de codes Reed-Muller binaires linéaires et imbriqués est proposée pour le canal gaussien et le canal à évanouissement de Rayleigh par blocs. Sur le canal gaussien, la construction est menée sur des partitions de dimensions 1,2 et 4. Pour chaque dimension, nous montrons comment obtenir le nombre suffisant de niveaux ainsi que les rendements des codes associés à chaque niveau. Nous montrons aussi comment l'augmentation de la dimension impacte les performances du système. Pour le canal à évanouissements par blocs, nous nous servons de la théorie des nombres algébriques qui permet d'obtenir la diversité égale à la dimension. Une fois le réseau de points algébrique obtenu, des opérations de rotation et de réduction de base nous permettent de nous ramener au cas d'un réseau d'entiers tourné, et par suite mener la construction multi-niveaux en imitant le cas gaussien.

Mots-clés : Communications numériques, Réseaux de points, Capacité du canal, Codage correcteur d'erreurs, Canal gaussien, Canal à évanouissements de Rayleigh, Shaping, Décodage par sphère, Codage/décodage multi-niveaux, Théorie des nombres algébriques

Abstract

In this work, we address the application of emergent structures, known as lattices, for the digital communications problem. Endowed with interesting properties such as periodicity and linearity, lattices have recently gained considerable attention as they solved the decades-old problem of achieving full capacity on the AWGN channel. Motivated by these promising results, this work is dedicated to the analysis of the lattice coding performance for point-to-point communications under different scenarios: the studies are carried out over both AWGN and Rayleigh fading channels. After introducing infinite lattices, along with their constructions and sphere decoding algorithm, we tackle the lattice shaping problem that selects a finite number of lattice points. A lattice code decoder that takes into account the shaping region is proposed, and lattices of 8 and 16 dimensions are compared to short packet LTE: lattice coding is proven to achieve better performance in terms of Frame Error Rate for high spectral efficiencies, provided that near-optimal sphere decoding is performed. Due to the high complexity of the sphere decoding algorithm for higher dimensions, lattices based on multilevel constructions are studied. More precisely, the multilevel lattice construction using a set of nested binary linear Reed-Muller codes is proposed for both Gaussian and Rayleigh block fading channels. On the AWGN channel, the construction is carried out over standard binary partition chains of dimensions 1, 2 and 4. For each dimension, we show how to obtain the sufficient number of levels, together with the component codes' rates assigned to each level. We also show how increasing the dimension affects the global lattice performance. For the Rayleigh block fading channel, we resort to algebraic number theory as it was proven to provide the maximum possible diversity. Once the algebraic lattice is built, a rotation and base reduction operations allow us to obtain a rotated version of the integer lattice, and thus carry out the construction by mimicking the AWGN case.

Keywords : Digital communications, Lattice theory, AWGN channel, Rayleigh fading channel, Channel capacity, Error-correcting codes, Shaping, Sphere decoding, Multilevel coding, Multistage decoding, Algebraic number theory

TELECOM Bretagne

n° d'ordre : 2016telb0371

Télécom Bretagne

Technopôle Brest-Iroise - CS 83818 - 29238 Brest Cedex 3

Tél : + 33(0) 29 00 11 11 - Fax : + 33(0) 29 00 10 00