# Nouvelle forme d'onde et récepteur avancé pour la télémesure des futurs lanceurs

Charles-Ugo Piat-Durozoi

# THÈSE

**En vue de l'obtention du**

# DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

**Délivré par :**

Institut National Polytechnique de Toulouse (Toulouse INP)

**Discipline ou spécialité :**

Informatique et Télécommunication

---

**Présentée et soutenue par :**

M. CHARLES-UGO PIAT-DUROZOI

le mardi 27 novembre 2018

**Titre :**

Nouvelle forme d'onde et récepteur avancé pour la télémesure des futurs lanceurs

---

**Ecole doctorale :**

Mathématiques, Informatique, Télécommunications de Toulouse (MITT)

**Unité de recherche :**

Institut de Recherche en Informatique de Toulouse (I.R.I.T.)

**Directeur(s) de Thèse :**

MME MARIE LAURE BOUCHERET

M. CHARLY POULLIAT

**Rapporteurs :**

M. EMMANUEL BOUTILLON, UNIVERSITE DE BRETAGNE SUD

M. FREDERIC GUILLOUD, IMT ATLANTIQUE

**Membre(s) du jury :**

M. CHRISTOPHE JEGO, UNIVERSITÉ DE BORDEAUX, Président

M. CHARLY POULLIAT, INP TOULOUSE, Membre

Mme MARIE LAURE BOUCHERET, INP TOULOUSE, Membre

Mme NATHALIE THOMAS, INP TOULOUSE, Membre

*L'aspiration à la vérité est plus précieuse*

*que l'assurance de sa possession.*

−Gotthold Lessing.

*Des chercheurs qui cherchent, on en trouve;*

*Des chercheurs qui trouvent, on en cherche.*

−Générale De Gaulle.

Vous me donnerez votre avis à la lecture de ce manuscrit.

*À mes Parents, Pascal et Isabelle, qui m'encouragèrent à faire cette thèse...*

# Remerciements

Je tiens à remercier mes directeurs de thèse Monsieur Charly Poulliat, Madame Nathalie Thomas et Madame Marie-Laure Boucheret pour leur aide et leurs conseils prodigués pendant la thèse. Plus particulièrement, je remercie Charly pour son abnégation envers ses thésards, sa disponibilité, ses éclairages et ses intuitions qui ont été déterminants dans l'avancement de ma thèse. Également, Nathalie qui fut disponible et pertinente lors des nombreuses relectures et points d'avancements. Enfin je remercie Marie-Laure pour son aide et son expertise apportée lors de la thèse. Mes remerciements vont également à mes encadrants du Centre National d'Étude Spatial (CNES) Monsieur Guy Lesthievent et Monsieur Emmanuel Bouisson.

Je remercie le CNES et l'Institut de Recherche en Informatique de Toulouse (IRIT) pour avoir financé cette thèse.

Je remercie le laboratoire Télécommunication Spatiales et Aéronautiques (TéSA) pour le cadre de travail formidable dont j'ai bénéficié pendant la thèse.

Je souhaite remercier Monsieur Christophe Jego pour avoir accepté de présider ma soutenance de thèse. Également, Monsieur Frédéric Guilloud et Monsieur Emmanuel Boutillon pour avoir rapporté ma thèse.

Enfin je souhaiterai remercier tous les membres du TéSA, Romain, Simonet, Barbara, Selma, Julien, Lorenzo, Sylvain, Adrien, Antoine, Raoul, Philippe, Serge, Bernard, Patrice, Jacques, Isabelle, Jean-Yves, Corinne... avec qui j'ai vécu trois merveilleuses années. L'environnement de travail au TéSA fut très agréable et les pauses animées du matin me manqueront. J'espère que j'aurais l'occasion de revenir de temps en temps au laboratoire. Pour finir je remercie mes parents qui m'ont soutenu dans le choix de m'orienter vers une thèse, mes cinq frères et soeurs qui étaient toujours curieux et préoccupés de l'avancement de mes travaux et Marianne de m'avoir supporté durant ces trois dernières années (et ce n'est que le début...).

# Résumé

Les *modulations à phase continue* (CPMs) sont des methodes de modulations robuste à la non-cohérence du canal de propagation. Dans un context spatial, les CPM sont utilisées dans la chaîne de transmission de télémesure de la fusée. Depuis les années 70, la modulation la plus usitée dans les systèmes de télémesures est la modulation CPFSK *continuous phase frequency shift keying* filtrée. Historiquement, ce type de modulation est concaténée avec un code Reed-Solomon (RS) afin d'améliorer le processus de décodage. Côté récepteur, les séquences CPM non-cohérentes sont démodulées par un détecteur Viterbi à sortie dure et un décodeur RS. Néanmoins, le gain du code RS n'est pas aussi satisfaisant que des techniques de codage moderne capables d'atteindre la limite de Shannon. Actualiser la chaîne de communication avec des codes atteignant la limite de Shannon tels que les codes en graphe creux, implique de remanier l'architecture du récepteur usuel pour un détecteur à sortie souple. Ainsi, on propose dans cette étude d' élaborer un détecteur treillis à sortie souple pour démoduler les séquences CPM non-cohérentes. Dans un deuxième temps, on concevra des schémas de pré-codages améliorant le comportement asymptotique du récepteur non-cohérent et dans une dernière étape on élabora des codes de parité à faible densité (LDPC) approchant la limite de Shannon.

# Abstract

*Continuous phase modulations* (CPM) are modulation methods robust to the non-coherency of propagation channels. In a space context, CPMs are used in the communication link between the rocket and the base stations. Since the 70's, the most popular telemetry modulation is the filtered continuous phase frequency shift keying (CPFSK). Traditionally, the CPFSK scheme is concatenated with a Reed-Solomon (RS) code to enhance the decoding process. At the receiver side, the non-coherent CPM sequences are demodulated through a hard Viterbi detector and a RS decoder. However, the RS's coding gain is no more satisfactory when directly compared to modern coding schemes enable to reach the Shannon limit. Updating the communication link to capacity achieving codes, as sparse graph codes, implies to redesign the receiver architecture to soft detector. In that respect, we propose in this study to design a trellis-based soft detector to demodulate non-coherent CPM sequences. In a second part, we will elaborate precoding schemes to improve the asymptotic behaviour of the non-coherent receiver and in a last step we will build low density parity check codes approaching the Shannon limit.

# Contents

# List of Figures

# List of Tables

# Table des sigles et acronymes

| | |
|---|---|
| **LLR** | Log-likelihood ratios |
| **TBR** | Trellis-based receiver |
| **BICM** | Bit-interleaved coded modulation |
| **EXIT** | Extrinsic information transfer |
| **NC** | Non-Coherent |
| **NB** | Non-Binary |
| **BER** | Bit error rate |
| **DE** | Density evolution |
| **SPA** | Sum-Product Algorithm |
| **BP** | Belief Propagation algorithm |
| **SISO** | Soft-Input Soft-Output |
| **ML** | Maximum Likelihood |
| **MAP** | Maximum A Posteriori |
| **RS** | Reed-Solomon |
| **CPFSK** | Continuous Phase Frequency Shift Keying |
| **FM** | Frequency Modulation |
| **PM** | Phase Modulation |
| **CCSDS** | Consultative Committee for Space Data Systems |
| **RCC** | Range Commanders Council |

**GMSK** Gaussian Minimum Sift Keying

**MSK** Minimum Sift Keying

**MSK** Frequency Sift Keying

**SOQPSK-TG** Shaped Offset Quadrature Phase Shift Keying-Telemetry Group

**SNR** Signal to Noise Ratio

**AWGN** Additive white Gaussian noise

# Introduction (French)

## Contexte

Le vol de la fusée est entièrement pré-programmé avant le lancement. Par conséquent, après la mise à feu des réacteurs, la trajectoire du lanceur ne peut plus être rectifiée. La seule contre-mesure encore disponible est l'auto-destruction afin d'empêcher la fusée de s'abîmer sur des zones sensibles. Aux États-Unis, cette contre-mesure suit un cahier des charges rigoureux certifié par le *45th Space Wing* de l'US Air Force. Afin de prendre cette décision irrémédiable, le lanceur doit informer le sol d'hypothétiques dysfonctionnements. Hormis ce scénario catastrophe, il est nécessaire que les opérateurs puissent suivre scrupuleusement les comportements de la fusée durant le vol. Ainsi, le *système de télémémesures* embarqué sur le lanceur transmet au sol de nombreuses données provenant des capteurs. Ensuite, les données sont enregistrées et traitées par les opérateurs sol. Au long de ce manuscrit, on se focalisera davantage sur le système de communication de la télémesure entre la fusée et le sol. Historiquement dans le domaine aéronautique et spatial, le système de télémémesures (TM) est une application requérant un faible débit d'environ 1Mbps en bande S. Ce débit décroît généralement avec l'éloignement de la fusée au cours du vol. Afin de garder le lien de communication ouvert pendant le vol, des stations de base sont positionnées le long de la trajectoire de la fusée. Ces stations enregistrent généralement plusieurs jeux de données identiques provenant du lanceur. En effet plusieurs antennes TM sont disposées sur la fusée avec une diversité de polarisation (Circulaire droite/gauche) ou de fréquence selon les cas. L'idée d'une telle architecture est de minimiser les risques d'une perte totale du lien de la communication entre le sol et la fusée. Ce risque est d'autant plus grand que le canal de propagation est souvent perturbé.

### Canal de propagation

Le canal de propagation entre le lanceur et le sol est un milieu très perturbé surtout pendant les phases de transition: allumage moteur, séparation des moteurs, de la coiffe... Ces phases génèrent des vibrations qui parcourent la fusée. Ce phénomène entraîne une dégradation du signal transmis. Un autre phénomène impactant le canal est les *effets de flamme*. Ces effets apparaissent lorsque le signal passe derrière les réacteurs de la fusée. Ce phénomène provoque une dérive de la phase voir même des sauts de la phase pendant la transmission. On appelle canal *non-cohérent* un canal de propagation entrainant une rotation de phase (par opposition au canal cohérent). Ces phénomènes accumulés peuvent conduire à une perte totale du lien de communication. Par conséquent, les méthodes de modulation utilisées dans le système de télémesures doivent être suffisamment robustes à de telles contraintes.

### Modulation à phase continue

Historiquement les modulations adoptées pour la télémesure sont des modulations de fréquence (FM) et/ou de phase (PM). Elles sont plus connues sous le nom de *modulation à phase continue*. Cette classe de modulations est robuste aux dérives de phase et aux distorsions non-linéaires générées par l'amplificateur lorsqu'il est poussé à sa puissance maximale de sortie. Depuis les années 70, la modulation la plus usitée dans les systèmes de télémesures est la modulation par impulsions et codage PCM/FM aussi appelée la *continuous phase frequency shift keying* (CPFSK) filtrée. Cependant la multiplication des capteurs, entraînant un besoin pressant pour le haut débit, rend ce type de modulation de plus en plus obsolète. De plus, les détecteurs CPFSK des standards de télémesure ne sont pas adaptés aux codes correcteurs à entrées souples ce qui limite le gain de codage atteignable par rapport aux techniques de codage moderne. Récemment, le *Range Commanders Council* (RCC) a proposé une série de standards pour la télémesure afin de favoriser la compatibilité des équipements en transmission, réception et traitement du signal dans le rapport *IRIG Standard 106-17* [Iri]. Ce rapport suggère de remplacer la CPFSK par d'autres modulations comme la *shaped offset quadrature*

*phase shift keying - telemetry group* (SOQPSK-TG). De même le *Consultative Committee for Space Data Systems* (CCSDS) développa une série de recommandations sur des techniques de modulation pour optimiser les largeurs de bande du standard de la télémesure pour des missions nécessitant du haut débit tel que la *Gaussian Minimum Shift Keying* (GMSK) précodée ou l'OQPSK filtrée. Un désavantage majeur des modulations OQPSK est que l'information utile est exclusivement comprise dans la phase donc si un déphasage d'une quantité inconnue se produit il est difficile de démoduler de manière non-cohérente. Généralement le déphasage est alors évalué récursivement pour permettre la démodulation. Dans le cas d'une modulation CPM classique, les bits d'information sont encodés dans la trajectoire de phase et décodés par un récepteur dit non-cohérent.

### Récepteur non-cohérent

Un canal de transmission caractérisé par un déphasage est appelé non-cohérent. Pour démoduler des séquences de manière non-cohérentes, les stations sol doivent être équipées de récepteurs dit non-cohérents. Certains de ces récepteurs fonctionnent sur le même principe que les détecteurs cohérents excepté que l'information de phase est inaccessible. Les récepteurs non-cohérents se distinguent des cohérents de part leurs métriques de branches. Les métriques non-cohérentes sont calculées en moyennant les métriques cohérentes sur la phase. Deux approches existent pour démoduler une séquence non-cohérente: soit par bloc soit par treillis. La détection par bloc est plus robuste aux sauts de phases que le treillis mais est moins performante. [RD99][VCT10] implémentèrent le récepteur par bloc et [DS90][CFR00] celui par treillis. Ces détecteurs sont évidemment moins performants que le récepteur cohérent mais ils peuvent s'approcher des résultats du cohérent en augmentant fortement la fenêtre d'observation. Historiquement dans les standards de la télémesure, le processus de décodage est amélioré par la concaténation de la CPFSK avec un code Reed-Solomon (RS) [RS60]. Cependant, le gain de codage d'un code RS n'est pas aussi satisfaisant que ceux utilisés dans les techniques de codages modernes.

### Schéma de codage

Berrou et al. [BG96] révolutionnèrent le domaine du codage en 1993. Ces derniers proposèrent un schéma de codage fonctionnant proche de la limite de Shannon. Le principe est de concaténer deux codes convolutifs séparés par un entrelaçeur et d'effectuer un décodage itératif joint. Le décodage joint fonctionne en connectant la sortie d'une premier code avec l'entrée du second et vise versa. Cette classe de code est appelée *Turbo-Codes*. Les Turbo-Codes sont bien adaptés aux modulations sans mémoire cependant ils deviennent redondants lorsqu'ils sont associés aux modulations CPM. En effet, il est possible d'effectuer un décodage itératif joint directement entre le code convolutif et le démodulateur CPM. Une autre classe de codes fonctionnant proche de la capacité de Shannon est la classe des codes de parité à faible densité appelés LDPC. Ils sont aussi performants que les turbo-codes tout en ayant une faible complexité de décodage. De plus, des études récentes ont montré que cette classe de codes est bien adaptée aux CPM [Ben15]. Ainsi on privilégiera, au long de ce manuscrit, le système composé de la concaténation série d'un code LDPC avec une modulation CPM. En revenant à notre contexte, ce schéma de codage devra convenir à la démodulation non-cohérente. Par ailleurs, il serait intéressant d'avoir un schéma de codage adapté à la fois au régime cohérent et non-cohérent ainsi il ne sera pas utile de modifier le codage au cours de la transmission suivant la cohérence du canal. Cependant, cela nécessite d'avoir des récepteurs avec des fonctions de transfert d'information mutuelle proches dans les deux régimes, ce qui est rarement le cas. Un moyen de changer la fonction de transfert sans modifier les performances du récepteur est d'utiliser un schéma de pré-codage.

### Schéma de pré-codage

Le pré-codage, associé à la CPM en régime cohérent, fut étudié dans de précédents travaux. [Ben+07]; [Per+10] proposèrent un pré-codage permettant d'atteindre le débit maximal sans itérer entre les schémas de codage. L'idée consiste essentiellement à changer le mapping entre les symboles d'information et les trajectoires de la CPM. Ainsi le débit maximal est

atteint sans itérer, et plus encore, le pré-codage change la forme de la fonction de transfert du démodulateur CPM sans perte de performances. Cependant, cette méthode est efficace uniquement pour les CPM binaires et n'est pas directement applicable au récepteur non-cohérent. Par conséquent, deux axes d'amélioration sont possibles:

1. élaborer une classe de pré-codages applicables quelque soit la cohérence du canal
2. sélectionner ceux de cette classe générant une fonction de transfert de l'information mutuelle proche dans les deux régimes.

## Structure du manuscrit et principales contributions

Dans ce manuscrit, on souhaite élaborer un détecteur CPM treillis basé sur une démodulation de type MAP en régime non-cohérent avec de bons gains de codage. L'application principale est le lien de communication entre le système de télémesures de la fusée et le sol. L'organisation de cette thèse et les principales contributions sont résumées ci-dessous.

**Chapitre 1:** Ce chapitre est un chapitre d'introduction à la CPM [Pro][AAS13] et aux récepteurs cohérent et non-cohérent qui lui sont associés. Dans un premier temps, on passe en revue tous les paramètres du signal CPM et leurs effets sur le spectre. Puis on présente le treillis originellement associé à la CPM [ARS81a][ARS81b] et les améliorations apportées par la décomposition de Rimoldi [Rim88]. Dans une dernière étape, on décrit les différentes méthodes existantes pour démoduler la CPM en régime cohérent et non-cohérent. Dans cette thèse, les contributions sont principalement sur la détection non-cohérente, ainsi on ne présente que le détecteur MAP treillis en régime cohérent. En régime non-cohérent, on aborde deux types d'approches pour démoduler des séquences CPM: soit par bloc soit par treillis. Les détecteurs treillis non-cohérents furent élaborés au travers de deux références majeures: [DS90] pour le cas ML et [CFR00] pour le critère MAP. La détection par bloc, pour sa part, fut formulée dans [RD99] et [VCT10]. Les résultats de ce chapitre ont été publiés dans:

- Conférence Nationale: *Piat-Durozoi, C. U.*, Poulliat, C., Thomas, N., Boucheret, M. L.,

Bouisson, E., & Lesthievent, G. (2017, Sept). Détection non-cohérente souple par bloc ou à mémoire des CPM. In Groupe de recherche sur le traitement du signal (GRETSI), Juan-Les-Pins France.

- Conférence Internationale: *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G. (2017, March). Multisymbol with memory noncoherent detection of CPFSK. In Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on (pp. 3794-3798). IEEE. New-Orleans USA.

**Chapitre 2:** Dans ce chapitre, on étudie le récepteur treillis non-cohérent. Précédemment, on a mis en lumière un désaccord entre les deux références majeures [DS90] et [CFR00] au sujet de l'espace d'état utilisé pour démoduler des séquences CPM non-cohérentes. Il apparaît que la cardinalité des états proposée pour la détection MAP [CFR00] est plus importante que celle introduite en [DS90] pour la détection ML. En effet, l'auteur en [CFR00] ajouta à l'espace d'état fourni en [DS90] l'information de phase générant un modèle étendu. Ainsi, on étudie dans ce chapitre le bénéfice et/ou la nécessité de cet écart en redérivant complètement les équations de la détection non-cohérente basée sur l'algorithme BCJR. On montre finalement que l'espace d'état étendu peut être réduit à la même formulation que [DS90] sans perte de performance. Cependant le model à espace d'état redondant est utlisable par les deux types de récepteurs (cohérent et non-cohérent). Par la suite, on propose de rectifier la formulation du détecteur non-cohérent MAP proposé en [CFR00] avec la formulation de l'espace d'état optimal donné en [DS90]. Les contributions de ce chapitre ont été publiées dans:

- Journal: *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G. Minimal state non-coherent symbol MAP detection of Continuous-Phase-Modulations. IEEE Communication Letter, 2018.

**Chapitre 3:** Dans ce chapitre, on se propose d'ajuster le comportement asymptotique de la CPM avec les contraintes du canal afin de maximiser les gains de codage. Le comportement asymptotique de la CPM est différent suivant que le régime soit cohérent ou non-cohérent.

En effet, la plupart des CPM utilisées présentent une fonction de transfert d'information mutuelle convergeant vers le point $(1, 1)$ en cohérent alors que ce n'est plus le cas en régime non-cohérent. Ainsi, les schémas de codage classiques (convolutif, LDPC) génèrent une pénalité de capacité et/ou un palier d'erreur en régime non-cohérent. Pour répondre à cette problématique, un pré-codage est élaboré. Ce dernier force la convergence de la fonction de transfert vers le point de coordonnées $(1, 1)$. Le pré-codage consiste essentiellement à changer le mapping entre les symboles d'information et les trajectoires de la CPM. Ce *pré-traitement* préserve ***le débit maximal atteignable*** tout en permettant un décodage itératif efficace. La contribution de ce chapitre a été publiée dans:

- Conférence Internationale: *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G (2018, October). Precoding for Non-coherent detection of continuous phase modulations. MILCOM, 2018 in process, Los-Angeles California USA.
- Journal: *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G. A novel nonbinary precoding for continuous phase modulations. En préparation.

**Chapitre 4:** Dans ce chapitre, on élabore des schémas de codage concaténés en série avec une modulation CPM qui donnent de bon gains pour les deux régimes. On se concentre principalement sur les codes de parité à faible densité (LDPC) sur GF(q). On a montré aux chapitres 2 et 3 que la fonction de transfert de l'information mutuelle à la sortie du détecteur non-cohérent dépendait de la cohérence du canal. En effet, dans le cas cohérent, la plupart des CPM ont une information mutuelle atteignant le point $(1, 1)$ alors que ce n'est plus le cas en régime non-cohérent si aucun pré-codage n'est considéré. Par conséquent, les profils et la condition de stabilité des codes en graphes creux seront différents d'un régime à l'autre. En fait, il apparaît qu'un schéma de codage élaboré pour le cas cohérent ne peut être utilisé en régime non-cohérent puisque les profils ne remplissent pas la condition de stabilité. Au contraire, un code conçu pour le non-cohérent est toujours stable en régime cohérent mais au

prix d'une perte de performance. En fait, pour avoir un schéma de codage opérant pour les deux canaux, il semble raisonnable d'élaborer en premier lieu un schéma de codage pour le cas le plus contraignant puis d'évaluer ces performances dans le cas le moins contraint. La contribution de ce chapitre a été publiée dans:

- Conférence Internationale: *Piat-Durozoi, C. U.*, Poulliat, C., Thomas, N., Boucheret, M. L., & Lesthievent, G. (2017, June). On sparse graph coding for coherent and non-coherent demodulation. In Information Theory (ISIT), 2017 IEEE International Symposium on (pp. 2905-2909). IEEE, Aachen Germany.

# Introduction

## Context

Flight phases of the launcher are entirely preprogrammed before take off. Consequently, after the engine-ignition, the rocket trajectory can not be rectified. The only countermeasure available is the self-destruction to prevent the rocket to crash over sensitive areas. In USA this countermeasure follows thorough and strict specifications certified by the *45th Space Wing* of the US Air Force. To take such decision, the launcher must inform the ground station of hypothetical failures. Apart from such a worst case scenario, it is important to the ground operators of having a full tracking of the rocket behaviour during the different flight phases. In that respect, the *Telemetry System* on-board the launcher transmits a wide variety of data coming from its sensors to the ground stations. Afterwards, the data are recorded and processed on ground by the operators. Along this manuscript, we focus more specifically on the communication unit between the rocket and the ground station. Traditionally in space/aeronautical domain, the telemetry (TM) is an application requiring low data rate, around 1Mbps in S-band. This rate usually decreases over the course of the rocket. To keep the communication link open over the flight, ground stations are positioned along the path of the launcher. Those stations may acquire several identical sets of data coming from the launcher. Indeed quite a few TM antennas are set in opposed position on the rocket with a polarization diversity (Right/Left Hand Circular ) or frequency diversity. The idea is to reduce the risk of a total communication loss by enabling multiple sights between the ground station and the rocket. This risk is significant because of the propagation channel features.

### Propagation channel

The propagation channel between the launcher and the ground is very disturbed especially during the transitional phases as: engine-ignition, boosters separation, rocket's nose separation... Actually, those phases generate a lot of vibration spreading over the launcher. This phenomenon leads to impairments of the rocket's signal during the transmission. Another phenomenon impacting the channel is called the *Flame Effects*. Those effects appear when the transmitted signal goes on behind the rocket's engine. As a result, the signal undergoes a phase shift or even phase hopping during the transmission. Usually, transmission channels incurring unknown phase shift are called non-coherent (by opposition to coherent channel). This set of phenomenons may lead to a total loss of the communication link. Consequently, modulations methods used in such transmissions must be sufficiently robust to deal with such constraints.

### Continuous phase modulation

Traditional modulation methods for telemetry are frequency modulation (FM) and phase modulation (PM). Among them, *continuous phase modulation* (CPM) are a class of broadly used modulation. This class of modulations is robust to channel phase shift and insensitive to nonlinear distortions generated by amplifiers at their maximal output power. Since 70's, the most popular telemetry modulation is the pulse code modulation PCM/FM also called filtered continuous phase frequency shift keying (CPFSK). However the increasing need for high data rates, generated by sensors installed on-board launchers, is less and less adapted to this modulation. Moreover the CPFSK detector does not suite well to soft-input error correcting code which limits the attainable coding gain with respect to modern coding scheme. Recently, the Range Commanders Council (RCC) has provided a telemetry standards to foster compatibility of transmitting, receiving, and signal processing equipment through the report *IRIG Standard 106-17* [Iri]. This report suggests to replace the CPFSK by other modulations providing the required bandwidth efficiency for higher bit rate such as shaped

offset quadrature phase shift keying (SOQPSK-TG). As well the Consultative Committee for Space Data Systems (CCSDS) developed in [Ccs] a series of recommendations for standard bandwidth efficient modulation techniques applicable to high rate missions, such as a precoded Gaussian Minimum Shift Keying (GMSK) or Filtered OQPSK. One drawback of OQPSK modulations is that the information bits are encoded in the phase so, if an unknown phase deviation occurs, it is complicated to recover the information bits without a recursive phase estimation. Whereas in classical CPM modulations, the information bits are encoded in the phase trajectory and decoded by a non-coherent receiver.

### Non-coherent receivers

Transmission channels undergoing unknown phase shift are called non-coherent. To demodulate non-coherent sequences, base stations must be equipped of non-coherent receivers. Some of these receivers work using the same principles as classical coherent receivers except that the information phase is not processed. What mathematically differentiates the coherent receivers from the non-coherent ones are the metrics used for enabling MAP/ML detection. The non-coherent metrics are evaluated by averaging the coherent ones over the signal's phase. Two main approaches exist to detect a non-coherent sequence based either on block or on trellis. Block detection is more robust to phase hopping than trellis-based detector but is less effective. [RD99][VCT10] implemented the receiver by block while [DS90][CFR00] focus on the non-coherent trellis-based detector. Such detectors are less effective than coherent ones but they might reach the same performance by strongly increasing their observation length. In telemetry standards, the decoding process is traditionally enhanced by concatenating the CPFSK scheme with a Reed-Solomon (RS) code [RS60]. However, the RS codes gain is no more satisfactory when directly compared to modern coding scheme.

### Coding scheme

In 1993, the field of channel coding was revolutionized by Berrou et al. [BG96]. They proposed a coding scheme performing close to the Shannon limit. The principle is to concatenate two convolutional codes separated by an interleaver and to perform a joint iterative decoding. The joint decoding works by connecting the so called soft output of the first convolutional code to the soft input of the second one and vise versa. This class of codes were named *Turbo-Codes*. Turbo-Codes are well suited to modulation without memory yet it seems redundant when associated to CPM modulation. Indeed, it is possible to perform the joint iterative decoding directly between the CPM demodulator and an inner convolutional code. Another class of codes, performing close to the Shannon capacity, are the low density parity check codes (LDPC). They exhibit competitive performance against turbo-codes while having low decoding complexity. Moreover, recent studies have shown that this class of codes is well suited to CPM [Ben15]. In that respect, system composed of an inner LDPC code serially concatenated with a CPM schemes will be favoured along this manuscript. Back to our context, the coding scheme must be adapted to non-coherent demodulation. Moreover, it would be interesting to have a coding scheme which suits to both coherent and non-coherent CPM receivers so that whatever the channel coherency, we would not be force to change the coding scheme during the transmission. However, this does require that both receivers feature close mutual information transfer functions which is rarely the case. A mean to change the transfer function, without modifying the receiver performance, is to use an adequate precoding scheme.

### Precoding scheme

Some previous works on precoding were carried out on CPM in coherent regime. [Ben+07][Per+10] proposed a precoding method to reach the maximal achievable rate without iterative decoding. Their idea was essentially to change the mapping between the information symbols and the CPM waveforms. As a result, the maximal achievable rate is reached without iterat-

ing but, above all, it changes CPM demodulator transfer function without performance loss. However the method is efficient only for binary CPM and it is not directly applicable to non-coherent receiver. Consequently, two axis must be investigated: 1. to design a class of precoding schemes applicable whatever the channel coherency is and 2. to find a family belonging to this class generating mutual information transfer function with close behaviours for both regimes.

## Dissertation outline and main contributions

In this manuscript, we aim to design a trellis-based detector to demodulate, with a MAP criterium, the non-coherent CPM transmitted sequences with good coding gain. The main application is the communication link between the launcher's telemetry unit and the ground stations. The dissertation outline and the principal contributions are summarized hereunder.

**Chapter 1:** This chapter gives an introduction to CPM [Pro][AAS13] and its associated receivers in coherent and non-coherent regimes. First, we review all the parameters of the CPM signal and their effects over the spectrum. Then we revisit the original trellis representation associated to the CPM [ARS81a][ARS81b] and the improvements achieved by the decomposition proposed by Rimoldi [Rim88]. In a third step, we describe the various CPM demodulation's methods existing in the literature for both regimes. In the thesis, our contributions are mainly on non-coherent detection. In non-coherent regime, we will address two approaches to detect a sequence based by block and by trellis. Non-coherent trellis based detector were designed in two major references for respectively a ML [DS90] and MAP criterium [CFR00]. The block detection, for its part, has been formulated by [RD99] and [VCT10]. The contribution of this chapter was published in:

- National Conference Paper: *Piat-Durozoi, C. U.*, Poulliat, C., Thomas, N., Boucheret, M. L., Bouisson, E., & Lesthievent, G. (2017, Sept). Détection non-cohérente souple par bloc ou à mémoire des CPM. In Groupe de recherche sur le traitement du signal

(GRETSI), Juan-Les-Pins France.

- Conference Paper: *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G. (2017, March). Multisymbol with memory noncoherent detection of CPFSK. In Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on (pp. 3794-3798). IEEE. New-Orleans USA.

**Chapter 2:** In this chapter, we mainly focus on non-coherent trellis based receiver. The introduction chapter has brought to light a disagreement between the two major trellis-based detector references [DS90] and [CFR00] about the state space employed to demodulate non-coherent CPM sequences. The state cardinality proposed for the MAP detection in [CFR00] is greater than the one presented in [DS90] for the ML detection. Indeed, the author in [CFR00] added to the state space provided in [DS90] the CPM's information phase, generating an extended state space. In that respect, we investigate in this chapter the benefit and/or the necessity of this discrepancy by rederiving completely the equations of the non-coherent detection based on the BCJR algorithm. As a result, it seems that the extended state space should reduced to the formulation given in [DS90] without performance loss. Yet this *redundant* state space model offers the benefit to suit to both regimes enabling an unified framework to derive both type of receivers. Afterwards, we modify the non-coherent symbol/MAP detector proposed in [CFR00] with the optimal state space formulation given in [DS90]. The contributions of this chapter were published in:

- Journal Paper : *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G. Minimal state non-coherent symbol MAP detection of Continuous-Phase-Modulations. IEEE Communication Letter, 2018.

**Chapter 3:** In this chapter, we adjust the asymptotic behaviour of CPM scheme to the channel constraints in order to maximize the asymptotic coding gain. It obviously appears that the asymptotic behaviour of CPM varies with respect to the channel coherency. Indeed,

most CPM in use in standards have mutual information transfer functions converging the point $(1,1)$ in coherent channel, whereas it is almost never the case in non-coherent regime. As a result, classical coding schemes (based on convolutional, LDPC codes) generate a capacity penalty and/or an error floor in the asymptotic regime leading to non-convergent systems in non-coherent regime. To solve this issue, a precoding is designed to force the convergence of the mutual information transfer function to point $(1,1)$. The precoding essentially consists in changing the mapping between the information symbols and the CPM waveforms. This preprocessing preserves ***the information rate*** while enabling efficient iterative decoding. The contributions of this chapter is published in:

- Conference Paper: *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G. (2018, October). Precoding for Non-coherent detection of continuous phase modulations. MILCOM, 2018 in process, Los-Angeles California USA.
- Journal: *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G. A novel nonbinary precoding for continuous phase modulations. In preparation .

**Chapter 4:** In this chapter we design good coding schemes, serially concatenated with CPM, performing well for both the coherent and the non-coherent regimes. We mainly focused on sparse graph codes such as low-density parity-check (LDPC) codes over GF(2) and their extension over GF(q). We have pointed out, along chapter 2 and 3 that the mutual information transfer function at the output of the CPM detector varies depending on the considered regime. Indeed, in the coherent case, most of the CPM have mutual information transfer functions reaching the point $(1,1)$ whereas it is no longer the case in non-coherent regime if no specific precoding is used. Consequently the profiles and the stability condition of the sparse graph code will be different from one regime to another. Actually, it seems that a coding scheme designed for the coherent case cannot be used for the non coherent setting, since the resulting profiles cannot be stable. On the contrary, a code designed for the non

coherent case is always stable under coherent decoding. In that respect, to have good coding schemes operating in both regimes, it seems reasonable to design first codes for the most binding case and then assess the performance in the less constraining case. Extension of this work to the precoded case is also investigate. The contributions of this chapter is published in:

- Conference Paper: *Piat-Durozoi, C. U.*, Poulliat, C., Thomas, N., Boucheret, M. L., & Lesthievent, G. (2017, June). On sparse graph coding for coherent and noncoherent demodulation. In Information Theory (ISIT), 2017 IEEE International Symposium on (pp. 2905-2909). IEEE, Aachen Germany.

## List of publications

### Journal paper

- *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G. Minimal state non-coherent symbol MAP detection of Continuous-Phase-Modulations. IEEE Communication Letter, 2018.

- Journal: *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G. A novel nonbinary precoding for continuous phase modulations. In preparation.

### International conference papers

- *Piat-Durozoi, C. U.*, Poulliat, C., Thomas, N., Boucheret, M. L., & Lesthievent, G. (2017, June). Multisymbol with memory noncoherent detection of CPFSK. In Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on (pp. 3794-3798). IEEE. New-Orleans USA.

- *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G. (2017, March). On sparse graph coding for coherent and noncoherent demodulation. In Information Theory (ISIT), 2017 IEEE International Symposium on (pp. 2905-2909). IEEE, Aachen Germany.

- *Piat-Durozoi, C. U.*, Poulliat, C., Boucheret, M. L., Thomas, N., Bouisson, E., & Lesthievent, G. (2018, October). Precoding for Non-coherent detection of continuous phase modulations. MILCOM, 2018 in process, Los-Angeles California USA.

### National conference paper

- *Piat-Durozoi, C. U.*, Poulliat, C., Thomas, N., Boucheret, M. L., Bouisson, E., & Lesthievent, G. (2017, Sept). Détection non-cohérente souple par bloc ou à mémoire des CPM. In Groupe de recherche sur le traitement du signal (GRETSI), Juan-Les-Pins France.

Chapter 1

# Continuous Phase Modulation

**Sommaire**



## 1.1 Résumé


Ce chapitre est un chapitre d'introduction aux modulations à phase continue et aux récepteurs qui leur sont associés. Les modulations à phase continue sont apparues dans le début des années 70 sous le nom de *minimum shift keying* (MSK) ou de *frequency shift keying* (FSK).

Par la suite, ce type de modulation fut formalisé par Tor Aulin and Carl-Erik Sundberg et rebaptisé pour la première fois *modulation à phase continue* dans [ARS81a],[ARS81b]. Les CPM devinrent vite populaire en communication numérique de par leur bonne efficacité et occupation spectrale. Elles sont d'ailleurs très utilisées aujourd'hui dans le domaine spatial [SG13], aéronautique [Geo00] et militaire [CFC10].

Du point de vue récepteur, les détecteurs CPM optimaux s'appuient sur un treillis. Le treillis originel, tiré des travaux de Tor Aulin, fut amélioré par Rimoldi [Rim88]. Cette nouvelle représentation permet de démoduler efficacement des séquences CPM cohérentes à partir de récepteurs basés sur des algorithmes de type Viterbi [Vit67] et BCJR [Bah+74].

Dans le cas non-cohérent, deux types de détection sont possibles: soit par bloc soit par treillis. La détection par bloc est plus robuste au changement de phase induit par le canal de propagation que par treillis mais est moins performante. [RD99] et [VCT10] implémentèrent respectivement, la détection souple et dure par bloc. La démarche consiste à comparer les échantillons reçus sur une fenêtre donnée avec toutes les combinaisons possibles. Une première implémentation du récepteur par treillis basée sur l'algorithme Viterbi fut proposée par [DS90]. Peu après, une détection par MAP basée sur l'algorithme de type BCJR fut développée par [DS90], à partir d'un espace d'état différent de celui instancié en [DS90]. Ces derniers détecteurs ont montré qu'ils pouvaient atteindre les performances de détecteurs cohérents lorsqu'on augmentait fortement la taille de la fenêtre d'observation.

## 1.2 Introduction

This chapter introduces continuous phase modulations (CPM) and their associated receivers. This modulation was invented in early 70's under the name of minimum shift keying (MSK) or frequency shift keying (FSK). Thereafter, their generalization has been formalized by Tor Aulin and Carl-Erik Sundberg. The term of *continuous phase modulation* was first introduced in their seminal paper [ARS81a],[ARS81b]. After its invention, CPM has become popular in digital communication because of its interesting characteristics regarding the power efficiency

and the spectral occupancy. Today, they are widely used in space [SG13], aeronautical [Geo00] and military [CFC10] communications, but for low rate applications.

Optimal CPM receivers are trellis based detectors. The original trellis was not fully effective to demodulate coherent CPM sequences. Afterwards, based on a well-thought decomposition of CPM modulation scheme, Rimoldi [Rim88] implemented an efficient trellis based detector which can be applied to coherent MAP sequences (Viterbi type algorithms, VA) or symbols/bits (BCJR type algorithm [Bah+74]) detection. The detection of partial CPM can quickly become complex according to the size of the CPM memory. Thereby, [Kal89] and later on [CR97], made use of Laurent/Mengali decomposition [Lau86]/[MM95] to reduce the trellis complexity for a maximum likelyhood (ML) and respectively MAP demodulation. Other method as per-survivor [MKF14] was implemented afterward to reduce the complexity of CPM soft detection.

In non-coherent regime, two main approaches exist to demodulate/detect a sequence based either by block or by trellis. Block detection can work for any value of modulation index and is robust to fast channel phase shifts when blocks are taken independently. Trellis based detection, for its part, is less robust to channel phase deviation but reaches better performance than block detection. [RD99] and [VCT10] implemented the receiver by block for a hard and respectively soft-decision demodulation. The process can be summarized as follows: the block receiver does the correlation between the block of received symbols and all existing combinations of the same block length. The condition required to use this method is the absence of phase shift between symbols belonging to the same block since the phase continuity is exploited within the blocks. Thus the channel coherence time must be at most of the order of the block size. A first trellis-based approach based on Viterbi algorithm was presented in [DS90]. Thereafter [CFR00] proposed a symbol MAP decoding algorithm similar to the well-known BCJR but based on a state space model different from [DS90]. Based on this literature, it seems that non-coherent trellis-based receivers might reach the performance of coherent detectors when the observation length increases.

## 1.3 CPM signal

CPM [Pro],[AAS13] is a particular modulation having a constant envelop waveform that means the transmitted carrier power is constant. This special feature leads to excellent power efficiency. A second important aspect of CPM is the phase continuity yielding better spectral occupancy. The phase of a CPM signal, for a given symbol interval, is determined by the cumulative phase of previous transmitted symbols known as the phase memory. Hence the decision taken on the current symbol must take into account the previous ones.

Two types of CPM can be distinguished, partial response CPM which has a memory strictly greater than one symbol and full response CPM whose memory is exactly equal to one. Another important element of CPM is the modulation index which could restrain, in a particular case, the set of the phase memory to a finite set. A well-known full response CPM is the continuous phase frequency shift keying (CPFSK) described by a rectangular phase response. This modulation suits well to low rate data transmission applications such as telemetry launchers (Ariane, Vega, Soyuz...). Partial response CPM are also used in several communication systems. For instance, Gaussian minimum shift keying (GMSK) with 3 symbols in memory and a gaussian pulse bandwidth of 0.3 ($BT$) is the modulation specified for GSM standard. Other types of applications concern aeronautical/satellite or military/tactical communications.

### 1.3.1 Representation

Let us consider a sequence of $N_s$ independent and identically distributed symbols $a_0^{N_s-1} = [a_0, ..., a_{N_s-1}]$ belonging to the $M$-ary alphabet $\{\pm 1, \pm 3, ..., \pm(M-1)\}$ if $M$ is even (respectively $\{\pm 0, \pm 2, ..., \pm(M-1)\}$ if odd) with $M$ the modulation order. Each symbol has the same probability $1/M$ and $M$ is assumed to be a power of 2. The CPM complex baseband representation of this sequence is given as follows:

$$s(t, a_0^{N_s-1}) = \sqrt{\frac{Es}{T}} e^{j2\pi h \sum_{n=0}^{N_s-1} a_n q(t-nT)} e^{j\xi_0} \tag{1.1}$$

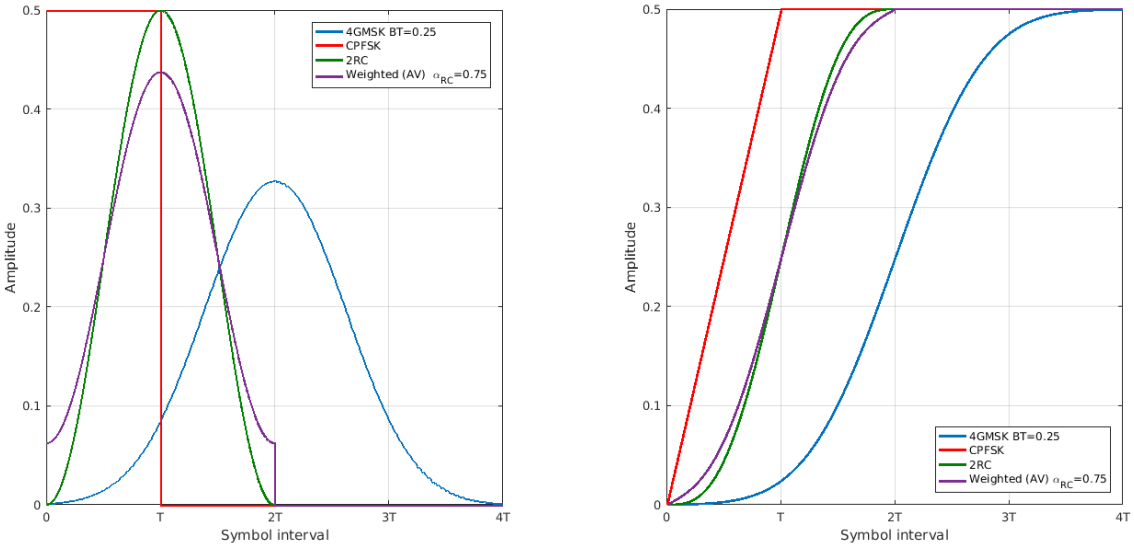
In the expression above, $T$ is the symbol period, $Es$ is the symbol energy, $\xi_0$ is the initial phase and $h$ is the modulation index ($h = P/Q$ , with $P$ and $Q$ are relatively prime). the function $q(t)$ is the phase response that satisfies:

$$q(t) = \begin{cases} 0, & t \leq 0 \\ \int_0^t g(u)du, & 0 < t \leq LT \\ \frac{1}{2}, & t > LT \end{cases} \tag{1.2}$$

$L$ is the CPM memory and $g(u)$ is the frequency pulse depending on the kind of used CPM. Various CPM schemes are provided in Table 1.1 according to their frequency pulse and some other parameters. Then their frequency pulse and phase response have been reproduced in Fig.1.1 (a.) and (b.) respectively for a set of parameters given in Table 1.2.

Table 1.1: CPM schemes and their frequency pulse

| CPM | Frequency pulse $g(t)$ | Parameters |
|---|---|---|
| LGMSK | $g(t) = \frac{1}{2T} \cdot \left[ \mathcal{G}\left(2\pi B \frac{t-\frac{T}{2}}{\sqrt{log(2)}}\right) - \mathcal{G}\left(2\pi B \frac{t+\frac{T}{2}}{\sqrt{log(2)}}\right) \right]$ <br> $\mathcal{G}(t) = \int_t^\infty \frac{1}{\sqrt{2\pi}} e^{\frac{-v^2}{2}} dv, \quad t \in \mathbb{R}$ | $0 \leq BT \leq 1$ <br> $L$ |
| LRC | $g(t) = \frac{1}{2LT}$, $t \in [0, LT)$ (also noted $g_{LRC}$) | $L$ |
| LREC | $g(t) = \frac{1}{2LT} \cdot \left[1 - cos(\frac{2\pi t}{LT})\right]$, $t \in [0, LT)$ (also noted $g_{LREC}$) | $L$ |
| Weighted average CPM [SG13] | $g(t) = \alpha_{RC} g_{2RC}(t) + (1 - \alpha_{RC}) g_{2REC}(t)$, $t \in [0, 2T)$ | $0 \leq \alpha_{RC} \leq 1$ |



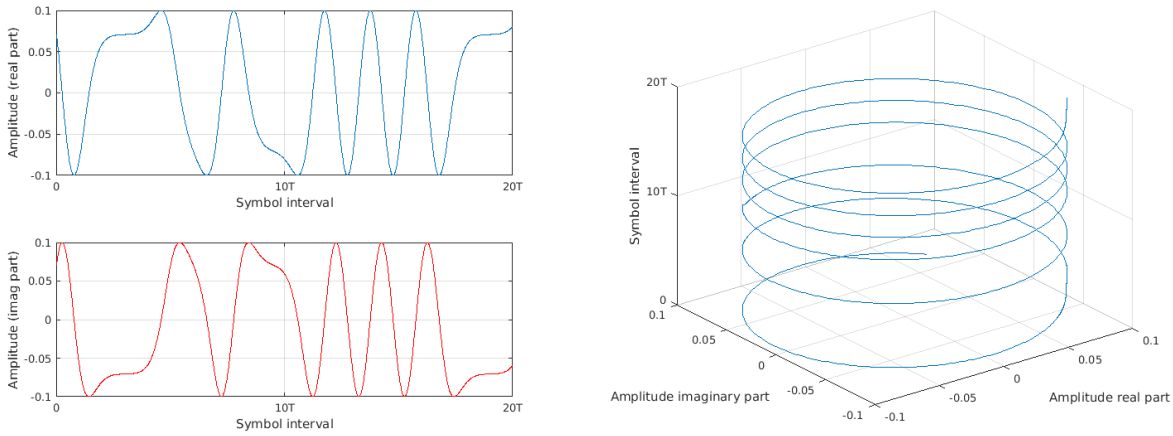

(a.) (b.)

Figure 1.1: (a.) Frequency pulse (b.) Phase response.

Table 1.2: Set of parameters of Fig.1.1

| CPM | Modulation order ($M$) | Modulation index ($h$) |
|---|---|---|
| GMSK | binary | 1/2 |
| 2RC | quaternary | 1/4 |
| 1REC (CPFSK) | quaternary | 5/7 |
| Weighted (AV) [SG13] | quaternary | 1/3 |

Fig.1.2, (a.) and (b.) illustrate respectively two important aspects of the CPM: the phase continuity between symbols and the constant envelop. The phase continuity yields excellent spectral occupancy. This latter can be shaped by CPM parameters as we would show in section 1.3.2. The constant envelop, for its part, is desirable when considering nonlinear/fading channels or communication tools generating perturbations (as nonlinearities). For instance in satellite communication, amplifiers on board satellite (TWTA) generate nonlinearities when reaching the saturation point. This effect occurs as well for unmanned aerial vehicle (UAV) communications. Then to prevent this unwanted effect, amplifiers are not used at their maximal power. However transmitted carrier with constant power (as CPM) are insensitive to nonlinear distortions which permits to push amplifiers at their maximal output power. For its numerous qualities, CPM are used in various contexts as in satellite communications [SG13], deep space communications [Sim05], Bluetooh data transmission [Lam+03], aeronautical telemetry systems [Geo00], automatic identification system (AIS) [Sco+10], GSM mobile communications [MPFBH92], tactical communications [CFC10], machine to machine communications [DL12] and optical communications [Det11].




(a.) (b.)

Figure 1.2: Binary 2GMSK $h = 1/2$, $BT = 0.25$ (a.) Amplitude (b.) Envelop.

### 1.3.2 Effects of some parameters

#### 1.3.2.1 Modulation index $h$

The modulation index $h$ is usually chosen as a rational number and smaller than one. In the literature, $h$ is frequently written as the ratio between two relatively prime numbers $P$ and $Q$ (such as $h = P/Q$) where $Q$ is associated to the cardinality of the finite set of phases taken by the CPM scheme in Rimodi's representation. If irrational modulation index is assumed, the set of phases becomes infinite. Such schemes are avoided here because of the high complexity induced by their associated receivers.

#### 1.3.2.2 CPM memory $L$

The CPM memory $L$ is the number of previous symbols required to determine the signal waveform of the current symbol. If $L$ is exactly equal to one, the CPM is called full response and when $L$ is strictly greater than one it is called partial response. The CPM memory specifies also the support of the frequency pulse (see Table 1.1). Large $L$ generally leads to complex receivers.

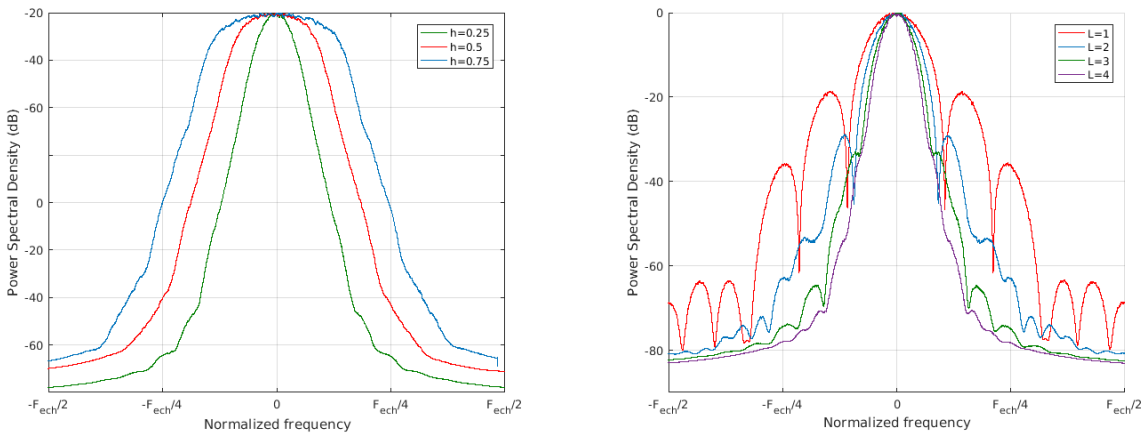
#### 1.3.2.3 The frequency pulse $g(u)$

As presented in Fig.1.1 (a.) and Table 1.1 the frequency pulse $g(u)$ determines the shape of the CPM (rectangular REC, raise cosine RC, gaussian GMSK, mixture of rectangular and raise cosine Weighted Average CPM...). Its primitive is the phase response $q(t)$, continuous over $[0,\ LT]$. This latter criterium keeps the phase of the CPM signal continuous.

#### 1.3.2.4 Influences of $h$, $L$ and $g(u)$ on the CPM scheme

$h$, $L$ and $g(u)$ can be tuned to shape the CPM bandwidth occupancy. It appears that small values of $h$ generate CPM signals with relatively small bandwidth occupancy while large

values result in CPM signals with large bandwidth occupancy. $L$ impacts also the CPM bandwidth, it seems that large values of $L$ lead to small bandwidth occupancy (but in a lower proportion than $h$) and reduce the side lobes sizes. Those effects are illustrated Fig.1.3 (a.) for the impact of the modulation index and (b.) for the impact of the CPM memory. However a greater complexity and a worse symbol-error-rate (SER) are the downside of having large $L$ and small $h$ respectively. Indeed the complexity of the receiver increases exponentially with $L$ and the SER are impaired when associated with small $h$ (see Fig.1.4 (b.)). $g(u)$, for its part, can modify the main lobe width and the side lobes size according to the chosen shape (RC, REC, gaussian...). Impacts of various frequency pulse shapes on the PSD are displayed Fig.1.4 (b.). It appears that the REC has the narrowest main lobe and RC the lowest out-of-band power.

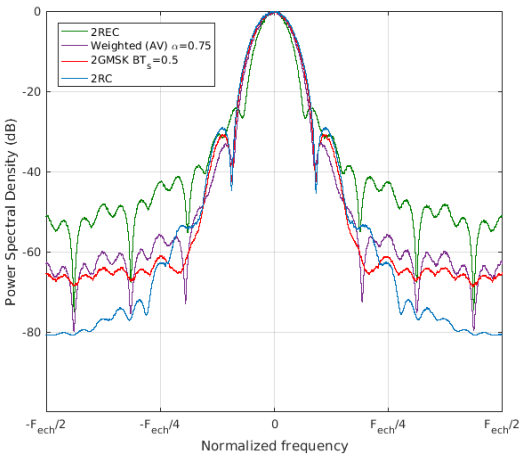



(a.) (b.)

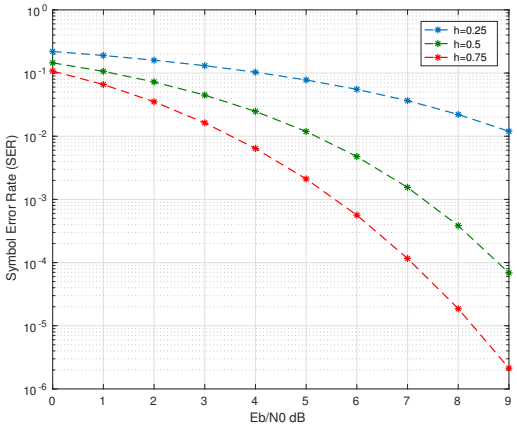
Figure 1.3: RC (a.) quaternary $L = 3$ (b.) binary $h = 0.5$

## 1.4 Trellis Representation

A trellis is a state diagram. It visually characterizes a set of states linked to each other by connection called transitions. The aim of this section is to present the original states and transitions of the CPM scheme trellis representation [ARS81a],[ARS81b]. Those latter can be identified by decomposing the information-carrying phase $\Xi$. Then $\forall$ t $\in [kT,\ kT + T[$,




(a.)




(b.)

Figure 1.4: (a.) PSD of various CPM $M = 2$, $h = 0.5$ (b.) GSMK $BT_s = 0.5$, $M = 2$, $L = 2$

$\Xi(t, a_0^k)$ is rewritten as follows:

$$\begin{aligned} \Xi(t, a_0^k) &= 2\pi h \sum_{n=0}^{k} a_n q(t - nT) \\ &= 2\pi h a_k q(t - kT) + 2\pi h \sum_{n=k-L+1}^{k-1} a_n q(t - nT) + \pi h \sum_{n=0}^{k-L} a_n \end{aligned} \tag{1.3}$$

$C = 2\pi h a_k q(t-kT)$ corresponds to the current symbol distribution, $D = 2\pi h \sum_{n=k-1-L}^{k-1} a_n q(t-nT)$ is the contribution of the $L-1$ last symbols and $\xi_k = \pi h \sum_{n=0}^{k-L} a_n$ is the phase state. By taking the phase state modulo $2\pi$, we can enumerate all its possible values:

- if $P$ is even:

$$\xi_k \in \{0, \frac{\pi P}{Q}, \frac{2\pi P}{Q}, ..., \frac{\pi(Q-1)P}{Q}\}$$

- if $P$ is odd:

$$\xi_k \in \{0, \frac{\pi P}{Q}, \frac{2\pi P}{Q}, ..., \frac{\pi(2Q-1)P}{Q}\}$$

A state space emerges from this decomposition. The transition has to depend on $C$ and the state must take into account $D$ and $\xi$. Consequently, at the $k^{th}$ symbol interval, the encoder state can be completely characterize by the L-tuple:

$$\sigma_k = [\xi_k, a_{k-L+1}, ..., a_{k-1}] \tag{1.4}$$

And the transmitted signal in the interval $[kT,\ \ kT+T[$ is completely deduced by the tuple $[\sigma_k, a_k]$. By recursion:

$$\begin{aligned} \sigma_{k+1} &= [\xi_{k+1}, a_{k-L+2}, ..., a_k] \\ \xi_{k+1} &\equiv \xi_k + \pi h a_{k-L+1} \mod 2\pi \end{aligned} \tag{1.5}$$

The original trellis representation has two drawbacks. First the phase trellis is time-varying because of the chosen symmetrical alphabet ($\{\pm 1, ..., \pm(M-1)\}$ ). It becomes obvious when we draw the phase trajectories of a 1RC ($M = 2$ and $h = 1/2$) in Fig.1.5. The phase trajectories in the even intervals (green circle on the figure) do not match with the odd intervals (red circle). Furthermore we notice that the size of the trellis depends on the parity of $P$ so for $P$ even, we obtain $Q \cdot M^{L-1}$ states and $Q \cdot M^L$ transitions and when $P$ is odd this number doubles for both states and transitions. Then, regarding those drawbacks, we conclude this part by saying that the original trellis is not entirely satisfactory.

## 1.5 Rimoldi's Decomposition

To address the original trellis' drawbacks, Rimoldi proposed an astute decomposition of the carrying-phase in [Rim88]. First of all, a time-invariant phase trellis $\Phi$ is defined. This latter depends on the traditional phase $\Xi$ plus a corrective term making the new phase trellis time-invariant. Then such a time-invariant phase trellis can be obtained in the following manner $\forall$ t $\in [kT,\ \ kT+T[$,

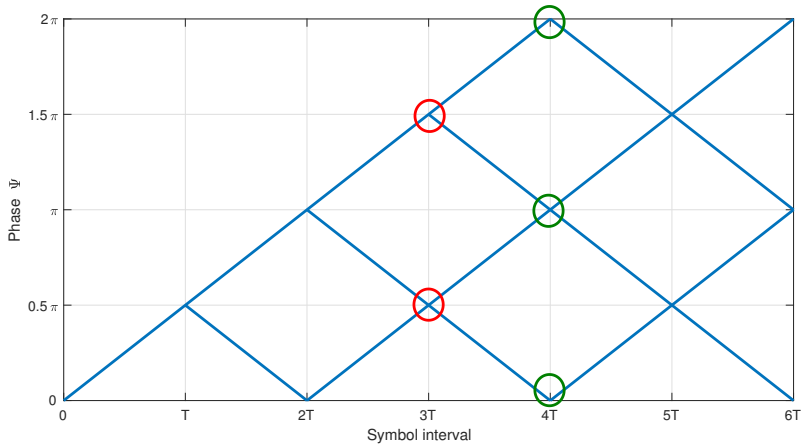



Figure 1.5: Modulo $2\pi$ Phase Tree of 1RC

$$\Phi(t, a_0^k) = \Xi(t, a_0^k) + \pi \frac{h(M-1)t}{T} \tag{1.6}$$

$\pi h(M-1)t/T$ can be absorbed in the signal's frequency term. For instance if $f_0$ is the original frequency, the new frequency $f_1$ is defined as $f_1 = f_0 - h(M-1)t/2T$. To distinguish between both $\Phi(t, a)$ and $\Xi(t, a)$, we denote $\Phi(t, a)$ the tilted phase. $\forall$ t $\in [kT,\ kT + T[$, $\Phi(t, a_0^k)$ is rewritten as follows:

$$\Phi(t, a_0^k) = 2\pi h \sum_{n=k-L+1}^{k} a_n q(t - nT) + \pi h \sum_{n=0}^{k-L} a_n + \frac{\pi h(M-1)t}{T} \tag{1.7}$$

We may introduce the modified data sequence defined by:

$$u_n = \frac{a_n + (M-1)}{2} \tag{1.8}$$

In the sequel we denote $u_n \in \{0, 1, ..., M-1\}$ the tilted symbol. The astute reader will notice that the modified data digits are just M-ary digits whatever the parity of $M$. We set $t = \tau + kT$, with $\tau\ \in ]0,\ T]$ and we develop the tilted phase's mathematical expression.

$$\begin{aligned}\Phi(\tau + kT, u_0^k) \quad &= 4\pi h \sum_{n=0}^{L-1} u_{k-n} q(\tau + nT) + 2\pi h \sum_{n=0}^{k-L} u_n + \frac{\pi h (M-1)\tau}{T} \\ &\quad -2\pi h (M-1) \sum_{n=0}^{L-1} q(\tau + nT) + (L-1)(M-1)\pi h\end{aligned} \tag{1.9}$$

$W(\tau) = \pi h(M-1)\left(\frac{\tau}{T} - 2\sum_{n=0}^{L-1} q(\tau + nT) + (L-1)\right)$ is the data-independent terms. This part is common to all the transmitted continuous-time waveform whatever the value of $k$. The data dependent (and time independent) term modulo $2\pi$ and modulo $Q$ is called the phase state $\phi_k = 2\pi h \sum_{n=0}^{k-L} u_n$. This term can take $Q$ possible values whatever the parity of $P$, which contrasts with the original representation where $\xi_k$ takes $2Q$ values for $P$ odd. In the sequel, we would note $\mathcal{Q}$ the set of phases taken by $\phi$. Finally at the $k^{th}$ symbol interval, the encoder state can be completely characterize by the L-tuple:

$$\delta'_k = [\phi_k, u_{k-L+1}, ..., u_{k-1}] \tag{1.10}$$

And the transmitted signal in the interval $[kT,\ kT+T[$ is completely deduced by the L-tuple $[\delta'_k, u_k]$. By recursion:

$$\begin{aligned}\delta'_{k+1} &= [\phi_{k+1}, u_{k-L+2}, ..., u_k] \\ \phi_{k+1} &\equiv \phi_k + 2\pi h u_{k-L+1} \mod 2\pi\end{aligned} \tag{1.11}$$

The encoding stage, leading to $\delta'_k$, is done through a linear continuous phase encoder (CPE) then, a memoryless modulator (MM) mapps $\delta'_k$'s symbols with the phase response $q(t)$ (see Fig. 1.6). Those two steps are described in the next section.

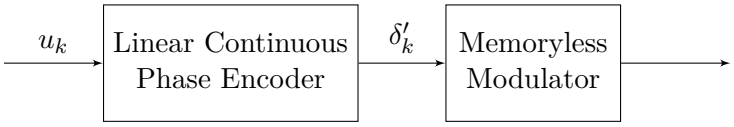



Figure 1.6: Block diagram of CPM system (CPE and MM).

### 1.5.1 CPM system

Taking into account the tilted phase and the tilted symbols, the signal is redefined as follows:

$$s(\tau, u_{k-L+1}^{k}) = \sqrt{\frac{Es}{T}} A(\tau) e^{j4\pi h \sum_{n=0}^{L-1} u_{k-n} q(\tau+nT)} e^{j\phi_k} \tag{1.12}$$

Where $A(\tau) = e^{jW(\tau)}$ is the Rimoldi's data independent term.

#### 1.5.1.1 The continuous-phase encoder (CPE)

As pointed out previously, at the $k^{th}$ symbol interval, the CPE updates the L-tuple $\delta'_k$ using the next symbol $u_k$ to generate the next MM input $\delta'_{k+1}$. The CPE can be designed by a shift register thereby, at each time clock, $T$ symbols are moved by one in the shift register (replacing $k$ by $k+1$ in $\delta'_{k-1}$) and $\phi_k$ is updated by adding $u_{k-L+1}$ (1.11). A possible realization of the CPE is proposed in Fig.1.7.

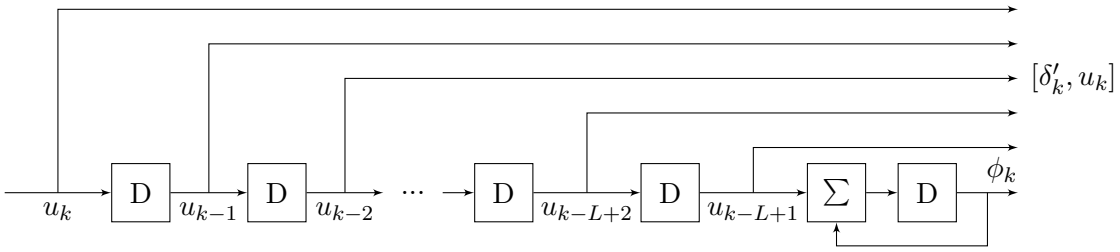



Figure 1.7: Continuous phase modulator.

#### 1.5.1.2 The memoryless modulator (MM)

The memoryless modulator maps the output of the CPE into a set $X$ of $M^L$ continuous-time waveforms with finite time support of length $LT$. At the $k^{th}$ symbol interval, the subset $u_{k-L+1}^{k} = \{u_{k-L+1}, ..., u_k\}$ matches $x_i(\tau)$ corresponding to the $i^{th}$ signal of $X = \{x_i(\tau), i = 0...M^L - 1\}$ with [Rim88],[VCT10]:

$$x_i(\tau) = \frac{A(\tau)}{\sqrt{T}} \cdot e^{j4\pi h \sum_{n=0}^{L-1} u_{k-n} q(\tau+nT)}, \tau \in [0, T), \tag{1.13}$$

The index $i$ is determined as follows.

$$i = \sum_{n=0}^{L-1} u_{k-n} \cdot M^{L-1-n} \tag{1.14}$$

It seems that some explanations are needed for the evaluation of index $i$. Since symbols belong to the M-ary alphabet, they may take $M$ possible value. For partial CPM, the phase trajectory depends on a set of $L$ symbols. Thus $X$ is composed of $M^L$ possible trajectories. To sort each of them in $X$, we decided to make use of index $i$. For instance the set $u_{k-L+1}^{k} = \{0, ..., 0\}$ corresponds to the first index $i = 0$ according to (1.14). As well the set $u_{k-L+1}^{k} = \{1, 0, ..., 0\}$ corresponds to the second index $i = 1$ and so on. We could equivalently replace index $i$ by its depending set of symbols $u_{k-L+1}^{k}$ and write $x_{u_{k-L+1}^{k}}$.

We will terminate this section showing a basic example of the Rimoldi's decomposition for the MSK modulation Fig.1.8. The set of samples is $\{\pm 1\}$ which corresponds in the tilted domain to the set $\{0, 1\}$. The cardinality of the tilted phase's set is divided by 2 when we compare to the cardinality of original phase's set , then $\phi \in \{0, \pi/2\}$.

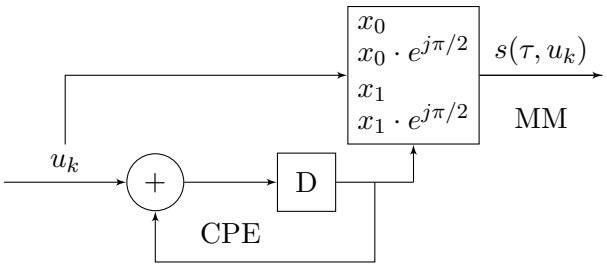



Figure 1.8: Recursive implementation of MSK following the CPE and MM representation.

## 1.6 CPM Receivers

Trellis-based receivers (TBR) are generally implemented to demodulate CPM schemes. They are by nature non-linear receiver. It exits various trellis based algorithms as the viterbi algorithms [Vit67] or the bit/symbol-wise BCJR (Bahl, Cocke, Jelinek and Raviv) algorithm [Bah+74]. Those algorithms are respectively based on the maximum likelihood (ML) and maximum a priori (MAP) criterium. Thus the viterbi algorithm returns the sequence with the largest probability whereas BCJR computes the probability of each symbol/bit regarding the whole sequence. The two algorithms give similar performances when no a priori are taken into account. In coherent regime a trellis-based receiver is optimal to demodulate the symbols. However the detection of partial CPM can quickly become complex according to the size of the CPM memory. Thereby, [Kal89] and later on [CR97], made use of Laurent/Mengali decomposition [Lau86]/[MM95] to reduce trellis complexity for a ML and respectively a MAP demodulation. In non-coherent regime, the accumulated phase of the CPM signal is unknown which adds a new difficulty against the coherent case. To deal with this constraint, non-coherent metrics are averaged over the CPM signal phase during the demodulation process. Generally, two types of demodulation are performed, based either on a trellis [CFR00] as in coherent regime or on a block processing [RD99]. The block receiver is more robust to phase shift than the trellis but less effective. Although many interesting studies have been achieved in the coherent detection, our work is more broadly focused on the non-coherent case. Thus we would delve more deeply in the non-coherent regime compare to the coherent one.

### 1.6.1 System model

A binary message vector $\mathbf{b} = [b_0, \cdots, b_{K_b-1}] \in GF(2)^{K_b}$ is mapped into a sequence $\mathbf{U} = [u_0, ..., u_{N_s-1}]$ belonging to the $M$-ary alphabet $\{0, ..., M-1\}$ (with M power of 2). Symbols are then modulated following the CPM modulation rule using Rimoldi's representation [Rim88]. This can be seen as the serial concatenation of a continuous phase encoder (CPE) and a memoryless modulator (MM). First, the CPE ensures the continuity between the trans-

mitted continuous-time waveforms by accumulating the phase of each modulated symbol.

$$\phi_{k+1} = \phi_k + 2\pi h u_{k-L+1} \tag{1.15}$$

Then, the memoryless modulator maps the output of the CPE into a set $X$ of $M^L$ continuous-time waveforms. At the $k^{th}$ symbol interval, the subset $u_{k-L+1}^k = \{u_{k-L+1}, ..., u_k\}$ matches $x_i(\tau)$ corresponding to the $i^{th}$ signal of $X = \{x_i(\tau), i = 0...M^L - 1\}$ with [Rim88][VCT10]

$$x_i(\tau) = \frac{A(\tau)}{\sqrt{T}} \cdot e^{j4\pi h \sum_{n=0}^{L-1} u_{k-n} q(\tau+nT)}, \tau \in [0, T), \tag{1.16}$$

As a reminder, $A(\tau)$ represents the Rimoldi representation's data independent terms and the index $i$ is given as follows

$$\begin{aligned} i &= \sum_{n=0}^{L-1} u_{k-n} \cdot M^{L-1-n} \\ &\text{and,} \\ A(\tau) &= e^{j\pi h(M-1)\left(\frac{\tau}{T} + (L-1) - 2\sum_{n=0}^{L-1} q(\tau+nT)\right)}. \end{aligned} \tag{1.17}$$

$X$ is composed of $M^L$ possible trajectories. To sort each of them in $X$, we decided to make use of index $i$. For instance the set $u_{k-L+1}^k = \{0, ..., 0\}$ corresponds to the first index $i = 0$ according to (1.17). As well the set $u_{k-L+1}^k = \{1, 0, ..., 0\}$ corresponds to the second index $i = 1$ and so on. We could equivalently replace index $i$ by its depending set of symbols $u_{k-L+1}^k$ and write $x_{u_{k-L+1}^k}$. It must be pointed out that the system model proposed in this section assumes a complex baseband representation of the CPM. In that respect, the CPM complex baseband representation of the transmitted continuous-time waveform during the $k^{th}$ symbol time of the observation interval is given by:

$$s_k(t) = \sqrt{Es} \cdot x_i(t) \cdot e^{j\phi_k} \tag{1.18}$$

The transmitted signal undergoes a phase rotation $\theta$ and it is corrupted by a complex additive white Gaussian noise (AWGN) channel with noise spectral density $N_0$. $\theta$ is assumed to be constant during the whole transmission and uniformly distributed on $[0, 2\pi[$. The channel is said to be non-coherent when $\theta$ is unknown. The corresponding complex-baseband received signal is given by

$\forall\, t \in [kT; (k+1)T)$,

$$r_k(t) = e^{j\theta} \cdot s_k(t) + n(t), \tag{1.19}$$

$n(t)$ in equation (1.19) corresponds to the complex AWGN. Perfect frequency and time synchronization is assumed. During the $k^{th}$ symbol interval, the received signal $r_k(t)$ is passed through a bank of $M^L$ matched filters whose impulse responses are given by $x_i^*(-t)$, $i = 0, ..., M^L - 1$ where $x_i^*(t)$ is the complex conjugate of $x_i(t)$ (see Fig. 1.9). The corresponding sufficient statistics will be used during the demodulation process.

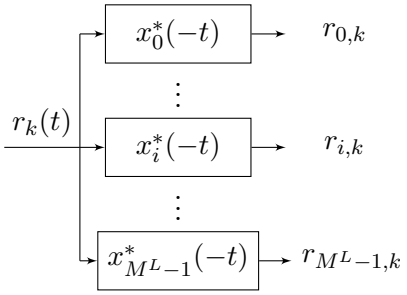



Figure 1.9: Complex matched filters of M-CPM modulation

Considering a perfect timing synchronization, $r_{i,k}$ is the element resulting from the correlation between $r_k(t)$ and $x_i^*(-t)$.

$$r_{i,k} = \int_0^T r_k(t) x_i^*(t) dt \tag{1.20}$$

In the sequel, we adopt the following notation $\boldsymbol{r}_k = [r_{0,k}, ..., r_{M^L-1,k}]$ and the set of observations is given by $\boldsymbol{r}_{L-1}^{N_s-1} = [\boldsymbol{r}_{L-1}, ..., \boldsymbol{r}_{N_s-1}]$. We will equivalently replace, in the following,

index $i$ by its depending set of symbols $u_{k-L+1}^{k}$ and rewrite $\boldsymbol{r}_k = [r_{\{0,...,0\},k}, ..., r_{\{M-1,...,M-1\},k}]$.

### 1.6.2 Coherent regime

In coherent regime, $\theta$ is assumed perfectly known and without loss of generality, we can consider $\theta = \phi_0 = 0$. The detection is done using the BCJR algorithm. The aim is to compute the conditional probability of a symbol given the observations noted $p(u_k|\boldsymbol{r}_{L-1}^{N_s-1})$.

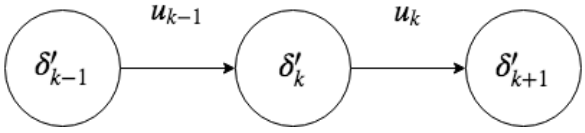



Figure 1.10: TBR States Diagram

Let $\delta'_k = [\phi_k, u_{k-L+1}, ..., u_{k-1}]$ be a state of the trellis taking into account the accumulated phase $\phi_k$ and a series of $L-1$ symbols $u_{k-L+1}^{k-1}$ (with $k \geq L-1$) coming from the memory required by the partial response (if full-response CPMs are considered, $L = 1$ and this amount is null). The transition $\{\delta'_k \rightarrow \delta'_{k+1}\}$ corresponds to the emitted symbol $u_k$ and fulfills to the subsequent equation $\phi_{k+1} = \phi_k + 2\pi h u_{k-L+1}$. The conditional probability is developed in the following manner.

$$p(u_k|\boldsymbol{r}_{L-1}^{N_s-1}) \propto \sum_{\{\delta'_k\}} \alpha_k(\delta'_k)\beta_{k+1}(\delta'_{k+1})\gamma(\delta'_k \rightarrow \delta'_{k+1}, \boldsymbol{r}_k)p(u_k) \tag{1.21}$$

Using sufficient statistics at the output of the filter bank of Fig. 1.9, the classical forward, backward and transition kernel probabilities (denoted $\alpha, \beta$ and $\gamma$ respectively) are computed as follows.

$$
\begin{aligned}
\alpha_k(\delta'_k) &\propto \sum_{\{\delta'_{k-1}\}} \alpha_{k-1}(\delta'_{k-1}) e^{\rho \cdot \Re\left( (e^{-j\phi_{k-1}}) \cdot r_{u_{k-L}^{k-1},k-1} \right)} p(u_{k-1}) \\
\beta_k(\delta'_k) &\propto \sum_{\{\delta'_{k+1}\}} \beta_{k+1}(\delta'_{k+1}) e^{\rho \cdot \Re\left( (e^{-j\phi_k}) \cdot r_{u_{k-L+1}^{k},k} \right)} p(u_k) \\
\gamma(\delta'_k \rightarrow \delta'_{k+1}, \boldsymbol{r}_k) &\propto e^{\rho \cdot \Re\left( (e^{-j\phi_k}) \cdot r_{u_{k-L+1}^{k},k} \right)}
\end{aligned}
\tag{1.22}
$$

Where $\Re(.)$ stands for the real part and $\rho = 2\sqrt{Es}/N_0$.

### 1.6.3 Non-coherent regime

In non-coherent regime the phase induced by the channel is unknown. Then we can't differentiate $\theta$ from $\phi$. To overcome this constraint, what is usually done is to remove the phase dependency from the detection. Thus the phase of the received signal is averaged on $[0, 2\pi[$ and the demodulation is processed afterwards. Two approaches have been proposed to demodulate in non-coherent regime based on the Rimoldi's decomposition: either by block or by trellis.

#### 1.6.3.1 Block receivers

[RD99] and [VCT10] have implemented the receiver by block for a hard and respectively soft-decision demodulation. Block detection is a sub-optimal detector, it can work for any value of modulation index and it is robust to channel phase shift between blocks when those are taken independently. Thus the coherent time must be at most of the order of a block size. This method is only adapted to full response CPM. However a *differential* detection by block for partial CPM can be performed [DS90] but associated with an exceedingly high complexity. The process of the block receiver can be summarized as follows. The block receiver does the correlation between the block of received symbols and all existing combinations of the same block length. The condition required to use this method is the absence of phase shift between symbols belonging to the same block since the phase continuity is exploited within

the blocks. If we suppose the phase shift is stable over a block of $N$ symbols, it exists $M^N$ possible combinations for a block of size $N$ (with $M$ the modulation order). The incoming signal is filtered by a bank of $M^N$ matched filters. Then the probability of the transmitted sequence of size $N$ under the condition of one of the possible combination is done for each existing combination. Eventually, if we assume a ML block detection as in [RD99], the decision is made in the favor of the largest conditional probability. If a MAP detection is preferred [VCT10] (done for CPFSK only), the demodulator computes the log-likelihood ratio (LLR) for each symbol/bit of the block based on the conditional and a priori probabilities.

The non-coherent block receiver computes the probability of a sequence of $N$ symbols given its observations. It may read as follows:

$$p(\boldsymbol{r}_{k-N+1}^{k}|u_{k-N+1}^{k}) \propto I_0\left(\rho \cdot \left|\mu(u_{k-N+1}^{k})\right|\right) \tag{1.23}$$

where $I_0$ is the modified zero order Bessel function of the first kind and

$$\mu(u_{k-N+1}^{k}) = \sum_{i=k-N+1}^{k} r_{u_i,i} \cdot e^{-j2\pi h \sum_{n=k-N+1}^{i-1} u_n} \tag{1.24}$$

For the ML detection, the decision is made in the favor of the largest probability among the $M^N$ combinations of $N$ symbols $u_{k-N+1}^{k}$ . For a MAP detection, the demodulator computes the LLR for each bit based on the probability $p(u_{k-N+1}^{k}|\boldsymbol{r}_{k-N+1}^{k})$. Taking for instance the $i^{th}$ bit in the symbols block $u_{k-N+1}^{k}$ (noted $b_i(u_{k-N+1}^{k})$), the LLR is given by

$$\lambda_i = log\left[\frac{\sum_{b_i(u_{k-N+1}^{k})=0} p(\boldsymbol{r}_{k-N+1}^{k}|u_{k-N+1}^{k}) \cdot p\left(b_{\setminus i}(u_{k-N+1}^{k})\right)}{\sum_{b_i(\boldsymbol{u}_{k-N+1}^{k})=1} p(\boldsymbol{r}_{k-N+1}^{k}|u_{k-N+1}^{k}) \cdot p\left(b_{\setminus i}(u_{k-N+1}^{k})\right)}\right] \tag{1.25}$$

Here, $b(u_{k-N+1}^{k})$ is the bits block associated to the symbols block $u_{k-N+1}^{k}$, thus $b_{\setminus i}(u_{k-N+1}^{k})$

is $b(u_{k-N+1}^{k})$ excluding $b_i(u_{k-N+1}^{k})$.

#### 1.6.3.2 Non-coherent Trellis based Receiver

A first trellis-based approach based on Viterbi algorithm [Vit67] had been presented in [RD99]. Thereafter [CFR00] proposed a symbol MAP decoding algorithm similar to the well-known BCJR [Bah+74]. The two solutions, based on different metrics, ML for [RD99] and MAP for [CFR00], lead to similar performance. The following relationship clarifies the difference between both ML and MAP metrics. Let us consider $\hat{u}_{ML,k}$ and $\hat{u}_{MAP,k}$ the estimated symbols obtained after a ML and respectively symbol MAP detection at the $k^{th}$ symbol interval.

$$\begin{aligned} \hat{u}_{ML,k} &= \arg\max_{u_0^{N_s-1}} \{p(\boldsymbol{r}_{L-1}^{N_s-1} | u_0^{N_s-1})\} \\ \hat{u}_{MAP,k} &= \arg\max_{u_k} \{p(u_k | \boldsymbol{r}_{L-1}^{N_s-1})\} \end{aligned} \tag{1.26}$$

The MAP maximizes the probability of each symbol whereas the ML returns the most likely sequence. Although the performance are similar between [RD99] and [CFR00], we notice that the state space used by both references are slightly different as we would show in the sequel.

***Maximum-Likelihood receiver***

[RD99] has proposed a non-coherent detector based of the Viterbi algorithm. This algorithm aims to compute $p(\boldsymbol{r}_{L-1}^{N_s-1} | u_0^{N_s-1})$. The state space associated to [RD99] is given by $\delta_k = \{u_{k-N-L+2}, ..., u_{k-1}\}$. This latter takes into account a series of $N+L-2$ symbols $u_{k-N-L+2}^{k-1}$ (with $k \geq N+L-2$). Based on this state space, we can differentiate the $L-1$ symbols coming from the memory required by the partial response and the $N-1$ additional symbols required when we extend the observation length in non-coherent regime to improve the performance. Those latter are called the correlated symbols in the sequel because they are used in the process of correlation between the observations and the existing combination of symbols. The transition between two states $\{\delta_k \rightarrow \delta_{k+1}\}$ corresponds to the transmitted

symbol $u_k$. The probability of the maximum likelihood sequence is computed as follows.

$$p(\boldsymbol{r}_{L-1}^{N_s-1}|u_0^{N_s-1}) \propto I_0\left(\rho \cdot \left|\mu(u_0^{N_s-1})\right|\right) \tag{1.27}$$

where

$$\mu(u_0^{N_s-1}) = \sum_{i=L-1}^{N_s-1} r_{u_{i-L+1}^i,i} \cdot e^{-j2\pi h \sum_{n=0}^{i-L} u_n} \tag{1.28}$$

***Symbol/bit Maximum a Priori receiver***

[CFR00] implemented a non-coherent detector based on the BCJR algorithm. This algorithm aims to compute $p(u_k|\boldsymbol{r}_{L-1}^{N_s-1})$. The state space presented by the authors, can be seen as an extended version of the one proposed in [RD99]. Indeed [CFR00] has taken into account the accumulated phase in addition to the symbols in memory $(L-1)$ and the correlated symbols $(N-1)$. Then $\delta_k'' = \{\phi_{k-N+1}, u_{k-N-L+2}, ..., u_{k-1}\}$ is called in the sequel the extended state. The transition $\{\delta_k'' \to \delta_{k+1}''\}$ corresponds to the emitted symbol $u_k$ and fulfills to the subsequent equation $\phi_{k+1} = \phi_k + 2\pi h u_{k-L+1}$. The conditional probability is developed as the following manner,

$$p(u_k|\boldsymbol{r}_{L-1}^{N_s-1}) \propto \sum_{\{\delta_k''\}} \alpha_k(\delta_k'')\beta_{k+1}(\delta_{k+1}'')\gamma(\delta_k'' \to \delta_{k+1}'', \boldsymbol{r}_{k-N+1}^k)p(u_k) \tag{1.29}$$

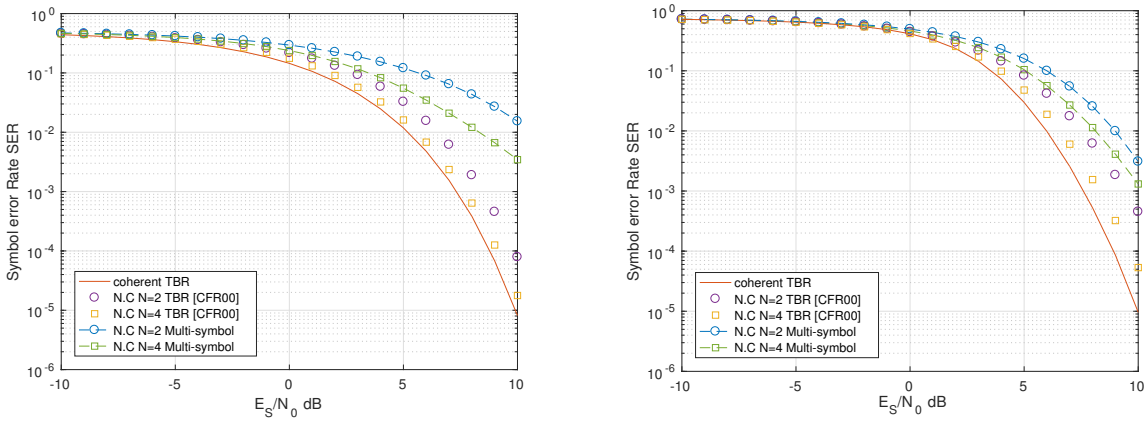
The forward-backward recursions and the transition kernel probabilities read as follows

$$
\begin{aligned}
\alpha_k(\delta''_k) &\propto \sum_{\{\delta''_{k-1}\}} \alpha''_{k-1}(\delta''_{k-1}) \frac{I_0\left(\rho\cdot\left|\mu(\boldsymbol{u}_{k-N-L+1}^{k-1})\right|\right)}{I_0\left(\rho\cdot\left|\mu(\boldsymbol{u}_{k-N-L+2}^{k-1})\right|\right)} p(u_{k-1}) \\
\beta_k(\delta''_k) &\propto \sum_{\{\delta''_{k+1}\}} \beta''_{k+1}(\delta''_{k+1}) \frac{I_0\left(\rho\cdot\left|\mu(\boldsymbol{u}_{k-N-L+2}^{k})\right|\right)}{I_0\left(\rho\cdot\left|\mu(\boldsymbol{u}_{k-N-L+2}^{k-1})\right|\right)} p(u_k) \\
\gamma(\delta''_k \to \delta''_{k+1}, \boldsymbol{r}_{k-N+1}^{k}) &\propto I_0\left(\rho\cdot\left|\mu(u_{k-N-L+2}^{k})\right|\right)
\end{aligned}
\tag{1.30}
$$

where

$$
\mu(u_{k-N-L+2}^{k}) = \sum_{i=k-N+1}^{k} r_{u_{i-L+1}^{i},i} \cdot e^{-j2\pi h \sum_{n=k-N-L+2}^{i-L} u_n}
\tag{1.31}
$$

Some finite length and asymptotic simulation are provided Fig. 1.11 to evaluate the performances of the previous receivers.

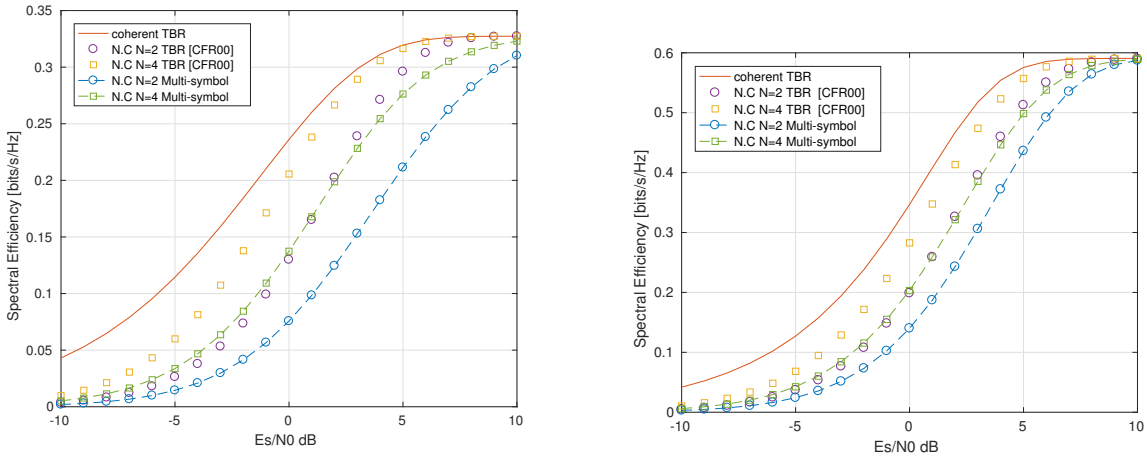



(a.) (b.)

Figure 1.11: SER of (a.) GMSK $h = 1/2$, $BT = 0.3$ $L = 1$ and (b.) CPFSK $L = 1$, $M = 4$, $h = 5/7$

## 1.7 Conclusion

In this chapter, we introduced the CPM modulation and its decomposition over a trellis. Then we did an overview of its associated receivers for the coherent and the non-coherent regime. In the sequel, we would mainly focus on the non-coherent symbol/bit MAP receiver and study




(a.) (b.)

Figure 1.12: Spectral Efficiency of (a.) GMSK $h = 1/2$, $BT = 0.3$ $L = 1$ and (b.) CPFSK $L = 1$, $M = 4$, $h = 5/7$

ways of improving it. In that respect, in the next chapter, we will see that the extended state space model introduced previously can suit to both coherent and non-coherent channel but is definitely not minimal. Actually, the authors included unnecessarily the accumulated phase to the non-coherent minimal state. Indeed the phase information is useless since it can not be exploited between states due to the unknown phase rotation undergone during the transmission. This will be corroborated mathematically and by finite length and asymptotic simulations in the sequel. Consequently, those results make sure that this state space for the MAP receiver should be reduced to the one given in [RD99].

Chapter 2

# On the link between coherent and non-coherent receivers

---

## Sommaire

## 2.1 Résumé

Au chapitre précédent, nous avons passé en revue les récepteurs non-cohérents de la littérature. Il existe deux catégories de récepteurs non-cohérents : les détecteurs par bloc et les détecteurs par treillis. Dans cette partie, on ne s'intéressera qu'aux détecteurs par treillis. Ces derniers peuvent être basés sur l'algorithme de type Viterbi avec un critère de maximum de vraisemblance (ML) [DS90] ou sur un algorithme de type BCJR avec un critère par maximum *a posteriori* (MAP) [CFR00]. Cependant, au chapitre 1, on a remarqué que l'espace d'états proposé en [CFR00] pour une détection MAP était plus grand que celui présenté en [DS90] par maximum de vraisemblance. L'auteur a pris en compte, dans ses états, l'information de phase. On parlera alors d'espace d'états *étendu*. On étudiera dans ce chapitre l'intérêt de cet ajout dans une démodulation non-cohérente. Dans un premier temps, on montrera par une dérivation analytique que cet espace d'états peut être réduit au même espace que [DS90] sans perte de performances. Notre analyse sera étayée par la simulation du taux d'erreur binaire et de l'information mutuelle pour différents paramètres de CPM. Ce résultat ne devrait pas être surprenant à première vue puisque la phase étant inconnue du fait de la non-cohérence du canal, cette dernière ne peut être exploitée entre les états du treillis. On dira alors que cet espace d'états est minimal et suffisant pour une taille d'observation donnée. Minimal puisque on ne peut réduire l'espace d'états du récepteur sans perte de performances et suffisant car prendre en compte la phase complexifie le récepteur sans gain de performance. Cependant on montrera que l'espace d'états étendu présente un avantage important : il est adapté à la fois au canal cohérent et non-cohérent. En effet, disposer d'un espace d'états redondant permet de démoduler avec le même espace d'états quelque soit la cohérence du canal. Seules changent les métriques utilisées.

## 2.2 Introduction

In chapter 1, we have presented two main approaches to demodulate in non-coherent regime, either by block or trellis. Block detection may work for any value of modulation index and is robust to fast channel phase shifts when blocks are taken independently. However trellis based detection gives better performance than block detection for lower complexity but is a bit more sensitive to phase deviation. In this chapter, we will focus solely on trellis based detection. The first trellis-based approach based on Viterbi algorithm was presented in [DS90]. Thereafter [CFR00] proposed a symbol MAP decoding algorithm similar to the well-known BCJR [Bah+74]. However we have seen in chapter 1 that the state cardinality proposed for the MAP detection in [CFR00] is greater than the one presented in [DS90] for the ML detection. Indeed the author added to the state provided in [DS90] the information phase given birth to a certain extent to an extended state space. In that respect, we investigate in what follows the benefit of this discrepancy over the non-coherent detection. To this end we derive the equations of the non-coherent receiver with the extended state space approach provided in [CFR00]. Then we analytically prove that we can reduced the extended state space model to its minimal form without performance loss. Our claim are also supported by numerical results. Indeed we explicitly show thereafter that, by considering the state space given by [DS90], the direct derivation of the symbolwise non coherent MAP receiver for CPMs leads to the same bit error rate performance than the extended state space approach usually considered in the literature. Furthermore an additional EXIT charts analysis shows that there is no information loss when considering the proposed reduced state space approach. This implies that [DS90] is effectively a sufficient state representation for non coherent detection, since no loss of information occurs. This result should be not surprising at the first sight: indeed it makes sense that the phase information cannot be exploited between states due to the unknown phase rotation undergone during the transmission. Therefore, taking into account the accumulated phase in the state formulation is "useless" in the non-coherent regime. By useless, we mean that there is no information to gain from this space expansion, which results undoubtedly in an avoidable increase of complexity. Moreover, we can also say that the trellis is minimal, in

the sense that, for a given observation length $N$, you cannot find a non coherent trellis MAP detector having less states without performance loss. This implies that this representation is the solely representation leading to minimal complexity for a given observation length. However, the extended state space model features one important advantage: it suits to both coherent and non-coherent channel. Indeed, since it is a "redundant" model, it is common to both regime, thereby it can deal with non-coherent channel and, at the same time, retrieve coherent performance when using the proper metrics on the trellis. Actually, in non-coherent regime, metrics are averaged over the received signal phase but in the event where the channel would become again coherent, it is possible, with this common model, to retrieve coherent performance without average. In a way, this model is a mean to guarantee the validity of equations in the non-coherent regime because it relies on the well-known coherent results. Consequently starting from this general model/framework, it is possible to derive both the coherent and non coherent BCJR metrics and thereafter to propose a state reduction according to the regime under consideration.

## 2.3 System model

The system model considered here, is the same than the one presented in the previous chapter (see part 1.6.1). So to sum up quickly, the CPM complex baseband signal transmitted during the $k^{th}$ symbol time of the observation interval is given as follows (with $i \in \{0...M^L - 1\}$):

$$s_k(t) = \sqrt{Es} \cdot x_i(t) \cdot e^{j\phi_k} \tag{2.1}$$

Then during the transmission the signal undergoes a phase rotation $\theta$ and it is corrupted by a complex additive white Gaussian noise (AWGN) channel with noise spectral density $N_0$. $\theta$ is assumed to be unknown, constant during the whole transmission and uniformly distributed on $[0, 2\pi[$. The channel is said to be non-coherent. The corresponding complex-baseband received signal is given by

$\forall\, t \in [kT; (k+1)T)$,

$$r_k(t) = e^{j\theta} \cdot s_k(t) + n(t), \tag{2.2}$$

$n(t)$ in equation (2.2) corresponds to the complex AWGN. Perfect frequency and time synchronization is assumed. During the $k^{th}$ symbol interval, the received signal $r_k(t)$ is passed through a bank of $M^L$ matched filters whose impulse responses are given by $x_i^*(t)$, $i = 0, ..., M^L - 1$ where $x_i^*(t)$ is the complex conjugate of $x_i(-t)$ (again see Fig. 1.9)). The corresponding sufficient statistics will be used during the demodulation process. Considering a perfect timing synchronization, $r_{i,k}$ is the element resulting from the correlation between $r_k(t)$ and $x_i^*(-t)$. In the sequel, we adopt the following notation $\boldsymbol{r}_k = [r_{0,k}, ..., r_{M^L-1,k}]$ and the set of observations is given by $\boldsymbol{r}_{L-1}^{N_s-1} = [\boldsymbol{r}_{L-1}, ..., \boldsymbol{r}_{N_s-1}]$. We will equivalently replace, in the following, index $i$ by its depending set of symbols $u_{k-L+1}^k$ and rewrite $\boldsymbol{r}_k = [r_{\{0,...,0\},k}, ..., r_{\{M-1,...,M-1\},k}]$.

## 2.4 TBR: An Extended state space approach

In this section, we investigate on the derivation of a symbol MAP receiver for both the coherent and non-coherent case. We derive an exact mathematical formulation for the trellis-based receiver based on the extended state space [CFR00] and a modified version of the BCJR algorithm [Bah+74]. This formulation is *valid for both regimes*. Then, we instantiate the coherent and non-coherent metrics for both regimes. Instantiating in the different regimes will lead to a ***state reduction*** according to the considered channel. In particular, we will show that we can recover the classical AWGN receiver when instantiated in coherent regime while we obtained a symbol MAP receiver with a state space similar to the model proposed in [RD99] for a ML criterion, but significantly reduced compared to [CFR00] when considering the non-coherent regime. Thus, the state space proposed in [CFR00] is not minimal for those two regimes. We will show at the end of this section the mathematical relationship between the metrics based on the minimal state space and the ones based on the extended state space in both regimes. For ease of understanding the main differences between both literature approaches ([CFR00],[RD99]) and the one proposed in this chapter are summarized in Table 2.1.

Table 2.1: Comparison with main existing approaches.

| Reference paper | Algorithm/Criteria | State Space | Metrics | Complexity |
|---|---|---|---|---|
| [DS90] | $\neq$ | $=$ | $\neq$ | $<$ |
| [CFR00] | $=$ | $\neq$ | $\neq$ | $>$ |

[1] $=$ (respectively $\neq$) the reference paper has the same (respectively different) feature ("Criteria/State model/Metric") than the proposed approach.
[2] $<$ (respectively $>$) the reference has a lower (respectively larger) complexity than the proposed approach.

### 2.4.1 Extended State Space: Mathematical Formulation

TBR extended state space is based on a trellis representation allowing us to use a modified version of the BCJR algorithm to compute the conditional probability of a symbol given the observations noted $p(u_k|\boldsymbol{r}_{L-1}^{N_s-1})$.

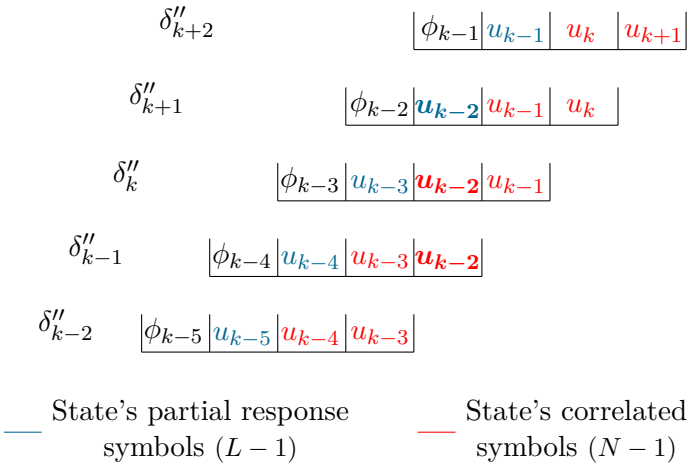



Figure 2.1: TBR Extended State Space Model for N=3 and L=2

Let $\delta''_k = \{\phi_{k-N+1}, u_{k-N-L+2}, ..., u_{k-1}\}$ be a state of the trellis taking into account the accumulated phase $\phi_{k-N+1}$ and a series of $N+L-2$ symbols $u_{k-N-L+2}^{k}$ (with $k \geq N+L-2$). Based on this extended state space, we can differentiate the $L-1$ symbols coming from the memory required by the partial response (if full-response CPMs are considered, $L=1$ and this amount is null) and the $N-1$ additional symbols required when we extend the observation length in non-coherent regime to improve the performance (if the observation length is one, this amount is null). The latter ones are called the correlated symbols in the sequel because they are used in the process of correlation between the observations and the existing combination of symbols (see Fig.A.1 in Section A). Stated more explicitly, we can decompose $\delta''_k$ in two parts: $\delta'_k = \{\phi_{k-N+1}, u_{k-L+1}, ..., u_{k-1}\}$ is the coherent optimal state and $\delta_k = \{u_{k-N-L+2}, ...u_{k-1}\}$ the optimal ML one in the non-coherent regime. The TBR extended state space model provides a common state for both regime which permits to derive the equations regardless the type of channel. Based on this extended state space model, we will derive a modified version of the BCJR algorithm. Finally, we will show how instantiating

the model to a given regime leads to a state reduction. The transition between two states $\{\delta''_k \to \delta''_{k+1}\}$ corresponds to the transmitted symbol $u_k$ and fulfills to the subsequent equation $\phi_{k-N+2} = \phi_{k-N+1} + 2\pi h u_{k-N-L+2}$.

As stated above, the BCJR algorithm has to be re-derived in order to take into account the extended TBR states. Based on Bayes theorem, the conditional probability of symbols given the observation $p(u_k|\boldsymbol{r}_{L-1}^{N_s-1})$ is proportional to $p(\boldsymbol{r}_{L-1}^{N_s-1}|u_k)$. At first, the conditional probability is developed in the following manner (the detailed proof is given in appendix in section A).

$$p(\boldsymbol{r}_{L-1}^{N_s-1}|u_k) = \sum_{\{\delta''_k\}} p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta''_k) p(\delta''_k) p(\boldsymbol{r}_{k-N+1}^{k}|\delta''_k, u_k) p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k}, \delta''_{k+1}) \tag{2.3}$$

The classical forward, backward and transition kernel probabilities (denoted $\alpha, \beta$ and $\gamma$ respectively) are identified in (2.3) as follows.

$$\begin{aligned} \gamma(\delta''_k \to \delta''_{k+1}, \boldsymbol{r}_{k-N+1}^{k}) &= p(\boldsymbol{r}_{k-N+1}^{k}|\delta''_k, u_k) \\ \alpha_k(\delta''_k) &= p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta''_k) p(\delta''_k) \\ \beta_{k+1}(\delta''_{k+1}) &= p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k}, \delta''_{k+1}) \end{aligned} \tag{2.4}$$

As with the BCJR algorithm, $\alpha_k$ can be calculated as

$$\alpha_k(\delta''_k) = \sum_{\{\delta''_{k-1}\}} p(\boldsymbol{r}_{L-1}^{k-N-1}|\boldsymbol{r}_{k-N}^{k-2}, \delta''_{k-1}) p(\delta''_k) p(\delta''_{k-1}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta''_k) p(\boldsymbol{r}_{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta''_k, \delta''_{k-1}) \tag{2.5}$$

where

$$
\begin{aligned}
p(\boldsymbol{r}_{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta_k'',\delta_{k-1}'') &= p(\boldsymbol{r}_{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta_{k-1}'',u_{k-1}) \\
&= \frac{p(\boldsymbol{r}_{k-N}^{k-1}|\delta_{k-1}'',u_{k-1})}{p(\boldsymbol{r}_{k-N+1}^{k-1}|\delta_{k-1}'',u_{k-1})} \\
&= \frac{\gamma(\delta_{k-1}''\rightarrow\delta_k'',r_{k-N}^{k-1})}{p(\boldsymbol{r}_{k-N+1}^{k-1}|\delta_{k-1}'',u_{k-1})}
\end{aligned}
\tag{2.6}
$$

Equation (2.6) is derived from Bayes' theorem and from equation (2.4). Moreover, we have

$$
\begin{aligned}
p(\delta_{k-1}''|\boldsymbol{r}_{k-N+1}^{k-1},\delta_k'')p(\delta_k'') &= p(\delta_{k-1}''|\delta_k'')p(\delta_k'') \\
&= p(\delta_k''|\delta_{k-1}'')p(\delta_{k-1}'') \\
&= p(u_{k-1})p(\delta_{k-1}'')
\end{aligned}
\tag{2.7}
$$

In equation (2.7), $p(\delta_{k-1}''|\boldsymbol{r}_{k-N+1}^{k-1},\delta_k'')$ is independent of observations $\boldsymbol{r}_{k-N+1}^{k-1}$. At last, after all terms have been collected, a recursion of $\alpha$ is obtained as follows

$$
\alpha_k(\delta_k'') = \sum_{\{\delta_{k-1}''\}}\alpha_{k-1}(\delta_{k-1}'')\frac{\gamma(\delta_{k-1}''\rightarrow\delta_k'',\boldsymbol{r}_{k-N}^{k-1})}{p(\boldsymbol{r}_{k-N+1}^{k-1}|\delta_{k-1}'',u_{k-1})}p(u_{k-1}) \tag{2.8}
$$

Similarly, $\beta$ can be calculated using a backward recursion.

$$
\beta_{k+1}(\delta_{k+1}'') = \sum_{\{\delta_{k+2}''\}}\beta_{k+2}(\delta_{k+2}'')p(\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1}'',\delta_{k+2}'')p(\delta_{k+2}''|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1}'') \tag{2.9}
$$

where

$$
\begin{aligned}
p(\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1},\delta''_{k+2}) &= p(\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1},u_{k+1}) \\
&= \frac{p(\boldsymbol{r}_{k-N+2}^{k+1}|\delta''_{k+1},u_{k+1})}{p(\boldsymbol{r}_{k-N+2}^{k}|\delta''_{k+1},u_{k+1})} \\
&= \frac{\gamma(\delta''_{k+1}\rightarrow\delta''_{k+2},\boldsymbol{r}_{k-N+2}^{k+1})}{p(\boldsymbol{r}_{k-N+2}^{k}|\delta''_{k+1},u_{k+1})}
\end{aligned}
\tag{2.10}
$$

Equation (2.10) is obtained from Bayes' theorem and from equation (2.4). $p(\delta''_{k+2}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1})$ being independent of observations $\boldsymbol{r}_{k-N+2}^{k}$, $p(\delta''_{k+2}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1})=p(\delta''_{k+2}|\delta''_{k+1})$, which leads to $p(\delta''_{k+2}|\delta''_{k+1})=p(u_{k+1})$. We get the recursion on $\beta$ as follows:

$$
\beta''_{k+1}(\delta''_{k+1}) = \sum_{\{\delta''_{k+2}\}}\beta''_{k+2}(\delta''_{k+2})\frac{\gamma(\delta''_{k+1}\rightarrow\delta''_{k+2},\boldsymbol{r}_{k-N+2}^{k+1})}{p(\boldsymbol{r}_{k-N+2}^{k}|\delta''_{k+1},u_{k+1})}p(u_{k+1}) \tag{2.11}
$$

Finally gathering $\alpha,\beta$ and $\gamma$ leads to

$$
p(u_k|\boldsymbol{r}_{L-1}^{N_s-1}) \propto \sum_{\{\delta''_{k}\}}\alpha_k(\delta''_k)\gamma(\delta''_k\rightarrow\delta''_{k+1},\boldsymbol{r}_{k-N+1}^{k})\beta''_{k+1}(\delta''_{k+1})p(u_k) \tag{2.12}
$$

Using sufficient statistics at the output of the filter bank of Fig. **??** we have ([VCT10])

$$
p(\boldsymbol{r}_k|u_{k-L+1}^{k},\psi_k) \propto e^{\rho\cdot\Re\left(e^{-j\psi_k}r_{u_{k-L+1}^{k},k}\right)} \tag{2.13}
$$

where $\Re(.)$ stands for the real part, $\psi_k=\phi_k+\theta$ and $\rho=2\sqrt{Es}/N_0$. The branch metric associated to the TBR extended state space model requires the computation of the conditional probability related to $\gamma$ and given for any CPM by

$$
\begin{aligned}
\gamma(\delta''_k\rightarrow\delta''_{k+1},\boldsymbol{r}_{k-N+1}^{k}) &= p(\boldsymbol{r}_{k-N+1}^{k}|\delta''_k,u_k) \\
&\propto e^{\rho\cdot\Re\left(e^{-j\psi_{k-N+1}}\mu(u_{k-N-L+2}^{k})\right)}
\end{aligned}
\tag{2.14}
$$

where

$$\mu(u_{k-N-L+2}^{k}) = \sum_{i=k-N+1}^{k} r_{u_{i-L+1}^{i},i} \cdot e^{-j2\pi h \sum_{n=k-N-L+2}^{i-L} u_n}. \tag{2.15}$$

Simplified expression for the case of CPFSK can be found similar to the expression given in [VCT10]. At this point, we are at a crossroad. The derived expressions are valid for both the coherent and the non-coherent regimes, linking both of them in a unified framework. To derive the final version of forward, backward and transition kernel probabilities, we have to instantiate the above expressions for each type of regime regarding its coherency.

### 2.4.2 Non-coherent Detection

In non-coherent regime the phase induced by the channel is unknown. Averaging over the random phase $\psi$ removes the channel phase dependency from the branch metric and yields the well known zero-order modified Bessel function [VCT10] as given by.

$$p(\boldsymbol{r}_{k-N+1}^{k}|\delta_k'', u_k) \propto I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k})\right|\right) \tag{2.16}$$

Then equations (2.8) and (2.11) can be rewritten as

$$\begin{aligned}
\alpha_k(\delta_k'') &\propto \sum_{\{\delta_{k-1}''\}} \alpha_{k-1}(\delta_{k-1}'') \frac{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k-1})\right|\right)}{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+1}^{k-1})\right|\right)} p(u_{k-1}) \\
\beta_k(\delta_k'') &\propto \sum_{\{\delta_{k+1}''\}} \beta_{k+1}(\delta_{k+1}'') \frac{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k})\right|\right)}{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k-1})\right|\right)} p(u_k)
\end{aligned} \tag{2.17}$$

After careful examination of the above expressions, it can be noticed that the accumulated phase in states is no longer useful for the derivation of $\alpha$, $\beta$ and $\gamma$ after the averaging. Indeed the transition kernel probability turns to be independent of $\phi$ since the normalization removes the phase. Moreover we remark that $\forall\, \phi \in \mathcal{Q}$, the $\alpha_k(\delta_k'')$ (respectively $\beta_k(\delta_k'')$) are equals

for an identical series of symbols $u_{k-N-L+2}^{k-1}$. Then the forward and backward probabilities can be also released from $\phi$. The idea is to propose a state reduction for which the start of the observations are processed independently of $\phi$. Thus the extended TBR state is reduced to $\delta_k = \{u_{k-N-L+2}, ..., u_{k-1}\}$ (noted non-coherent TBR state). It corresponds to the state space initially proposed in [DS90] and leads to a state space complexity reduction compared to [CFR00]. The relationship with the former extended state space is the following. (The detailed computation is provided section A.0.2):

$$\begin{aligned}
\gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}_{k-N+1}^k) &= \tfrac{1}{Q} \cdot \sum_{\{\phi_{k-N+1}\}} \gamma(\delta_k'' \to \delta_{k+1}'', \boldsymbol{r}_{k-N+1}^k) \\
\alpha_k(\delta_k) &= \sum_{\{\phi_{k-N+1}\}} \alpha(\delta_k'') \\
\beta_{k+1}(\delta_{k+1}) &= \tfrac{1}{Q} \cdot \sum_{\{\phi_{k-N+2}\}} \beta_{k+1}(\delta_{k+1}'')
\end{aligned} \tag{2.18}$$

We now give *a new formulation* of $\alpha$, $\beta$ and $\gamma$.

$$\begin{aligned}
\gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}_{k-N+1}^k) &= p(\boldsymbol{r}_{k-N+1}^k | \delta_k, u_k) \\
\alpha_k(\delta_k) &= p(\boldsymbol{r}_{L-1}^{k-N} | \boldsymbol{r}_{k-N+1}^{k-1}, \delta_k) p(\delta_k) \\
\beta_{k+1}(\delta_{k+1}) &= p(\boldsymbol{r}_{k+1}^{N_s-1} | \boldsymbol{r}_{k-N+2}^k, \delta_{k+1})
\end{aligned} \tag{2.19}$$

After calculation, it leads to the following conditional probability

$$p(\boldsymbol{r}_{L-1}^{N_s-1} | u_k) = \sum_{\{\delta_k\}} \alpha_k(\delta_k) \gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}_{k-N+1}^k) \beta_{k+1}(\delta_{k+1}) \tag{2.20}$$

The transition kernel probability remains unchanged with regards to the TBR extended state space model since $\gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}_{k-N+1}^k) = \gamma(\delta_k'' \to \delta_{k+1}'', \boldsymbol{r}_{k-N+1}^k)$. Following the same reasoning than the extended state space approach, $\alpha$ and $\beta$ are computed in the following manner:

$$\begin{aligned}
\alpha_k(\delta_k) &\propto \sum_{\{\delta_{k-1}\}} \alpha_{k-1}(\delta_{k-1}) \frac{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+1}^{k-1})\right|\right)}{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k-1})\right|\right)} p(u_{k-1}) \\
\beta_k(\delta_k) &\propto \sum_{\{\delta_{k+1}\}} \beta_{k+1}(\delta_{k+1}) \frac{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k})\right|\right)}{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k-1})\right|\right)} p(u_k)
\end{aligned} \tag{2.21}$$

N.B. $I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k-1})\right|\right)$ is seen as a normalization factor which is not taken into account in [DS90].

### 2.4.3 Coherent Detection

For a coherent channel, $\theta$ is known and without loss of generality we would consider $\theta = 0$. Furthermore equations (2.8) and (2.11) are derived as

$$\begin{aligned}
\alpha_k(\delta_k'') &\propto \sum_{\{\delta_{k-1}''\}} \alpha_{k-1}(\delta_{k-1}'') \frac{e^{\rho \cdot \Re\left(e^{-j\phi_{k-N}\mu(u_{k-N-L+1}^{k-1})}\right)}}{e^{\rho \cdot \Re\left(e^{-j\phi_{k-N+1}\mu(u_{k-N-L+2}^{k-1})}\right)}} p(u_{k-1}) \\
\beta_k(\delta_k'') &\propto \sum_{\{\delta_{k+1}''\}} \beta_{k+1}(\delta_{k+1}'') \frac{e^{\rho \cdot \Re\left(e^{-j\phi_{k-N+1}\mu(u_{k-N-L+2}^{k})}\right)}}{e^{\rho \cdot \Re\left(e^{-j\phi_{k-N+1}\mu(u_{k-N-L+2}^{k-1})}\right)}} p(u_k)
\end{aligned} \tag{2.22}$$

Numerator and denominator terms in (2.22) can be cancelled, then the forward-backward probabilities are reduced as follows.

$$\begin{aligned}
\alpha_k(\delta_k'') &\propto \sum_{\{\delta_{k-1}''\}} \alpha_{k-1}(\delta_{k-1}'') e^{\rho \cdot \Re\left(e^{-j\phi_{k-N}} r_{u_{k-N-L+1}^{k-N}, k-N}\right)} p(u_{k-1}) \\
\beta_k(\delta_k'') &\propto \sum_{\{\delta_{k+1}''\}} \beta_{k+1}(\delta_{k+1}'') e^{\rho \cdot \Re\left(e^{-j\phi_k} r_{u_{k-L+1}^{k}, k}\right)} p(u_k)
\end{aligned} \tag{2.23}$$

We notice that equation (2.23) that the $N$ correlated symbols in states are no longer useful for the derivation of the forward probabilities. Indeed, $\alpha_k(\delta_k'')$ only depends on $\alpha_{k-1}(\delta_{k-1}'')$ and the observation corresponding to the first $L-1$ symbols of $\delta_{k-1}''$ and the symbol $u_{k-1}$ doing the transition $\{\delta_{k-1}'' \rightarrow \delta_k''\}$. Likewise $\beta_k(\delta_k'')$ only depends on $\beta_{k+1}(\delta_{k+1}'')$ and the

observation corresponding to the last $L-1$ symbols of $\delta''_k$ and the symbol $u_k$ doing the transition $\{\delta''_k \rightarrow \delta''_{k+1}\}$. Thus the forward (respectively backward) probability can be released from its $N-1$ last symbols (respectively from its $N-1$ first symbols). In a way, this trellis-based detector does the transition between state $\{\phi_{k-N+1}, u_{k-N-L+2}, ..., u_{k-N}\}$ and state $\{\phi_{k+1}, u_{k-L+1}, ..., u_k\}$ following the path $u_{k-N+1}^{k}$. Then, the extended TBR states can be obviously reduced to $\delta'_k = \{\phi_k, u_{k-L+1}, ..., u_{k-1}\}$ (noted Coherent TBR state). It corresponds to the state space described by Rimoldi [Rim88]. The relationship with the former extended state space is the following (for a detailed proof we refer the reader to the appendix A.0.3):

$$\begin{aligned}
\alpha_k(\delta'_k) &= \sum_{\{u_{k-N-L+2}^{k-L}\}} \alpha_k(\delta''_k) p(\boldsymbol{r}_{k-N+1}^{k-1} | \delta''_k) \\
\beta_{k+1}(\delta'_{k+1}) &= \beta_{k+1}(\delta''_{k+1}) \\
\gamma(\delta'_k \rightarrow \delta'_{k+1}, \boldsymbol{r}_{k-N+1}^{k}) &= \frac{\gamma(\delta''_k \rightarrow \delta''_{k+1}, \boldsymbol{r}_k)}{p(\boldsymbol{r}_{k-N+1}^{k-1} | \delta''_k)}
\end{aligned} \tag{2.24}$$

We now give *a new formulation* of $\alpha$, $\beta$ and $\gamma$ as follows:

$$\begin{aligned}
\gamma(\delta'_k \rightarrow \delta'_{k+1}, \boldsymbol{r}_k) &= p(\boldsymbol{r}_k | \delta'_k, u_k) \\
\alpha_k(\delta'_k) &= p(\boldsymbol{r}_{L-1}^{k-1} | \delta'_k) p(\delta'_k) \\
\beta_{k+1}(\delta'_{k+1}) &= p(\boldsymbol{r}_{k+1}^{N_s-1} | \delta'_{k+1})
\end{aligned} \tag{2.25}$$

It leads to the following conditional probability.

$$p(\boldsymbol{r}_{L-1}^{N_s-1} | u_k) = \sum_{\{\delta'_k\}} \alpha_k(\delta'_k) \gamma(\delta'_k \rightarrow \delta'_{k+1}, \boldsymbol{r}_k) \beta_{k+1}(\delta'_{k+1}) \tag{2.26}$$

Following the same reasoning than the extended state space approach, $\alpha$, $\beta$ and $\gamma$ are computed in the following manner:

$$
\begin{aligned}
\alpha_k(\delta'_k) &\propto \sum_{\{\delta'_{k-1}\}} \alpha_{k-1}(\delta'_{k-1}) e^{\rho \cdot \Re\left( (e^{-j\phi_{k-1}}) \cdot r_{u_{k-L}^{k-1}, k-1} \right)} p(u_{k-1}) \\
\beta_k(\delta'_k) &\propto \sum_{\{\delta'_{k+1}\}} \beta_{k+1}(\delta'_{k+1}) e^{\rho \cdot \Re\left( (e^{-j\phi_k}) \cdot r_{u_{k-L+1}^{k}, k} \right)} p(u_k) \\
\gamma(\delta'_k \rightarrow \delta'_{k+1}, \boldsymbol{r}_k) &\propto e^{\rho \cdot \Re\left( (e^{-j\phi_k}) \cdot r_{u_{k-L+1}^{k}, k} \right)}
\end{aligned}
\tag{2.27}
$$

## 2.5 Non-coherent TBR: a novel approach

In the previous section, we showed that the authors added unnecessarily to the state space provided in [DS90] the information phase generating a *redundant* state space. As a results, we were able to average the metrics, arising from the extended state space approach, over the accumulated phase without performance loss (see Fig. 2.3 b.). In the following, to sum-up and clarified the non-coherent TBR equations implementation, we rederive the metrics of this receiver, but now, directly based on the optimal state space originally proposed in [DS90].

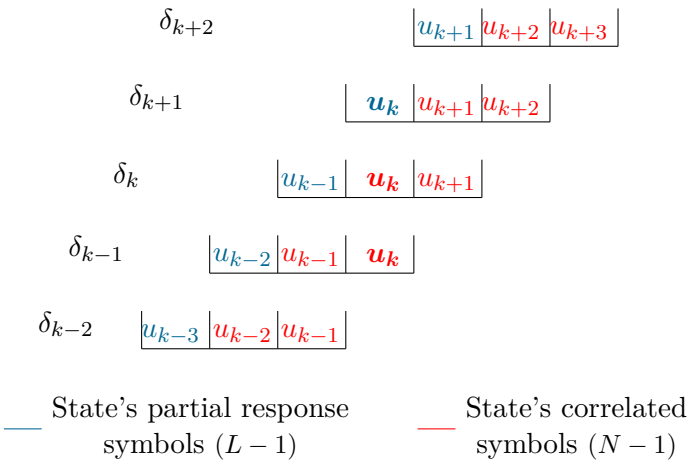



Figure 2.2: Non-coherent TBR State Space Model for N=3 and L=2

We start from the conditional probability of a symbol given the observations noted $p(u_k|\boldsymbol{r}_{L-1}^{N_s-1})$ and the sufficient and the minimal state space: $\delta_k = \{u_{k-N-L+2}, ..., u_{k-1}\}$. The transition between two states $\{\delta_k \rightarrow \delta_{k+1}\}$ corresponds to the transmitted symbol $u_k$. Based

on this minimal state space model, we re-derive a modified version of the BCJR algorithm. At first, the conditional probability is developed as follows.

$$p(u_k|\boldsymbol{r}_{L-1}^{N_s-1}) \propto \sum_{\{\delta_k\}} \alpha_k(\delta_k)\beta_{k+1}(\delta_{k+1})\gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}_{k-N+1}^{k})p(u_k) \tag{2.28}$$

where $\gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}_{k-N+1}^{k}) = p(\boldsymbol{r}_{k-N+1}^{k}|\delta_k, u_k)$, $\alpha_k(\delta_k) = p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta_k)p(\delta_k)$, $\beta_{k+1}(\delta_{k+1}) = p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k}, \delta_{k+1})$.

The forward-backward recursions can be calculated as

$$\begin{aligned} \alpha_k(\delta_k) &= \sum_{\{\delta_{k-1}\}} \alpha_{k-1}(\delta_{k-1}) \frac{\gamma(\delta_{k-1}\to\delta_k, \boldsymbol{r}_{k-N}^{k-1})}{p(\boldsymbol{r}_{k-N+1}^{k-1}|u_{k-1}, \delta_{k-1})} p(u_{k-1}) \\ \beta_k(\delta_k) &= \sum_{\{\delta_{k+1}\}} \beta_{k+1}(\delta_{k+1}) \frac{\gamma(\delta_k\to\delta_{k+1}, \boldsymbol{r}_{k-N+1}^{k})}{p(\boldsymbol{r}_{k-N+1}^{k-1}|u_k, \delta_k)} p(u_k) \end{aligned} \tag{2.29}$$

The branch metric can be computed as

$$\begin{aligned} \gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}_{k-N+1}^{k}) &= p(\boldsymbol{r}_{k-N+1}^{k}|\delta_k, u_k) \\ &\propto I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k})\right|\right) \end{aligned} \tag{2.30}$$

where $\rho = 2\sqrt{Es}/N_0$ and $I_0$ the modified zero order Bessel function of the first kind,

$$\mu(u_{k-N-L+2}^{k}) = \sum_{i=k-N+1}^{k} r_{u_{i-L+1}^{i},i} \cdot e^{-j2\pi h \sum_{n=k-N-L+2}^{i-L} u_n} \tag{2.31}$$

which finally gives the following recursions

$$\begin{aligned}
\alpha_k(\delta_k) &\propto \sum_{\{\delta_{k-1}\}} \alpha_{k-1}(\delta_{k-1}) \frac{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+1}^{k-1})\right|\right)}{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k-1})\right|\right)} p(u_{k-1}) \\
\beta_k(\delta_k) &\propto \sum_{\{\delta_{k+1}\}} \beta_{k+1}(\delta_{k+1}) \frac{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k})\right|\right)}{I_0\left(\rho \cdot \left|\mu(u_{k-N-L+2}^{k-1})\right|\right)} p(u_k)
\end{aligned} \tag{2.32}$$

The full details of this derivation is provided in the appendix B.

## 2.6 Complexity computation

This section aims to compare the computational effort associated to the extended state space approach with the proposed minimal state space in non-coherent regime.

### 2.6.1 Extended state space

The computational effort associated to the detection algorithm based on the extended state space approach may be determined by the analysis of 4 equations: The computation of the conditional probability, the kernel probability and the forward-backward probabilities recursion.

We start by evaluating the number of operations required by the kernel probability $I_0(\rho \cdot |\mu(u_{k-N-L+2}^k)|)$. The well known zero-order modified Bessel function has to be evaluated for each $M^{N+L-1}$ values taken by $\rho \cdot |\mu(.)|$. $I_0(.)$ may be stored in a table for once and for all since it does not depend on the observations. Then the magnitude taken by each $\mu(.)$ is multiplied by the scaling factor $\rho$. Likewise it corresponds to $M^{N+L-1}$ multiplications. Thereafter the number of operations required by the computation of $\mu(.)$ is more complex to evaluate. $\mu(u_{k-N-L+2}^k) = \sum_{i=k-N+1}^{k} r_{u_{i-L+1}^i, i} \cdot exp(-j2\pi h \sum_{n=k-N-L+2}^{i-L} u_n)$ is the sum over $N$ complex terms that corresponds to $N-1$ complex additions or $2(N-1)$ real additions. The complex factor $exp(-j2\pi h \sum u_n)$ may be stored in a table since it is not a function of the received signal. Then each of the $N$ complex terms is the product of two complex num-

bers. Thus it generates $N$ complex multiplications equivalent to $4N$ real multiplications and $2N$ real additions. Those operations are repeated for each of the $M^{N+L-1}$ series of correlated symbols. To sum up, the kernel generates $8NM^{N+L-1}$ operations. The normalization factor $I_0(\rho \cdot |\mu(u_{k-N-L+2}^{k-1})|)$ found in the forward-backward probabilities can be computed together with $I_0(\rho \cdot |\mu(u_{k-N-L+2}^{k})|)$ since the derivation of $\mu(u_{k-N-L+2}^{k})$ follows the derivation of $\mu(u_{k-N-L+2}^{k-1})$. Only the $M^{N+L-2}$ multiplications with $\rho$ and $M^{N+L-2}$ evaluations of $I_0(.)$ must be taken into account as for the kernel. Thereafter we evaluate the number of operations requires for the forward probability.

$\alpha_k(\delta_k'')$ is the sum of $M$ terms and each therm is the product of 4 elements. Thus $\alpha_k(\delta_k'')$ requires $M-1$ additions and $3M$ multiplications. Those operations are repeated for all the possible values taken by $\delta_k''$ thus $QM^{N+L-2}$ times. To sum up, the forward probability requires $QM^{N+L-2}(4M-1)$ operations. The same amount of operations are necessary to compute the backward probability. The complexity of the conditional probability is computed as follows.

$p(u_k|\boldsymbol{r}_{L-1}^{N_s-1})$ is the sum of $QM^{N+L-3}$ real terms and each term is the product of 4 elements. Thus it corresponds to $QM^{N+L-3}-1$ additions and $3QM^{N+L-3}$ multiplications. $p(u_k|\boldsymbol{r}_{L-1}^{N_s-1})$ is computed for each of the $M$ possible symbols. Again to sum up, the conditional probability requires $M(4QM^{N+L-3}-1)$ operations. As for the zero-order modified Bessel function, $p(u_k)$ may be store in a table and its computation is not taken into account in the calculus of the algorithm complexity.

Finally the amount of operations per CPM symbol used by this algorithm is $2M^{N+L-2}(4NM+4QM+Q+1)-M$. ***Thus the complexity of the algorithm is*** $\mathcal{O}(8(Q+N)M^{N+L-1})$.

### 2.6.2 Non-coherent TBR

The kernel metric has the same complexity than the extended state space in non-coherent regime. Thus we study the computational effort associated to the conditional probability and the forward-backward probabilities

$\alpha_k(\delta_k)$ is the sum of $M$ terms and each term is the product of 4 elements. Thus $\alpha_k(\delta_k)$ ne-

cessitates $M-1$ additions and $3M$ multiplications. Those operations are repeated for all the possible values taken by $\delta_k$ thus $M^{N+L-2}$ times. To sum up, the forward probability requires $M^{N+L-2}(4M-1)$ operations. The same amount of operations are necessary to compute the backward probability. The complexity of the conditional probability is computed as follows.

$p(u_k|\boldsymbol{r}_{L-1}^{N_s-1})$ is the sum of $M^{N+L-3}$ real terms and each term is the product of 4 elements. Thus it corresponds to $M^{N+L-3}-1$ additions and $3M^{N+L-3}$ multiplications. $p(u_k|\boldsymbol{r}_{L-1}^{N_s-1})$ is computed for each of the $M$ possible symbols. Again to sum up, the conditional probability requires $M(4M^{N+L-3}-1)$ operations. As for the zero-order modified Bessel function, $p(u_k)$ may be store in a table and its computation is not taken into account in the calculus of the algorithm complexity.

Finally the amount of operations per CPM symbol used by this algorithm is $4M^{N+L-2}(2NM+2M+1)-M$. ***Thus the complexity of the algorithm is*** $\mathcal{O}(8NM^{N+L-1})$.

For recall, this approach is the MAP extension of [DS90] that considers a Viterbi decoding. Obviously, we roughly have twice the complexity of the ML detection, that can be seen as a forward only version. Note that we address the symbolwise MAP decoding for enabling soft iterative non-coherent detection of CPMs signal (see chapter 4), which cannot be done using a Viterbi based detection. Finally, the non-coherent TBR has a lower complexity than [CFR00] evaluated as $\mathcal{O}(8(N+Q)M^{N+L-1}$.

## 2.7 Asymptotic Analysis

In this section, we perform an asymptotic analysis of the proposed receivers based on the three previous state spaces. First of all, the achievable information rate is derived based on the mutual information between the input and output process. The information rate gives the number of bits per second and per hertz the system can send per channel. Then, the spectral efficiency might be obtained by taking into account the bandwidth that contains a certain amount of power (generally 99%). Those asymptotic elements are very crucial when

we want to implement a coding scheme optimized for a given system. Indeed, the information rate corresponds to the maximal rate one can transmit over the channel to achieve reliable communication. De facto, it gives a good idea of how much the coding scheme may be improved. Thereafter we perform an Extrinsic Information Transfer (EXIT) charts analysis of the systems. EXIT chart is an efficient tool used to study the exchange of extrinsic information between the soft input and the soft output of decoders. More broadly, it is used to analyze the convergence behaviour of codes and to derive in certain cases an approximation of the achievable rate by computing the area under the curve (see [AKB04][Hag04]). Thus we will show in the sequel that the approximation by EXIT charts of the achievable rate and its theoretical value are the same. Moreover, we will gain

### 2.7.1 Mutual Information Rate

The mutual information rate of finite-state channels have been studied in [Arn+06] and [Yan04]. The mutual information rate between the channel input source $\mathcal{U}$ and the channel output $\mathcal{R}$ can be described as follows [YKT05].

$$I(\mathcal{U},\mathcal{R}) = \lim_{N_s\to\infty} \frac{1}{N_s} I(u_0^{N_s-1}, \boldsymbol{r}_{L-1}^{N_s-1} | \delta''_{N+L-2}) \tag{2.33}$$

We denote by $I(u_0^{N_s-1}, \boldsymbol{r}_{L-1}^{N_s-1} | \delta''_{N+L-2})$ the mutual information between the input process $u_0^{N_s-1}$ and the output process $\boldsymbol{r}_{L-1}^{N_s-1}$ conditioned in the initial state $\delta''_{N+L-2}$. The expression of $I(u_0^{N_s-1}, \boldsymbol{r}_{L-1}^{N_s-1} | \delta''_{N+L-2})$ can be derived as follows [CT12] (see appendix A.0.4 for more details)

$$\begin{aligned} I(u_0^{N_s-1}, \boldsymbol{r}_{L-1}^{N_s-1} | \delta''_{N+L-2}) = {} & \big(N_s - (N+L-2)\big)\, log_2(M) \\ & \cdot E\left[ log_2\left( p(u_0^{N_s-1} | \boldsymbol{r}_{L-1}^{N_s-1}, \delta''_{N+L-2}) \right) \right] \end{aligned} \tag{2.34}$$

$p(u_0^{N_s-1} | \boldsymbol{r}_{L-1}^{N_s-1}, \delta''_{N+L-2})$ which appears in equation (2.34) may be reduced to $p(u_{N+L-2}^{N_s-1} | \boldsymbol{r}_{L-1}^{N_s-1}, \delta''_{N+L-2})$ since $\delta''_{N+L-2} = \{\phi_{L-1}, u_0, ..., u_{N+L-3}\}$. It may be evaluated as follows.

$$p(u_{N+L-2}^{N_s-1}|\boldsymbol{r}_{L-1}^{N_s-1},\delta''_{N+L-2}) = \prod_{k=N+L-2}^{N_s-1} p(u_k|\delta''^{k}_{N+L-2},\boldsymbol{r}_{L-1}^{N_s-1}) \tag{2.35}$$

Computation of $p(u_k|\delta''^{k}_{N+L-2},\boldsymbol{r}_{L-1}^{N_s-1})$ in equation (2.35) can be found in [Pad+05]. The idea is to compute the probability of a symbol knowing perfectly all the previous states from the beginning of the transmission. This is equivalent to perform the BCJR algorithm as usual but taking into account the complete knowledge of the forward recursion i.e $\alpha$ is fixed to 1 for the correct state and 0 to all other states. $\gamma$ and $\beta$ remained unchanged beside the traditional BCJR. Computing $p(u_k|\delta''^{k}_{N+L-2},\boldsymbol{r}_{L-1}^{N_s-1})$ leads to complete calculation of the mutual information given by

$$\begin{aligned} I(u_0^{N_s-1},\boldsymbol{r}_{L-1}^{N_s-1}|\delta''_{N+L-2}) &= \big(N_s-(N+L-2)\big)\, log_2(M) \\ &\quad + E\left[\sum_{k=N+L-2}^{N_s-1} log_2\Big(p(u_k|\delta''^{k}_{N+L-2},\boldsymbol{r}_{L-1}^{N_s-1})\Big)\right] \end{aligned} \tag{2.36}$$

Then applying the reduced state proposed in sections 2.4.2 and 2.4.3 for both regimes we get.

$$\begin{cases} I_{nc}(u_0^{N_s-1},\boldsymbol{r}_{L-1}^{N_s-1}|\delta_{N+L-2}) = \big(N_s-(N+L-2)\big)\, log_2(M) \\ \qquad\qquad + E\left[\sum_{k=N+L-2}^{N_s-1} log_2\Big(p(u_k|\delta^{k}_{N+L-2},\boldsymbol{r}_{L-1}^{N_s-1})\Big)\right] \\ I_c(u_0^{N_s-1},\boldsymbol{r}_{L-1}^{N_s-1}|\delta'_{L-1}) = \big(N_s-(L-1)\big)\, log_2(M) \\ \qquad\qquad + E\left[\sum_{k=L-1}^{N_s-1} log_2\Big(p(u_k|\delta'^{k}_{L-1},\boldsymbol{r}_{L-1}^{N_s-1})\Big)\right] \end{cases} \tag{2.37}$$

Where $I_{nc}$ (respectively $I_c$) is the mutual information in the non-coherent (respectively co-

herent) regime. Then dividing by the source length ($N_s$), we get the mutual information rate.

### 2.7.2 Extrinsic Information Transfer charts

EXIT chart has been first introduced by Ten Brink in [TB01]. It visually characterizes the exchange of extrinsic information between the soft input and the soft output of decoders. More broadly, EXIT charts are used to analyze the convergence behaviour of codes. Hagenauer has pointed out in [Hag04] that the maximum achievable rate is approximately equal to the area under the EXIT curve for a given operating point. This property has been formally proven over the erasure channel [AKB04] for the case of Low-Density Parity-Check Codes (LDPC). In other words, with the experimental maximum achievable rate noted $\mathcal{R}^*$ and $I_e$ the mutual information at the output of the decoder

$$\mathcal{R}^* \simeq log_2(M) \cdot \int_0^1 I_e(x)\mathrm{d}x \tag{2.38}$$

To effectively compare various CPM, it is necessary to compute the achievable information rate under a bandwidth constraint. 1 Hz of available bandwidth is usually taken. Thus we define the normalized bandwidth as

$$\mathcal{B}_n = \mathcal{B}_{99} \cdot T \tag{2.39}$$

T is taken equal to 1 (since only 1 Hz bandwidth is available) and $\mathcal{B}_{99}$ is given as the bandwidth that contains 99% of the power of the uncoded complex baseband signal $\mathbf{s}$. If we note $\Gamma_s$ the power spectral density of the signal $\mathbf{s}$, then $\mathcal{B}_{99}$ is computed in the following manner (see for example [VCT10])

$$\int_{-\mathcal{B}_{99}/2}^{\mathcal{B}_{99}/2} \Gamma_s(f)\mathrm{d}f = 0.99 \int_{-\infty}^{\infty} \Gamma_s(f)\mathrm{d}f \tag{2.40}$$

Then the spectral efficiency is obtained by dividing the achievable information rate by the normalized bandwidth. This step permits, in some way, to quantify the influence of the system parameters $(h, L...)$ upon a bandwidth constraint.

## 2.8 Simulation results

In this section we present a performance evaluation of the receivers based on the previous state space. Three types of curves are displayed here under: EXIT charts trajectories, spectral efficiency and bit error rate (BER). Those simulations are provided for various type of CPM and parameters as exposed in Tab.1.1.

Fig.2.3 shows binary GMSK and quaternary 2RC EXIT charts for an operating point of $E_s/N_0 = 0\,dB$ and an observation length of 3 symbols (depending on the considered model and channel). Curves have been drawn in non-coherent regime based on: (a) the extended state space from [CFR00] noted N.C ESS, (b) the optimal state space model based on the ML metric presented in [DS90] noted N.C ML and (c) the N.C TBR state space proposed in section 2.4.2. The first obvious comment which can be made regarding the EXIT trajectories is that whatever the considered state space (the extended or the minimal one) those latter do not converge to the point (1,1) in non-coherent regime unlike in the coherent one. We would show in the sequel, to what extend this behaviour affects the design of the coding scheme.

Then, EXIT charts bring to light two major aspects of this chapter. First the receiver proposed in [CFR00] and the non-coherent TBR curves are superimposed, meaning the two models are equivalent from both the asymptotic behaviour and performance perspective (see Fig. 2.3 and 2.4). Yet the state space reduction permits to reach the minimal state space required for an optimal detection for a given complexity (ie. the number of correlated symbols). Thus the model proposed in [CFR00] may be reduced for equal performance (i.e the number

of state can be reduced from $Q \cdot M^{N+L-2}$ to $M^{N+L-2}$). Secondly, the ML metric ([DS90]) differs by the absence of the normalization part when compared with the MAP presented in (2.21). Both metrics have similar performance exclusively for zero a priori ($I_a = 0$). When the *a priori* information increases the non-coherent TBR outperforms [DS90]. It means iterative decoding will be more efficient with the non-coherent TBR metric than with the one provided in [DS90].

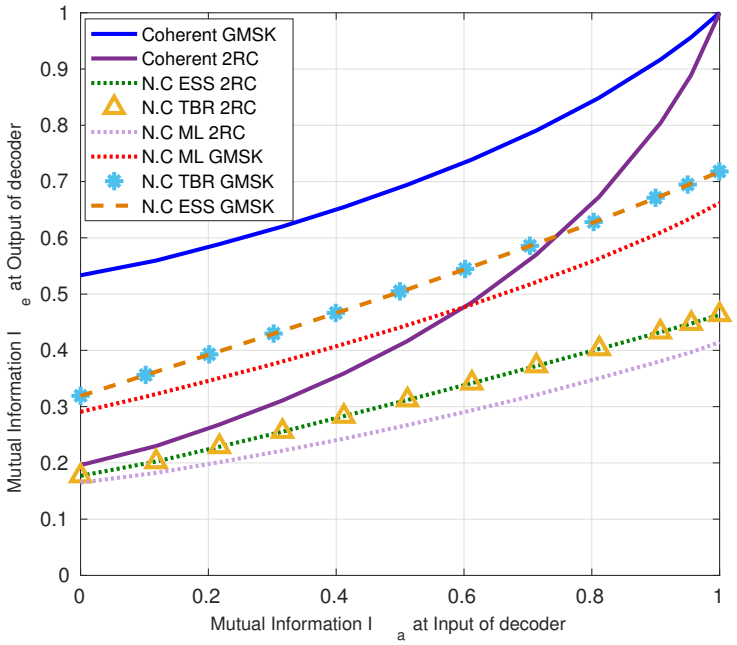
The spectral efficiency (based on (2.38) and EXIT chart approximation) and the bit error rate (BER) have been plotted in Fig. 2.4 and 2.3 (b.) for several types of CPM widely used in satellite and aeronautical communications [SG13]. First of all, it appears that the approximation by EXIT charts is relatively accurate when compared to direct calculation of the spectral efficiency. Secondly, those figures bring to light two important aspects: there is no information to gain from the state space expansion and the performance is improved when $N$ increases. Thus state $\delta_k$ is sufficient but also minimal since for a given observation length $N$, you cannot find a non coherent trellis MAP detector having less states without performance loss.

## 2.9 Phase deviation and frequency synchronization

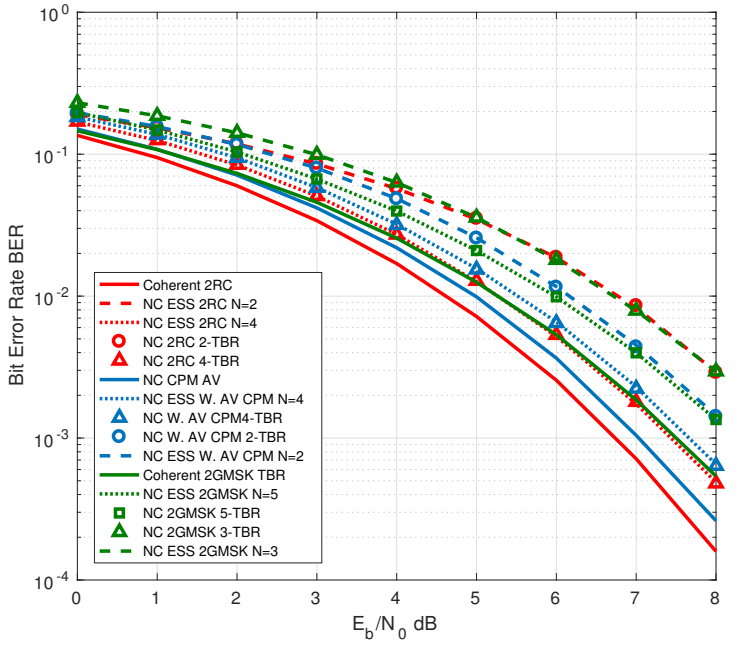
To conclude this chapter we evaluate the degradation of the non-coherent TBR to a phase deviation and a frequency synchronization error.

### 2.9.1 Phase Deviation

In this section, we evaluate the non-coherent TBR robustness to a channel undergoing a phase deviation. In that respect we redefined our phase model following the recommendation of [DS90]. The transmission of the CPM signal is affected by a phase noise noted $\theta(t)$ timely varying. The phase variation $\Delta_k$ between successive symbols is designed by an independent normal random variable with zero mean and a specified variance $\sigma_\Delta^2$ such as $\theta_{k+1} = \theta_k + \Delta_k$.

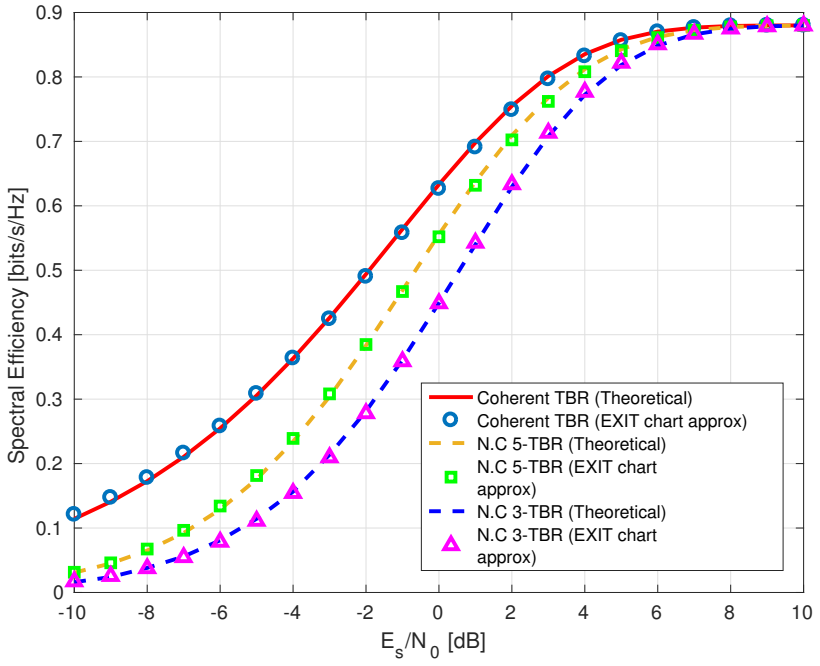



(a.)

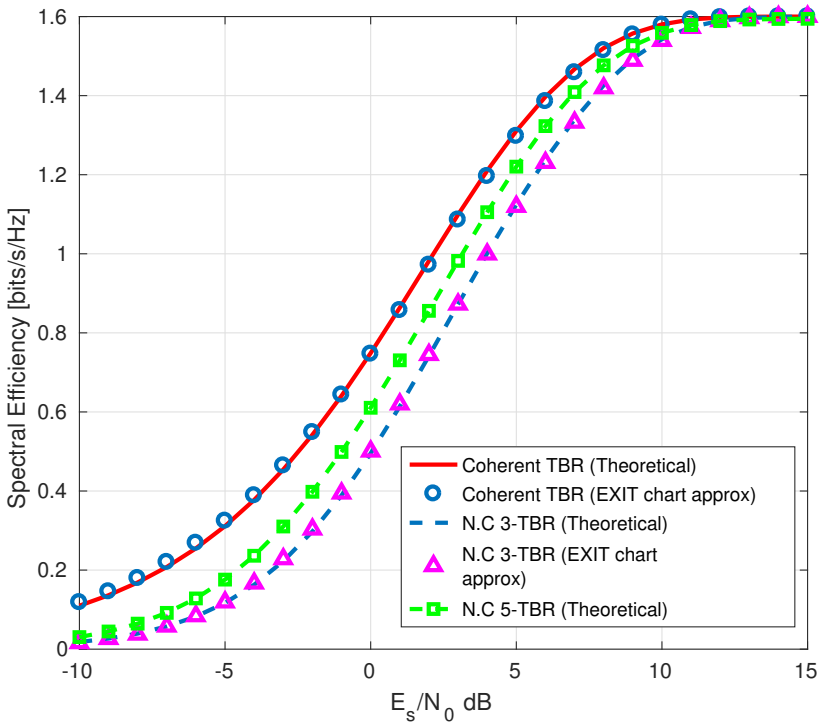



(b.)

Figure 2.3: (a) EXIT charts: 2GMSK $h = 1/2$, $M = 2$ and $BT = 0.25$ and 2RC with $h = 1/4$, $M = 4$ ($N = 3$ for both) (b). BER: 2GMSK with $h = 1/2, M = 2, BT = 0.25$, Weigthed AV CPM [SG13] $h = 1/3, M = 4$, 2RC $h = 1/3, M = 4$.




(a.)



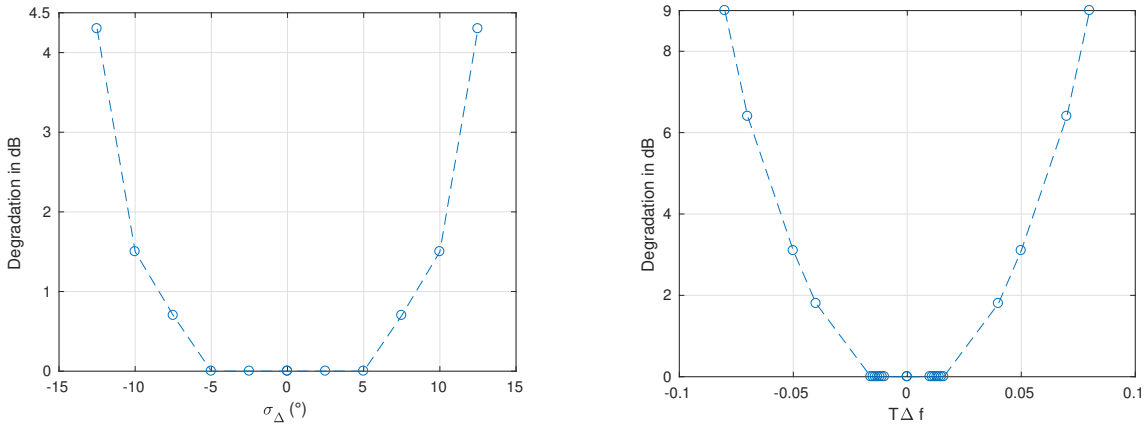

(b.)

Figure 2.4: SE of (a.) binary GMSK with $h = 1/2$, $L = 2$ and $BT = 0.25$ (b.) quaternary 2RC with $h = 1/4$.

Fig. 2.5 (a.) shows the degradation of the non-coherent symbol/MAP TBR for a 2GMSK (BER= $10^{-5}$) generated by a phase deviation. For instance for the 2GMSK ($N = 2, h = 1/2, M = 2, BT = 0.25$), a standard deviation of $\sigma_\Delta = 8°$ (respect. $12°$) generates 1dB (respect. 3dB) loss for a BER equal to $10^{-5}$.

### 2.9.2 Frequency Synchronization

In this section, we evaluate the non-coherent TBR robustness to the impact of a frequency synchronization error. To this end, we consider, in the following, that the frequency synchronization is not perfect and we try to study the behaviour of our algorithm. Consequently, we redesign our model to take into account a frequency shift $\Delta f$. The received signal can be rewritten as follows: $r_k(t) = e^{2j\pi\Delta f t}e^{j\theta} \cdot s_k(t) + n(t)$. Fig. 2.5 (b.) shows the degradation of the non-coherent symbol/MAP TBR for a 2GMSK (BER= $10^{-5}$) generated by a frequency shift. For instance for a 2GMSK ($N = 2, h = 1/2, M = 2, BT = 0.25$), a frequency shift of $T\Delta f = 2\%$ the symbol rate (respect. $3\%$) generates 0.5dB (respect. 1dB) loss for a BER equal to $10^{-5}$.




(a.) (b.)

Figure 2.5: Impact of (a.) phase deviation (b.) frequency synchronization in non-coherent regime over a binary GMSK, $E_b/N_0 = 12.8\,dB$ ($h = 1/2$ and $L = 2$)

## 2.10 Conclusion

The extended state space approach is a good tool to study in the meantime the coherent and the non-coherent regime. It was first proposed in [CFR00] to design the non-coherent symbol/MAP receiver but this state space was shwon not minimal. The authors included unnecessarily the accumulated phase to the non-coherent minimal state generating an extended state space. Indeed the phase information is useless since it can not be exploited between states due to the unknown phase rotation undergone during the transmission. Consequently, the state space might be reduced in non-coherent regime. The non-coherent minimal state space, for its part, was introduced by [RD99] to design the ML receiver. This chapter aimed to make the link between both state spaces and derived an optimal model for the non-coherent symbol/MAP detection. In that respect, we started from the extended state space and rederived completely the metrics associated to both regimes. Then, we instantiate the coherent and non-coherent metrics leading us to a ***state reduction*** specific to the channel.

Chapter 3

# CPM Precoding schemes

## Sommaire

## 3.1 Résumé

Dans ce chapitre, on souhaite adapter le comportement asymptotique des CPMs à l'utilisation de codes convolutifs dans un canal non-coherent. On a constaté au chapitre 2 que le comportement asymptotique des CPM différait suivant la cohérence du canal. En effet, dans un canal cohérent, la courbe d'information extrinsèque en sortie du démodulateur converge vers le point de coordonnées $(1, 1)$. Ce comportement permet de réaliser un décodage *itératif* sans palier d'erreur avec un code convolutif. Cependant, cela n'est plus le cas en régime non-cohérent où le point de convergence de l'information extrinsèque est en-deçà du point de coordonnées $(1, 1)$. Une solution possible serait de forcer la convergence de l'information extrinsèque vers le point $(1, 1)$ en régime non-cohérent. En s'inspirant des travaux réaliser par [Ben+07][Per+10] sur le pré-codage, on peut montrer qu'un pré-traitement des données peut modifier la trajectoire de l'information extrinsèque. Initialement, leur recherche visait à rendre les courbes d'information extrinsèque plates en régime cohérent afin d'atteindre le débit d'information maximal sans avoir à itérer. Cependant, il est à noter que leur méthode n'est efficace que pour les CPM binaires et n'est pas applicable au cas non-cohérent. Ainsi, au long de ce chapitre on tentera d'apporter une solution pour ces deux limitations. Dans un premier temps on montrera qu'il est possible de rendre, sous une certaine contrainte, les courbes d'information extrinsèque des CPMs non-binaires quasiment plates. Dans un deuxième temps, on proposera un précodage adapté au cas non-cohérent en vue de modifier la trajectoire de l'information extrinsèque en la faisant converger vers le point $(1, 1)$.

## 3.2 Introduction

In this chapter, we investigate an optimization of precoding schemes for CPMs. Traditionally those schemes were used to maximize the bit error rate (BER) of CPMs by carrying a processing of the symbols/bits. In that respect, [LT06] elaborated a precoding which increases the BER of the binary GMSK (with $h = 1/2$) while preserving the spectral efficiency. The authors' precoding was also extended to any 4-ary GMSK by expressing the 4-ary GMSK as a product of two 2-ary precoded GMSK. However the precoding is not restricted to BER improvement but may have various uses. [Mes+16] proposed precoded ternary CPMs with improved spectral efficiency. The authors' scheme is based on the concatenation of a binary input ternary output precoder and a ternary CPM. The binary to ternary mapping enable them to increase the minimal distance between symbols. This step leads to an improvement of the information rate. Then, their precoding scheme is able to restrict some transitions which are accountable to the bandwith expansion. Increase the information rate while limiting the spectrum occupation enables them to improve the spectral efficiency. The complexity of their scheme was thereafter reduced in [OSL17] based on the PAM decomposition and a duobinary encoding.

In what follows, we focus mainly on a precoding scheme enabling to adjust the asymptotic behaviour of the CPM and its demapper to classical convolutional code in non-coherent regime while preserving the spectral efficiency. A technique to adjust the asymptotic behaviour of the CPM demapper, was already studied in a different context in [Ben+07][Per+10]. The authors proposed a precoding to reach the information rate without iterative decoding. The approach consisted in modifying the CPE structure by connecting the partial response symbols and the accumulated phase to the input of the CPE. As a result, the obtained scheme serially concatenated with capacity achieving code performs close to the capacity without iterative decoding. Usually, extrinsic information transfert (EXIT) chart is favoured to describe the asymptotic behaviour of soft-input soft-output (SISO) decoder because of its low complexity. An analysis by EXIT aims to evaluate the transfer function of the SISO decoder. The transfer function takes as input an *a priori* information and generates at the output

an extrinsic information over the symbols/bits that needed to be decoded. The extrinsic information plotted against the *a priori* information is what we call an EXIT chart. Back to the precoding proposed in [Ben+07] and [Per+10], the precoding was able to flatten the CPM EXIT trajectories while preserving the area under the curve. [Hag04] pointed out that the achievable rate is approximately equal to the area under the EXIT curve for a given operating point. Based on this observation, the information rate under a flat EXIT curve is the same at any point, consequently there is indeed no longer need to iterate to increase the performance. Anyhow, the most useful point we retained from this work which will be useful in the sequel is: the precoding is able to change EXIT trajectories while keeping ***the same information rate***. In what follows, we call *coherent precoding*, the precoding drawn from [Ben+07] and *non-coherent precoding*, a precoding enabling efficient iterative detection based on classical convolutional schemes in non-coherent regime. As a first step, we formally introduce the EXIT charts and we deeply describe the precoding proposed in [Ben+07]. Then we design a precoding scheme which adjust the asymptotic behaviour of the CPM to classical convolutional code in non-coherent regime.

## 3.3 Extrinsic Information Transfer Chart

EXIT chart [Hag04], [TB01] helps to visually characterize the exchange of extrinsic information between soft input-soft output (SISO) concatenated components. The idea is to predict the behaviour of the iterative decoding solely by looking at the input/output relations of each of those components individually. The EXIT curve is obtained by setting the extrinsic information (noted $I_e$) of a given decoder on the ordinate and the *a priori* information (noted $I_a$) of this decoder on the abscissa. $I_e$ is the mutual information between the send symbols/bits and the extrinsic log-likelihood ratio generated at the output of the decoder noted $\mathcal{L}_e$. $I_a$ is the mutual information between the send symbols/bits and the LLRs distribution at the input of the decoder noted $\mathcal{L}_a$. In what follows, we assume the ergodicity of the *a priori* and *a posteriori* probabilities.

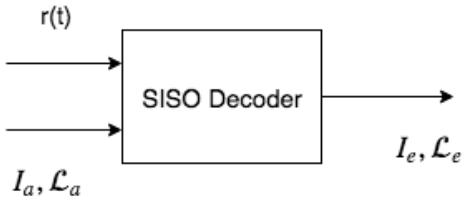



Figure 3.1: SISO Decoder

### 3.3.1 binary EXIT Chart

Let us denote $B$ the binary random variable associated to the transmitted bits with realizations $b \in \{\pm 1\}$. The random variable LLR $\mathcal{L}_a$ with realizations denoted $l_a$ associated to $B$ is designed as follows:

$$l_a = \frac{\sigma_a^2}{2} \cdot b + n_a \tag{3.1}$$

$n_a$ is an independent Gaussian random variable with zero mean and variance $\sigma_a^2$. Consequently the probability density function belonging to $\mathcal{L}_a$ denotes as:

$$p_{\mathcal{L}_a}(l_a|B=b) = \frac{1}{\sqrt{2\pi}\sigma_a} \cdot e^{-\frac{\left(l_a - \frac{\sigma_a^2}{2}\cdot b\right)^2}{2\sigma_a^2}} \tag{3.2}$$

The mutual information between the transmitted bits $B$ and the *a priori* LLR $\mathcal{L}_a$ is given as [CT12],[Ham86]:

$$I_a = I(B, \mathcal{L}_a) = \frac{1}{2} \sum_{b=\pm 1} \int_{-\infty}^{+\infty} p_{\mathcal{L}_a}(l_a|B=b) log_2 \left( \frac{2p_{\mathcal{L}_a}(l_a|B=b)}{p_{\mathcal{L}_a}(l_a|B=1) + p_{\mathcal{L}_a}(l_a|B=-1)} \right) dl_a. \tag{3.3}$$

Based on (3.1), the consistency condition i.e $p_{\mathcal{L}_a}(l_a|B=b) = p_{\mathcal{L}_a}(-l_a|B=b)e^{bl_a}$ and the symmetrical condition i.e $p_{\mathcal{L}_a}(l_a|B=b) = p_{\mathcal{L}_a}(-l_a|B=-b)$, the *a priori* information becomes

(under the hypothesis $B = b$):

$$\begin{aligned} I(B, \mathcal{L}_a) &= 1 - \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_a} \cdot e^{-\frac{(l_a - \frac{\sigma_a^2}{2} \cdot b)^2}{2\sigma_a^2}} log_2\left(1 + e^{-l_a}\right) dl_a \\ &= 1 - E\left[1 + e^{-\mathcal{L}_a}\right] \end{aligned} \tag{3.4}$$

The *a priori* information is a function of $\sigma_a$ so we denote $I_a(\sigma_a) = I(B, \mathcal{L}_a)$. More broadly, the mutual information $I(X, \mathcal{L})$ is a function of a given standard deviation $\sigma$ called the $J$-function. In our case we may write $I_a(\sigma_a) = J(\sigma = \sigma_a)$. A piece-wise polynomial approximation of the $J$-function has been given in [TBKA04].

$$J(\sigma) \simeq \begin{cases} 0.0421061\sigma^3 + 0.209252\sigma^2 - 0.00640081\sigma, & 0 \leq \sigma \leq 1.6363 \\ 1 - \exp\left(0.00181491\sigma^3 + 0.142675\sigma^2 + 0.0822054\sigma + 0.0549608\right), & 1.6363 < \sigma \leq 10 \\ 1, & 10 < \sigma \end{cases} \tag{3.5}$$

$$\sigma = J^{-1}(I) \simeq \begin{cases} 1.09542\ I^2 + 0.214217\ I + 2.33727\ \sqrt{I}, & 0 \leq \sigma \leq 0.3646 \\ -0.706692\ log[0.386013(1 - I)] + 1.75017\ I, & 0.3646 < \sigma \leq 1 \end{cases} \tag{3.6}$$

$I_e$ is the mutual information between the transmitted bits $B$ and the extrinsic LLR $\mathcal{L}_e$ generated at the output of the SISO decoder. In other words,the extrinsic LLRs are computed based on the binary probability vector returned at the output of the decoder. Similarly to

$I_a$, the extrinsic information is computed based on (3.4).

### 3.3.2 nonbinary EXIT Chart

In the binary case, we have seen that the mutual information between the transmitted bits and the LLR is well described by the variance of the LLR messages. In the nonbinary case, it becomes quite delicate because the LLR values exchanged between the concatenated components are not anymore single scalar values but are now represented by vectors which size is given by the modulation order. [KNH06] and [BB06] have proposed an effective method to compute the nonbinary EXIT chart. The mutual information is evaluated based on multivariate Gaussian distributed messages. Apart from the normalization factor (matching with the field or the modulation order) involved in the mutual information given by [BB06], the two formulations give similar results. The $J$-function associated to the mutual information is evaluated similarly to the binary counterpart but this time based on a co-variance matrix. With this in mind, the nonbinary EXIT chart computation process is performed in what follows. For ease of reading, we would reuse almost the same notation than the binary case.

Let us denote $C$ the symbol random variable associated to the transmitted symbols with realizations $c$ into the $q$-ary alphabet $\{0, ..., q-1\}$ ($q$ being a power of 2). Henceforth $\mathcal{L}_a$ is a multivariate Gaussian distributed with mean $m$ and a co-variance matrix $\Sigma$ such as ([BB06]):

$$m = \begin{pmatrix} \sigma_a^2/2 \\ \vdots \\ \sigma_a^2/2 \end{pmatrix} \quad \Sigma = \begin{pmatrix} \sigma_a^2 & \dots & \sigma_a^2/2 \\ \vdots & \ddots & \vdots \\ \sigma_a^2/2 & \dots & \sigma_a^2 \end{pmatrix} \tag{3.7}$$

$m$ is a $q-1$-dimensional vector and $\Sigma$ a $q-1 \times q-1$-dimensional matrix whose diagonal elements are equal to $\sigma_a^2$ and $\sigma_a^2/2$ otherwise. Consequently $\mathcal{L}_a$ is determined by the simple parameter $\sigma_a$. $\mathcal{L}_a$'s realization associated to the code word zero $C = 0$ is computed in the

subsequent manner:

$$l_a = G \cdot n_a + m \tag{3.8}$$

where $n_a$ is a multivariate Gaussian variable with zero mean and a co-variance matrix equal to the identity matrix and G results from the Cholesky decomposition of $\Sigma$ such as $\Sigma = GG^{\mathsf{H}}$. N.B this method computes the LLR for a transmitted message composed solely of the code symbol zero. Consequently, a circular permutation of the probability vector associated to $\mathcal{L}_a$ is needed. For ease of reading, let us present the following two operators: $LLR(.)$ and $LLR^{-1}(.)$. $LLR(.)$ computes the LLR based on a given probability vector. For instance, we apply this operator on the probability vector $\mathcal{P}_{\mathcal{L}_a} = [p_{\mathcal{L}_a}(l_a|C=0), ..., p_{\mathcal{L}_a}(l_a|C=q-1)]^\top$.

$$\begin{aligned} LLR(\mathcal{P}_{\mathcal{L}_a}) &= \left[ log\left(\frac{p_{\mathcal{L}_a}(l_a|C=0)}{p_{\mathcal{L}_a}(l_a|C=1)}\right), ..., log\left(\frac{p_{\mathcal{L}_a}(l_a|C=0)}{p_{\mathcal{L}_a}(l_a|C=q-1)}\right)\right]^\top \\ &= \left[\mathcal{L}_{a_1}, ..., \mathcal{L}_{a_{q-1}}\right]^\top \\ &= \mathcal{L}_a \end{aligned} \tag{3.9}$$

As well $LLR^{-1}(.)$ computes the probability vector from a given LLR vector. Applying this operator on $\mathcal{L}_a$ gives:

$$\begin{aligned} LLR^{-1}(\mathcal{L}_a) &= \frac{1}{1 + \sum_{k=1}^{q-1} e^{-\mathcal{L}_{a_k}}} \cdot \left[1, e^{-\mathcal{L}_{a_1}}, ..., e^{-\mathcal{L}_{a_{q-1}}}\right]^\top \\ &= \mathcal{P}_{\mathcal{L}_a} \end{aligned} \tag{3.10}$$

Thus, if the transmitted code word is $C = c$ then the rows of the probability vector $\mathcal{P}_{\mathcal{L}_a} = LLR^{-1}(\mathcal{L}_a)$ must be circularly permuted of a quantity equal to $c$. For ease of understanding,

we give the following example. Let us assume a vector $v_{ex} = [1\ 0\ 0]^\top$, a circular permutation $c = 2$ of vector $v_{ex}$ gives $[0\ 0\ 1]^\top$. Then after having permuted the probability vector to match the transmitted symbols, the associated permuted LLRs are obtained by the $LLR(.)$ function applied to the permuted probability vector. The mutual information between the transmitted symbols $C = 0$ and the *a priori* LLR $\mathcal{L}_a$ is given as [BB06]:

$$I(C = 0, \mathcal{L}_a) = 1 - E\left[log_q\left(1 + \sum_{k=1}^{q-1} e^{-\mathcal{L}_{a_k}}\right)\right] \tag{3.11}$$

Else, $I(C = c, \mathcal{L}_a) = I(C = 0, \mathcal{L}_a) + E\left[log_q\left(e^{-\mathcal{L}_c}\right)\right] \forall \quad c = 1...q-1$. Again the *a priori* information is a function of $\sigma_a$ so we denote $I_a(\sigma_a) = I(C, \mathcal{L}_a)$. Again we can write the mutual information $I(X, \mathcal{L})$ as a function of a given variance $\sigma$ like that $I_a(\sigma_a) = J(\sigma = \sigma_a)$. The $J$-function is defined similarly to its binary counterpart based on (3.11) and for a given order $q$. A piece-wise polynomial approximation or an approximation by splines might be set up to evaluate the $J$-functions. Similarly to the *a priori* case, $I_e$, the mutual information between the transmitted bits $C$ and the extrinsic LLR $\mathcal{L}_e$ generated at the output of the SISO decoder is computed based on (3.11). In other words, the extrinsic LLRs are computed based on the symbol probability vectors returned at the output of the decoder. Another method to compute the mutual information was proposed in [KNH06] based on the conditional probability. The authors' method is easier to use because the formulation is the same $\forall \quad c = 0...q-1$ contrary to [BB06]. Another difference is that their formulation is not normalized by $log_2(q)$ as done in [BB06]. Aside from this difference, the two methods give the same information rate.

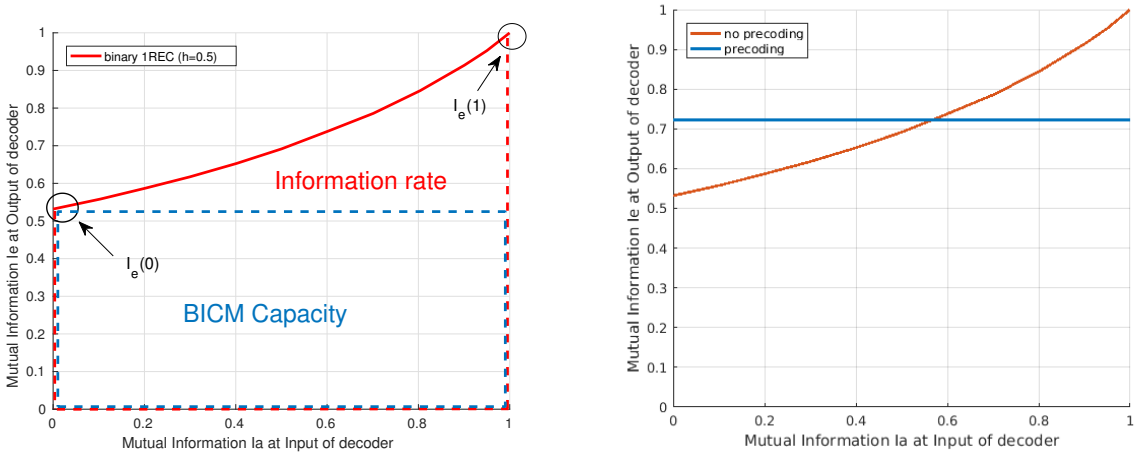
## 3.4 Precoding for maximization of the BICM capacity

### 3.4.1 Concept

We showed in chapter 2 that, if no specific precoding is used, most of CPM in use have EXIT transfer functions reaching the point $(1, 1)$. This enables the use of serially concatenated

convolutional outer codes or sparse graph codes to design simple BICM CPM schemes with no asymptotic error floors. In other words, there exists a SNR threshold above which we can achieve an arbitrary low probability of error. However the *sine qua none* condition to reach this low probability of error is to iterate between the CPM and the coding scheme. Consequently the decoding step complexity is enhanced by the number of required iterations to converge to the lowest BER at a given SNR. [Ben+07] proposed a precoding carried out on CPM to avoid the iteration process by reaching immediately the lowest BER at a given SNR. The aim was to obtain a scheme serially concatenated which performed close to the information rate without iterative decoding. In that respect, the precoding is used to maximize the non iterative BICM capacity referred to as the *pragmatic* capacity. From an EXIT charts perspective, the BICM capacity equals to the rate associated to the zero a priori ($I_a = 0$). It corresponds to the area under the curve $\mathcal{R}_{BICM} = I_e(0)$ on [0 1] while the information rate corresponds to the area under the whole EXIT chart ($\mathcal{R}^* \simeq \int_0^1 I_e(x)\mathrm{d}x$) so $\mathcal{R}_{BICM} \leq \mathcal{R}^*$ (see Fig. 3.2 (a.)). The precoding presented in [Ben+07] modifies the EXIT trajectories while keeping the same information rate (same area under the exit curve) and it can, for some specific cases as binary CPM, generate flat EXIT charts (see Fig. 3.2 (b.)) leading to $\mathcal{R}^* = I_e(I_a)\ \forall\, I_a \in [0\ 1]$ whence $\mathcal{R}^* = \mathcal{R}_{BICM}$. Thus, the BICM capacity and the information rate coincide for a specific precoding in use. De facto, iterative decoding becomes useless if capacity achieving error correcting schemes are used such as turbo LDPC or Polar Codes.

The precoding process proposed in [Ben+07] is operated inside the CPE structure. For recall, in Rimoldi's representation [Rim88], symbols are modulated through the serial concatenation of a continuous phase encoder (CPE) and a memoryless modulator (MM). This decomposition induces a natural mapping between information symbols and CPM waveforms. [Ben+07] has shown that it may be possible to improve the BICM capacity by changing this mapping. The approach consists in modifying the CPE structure by adding a state-dependent input mapping. The new mapping is called in [Ben+07] *dynamic mapping* and can be seen as a rate one precoding. The precoded input symbols are obtained by combining the information




(a.) (b.)

Figure 3.2: EXIT charts Binary 1REC $h = 1/2$ $E_b/N_0 = 0\,dB$ (***a.***) Information Rate (***b.***) Precoding Effect

input symbols, the precoded symbols required by the partial response and the accumulated precoding symbols taken from the start of the message. This solution preserves the phase continuity of the CPM signal and allows us to vary the input symbol mapping as a function of the symbols in memory and the accumulated phase. Then, the obtained symbols classically matched the transmitted continuous-time waveforms in the MM structure.

### 3.4.2 Modelization

Henceforth we denote $\bar{\delta_k}'' = \{\bar{\phi}_k, \bar{u}_{k-L+1}, ..., \bar{u}_{k-1}\}$ the coherent precoded TBR state (with $\bar{\phi}_{k+1} = \bar{\phi}_k + 2\pi h \cdot \bar{u}_{k-L+1}$). Precoded symbols are computed as follows:

$$\bar{u}_k = u_k \oplus d_k \tag{3.12}$$

Here $\oplus$ is the sum over $\mathbb{Z}_2^m$, so the $m$ bits of $u_k$ are added modulo 2 to the $m$ bits of $d_k$.

$$\boldsymbol{d}_k = [\bar{\boldsymbol{u}}_{k-1}, ..., \bar{\boldsymbol{u}}_{k-L+1}, \boldsymbol{\nu}_k] \cdot \mathbf{F}^\top \tag{3.13}$$

where

$$\nu_k = \sum_{i=0}^{k-L}{}_{Q} \bar{u}_i \tag{3.14}$$

$\bar{\boldsymbol{u}}_i$ is the *right-to-left* binary representation of symbol $\bar{u}_i$ such as $\bar{\boldsymbol{u}}_i = [\bar{u}_{0,i}, ..., \bar{u}_{j,i}, ..., \bar{u}_{m-1,i}]$ with $\bar{u}_{j,i}$ the $j^{th}$ bit of $\bar{u}_i$ (the same applies for $d_k$ with $\boldsymbol{d}_k$, $u_k$ with $\boldsymbol{u}_k$ and $\nu_k$ with $\boldsymbol{\nu}_k$). $\nu_k$ is the accumulated symbols directly related to the accumulated phase by the following relationship $\phi_k = 2\pi h \cdot \nu_k$ (with the initial phase $\phi_{L-1} = 0$) and $\sum_Q$ is the sum *modulo* $Q$. $\mathbf{F}^\top$ is the transpose of the $m \times r$-dimensional matrix $\mathbf{F}$ which components belong to $\mathbb{Z}_2$ ($\mathbf{F} \in \mathcal{M}_{m,r}(\mathbb{Z}_2)$) such as

$$r = m \cdot (L-1) + \lceil log_2(Q) \rceil \tag{3.15}$$

where $\lceil \cdot \rceil$ rounds to the next larger integer and $m$ is the number of bits per symbols ($m = log_2(M)$). Visually, $r$ corresponds to the number of branches connecting the precoded partial response symbols and the accumulated ones to the CPE input. For ease of presentation, an example of a precoding scheme inside the CPE structure is given Fig. 3.3 for a binary 2CPM with $Q = 2$ and $\mathbf{F} = [1\ 1]$.

### 3.4.3 Optimization procedure

The optimization procedure consists in selecting among the various precoding matrices $\mathbf{F}$ the ones maximizing the pragmatic capacity. From EXIT charts perspective, it is equivalent to select the precoding matrices generating flat EXIT charts. Back to our previous example, matrix $\mathbf{F} = [1\ 1]$ was able to flatten the EXIT trajectories of the binary 2GMSK.

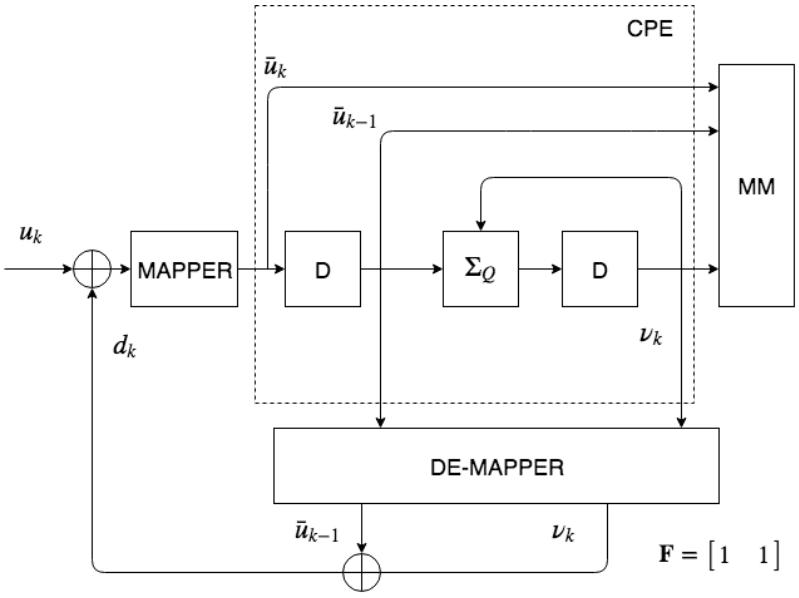
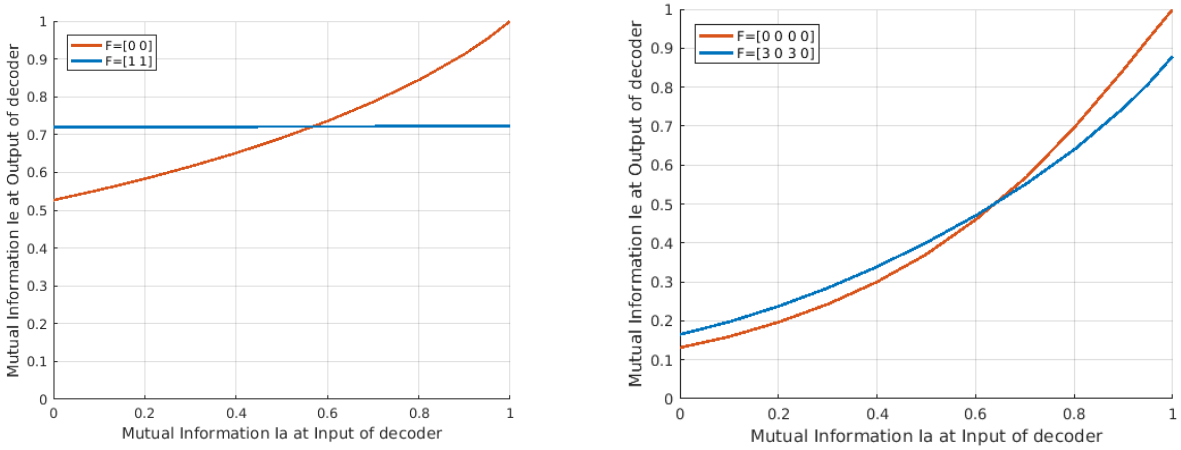



Figure 3.3: binary 2CPM ($Q = 2$) CPE diagram




(a.) (b.)

Figure 3.4: Exit charts of (a.) binary 2GMSK ($h = 0.5$) (b.) quaternary 2GMSK ($h = 0.25$, $BT = 0.5$) for $E_s/N_0 = 0\,dB$

### 3.4.4 Simulations and conclusions

Fig. 3.4 (a.) depicts the effect of the precoding over the binary 2GMSK for a $E_b/N_0 = 0\ dB$ from an EXIT point of view. As presented previously the Exit chart without precoding converges to the point $(1, 1)$ whereas the one provided by the modified CPE is completly flat. The two EXIT trajectories offer the same area under the curve. Consequently, the obtained BICM capacity is equal to the information rate and no iterative procedure is required. However the method is efficient only for binary CPM, indeed a significant gap appeared between the $M$-ary BICM capacity and the $M$-ary achievable rate (see Fig. 3.4 (b.)). An example of a $M$-ary CPM precoding scheme inside the CPE structure is given Fig. 3.5 for a quaternary 2CPM with $Q = 4$. In the M-ary configuration, regarding the best senario, the precoding will increase the information rate for a zero a priori and decrease the convergence point but without getting a flat EXIT curve (see Fig. 3.4 ***b.***). Then the BICM capacity does not reach the maximal achievable rate. De facto, iterating decoding will still be necessary to reach the maximal achievable rate. Furthermore, this method seems counterproductive for M-ary CPM after what has been presented in chapter 2. Iterate over EXIT charts which do not converge to point (1,1) generates an error with a convolutional codes. In this case, it is necessary to optimize a LDPC code based on profiles respecting specific constraints. Another drawback to this approach is that it is not suited to the non-coherent case. Indeed the coherent precoding exploits the phase information to map the input symbol $u_k$. Consequently it is necessary to know perfectly the accumulated phase to demodulate.

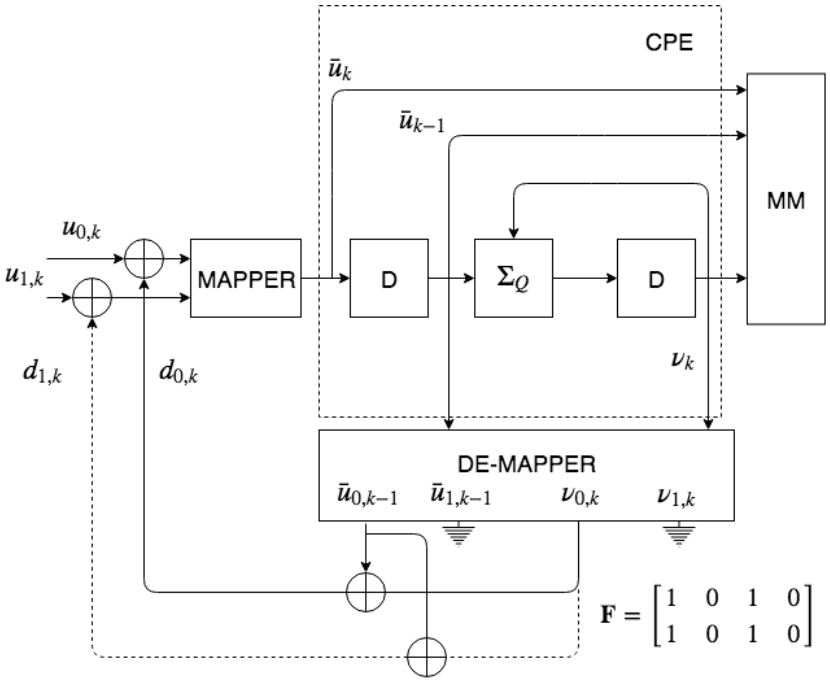



Figure 3.5: quaternary 2GMSK ($h = 0.25$, $BT = 0.5$) CPE diagram

## 3.5 A novel approach for non-binary CPM precoding for coherent detection

### 3.5.1 Modelization

We have seen along the previous section that the precoding proposed in [Ben+07] was efficient only for binary CPM, indeed a significant gap appeared between the $M$-ary CPM's BICM capacity and the $M$-ary CPM's information rate. Consequently, the authors' precoding was not able to flatten non-binary CPM EXIT curves. In our new approach we have assumed that since the precoding works well for binary CPM, the optimization dedicated to the M-ary case must be an generalized transposition of its binary counterpart. *De facto*, it appeared to us that the method implemented to maximize the CPM pragmatic capacity was not suited to M-ary CPM for two reasons. First, the authors carried out the precoding at a bit level whatever the modulation order. So they were obliged to convert the symbols to their bits form in the CPE block to apply their precoding. We assume here that the precoding should

be done at a symbol level without going back to bits by using an appropriate precoding vector. Secondly, the coding scheme, concatenated with the precoding message vector, was only considered over GF(2). However, an asymptotic study of the system through non-binary EXIT chart reveals that an adequat non-binary precoding may flatten the non-binary EXIT curves. Concequently, non-binary coding scheme should be taken into account in the precoded system to reach the information rate without iterative decoding. In what follows, we proposed a precoding scheme over $\mathbb{Z}_M$ allowing the $M$-ary CPM BICM capacity to reach the information rate. For ease of reading, we would reuse almost the same notation than in the binary case.

In that respect, precoded symbols are computed as follows:

$$\bar{u}_k = u_k \oplus d_k \tag{3.16}$$

Here $\oplus$ is the sum over $\mathbb{Z}_M$, so $u_k$ is added modulo $M$ to $d_k$.

$$d_k = [\bar{u}_{k-1}, ..., \bar{u}_{k-L+1}, \boldsymbol{\nu}_k] \cdot \mathbf{F}^\top \tag{3.17}$$

Where,

$$\nu_k = \sum_{i=0}^{k-L}{}_Q \, \bar{u}_i \tag{3.18}$$

$\boldsymbol{\nu}_k$ is the M-ary representation of $\nu_k$ such as $\boldsymbol{\nu}_k = [\nu_{0,k}, ..., \nu_{j,k}, ..., \nu_{q-1,k}]$ with $\nu_{j,k}$ the $j^{th}$ M-ary symbol of $\nu_k$ and $q = \lceil log(Q)/log(M) \rceil$. The astute reader will notice that $\nu_k$ is obtained modulo $Q$ whereas the symbols are M-ary. Thus, if $Q \leq M$, one symbol is enough to map $\nu_k$ but several symbols are needed when $Q > M$. $\mathbf{F}^\top$ is the transpose of the r-dimensional vector $\mathbf{F}$ which components belong to $\mathbb{Z}_M$ ($\mathbf{F} \in \mathbb{Z}_M^r$) with $r = (L-1) + q$. For the ease of presentation, an example of precoding scheme inside the CPE structure is given Fig.3.5 for a quaternary CPM with $Q = 4$ and $L = 2$, $\mathbf{F} = [2\ 3]$. $r$ branches connect the precoded partial

response symbols and the accumulated ones to the CPE input. Each branch is assigned a weight identified by one of the components of $\mathbf{F}$. Unlike the dynamic mapping proposed in [Ben+07], the proposed precoding is carry out at a symbol level. Consequently, the elements of $\mathbf{F}$ belongs to $\mathbb{Z}_M$. Note that for $M = 2$, we retrieved the same precoding scheme than [Ben+07].

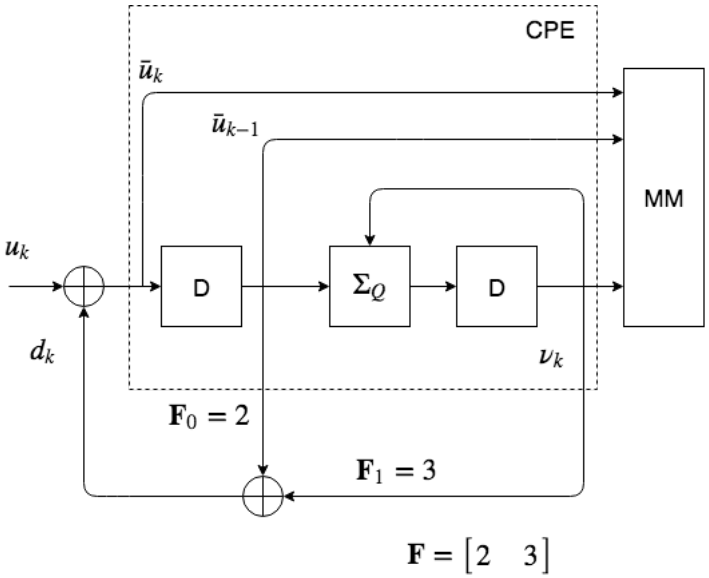
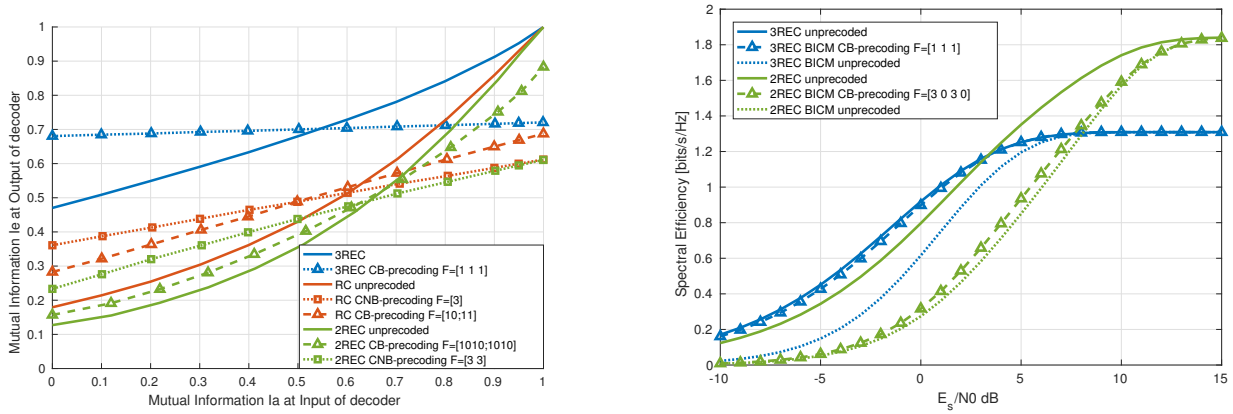



Figure 3.6: CPE diagram quaternary CPM with $\mathbf{F} = [3\,3]$, $L = 2$ and $Q = 4$

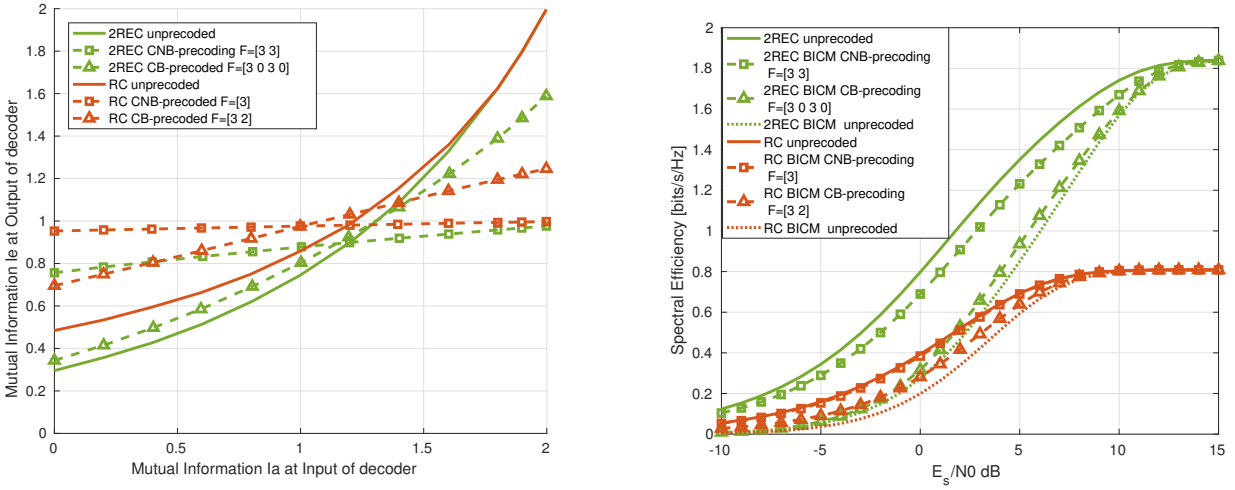
### 3.5.2 Optimization procedure

The optimization procedure is proceed as follows. We choose, among the $M^r$ possible precoding schemes, the one enabling a flat EXIT chart over GF($M$) or at least the one maximizing the extrinsic information for a null *a priori*. However, maximizing the pragmatic capacity through non-binary EXIT charts generates an additional constraint on the coding scheme. This latter must be designed over the order of finite field (GF($M$)) to preserve the special property of the precoded CPM scheme brought to light by the EXIT chart analysis. Eventually, based on this optimization procedure, we show some interesting results below.




(a.) (b.)

Figure 3.7: 3REC $h = 1/2$ $M = 2$ and 2REC $h = 1/4$, $M = 4$ (a.) Binary EXIT charts at $E_s/N_0 = 0\,dB$ (b.) Spectral Efficiency




(a.) (b.)

Figure 3.8: 2REC $h = 1/4$, $M = 4$, RC $h = 1/4$ $M = 4$ (a.) Non-binary EXIT charts at $E_s/N_0 = 0\,dB$ (b.) Spectral Efficiency

### 3.5.3 Simulations and conclusions

Fig.3.6 (a.) shows binary 3REC and quaternary 2REC EXIT charts at $E_s/N_0 = 0\ dB$ operating point. Two cases were considered: firstly, curves were plotted without [Ben+07]'s coherent precoding and secondly they were traced with an optimized matrix $\mathbf{F}$ based on [Ben+07]'s precoding (noted CB-precoding Fig.3.7 and 3.8) and called in the sequel coherent *binary* precoding to constrast with the coherent non-binary's one. As expected, an optimized CB-precoding combined with the binary 3REC offers a flat EXIT chart contrary to the quaternary 2REC. As a result, the binary CPM's BICM capacity superimposes the information rate whereas the $M$-ary CPM's one draws around $4dB$ far from the information rate at $E_s/N_0 = 0\ dB$ (see Fig.3.7). *De facto*, it is still necessary to iterate between the decoders in the $M$-ary case to reach the information rate even with CB-precoding turn on. The alternative possibility is to implement a precoding better suited to $M$-ary CPM as we did in this section. Again, the idea is to work entirely over $\mathbb{Z}_M$. The precoding and the CPM alphabet belong to $\mathbb{Z}_M$ and the coding to GF(M). In that respect, we have plotted, Fig.3.8 (b.), EXIT charts over GF(M) of quaternary 2REC and 1RC at $E_s/N_0 = 0\ dB$ operating point. For fair comparison we have drawn those curves considering both non-binary's (noted CNB Fig.3.8) and CB-precoding. It appears that the CNB-precoding considerably increases the extrinsic information at a zero *a priori* when directly compared to CB-precoding. The quaternary 1RC non-binary EXIT chart combined with the CNB-precoding is completely flat and nearly flat for the quaternary 2REC. As a result, the 1RC's BICM capacity with CNB-precoding superimposes the information rate. As well the 2REC's BICM capacity with CNB-precoding draws around $1dB$ far from the information rate as observed at $E_s/N_0 = 0\ dB$ Fig.3.8. Finally, the CNB-precoding bring the BICM capacity $3dB$ closer to the information rate than [Ben+07] at $E_s/N_0 = 0\ dB$. Those results are also confirmed in Fig.3.9 at finite length. To conclude, it seems that the best combination Precoding/CPM parameters is for $L = 1$, $h = 1/M$ and $F = [M]$. We can criticize our results by saying that, even if some CPM parameters combined with the CNB-precoding give very good results, we can't get systematically flat or nearly flat ($\leq$ 1dB from the information rate) non-binary EXIT charts but we still get an improvement

when directly compared to [Ben+07].

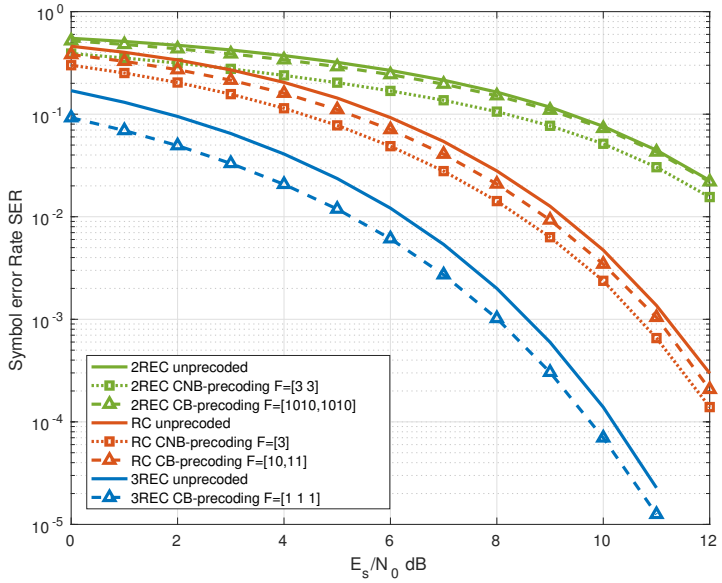



Figure 3.9: SER : 3REC $h = 1/2$ $M = 2$, 2REC $h = 1/4$, $M = 4$ and *RC* $h = 1/4$ $M = 4$

## 3.6 A novel approach to non-coherent precoding

### 3.6.1 Concept

We have seen in section 3.4.2 that the precoding proposed in [Ben+07] was able to change the EXIT charts trajectories in coherent regime while keeping the same area under the curve. Again, the autors' aim was to obtain a BICM capacity close to the information rate by adjusting the precoding. Visually, they turned the (1,1)-convergent EXIT chart into a flat chart in the case of binary CPM only. In non-coherent regime, a natural mapping (i.e no precoding) generates an EXIT chart which does not converge to point (1,1) as shown in Fig. 3.10 (a.). Consequently, an error floor appeared when iterating between the convolutional outer coding scheme and the CPM demodulator. Thus the solution to prevent such impairments are to make the EXIT chart converge to point (1,1) if an iterative process is feasible. We tested both CB and CNB-precoding of section 3.4 in non-coherent regime but it didn't worked

properly (see Fig. 3.10 (b.)). Indeed, the coherent precodings necessitate to know perfectly the accumulated phase to demodulate. This information is obviously missing in non-coherent regime since the CPM signal undergoes a unknown phase shift during the transmission. Consequently, we present, in what follows, a new precoding called *non-coherent binary (NCB) precoding* taking into account the constraints induced by the channel non-coherency. For ease of reading, we would reuse almost the same notation as for the coherent precodings.

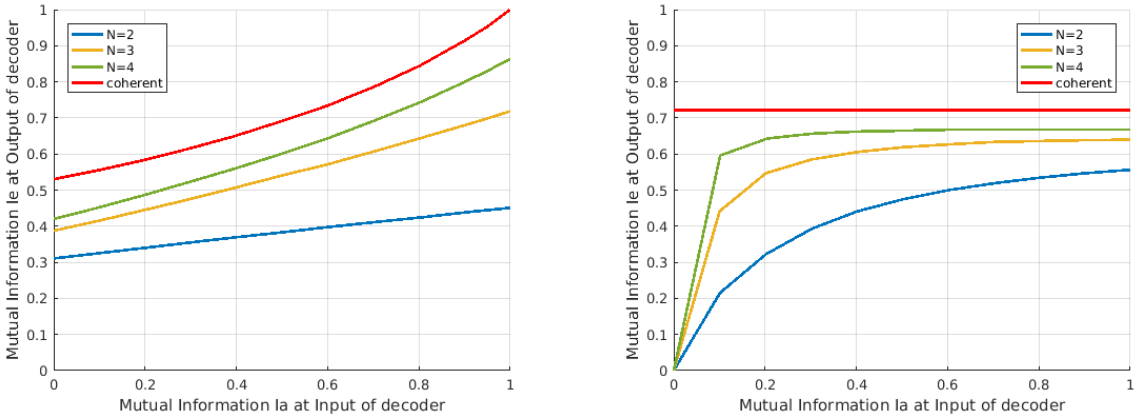



Figure 3.10: NC binary 2GMSK ($h = 0.5$ and $BT = 0.5$) (***a.***) $F = [0\,0]$ (***b.***) $F = [1\,1]$

### 3.6.2 Modelization of non-coherent binary precoding

The non-coherent binary precoding also called NCB-precoding for ease of reading is designed as follows. Henceforth we denote $\bar{\delta}_k = \{\bar{u}_{k-N-L+2}, ..., \bar{u}_{k-1}\}$ the precoded non-coherent TBR state where $\bar{\phi}_{k-N+2} = \bar{\phi}_{k-N+1} + 2\pi h \bar{u}_{k-N-L+2}$. Precoded symbols are computed as follows:

$$\bar{u}_k = u_k \oplus d_k \tag{3.19}$$

As for the coherent case, $\oplus$ is the sum over $\mathbb{Z}_2^m$, so the $m$ bits of $u_k$ are added modulo 2 to the $m$ bits of $d_k$.

$$\boldsymbol{d}_k = [\bar{\boldsymbol{u}}_{k-1}, ..., \bar{\boldsymbol{u}}_{k-L+1}, \tilde{\boldsymbol{\nu}}_{\boldsymbol{k}}] \cdot \mathbf{F}^{\top} \tag{3.20}$$

where

$$\tilde{\nu}_k = \sum_{i=k-N-L+2}^{k-L}{}_{Q} \; \bar{u}_i \tag{3.21}$$

$\tilde{\boldsymbol{\nu}}_k$ is the *right-to-left* binary representation of symbol $\tilde{\nu}_k$ such as $\tilde{\boldsymbol{\nu}}_k = [\tilde{\nu}_{0,k}, ..., \tilde{\nu}_{j,k}, ..., \tilde{\nu}_{m-1,k}]$ with $\tilde{\nu}_{j,k}$ the $j^{th}$ bit of $\tilde{\nu}_k$. $\tilde{\nu}_k$ is the accumulated symbols taken from the start of state $\bar{\delta}_k$. It corresponds to the $N$ correlated symbols belonging to $\bar{\delta}_k$. Parameters $\boldsymbol{d}_k$, $\mathbf{F}$, $\mathbf{F}^{\top}$, $\bar{\boldsymbol{u}}_i, \sum_Q$ are defined exactly as in the coherent case given section 3.4.2. For ease of presentation, an example of a precoding scheme inside the CPE structure is given Fig. 3.11 for a quaternary CPM with $\mathbf{F} = [1\,3\,1\,3]$, $L = 2$, $Q = 4$ and $N = 3$.

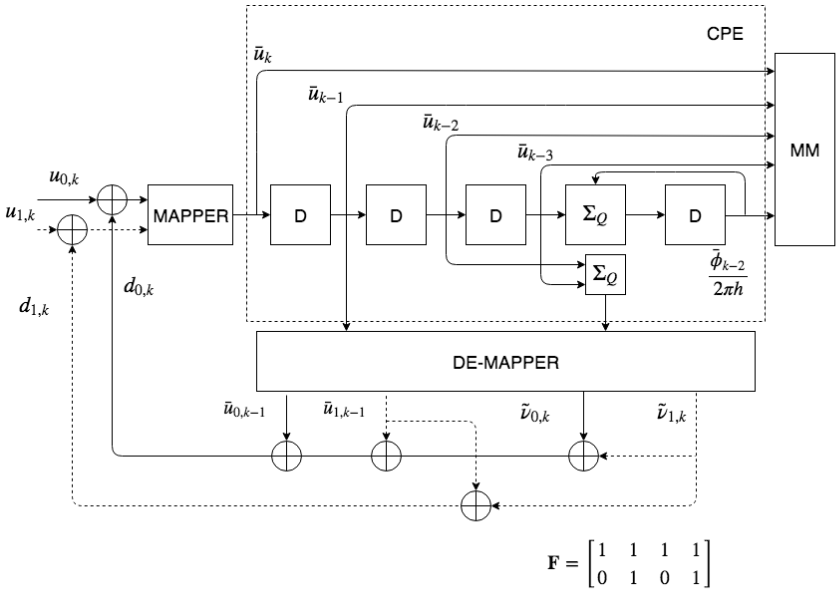



Figure 3.11: CPE diagram quaternary CPM with $\mathbf{F} = [1\,3\,1\,3]$, $L = 2$, $Q = 4$ and $N = 3$

The NCB-precoding equations are quite close to the coherent ones. The difference between the dynamic mapping proposed in [Ben+07] and the proposed NCB-precoding scheme dwells in the accumulated phase: since the coherent time is limited to the observation length in

non-coherent regime, only the accumulated symbols from the start of the observation are taken into account in the precoding scheme. Moreover, it seems not possible to make the EXIT charts completely flat based on the NCB-precoding. However we can manage to find a precoding matrix which makes the EXIT charts converge to point $(1, 1)$ for binary and non-binary CPM (see the optimization procedure section 3.6.4).

### 3.6.3 Modelization of non-coherent non-binary precoding

The non-coherent non-binary precoding also called NCNB-precoding, for ease of reading, is designed similarly to the coherent case. We give in the sequel the non-binary dual of the non-coherent precoding. First of all, the NCNB-precoding will not help getting flat EXIT charts contrary to its coherent counter part but it will enable the converge of EXIT charts as to its binary counter part. Thus it might be useful in the case of symbol interleaved coded modulation (SICM) systems in a non-coherent context. For ease of reading, we would reuse almost the same notation than the binary precoding. In that respect, precoded symbols are computed as follows:

$$\bar{u}_k = u_k \oplus d_k \tag{3.22}$$

Here $\oplus$ is the sum over $\mathbb{Z}_M$, so $u_k$ is added modulo $M$ to $d_k$.

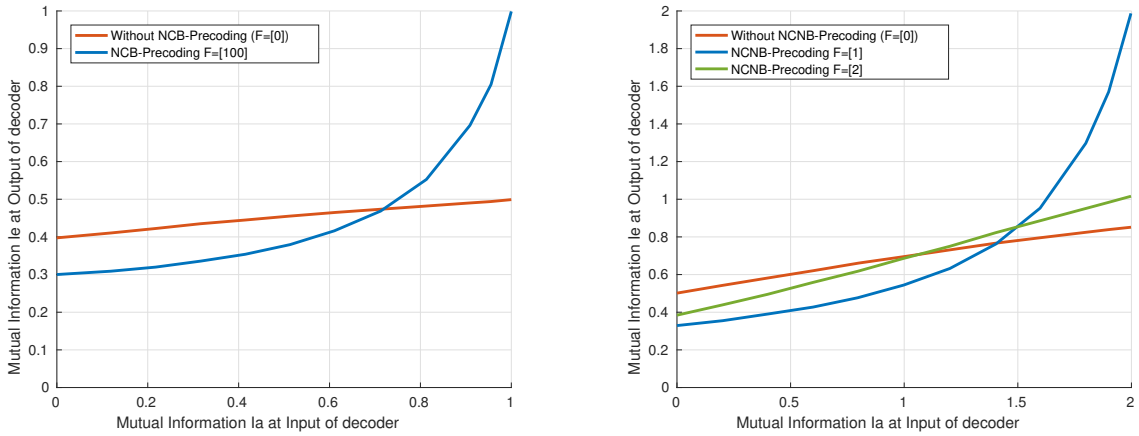
$$d_k = [\bar{u}_{k-1}, ..., \bar{u}_{k-L+1}, \tilde{\boldsymbol{\nu}}_k] \cdot \mathbf{F}^\top \tag{3.23}$$

Where,

$$\tilde{\nu}_k = \sum_{i=k-N-L+2}^{k-L}{}_{Q} \bar{u}_i \tag{3.24}$$

$\tilde{\boldsymbol{\nu}}_k$ is the M-ary representation of $\tilde{\nu}_k$ such as $\tilde{\boldsymbol{\nu}}_k = [\tilde{\nu}_{0,k}, ..., \tilde{\nu}_{j,k}, ..., \tilde{\nu}_{q-1,k}]$ with $\tilde{\nu}_{j,k}$ the $j^{th}$

M-ary symbol of $\tilde{\nu}_k$. $\tilde{\nu}_k$ represents the accumulated symbols taken from the start of state $\bar{\delta}_k$. It corresponds to the $N$ correlated symbols belonging to $\bar{\delta}_k$. Parameters $q, \mathbf{F}$, $\mathbf{F}^\top$, $\sum_Q$ are defined exactly as in the coherent case given in section 3.5. For ease of presentation, an example of a precoding scheme inside the CPE structure is given by Fig. 3.13 for a quaternary CPM with $\mathbf{F} = [2\,3\,1]$, $L = 2$, $Q = 5$ and $N = 3$.




(a.) (b.)

Figure 3.12: (a.) EXIT charts NCB-precoding CPFSK $h = 5/7$ and $M = 2$ (b.) non-binary EXIT charts NCNB-precoding CPFSK $h = 3/4$ and $M = 4$

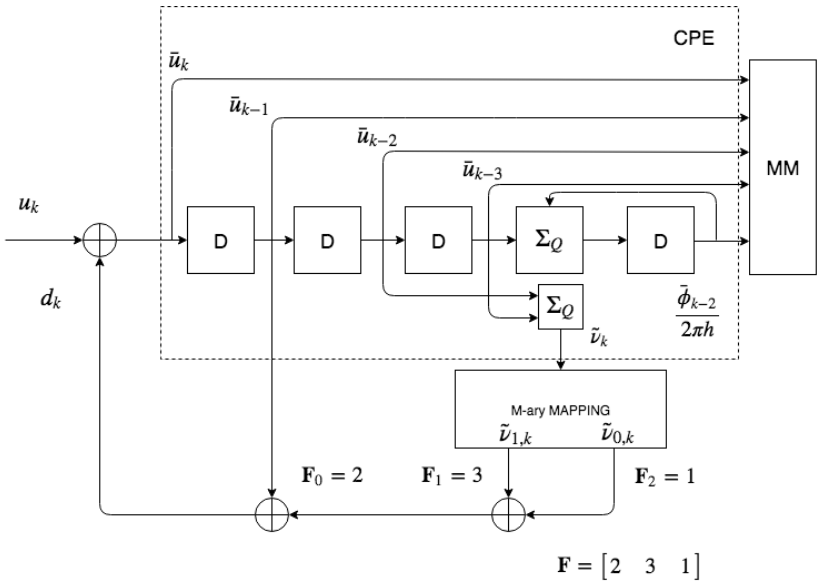



Figure 3.13: quaternary CPM with $\mathbf{F} = [2\,3\,1]$, $L = 2$, $Q = 5$ and $N = 3$

### 3.6.4 Optimization procedure

Before proposing an optimization procedure, we want to emphasize one important thing. In the coherent case, we have seen that if no specific precoding is used, most of CPMs in use have EXIT transfer functions reaching the point $(1, 1)$. This enables the use of serially concatenated convolutional outer codes to design simple BICM CPM schemes with no asymptotic error floors. The same thing can be achieved with sparse graph codes [TBKA04]. However, this is no longer the case in the non-coherent regime where EXIT curves do not reach the point $(1, 1)$ as it can be seen in Fig. 3.10 (a.). It will lead to an asymptotic error floor due to a crossing point between the outer convolutional code and the inner SISO CPM detector (Fig. 3.14), since the decoding process cannot be enhanced: the probability of error is decreasing with an increasing $E_s/N_0$ but it is bounded away from zero. We can ensure efficient decoding by tuning the non-coherent precoding to force the EXIT transfer function converge to the point $(1, 1)$. The idea is to select among the $2^{rm}$ possible precoding schemes, those converging to point $(1, 1)$. It was observed that, based on the proposed precoding procedure, various EXIT trajectories with equal information rate can be obtained for a given modulation and operating point. So, additionnaly, we select the EXIT curves with the highest extrinsic information corresponding to zero apriori information. Consequently an optimize $\mathbf{F}$ matrix must be included in the set of matrices offering convergent EXIT charts to point $(1, 1)$ (noted $\mathcal{F}_c$) and should have the highest extrinsic information for zero apriori among this set. The latter condition offers the best BICM capacity among $\mathcal{F}_c$, ie. if no iterative decoding is used. Various CPM optimal matrix $\mathbf{F}$ are displayed in Table 3.1 for the NCB-precoding.

### 3.6.5 Simulations

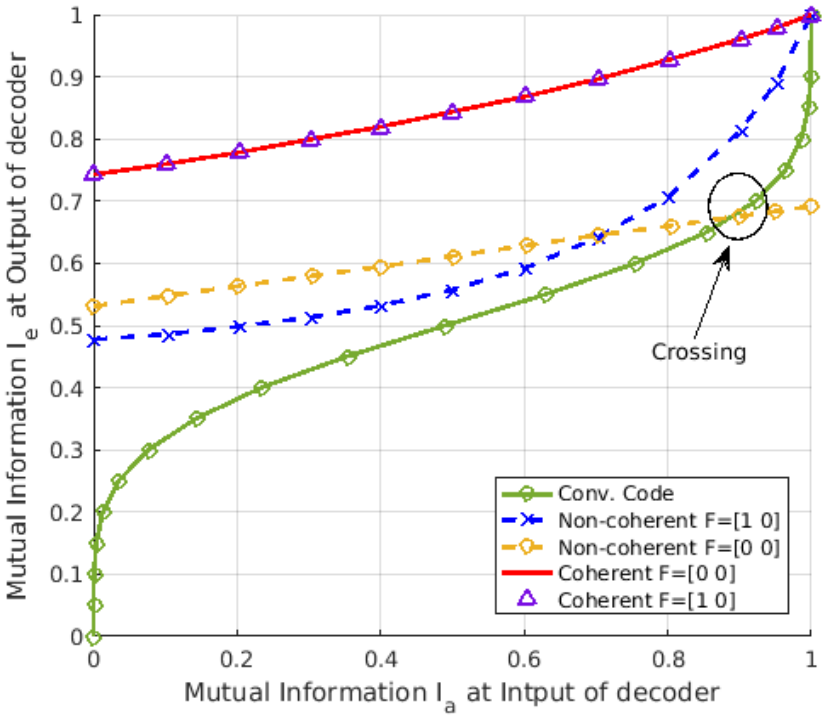
To illustrate our say, we plot on Fig. 3.14 the EXIT transfer function of: (a.) a binary GMSK for an operating point of $E_s/N_0 = 2.5\,dB$ and (b.) a quaternary 2RC for an operating point of $E_s/N_0 = 4.5\,dB$ for an observation length of 2 symbols. Two cases were considered in the non-coherent regime: a first exit curve has been plotted without precoding and a second one

Table 3.1: CPM optimal matrix for NCB-precoding

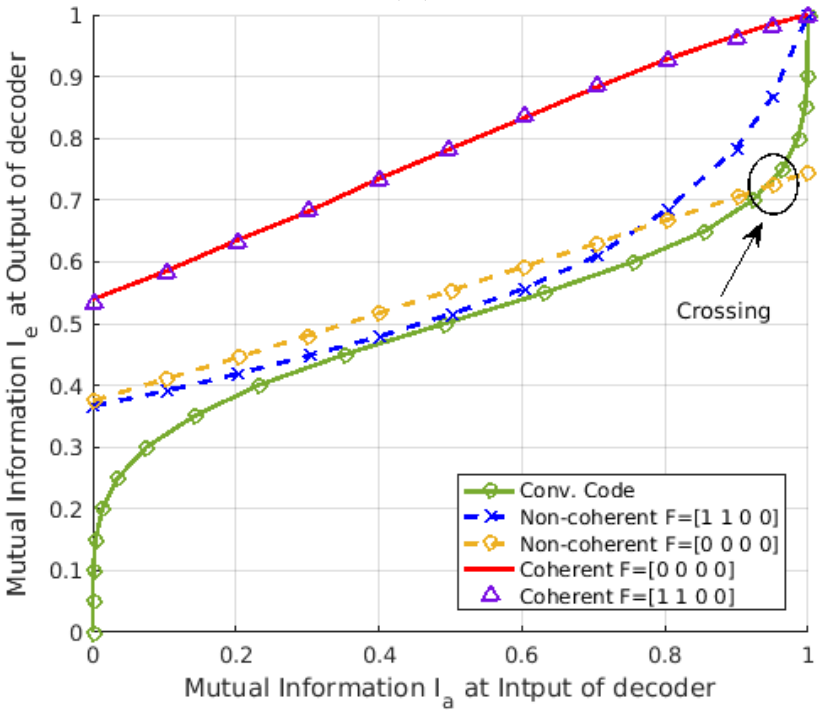
| CPM | **F** | $h$ | $M$ | $L$ |
|---|---|---|---|---|
| [SG13] Weighted (AV) ($\alpha_{rc} = 0.75$) | [1 1 0 0] | 1/3 | Quaternary | 2 |
| GMSK ($BT = 0.25$) | [1 0] | 1/2 | Binary | 2 |
| RC | [1 1 0 0] | 1/4 | Quaternary | 2 |
| CPFSK | [1 1 0] | 5/7 | Quaternary | 1 |

was plotted with an optimized matrix $\mathbf{F} = [1\,0]$ (respectively $\mathbf{F} = [1\,1\,0\,0]$ with the 2RC modulation). This matrix was obtained by the previous optimization procedure. Convolutional code EXIT chart of rate one-half with polynomials (5,7) in octal, has been added to Fig. 3.14 (a.) and (b.). It appears that the non-precoding EXIT curve intersects the convolutional code contrary to the one with the optimal precoding. Consequently, without precoding, the iterative procedure between the outer convolutional code and the CPM demodulator is stopped at the crossing level. This interruption in the decoding process will trigger an error floor. Non-coherent precoded and unprecoded EXIT curves in coherent regime were also studied. It seems that depending of the case the non-coherent precoding can have no effect on the coherent TBR.

The GMSK and REC bit error rates has been plotted on Fig. 3.15 (a.) and (b.) for the non-coherent regime. Binary messages were encoded by the considered rate one-half convolutional encoder with polynomials (5,7) in octal, for several coding length ($N_b = 512,\ 2048$, and 16000 bits). The unprecoded case is also reported. As expected, performance are enhanced by the precoded scheme, underlying the relevant use of CPM precoding in non-coherent regime. We can clearly observe that the BERs of precoded systems are characterized by the so called *waterfall* transition, whereas an error floor impairs the unprecoded systems performance. In addition, we can also notice that the coding length has an impact on the system performance.



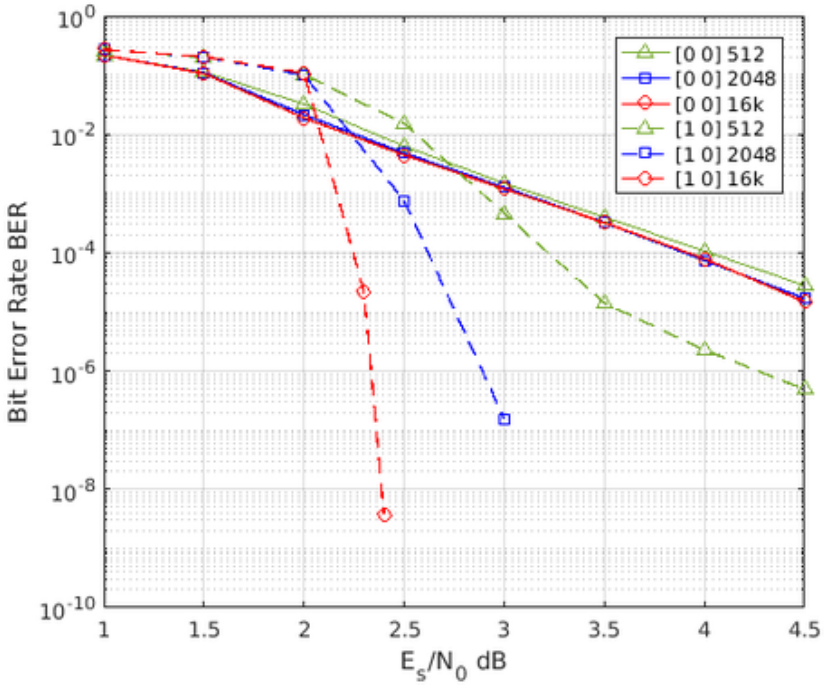

(a.)



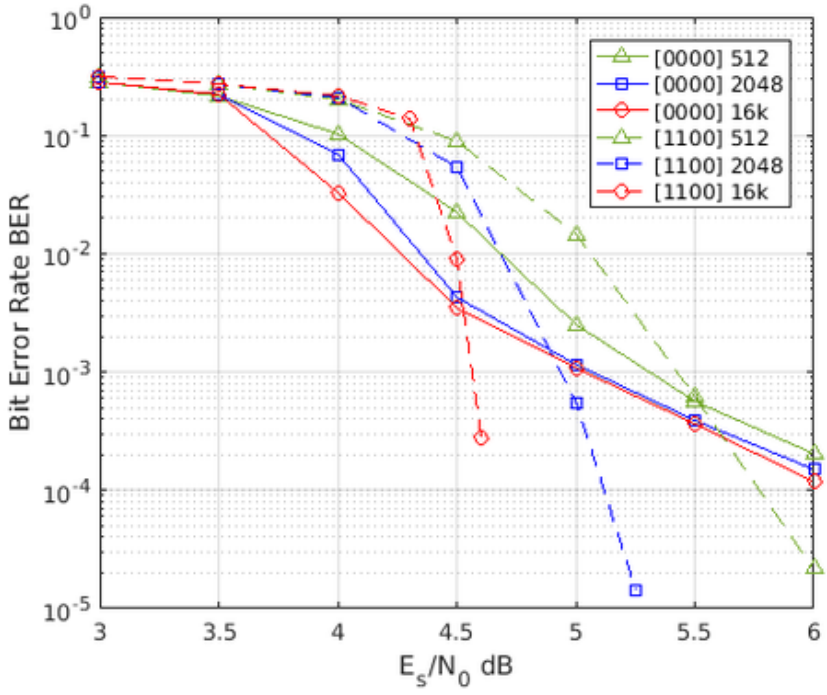

(b.)

Figure 3.14: Exit charts of NCB-precoding systems (a.) binary GMSK with $h = 1/2$, $L = 2$ and $BT = 0.25$ (b.) quaternary 2RC with $h = 1/4$

A large coding length behaves better than a smaller one.




(a.)




(b.)

Figure 3.15: NC BER of NCB-precoding systems (a.) binary GMSK with $h = 1/2$, $L = 2$ and $BT = 0.25$ (b.) quaternary 2RC with $h = 1/4$

Chapter 4

# CPM coding schemes

**Sommaire**



## 4.1 Résumé


Dans ce chapitre on souhaite élaborer des codes concaténés en série avec les modulations CPM fonctionnant efficacement, à la fois dans un canal de propagation cohérent et non-cohérent. À cet effet, on orientera notre recherche sur les codes en graphes creux de type LDPC. Ces

derniers appartiennent à la classe des codes en bloc et fonctionnent proche de la limite de Shannon. Cette limite peut être approchée grâce à un décodage itératif conjoint entre le code LDPC et la modulation CPM. Il est alors nécessaire que le décodeur LDPC et le démodulateur CPM présentent un comportement asymptotique similaire. Généralement, on fixe les paramètres de la modulation en premier puis on recherche un profil de code qui lui correspond. L'étude du comportement asymptotique est réalisé par l'analyse des courbes de transfert d'information extrinsèque (EXIT). Ces courbes correspondent à la fonction de transfert d'un décodeur entre l'information *a priori* des bits d'entrées et l'information extrinsèque des bits de sorties (pour un rapport signal-sur-bruit donné). On choisit alors parmi les profils de codes possibles celui faisant correspondre les courbes EXIT du décodeur LDPC avec celles du récepteur CPM. Pour rappel, notre intention est de trouver un code LDPC fonctionnant bien dans les deux régimes (cohérent et non-cohérent). Cependant, on a montré au chapitre 2 que les courbes EXIT différaient suivant le cohérence du canal. Ainsi un code optimisé pour un canal de propagation cohérent ne sera pas adapté au canal de propagation non-cohérent et vice-versa. Cela peut conduire, dans le meilleur cas, à une pénalité en débit et, dans le pire cas, à un palier d'erreur. On tentera au long de ce chapitre d'analyser et d'optimiser le codage pour ces deux regimes et de trouver des solutions permettant d'avoir des schémas efficaces dans les deux modes.

## 4.2 Introduction

In this chapter we aim to design coding schemes serially concatenated with CPM performing well for both the coherent and the non-coherent regimes. In that respect, we mainly focused on sparse graph codes such as low-density parity-check (LDPC) codes over GF(q) (with q power of 2). LDPC codes belong to the class of linear block codes and are described by the *low density* of their parity check matrices. They were originally invented by Gallager in his PhD thesis [Gal63] but remained not successfully considered for 35 years due to their huge complexity. Except for Tanner work [Tan81] who introduced the graphical representation of LDPC codes

and called later on Tanner graph. They were rediscovered by MacKay [MN96],[MMC98] and by Richardson & Urbanke [RU01] after the apparition of turbo codes. Those latter were invented by Berrou et al. in 1993 [BG96], and revolutionized the field of coding by proposing a coding scheme performing close to the Shannon limit. The principle was to concatenate two convolutional codes separated by an interleaver and to perform a joint iterative decoding. Over that period, the concept of iterative decoding greatly benefited to CPM systems. As wells as Turbo-Codes, LDPC can operate close to the Shannon limit. In that context we aim to optimize the joint (iterative) decoding between the CPM and the LDPC scheme while being suited to both regimes. First of all, an optimal joint decoding requires a similar convergence behaviour between the coding schemes. Thereby, it is necessary to carry out an asymptotic analysis of both coding schemes and see if their iterative features match. Usually, the CPM scheme is fixed and we study the coding scheme which would suit it. It exists several tools to study the convergence behaviour of codes such as the density evolution (DE) or the extrinsic information transfert (EXIT) chart. Those two are very popular, they were developed respectively by Richardson [RU01] and Ten Brink [TB01]. However, due to the computational complexity of the DE, we mainly focuse on EXIT chart to design the joint decoding of the concatenated system. An analysis by EXIT aims to evaluate the transfer function of the soft-input soft-output (SISO) decoder. The transfer function takes as input an *a priori* information and generates at the output an extrinsic information over the symbols/bits that needed to be decoded. The extrinsic information plotted against the *a priori* information is called EXIT chart. In the case of LDPC codes, the EXIT chart analysis allows the generation of *code profiles* essential to the construction of the parity check matrix [Ben15][Ben+14a]. This kind of optimization process is quite classical, usually the constraint is located on the design of the parity-check matrix based on those profiles. Going back to our issue, we aim to design a LDPC code serially concatenated with CPM suited to the coherent and the non-coherent regimes. We have pointed out along chapter 2 and 3 that EXIT trajectories of the CPM detector vary depending on the considered regime. Indeed, in the coherent case, most of the CPM have EXIT transfer functions reaching the point $(1, 1)$, whereas it is no longer the case in non-coherent regime if no specific precoding is used. The

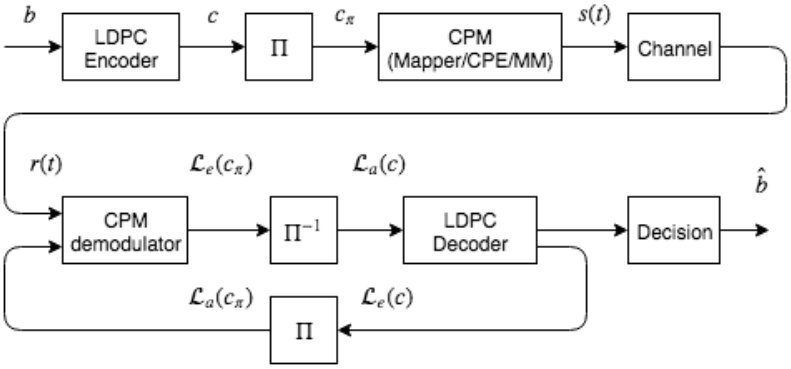



Figure 4.1: Coded interleaved CPM scheme with iterative decoding

profiles and the stability condition of the sparse graph code will be different from one regime to another. Consequently, a coding scheme optimized in coherent regime will not be suited in non-coherent regime and *vice versa*. In the best cases, a coding scheme unsuitable to the propagation channel will generate a capacity penalty and in the worst cases, an error floor. Then finding a good coding scheme performing well in both regimes seems difficult at first sight but we will show in the sequel how to override this problem.

## 4.3 Serially concatenated coded CPM scheme

In this chapter, we consider a serially concatenated coded scheme where an LDPC encoder is concatenated with a CPM modulator as depicted in Fig. 4.1. The log-likelihood ratios (LLR): extrinsic, *a priori* and *a posteriori* are respectively noted $\mathcal{L}_e$, $\mathcal{L}_a$ and $\mathcal{L}_{ap}$ Fig. 4.1.

At the transmitter side, a binary message vector $\mathbf{b} = [b_0, \cdots, b_{K_b-1}] \in GF(2)^{K_b}$ is encoded into a codeword $\mathbf{c} = [c_0, \cdots, c_{N_b-1}] \in GF(2)^{N_b}$ using an LDCP encoder of rate $R = K_b/N_b$. Each binary codeword $\mathbf{c}$ is then interleaved, mapped into a sequence of $N_s$ $M$-ary symbols from the considered modulation. Let $u_0^{N_s-1} = \{u_0, ..., u_{N_s-1}\}$ be the resulting set of $N_s$ symbols belonging to the $M$-ary alphabet $\{0, ..., M-1\}$. Symbols are then modulated following the CPM modulation rule. At the receiver side, a joint iterative decoding is performed between the LDPC decoder and the CPM modulator, then a decision over the bits is carried out based on the LLR *a posteriori* generated at the output of the LDPC decoder.

## 4.4 Introduction to LDPC codes

### 4.4.1 binary LDPC codes

#### 4.4.1.1 General presentation

LDPC codes are sparse graph codes providing performance close to the Shannon limit. They belong to the class of linear block codes and are described by the *low density* of their parity check matrices (noted **H**). **H** is a $\mathcal{M} \times \mathcal{N}$ matrix which is said full rank when the code rate $\mathcal{R} = 1 - \mathcal{M}/\mathcal{N}$. **H** is a binary matrix and the number of 1's in a row (column) is called the degree of that row (column). From a Tanner graph point of view, the rows (columns) match with nodes called variable (check) nodes and the variable (check) node degree corresponds to the number of edge connecting the variable (check) node to the check (variable) nodes. A polynomial representation of the variable and check nodes degree distribution from the edge perspective might be given by $\lambda(x)$ and $\rho(x)$ respectively.

$$\lambda(x) = \sum_{i=dv_{min}}^{dv_{max}} \lambda_i x^{i-1} \qquad \rho(x) = \sum_{j=dc_{min}}^{dc_{max}} \rho_j x^{j-1} \tag{4.1}$$

Where $\lambda_i$ is the number of 1's in *all* degree-i columns over the total number of 1's in **H**. As well, $\rho_j$ is the number of 1's in *all* degree-j row over the total number of 1's in **H**. $dv_{max}$ ($dv_{min} \geq 1$) is the highest (lowest) degree of the variable node. The same goes for the check nodes with $dc_{max}$ ($dc_{min} \geq 2$). $\lambda$ and $\rho$ are used in the evaluation of the LDPC code rate (equal for full rank).

$$\mathcal{R} \geq 1 - \frac{\int_0^1 \rho(x)dx}{\int_0^1 \lambda(x)dx} = 1 - \frac{\sum_{i=dc_{min}}^{dc_{max}} \frac{\rho_j}{j}}{\sum_{i=dv_{min}}^{dv_{max}} \frac{\lambda_i}{i}} \tag{4.2}$$

We give in the following an example of a LDPC code. For ease of reading the parity check matrices is given for a very short block length. In our example, **H** is full rank and thus

$\mathcal{R} = 4/7$.

$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \tag{4.3}$$

The polynomials profiles from the edge perspective associated to $\mathbf{H}$ are given by:

$$\lambda(x) = \frac{1}{4} + \frac{1}{2}x + \frac{1}{4}x^2 \quad \rho(x) = x^3 \tag{4.4}$$

Parity matrices are built based on the profiles. $\lambda(x)$ and $\rho(x)$ give the distribution of variable and check node degrees from the edge perspective in the parity check matrix. Matrix $\mathbf{H}$ are obtained from algorithms like the progressive edge growth (PEG) [HEA05]. Moreover $\mathbf{H}$ is said regular if all variable (respectively check) nodes have the same degree in the Tanner graph, otherwise $\mathbf{H}$ is said irregular. Back to our example, $\mathbf{H}$ is irregular, its tanner graph representation is given Fig. 4.2

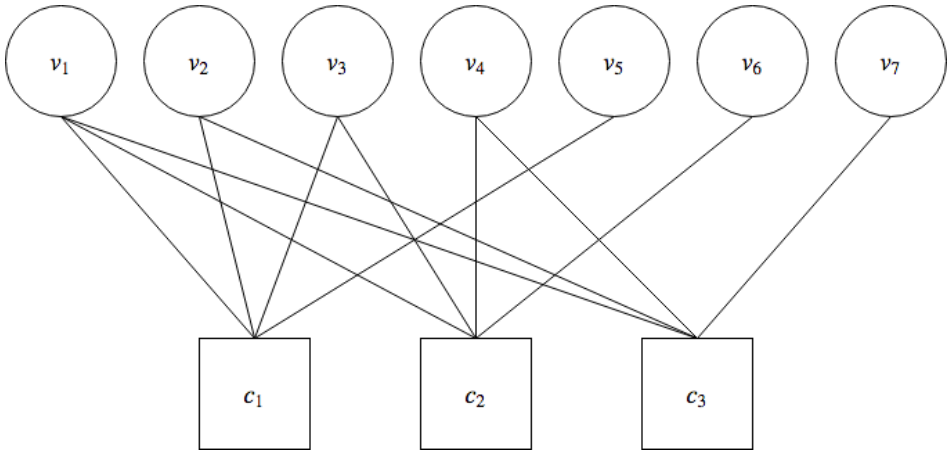



Figure 4.2: Tanner graph binary LDPC code

#### 4.4.1.2 Decoding procedure

The decoding is processed through the so called *sum-product algorithm* (SPA) also referred to as the *belief propagation algorithm* (BP) [Gal63]. Let us consider a binary input message $\boldsymbol{b} = [b_0, ..., b_{N-1}]$ transmitted along a binary-input additive white Gaussian noise (BIAWGN) channel. The received sequence is given as follows $\boldsymbol{r} = \boldsymbol{b} + \boldsymbol{n}$ where $\boldsymbol{n}$ is the AWGN. After receiving $\{r_i\}_{i\in\{i=0,...,N-1\}}$, the demapper outputs the LLRs noted $\mathcal{L}_{ch}$ with convention $\{'0' \leftrightarrow' +1',' 1' \leftrightarrow' -1'$.

$$\mathcal{L}_{ch_i} = log\left(\frac{p(b_i = +1|\boldsymbol{r})}{p(b_i = -1|\boldsymbol{r})}\right) \tag{4.5}$$

Where $p(b_i = +1|\boldsymbol{r})$ is the *a posteriori* probability of bit $b_i$. The decoding procedure, at a given iteration, is divided in two steps: the variable node update and the check node update. A variable node $v_i$ of degree $dv_i$ has $dv_i$ edges connected to check nodes and one edge connected to the demapper. The LLR information passing through the edge connecting variable node $v_i$ to its adjacent check nodes is noted $\{\mathcal{L}_{v_i,c_j}\}_{j\in\mathcal{V}_i}$. $\mathcal{V}_i$ denotes the set of check nodes adjacent to $v_i$. The LLR information carrying by the channel to variable node $v_i$ is noted $\mathcal{L}_{v_i,ch_i}$.

By convention, at the first iteration the LLRs $\{\mathcal{L}_{v_i,c_j}\}_{j\in\mathcal{V}_i}$ are equiprobable. The update of the LLR message passing through the edge connecting $v_i$ to $c_{\mathcal{V}_{i,j}}$ is given as follows:

$$\mathcal{L}_{v_i,c_{\mathcal{V}_{i,j}}} = \mathcal{L}_{v_i,ch_i} + \sum_{k=1,k\neq j}^{dv_i} \mathcal{L}_{v_i,c_{\mathcal{V}_{i,k}}} \tag{4.6}$$

As well, a check node $c_j$ of degree $dc_j$ has $dc_j$ edges connected to variable nodes. The LLR information passing through the edge connecting check node $c_j$ to its adjacent variable nodes is noted $\{\mathcal{L}_{c_j,v_i}\}_{i\in\mathcal{C}_j}$. $\mathcal{C}_j$ denotes the set of variable nodes adjacent to $c_j$.

The update of the LLR messages passing through the edge connecting $c_j$ to $v_{\mathcal{C}_{j,i}}$ is given as follows:

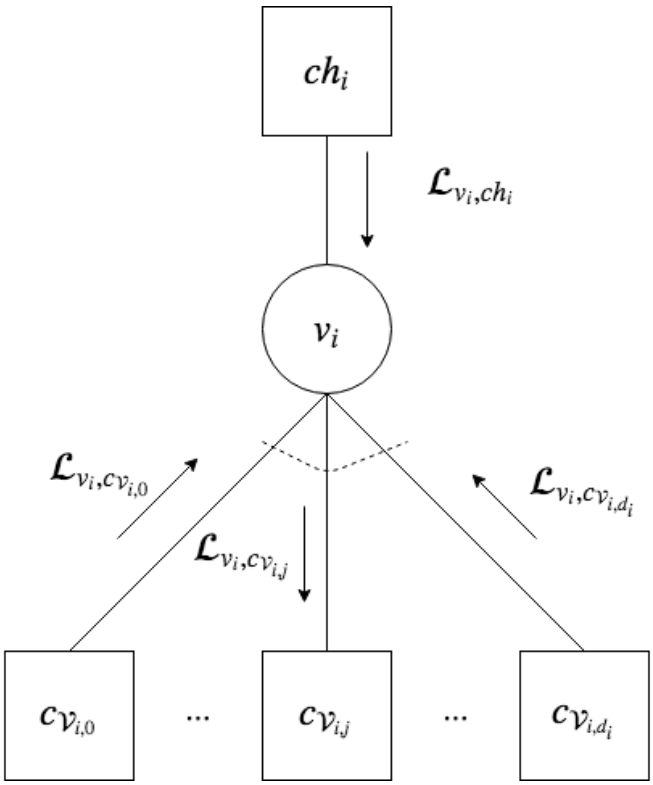



Figure 4.3: Variable node update

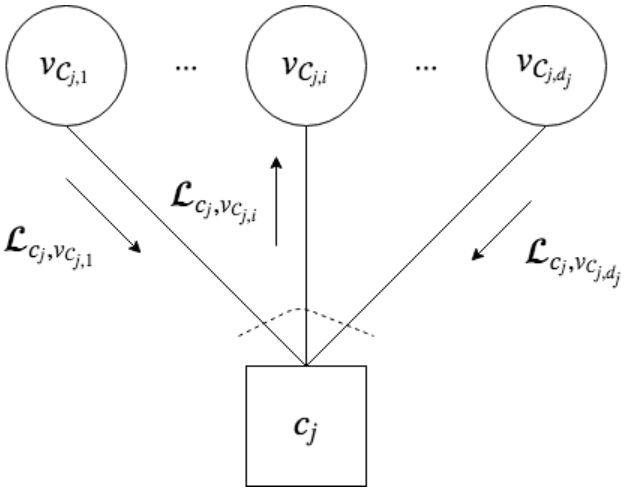



Figure 4.4: Check node update

$$\tanh\left(\frac{\mathcal{L}_{c_j,v_{C_{j,i}}}}{2}\right) = \prod_{k=1,k\neq i}^{dc_j} \tanh\left(\frac{\mathcal{L}_{c_j,v_{C_{j,k}}}}{2}\right) \tag{4.7}$$

### 4.4.2 Non-binary LDPC codes

#### 4.4.2.1 General presentation

As for the binary case, non-binary LDPC code is a sparse graph code performing close to the Shannon limit and belonging to the class of linear block codes. Its parity check matrix (noted **H**) is a $\mathcal{M} \times \mathcal{N}$ matrix. Consequently, the Tanner graph associated to **H** is composed of $N$ variable nodes and $M$ check nodes connected with each other by edges. For ease of reading, we would reuse almost the same notation than the binary case. Two major differences distinguish a non-binary LDPC code from its binary counterpart. First of all, each variable node $j$ is assigned a symbol $u_j$ taken from the GF(q) field. Indeed, it is not anymore a binary vector which passes through the LDPC encoder but symbols. Secondly, each edge $e_{i,j}$ of the tanner graph connecting variable node $i$ to check node $j$ is assigned a weight (noted $g_{i,j}$) belonging to GF(q). Thus a word **u** with components in GF(q) is a codeword if it satisfies the subsequent condition at each check node $j$:

$$\sum_{i\in\mathcal{C}_j} g_{i,j}u_i = 0 \tag{4.8}$$

Where $\mathcal{C}_j$ denotes the set of variable nodes connected to check node $j$. The summation and the multiplication is performed over GF(q). The polynomial representation of the variable and check nodes degree distribution from the edge perspective given by $\lambda(x)$ and $\rho(x)$ is defined in a way similar to the binary counterpart. We give in the following an example of a LDPC parity check matrix over GF(4):

$$\mathbf{H} = \begin{pmatrix} 2 & 1 & 1 & 2 & 0 \\ 3 & 1 & 3 & 1 & 0 \\ 1 & 0 & 2 & 2 & 1 \end{pmatrix} \tag{4.9}$$

The tanner graph associated to the parity check matrix **H** is the following:

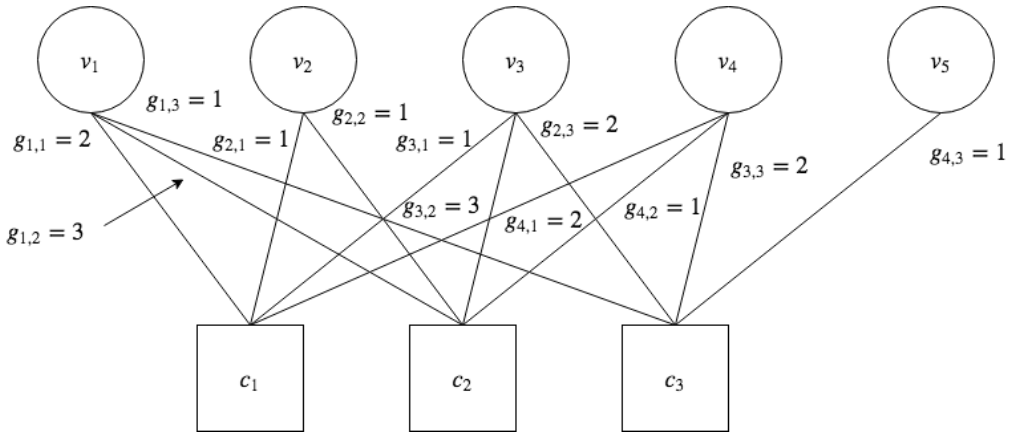



Figure 4.5: Tanner graph non-binary LDPC code

The polynomials profiles from the edge perspective associated to **H** are given by:

$$\lambda(x) = \frac{1}{12} + \frac{1}{6}x + \frac{3}{4}x^2 \quad \rho(x) = x^3 \tag{4.10}$$

#### 4.4.2.2 Decoding procedure

The *belief propagation algorithm* (BP) [Gal63] applied to non-binary LDPC is a generalization of its binary counterpart. The decoder attempts to recover from the channel observations $\{r_i\}_{i\in\{0,...,N-1\}}$ the message $\boldsymbol{u} = [u_0, ..., u_{N-1}]$ belonging to the finite field GF(q). Element of GF(q) belongs to the set $\{0, 1, \alpha, ..., \alpha^{q-2}\}$ where $\alpha$ is a primitive element of GF(q). $\alpha$ is defined as one of the roots of a minimal polynomial of GF(q). The representation by the primitive element $\alpha$ would allow the multiplication operations associated to the edge weights to be converted into a permutation. Consider Eq. (4.8) again, with the primitive representation of element belonging to GF(q) with have:

$$\sum_{i \in \mathcal{C}_j} g_{i,j} u_i = \sum_{i \in \mathcal{C}_j} \alpha^{k_{i,j}} \alpha^{k_i} = \sum_{i \in \mathcal{C}_j} \alpha^{k_{i,j}+k_i} \tag{4.11}$$

Where $k_{i,j}$ and $k_i$ are respectively the $\alpha$ order associated to weight $g_{i,j}$ and symbol $u_i$. Hence the product $g_{i,j}u_i$ is equivalent to permute symbol $u_i$ of a quantity $k_{i,j}$ yielding another symbol belonging to GF(q).

For convenience, the decoding algorithm is described from a vector probability level. However the probability message can be converted back to its LLR representation with the LLR(.) function defined in (3.9). We defined the probability message $\mathcal{P}_{v_i,ch_i}$ sent from the channel to variable node $v_i$ as:

$$\mathcal{P}_{v_i,ch_i} = \begin{pmatrix} p(u_i = 0|r_i) \\ p(u_i = 1|r_i) \\ p(u_i = \alpha|r_i) \\ \vdots \\ p(u_i = \alpha^{q-2}|r_i) \end{pmatrix} \tag{4.12}$$

The decoding procedure, at a given iteration, is divided in two steps: the variable node update and the check node update. A variable node $v_i$ of degree $dv_i$ has $dv_i$ edges connected to the check nodes and one edge connected to the demapper. The probability vector passing through the edge connecting variable node $v_i$ to its adjacent check nodes is noted $\{\mathcal{P}_{v_i,c_j}\}_{j \in \mathcal{V}_i}$. $\mathcal{V}_i$ denotes the set of check nodes adjacent to $v_i$. By convention, at the first iteration the components of the probability vector $\{\mathcal{P}_{v_i,c_j}\}_{j \in \mathcal{V}_i}$ are equiprobable. As well, we defined $\{\mathcal{P}_{c_j,v_i}\}_{i \in \mathcal{C}_j}$ as the probability vector passing through the edge connecting check node $c_j$ to its adjacent variable nodes. $\mathcal{C}_j$ denotes the set of variable nodes adjacent to $c_j$.

The update of the message passing through the edge connecting $v_i$ to $c_{\mathcal{V}_{i,j}}$ is given as follows:

$$\tilde{\mathcal{P}}_{v_i,c_{\mathcal{V}_{i,j}}} = \mathcal{P}_{v_i,ch_i} \prod_{l=0, l \neq j}^{d_i} \tilde{\mathcal{P}}_{v_i,c_{\mathcal{V}_{i,l}}} \tag{4.13}$$

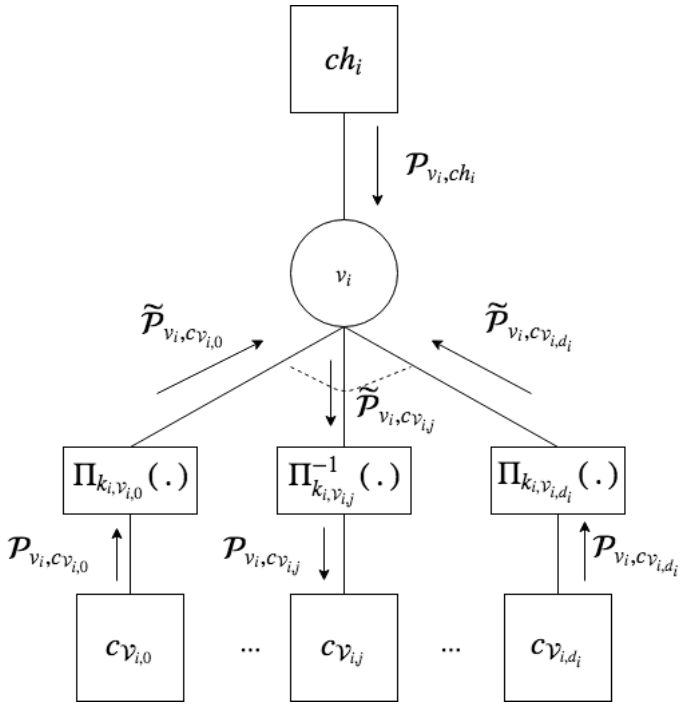



Figure 4.6: Variable node update

We define $\tilde{\mathcal{P}}_{v_i,c_j} = \Pi_{k_{i,j}}\left(\mathcal{P}_{v_i,c_j}\right)$ where $\Pi_{k_{i,j}}$ is a circular permutation of quantity $k_{i,j}$ resulting from the multiplication of weight $g_{i,j} = \alpha^{k_{i,j}}$ associated to the edge connected variable node $v_i$ and check node $c_j$. If $g_{i,j} = 1$ then $k_{i,j} = 0$ and $\Pi_0$ is the identity permutation. For instance, if we apply a permutation $\Pi_k$ to $\mathcal{P}_{v_i,ch_i}$ we obtain:

$$\tilde{\mathcal{P}}_{v_i,ch_i} = \Pi_k\left(\mathcal{P}_{v_i,ch_i}\right) = \begin{pmatrix} p(u_i = 0|r_i) \\ p(u_i = \alpha^{q-2-k+1}|r_i) \\ \vdots \\ p(u_i = \alpha^{q-2}|r_i) \\ p(u_i = 1|r_i) \\ \vdots \\ p(u_i = \alpha^{q-2-k}|r_i) \end{pmatrix} \tag{4.14}$$

Permutation $\Pi_k$ does not change the position of symbol zero. It is obvious that multiplying the edge weight by symbol zero gives symbol zero. The update of the LDPC check nodes is

more complex than its variable node counterpart. A check node $c_j$ of degree $dc_j$ has $dc_j$ edges connected to the variable nodes.

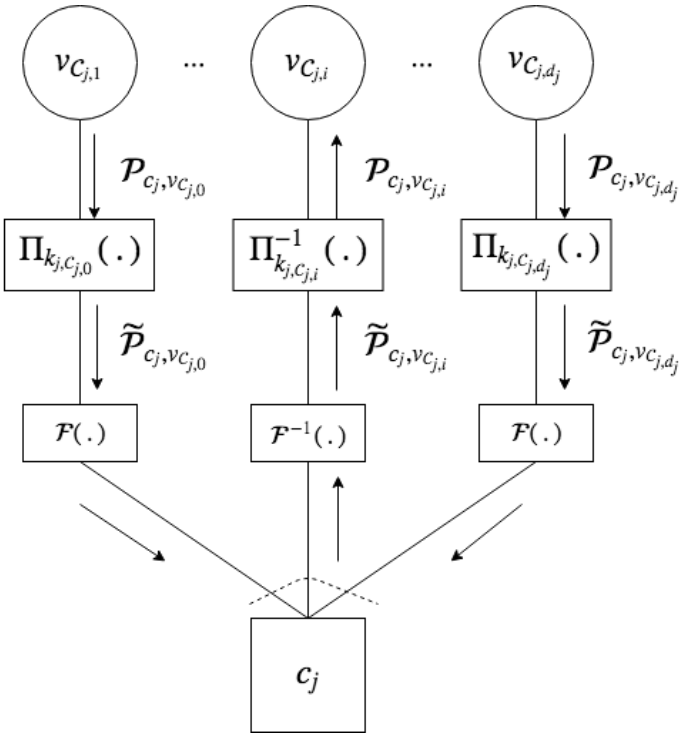



Figure 4.7: Check node update

The message $\mathcal{P}_{c_j,v_{c_{j,i}}}$ associated to the edge connecting check node $c_j$ to variable node $v_{c_{j,i}}$ is updated as follows:

$$\tilde{\mathcal{P}}_{c_j,v_{c_{j,i}}} = \mathcal{F}^{-1}\left(\prod_{k=1,k\neq i}^{dc_j} \mathcal{F}\left(\tilde{\mathcal{P}}_{c_j,v_{c_{j,k}}}\right)\right) \tag{4.15}$$

$\prod$ is the term by term product and $\mathcal{F}$ is the Fourier transform over the finite field $(\mathbb{Z}_2^q, +)$. We note $\boldsymbol{F}$ its matrix representation. $\boldsymbol{F}$ is a Walsh matrix made up entirely of 1 and $-1$ such as $\boldsymbol{F}_{m,n} = (-1)^{\sum_j m_j n_j}$ where the $m_j$ and $n_j$ are the binary digits of the indices $m$ and $n$. For instance the Walsh matrix $\boldsymbol{F}$ associated to $q = 4$ is evaluated as:

$$\boldsymbol{F} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

As well, $\mathcal{F}^{-1}$ is the inverse Fourier transform over $(\mathbb{Z}_2^q, +)$. We note $\boldsymbol{F}^{-1}$ its matrix representation. $\boldsymbol{F}^{-1}$ is also a Walsh matrix where each component $\boldsymbol{F}^{-1}_{m,n} = \frac{1}{q} \cdot (-1)^{-\sum\limits_{j} m_j n_j}$. For instance the Walsh matrix $\boldsymbol{F}^{-1}$ associated to $q = 4$ is evaluated as:

$$\boldsymbol{F} = \frac{1}{4} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

Consequently, we rewrite (4.15) as:

$$\tilde{\mathcal{P}}_{c_j, v_{\mathcal{C}_{j,i}}} = \boldsymbol{F}^{-1} \cdot \left( \prod_{k=1, k \neq i}^{dc_j} \boldsymbol{F} \cdot \tilde{\mathcal{P}}_{c_j, v_{\mathcal{C}_{j,k}}} \right) \tag{4.16}$$

## 4.5 Asymptotic analysis of serially concatenated systems

### 4.5.1 Asymptotic analysis of binary LDPC codes

Density evolution (DE) [RU01] and extrinsic information transfer chart [TB01] are two popular tools used to study the design and the convergence behaviour of codes. DE consists of tracking the probability density function of LLR messages along the edges of the Tanner graph to compute the convergence threshold. This latter corresponds to the lowest SNR permitting a reliable iterative decoding for long block length. However this method has the weakness of being computationally complex. Thereafter, EXIT charts have been developed to

study the performance of codes under various modulation scheme. This tool became popular, because less complex than DE. In the context of LDPC code, the idea is to compute the average mutual information between variable nodes, check nodes and demapper outputs in order to evaluate the convergence threshold and the best set of profiles ($\lambda(x)$ and $\rho(x)$). As for the decoding procedure, the EXIT chart asymptotic analysis is processed in two steps: the variable node update and the check node update.

The average combined mutual information of the variable nodes (noted $I^l_{vn,cn}$ at iteration $l$) is computed under a Gaussian approximation. Thus at a given iteration $l$, $I^l_{vn,cn}$ is evaluated as follows:

$$I^l_{vn,cn} = \sum_{i=dv_{min}}^{dv_{max}} \lambda_i J\left(\sqrt{(i-1)[J^{-1}(I^{l-1}_{vn,cn})]^2 + J^{-1}(T(J(\sqrt{i}J^{-1}(I^{l-1}_{cn,vn}))))^2}\right) \tag{4.17}$$

T(.) is the EXIT chart transfer function of the demapper (in practice approximated by a polynomial curve fitting). The average combined mutual information of the check nodes (noted $I^l_{cn,vn}$ at iteration $l$) is computed under a Gaussian approximation. Thus at a given iteration $l$, $I^l_{cn,vn}$ is evaluated as follows:

$$I^l_{cn,vn} = 1 - \sum_{j=dc_{min}}^{dc_{max}} \rho_j J\left(\sqrt{(j-1)}[J^{-1}(1-I^l_{vn,cn})]\right) \tag{4.18}$$

It results from the combination of (4.17) and (4.18) a linear function with respect to the $\{\lambda_k\}$ for a given concentrated check nodes profiles (i.e $dc_{min} = dc_{max} - 1$) and for a given $Es/N_0$. The optimization procedure consists in solving equation (4.17) under some specific constraints so that the rate $\mathcal{R}$ is maximized. In that respect, the rate maximization design is equivalent to maximizing the quantity $\sum_i \frac{\lambda_i}{i}$ (since $\rho$ is fixed see Eq. (4.2)) subject to the following constraints:

$$\begin{cases} \text{Mixture} : \sum_i \lambda_i = 1 \\ \text{Convergence} : I^l_{vn,cn} > I^{l-1}_{vn,cn} \end{cases} \tag{4.19}$$

This solving can be completed by adding a constraint on the degree-1 variable nodes proportion. Thus the optimization algorithm might be processed as follows [Ben15].

| **Optimization Procedure** |
|---|
| Fix a SNR |
| Compute the CPM EXIT curve T(.) |
| Fix $dv_{max}$ and $dv_{min}$ |
| Initialize the code rate $\mathcal{R}_{opt} = 0$ |
| Fix $\rho$ by generating a set $\mathcal{C}$ of concentrated check nodes degree |
| For each $\rho$ in $\mathcal{C}$ do |
| Solve linear programming (4.17) and get the $\{\lambda_i\}$ |
| Compute the new rate $\mathcal{R}$ from (4.2) |
| If $\mathcal{R}_{opt} < \mathcal{R}$ |
| $\mathcal{R}_{opt} = \mathcal{R}$, $\{\lambda_i\}_{opt} = \{\lambda_i\}$ and $\{\rho_i\}_{opt} = \{\rho_i\}$ |
| End if |
| End for |

Table 4.1: LDPC optimization procedure

### 4.5.2 Asymptotic analysis of non-binary LDPC codes

Similarly to the binary counterpart, the asymptotic analysis of non-binary LDPC code is proceed by non-binary EXIT chart. To this end, the average mutual information between variable nodes, check nodes and demapper outputs is computed to evaluate the convergence threshold and the best set of profiles $\lambda(x)$ and $\rho(x)$. The process is exactly the same than the binary counterpart except that the mutual information functions are adapted to the non-binary case (see [BB06]). In other word, the $J$ function is the one described in section 3.3.2. The analysis is processed in two steps: the variable node update and the check node update. The average combined mutual information passed from check nodes to variable nodes (noted

$I^l_{vn,cn}$ at iteration $l$) is obtained as follows:

$$I^l_{vn,cn} = \sum_{k=dv_{min}}^{dv_{max}} \lambda_k J\left(\sqrt{(k-1)[J^{-1}(I^{l-1}_{vn,cn})]^2 + J^{-1}(T(J(\sqrt{k}J^{-1}(I^{l-1}_{cn,vn}))))^2}\right) \tag{4.20}$$

T(.) is the non-binary EXIT chart transfer function of CPM demapper (in practice approximated by a polynomial curve fitting). $J(.)$ is the mutual information function between variable node to check node messages introduced in [BB06]. It takes as parameter the co-variance matrix of the variable node to check node messages. This function is the one described in section 3.3.2. This analysis was improved in [BB06] by replacing the $J(.)$ by a so called $J_R(.)$ function in [BB06]. The average combined mutual information passed from variable nodes to check nodes (noted $I^l_{cn,vn}$ at iteration $l$) is computed as follows:

$$I^l_{cn,vn} = 1 - \sum_{i=dc_{min}}^{dc_{max}} \rho_i J\left(\sqrt{(i-1)}[J^{-1}(1 - I^l_{vn,cn})]\right) \tag{4.21}$$

Again, the combination of (4.20) and (4.21) is a linear function with respect to the $\{\lambda_k\}$ for a given concentrated check nodes profiles (i.e $dc_{min} = dc_{max} - 1$) and for a given $Es/N_0$. The optimization procedure consists in solving the equation (4.20) under some specific constraints so that the rate $\mathcal{R}$ is maximized. The optimization algorithm is proceeded in the same way as the binary case 4.1.

## 4.6 Binary LDPC codes for coherent and noncoherent channel

In this section, we attempt to design good coding schemes performing well for both the coherent and the non-coherent settings. This design should be done carefully. Usually, the problem is solved sub-optimally by first considering the design of a good coding scheme for the coherent regime and then, it is applied to the non-coherent case. We will see that it may

be not the best option. Indeed, it will be shown in the sequel that codes designed for the coherent case may be not “stable” for the non-coherent case, preventing the use of "optimal" schemes in the coherent case for the non coherent setting. Thus, we will show that the best option is to design first codes for the non coherent case and then assess to the performance in the coherent case.

However, even this approach is not entirely satisfactory since a capacity penalty still appear in the coherent regime with the optimized non-coherent coding scheme. In that respect, we propose, in a second part, to reduce the penalty by making use of precoding schemes.

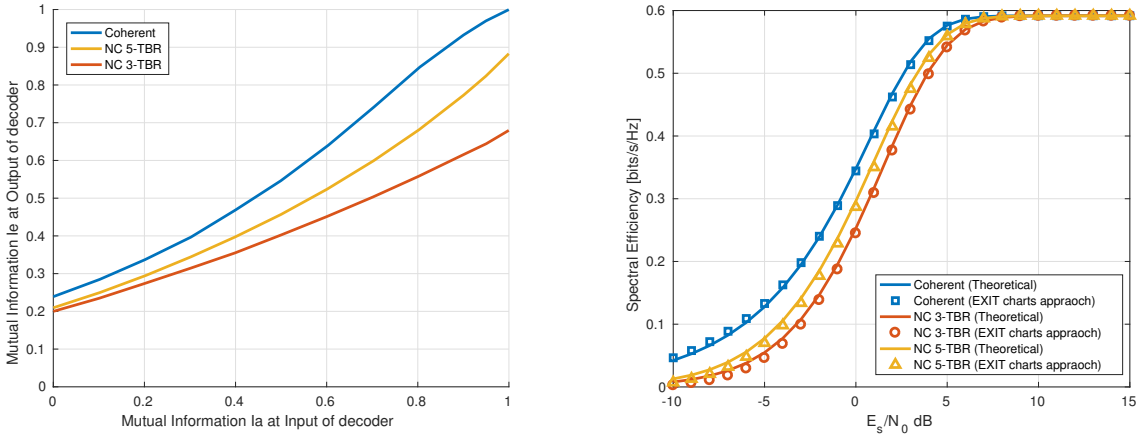
### 4.6.1 Optimization of LDPC profiles

Just as a reminder, [Hag04] pointed out that the achievable rate is approximately equal to the area under the EXIT curve for a given operating point (proven over the erasure channel [AKB04]). In other words, for binary EXIT chart, the information rate noted $\mathcal{R}^*$ is linked with $I_e$, the extrinsic information at the output of the decoder:

$$\mathcal{R}^* \simeq m \cdot \int_0^1 I_e(x)\mathrm{d}x \tag{4.22}$$

For non-binary EXIT charts $\mathcal{R}^* \simeq (1/m) \int_0^m I_e(x)\mathrm{d}x$ and $m = log_2(M)$. As shown in Fig. 4.8 (b.) for the case of CPFSK, the approximation is relatively accurate when compared to direct calculation of the mutual information rate.

This suggests that good coding schemes under iterative decoding can be designed by state-of-the-art curve fitting methods to approach these maximum achievable rates. We use in the following the procedure given Tab. 4.1 to design optimized LDPC profiles based on EXIT chart. EXIT trajectories have been drawn in Fig. 4.8 (a.) for $E_s/N_0 = 0$ dB. It is shown that only the coherent receiver is converging to the point $(1, 1)$, which is a common feature for non-linear modulations with memory such as CPM or differential modulations (if no specific




(a.) (b.)

Figure 4.8: CPFSK $M = 4$ and $h = 5/7$ ($\approx 0.715$) (a.) Exit charts (b.) Spectral Efficiency

precoding is used) or also for precoded linear channels with memory. This issue will strongly impact the code design when iterative decoding is used.

The remaining question is how to design good codes that can perform fairly well in both regimes (coherent and non coherent). Usually, the design of codes for CPM schemes is carried out based on the coherent settings then the performance is assessed in the non-coherent case. We will see that it may be not the best option. We first consider the non-coherent case. As in most serially concatenated systems, EXIT curves do not reach the point $(1, 1)$ and so methods as in [TBKA04][Ben+14b] can be used for LDPC code design. This leads to an upper bound on the degree 2 nodes associated with the Tanner graph of the code. This is due to the so-called *stability condition*. We now consider the coherent case. For this scheme, the EXIT curves reach the point $(1, 1)$. Several approaches are possible. Mainly, non systematic low-density generator matrix based coding schemes have been considered as in [CVT09]; [CVT07] using some analogies with irregular-repeat accumulate coding schemes. For the case of LDPC codes, it has been shown in [Ben15]; [Ben+14a] that some good LDPC codes can be designed if some degree one nodes are carefully introduced satisfying a stability condition constraining degree one nodes only. In that case, degree 2 nodes are not anymore constrained. It follows, from this simple fact, that the coding scheme designed for the coherent case cannot be used for the non-coherent setting since the resulting profiles cannot be stable. On the contrary, a code

| Channel \Coding scheme | Coherent optimization | Non-coherent optimization |
|---|---|---|
| Coherent regime | stable | stable |
| Non-coherent regime | unstable | stable |

Table 4.2: Summation of coding scheme stability

designed for the non-coherent case is always stable under coherent decoding. Of course, for the coherent case, one can also avoid degree one nodes in the design. Unfortunately, based on our experiments, it appears that, for the same information rate, the fraction of degree two nodes for code profiles optimized in the coherent case is always greater than the fraction of degree two nodes for a code profile optimized in the non coherent case. Here, again the code may be not stable under non-coherent decoding. To have good coding schemes operating in both regimes, it seems reasonable to design first codes for the non-coherent case and then assess the performance in the coherent case. For ease of viewing, Table.4.2 sum up the stability of optimized coding schemes for both regimes.

Table 4.3: Degree distribution used in Fig.4.9 ($dv_{min} \geq 2$,$R = 1/2$).

| | $d_c$ | $d_v$ | $\rho$ | $\lambda$ |
|---|---|---|---|---|
| Coherent | $\{4, 5\}$ | $\{2, 8\}$ | $\{0.14, 0.86\}$ | $\{0.87, 0.13\}$ |
| Noncoherent 3-TBR | $\{4, 5\}$ | $\{2, 8\}$ | $\{0.49, 0.51\}$ | $\{0.81, 0.19\}$ |

Table.4.3 provides LDPC profiles optimized for the CPFSK modulation. Based on those profiles, we have compared in Fig. 4.9 the bit error rate (BER) performances of different optimized coding schemes for rate one-half codes with $N_b = 4096$ coded bits following [Ben15][Ben+14b]. For fair comparison, we consider the optimization in the coherent case without degree one nodes. Two major notices can be made from the finite length simulation displayed by Fig. 4.9. First, the coherent coded BER, based on the profiles optimized with non-coherent settings, has relatively small loss compared to the optimal coherent case, with no degree one nodes allowed. Secondly, coherent schemes optimized with degree 2 variable nodes constrained have a capacity penalty of $3dB$. Thus, those profiles are not entirely satisfactory. LDPC codes designed with degree one nodes should be investigated while satisfying

the stability condition.

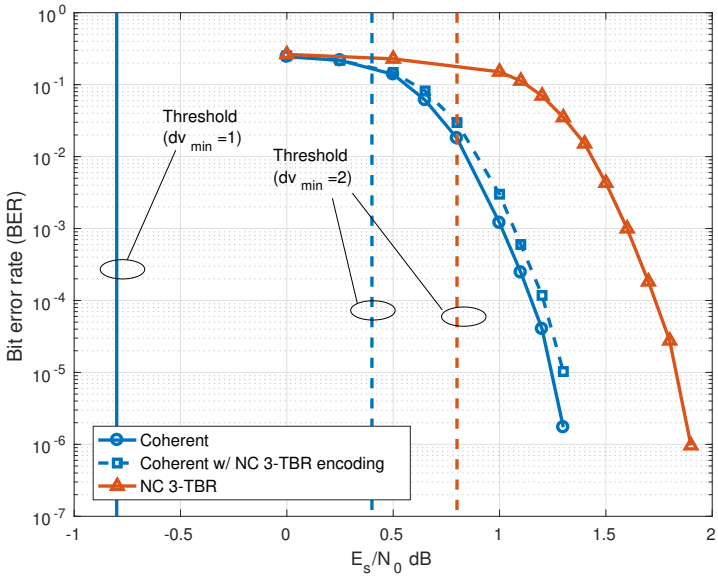



Figure 4.9: Coded BER: quaternary CPFSK with $h = 5/7$, $N = 3$ and $R = 1/2$.

### 4.6.2 Code profile optimization based on precoding scheme

In the previous section, we have seen that we were not able to design conjoint LDPC profiles with degree one variable nodes because of the EXIT charts convergence issue in non-coherent regime. In that respect, we were obliged to consider profiles constrained with degree two variable nodes to obtain a coding scheme suited to both regimes. However, usually capacity achieving LDPC codes can be designed if some degree one nodes are carefully introduced. In our previous case, since only degree 2 constrained variable node have been taken into account, the coding scheme generates a capacity penalty in coherent regime as displayed Fig. 4.9. This issue can be solved through the use of precodings. We showed along chapter 3 that a precoding scheme was able to change the EXIT trajectory and above all to force the convergence to the point $(1, 1)$. This enable us to design better conjoint LDPC profiles with degree one variable nodes. Fig. 4.10 (a.) display several EXIT trajectories at rate one-half for 2RC, 2GMSK and CPFSK modulations. Those curves were evaluated based on optimized precoding matrices $\mathbf{F}$ seen in chapter 3. As expected, the precoding has made the EXIT charts converge to point $(1, 1)$ in non-coherent regime. On interesting point is the effect of the non-coherent binary precoding on the coherent regime. It appears that the non-coherent precoding has no impact in coherent regime on the 2RC and the 2GMSK EXIT charts for their corresponding set of parameters. However, it is not the case with the CPFSK where the modulation index ($h = 5/7$) is a bit less conventional than the ones chosen for the 2RC and 2GMSK (respectively $h = 1/2$ and $1/4$). Table.4.4 shows the updated profiles of those precoded modulations for a rate one-half. Two major notices can be made from this table. First it appears that the degree one variable nodes distribution of the CPFSK code profiles (*unconstrained case*) is huge. If we convert the degree distribution from an edge perspective to a node prospective, we will see that the amount of degree ones exceeds 50%. Codes built from such profiles are rarely considered because not enough efficient ($d_{min} = 2$). Thus, it is necessary to strongly constrained the degree ones to get an efficient LDPC code (yet leading to a capacity penalty) or used other types of codes as low-density generator matrix (LDGM). Second remark, again the most constrained profiles belong to the non-coherent precoded case

for each of the considered case (2RC, 2GMSK and CPFSK). The non-coherent regime offers less degree one in its distribution than the coherent case. Thus it follows that a coding scheme designed for the coherent case cannot be used for the non-coherent setting since the resulting profiles cannot be stable. On the contrary, a code designed for the non-coherent precoded case is always stable under coherent decoding. For ease of viewing, Table.4.3 sum up the stability of optimized coding schemes for both regimes. In that respect, it is necessary to design, first, codes for the non-coherent precoded case and then assess the performance in the coherent case to implement good conjoint coding schemes. The rate achievable by optimized LDPC code are plotted Fig. 4.11 and 4.12 for the set of modulations previously considered. First off, it appears on the viewing of Fig. 4.11 (b.), that constraining to much the degree ones, to build an optimized LDPC code (with a degree one variable nodes distribution lower than 50%), has generated a capacity penalty of 1dB. So finally, the coherent BER curves plotted Fig. 4.9 were not so far from the threshold of the rate one-half achievable by an optimized LDPC code. Secondly we can observe some improvements thanks to the NCB-precoding scheme. In coherent regime, the scheme *precoding* + $dv_{min} = 1$ + *code optimized for the non-coherent regime* has a 0.2dB (respectively 1dB) threshold better than the scheme $dv_{min} = 2$ + *code optimized for the non-coherent regime* for the 2GMSK and the CPFSK (respectively 2RC) at rate one-half. In non-coherent regime, the scheme *precoding* + $dv_{min} = 1$ + *code optimized for the non-coherent regime* has a 0.4dB (respectively 1.1dB) threshold better than the scheme $dv_{min} = 2$ + *code optimized for the non-coherent regime* for the 2GMSK (respectively 2RC) and it worsen the case CPFSK of less than 0.1dB at rate one-half. So, we can conclude by saying that the combination *precoding+LDPC* is not as significant than with convolutional codes but we definitively have enhancements depending of the cases.

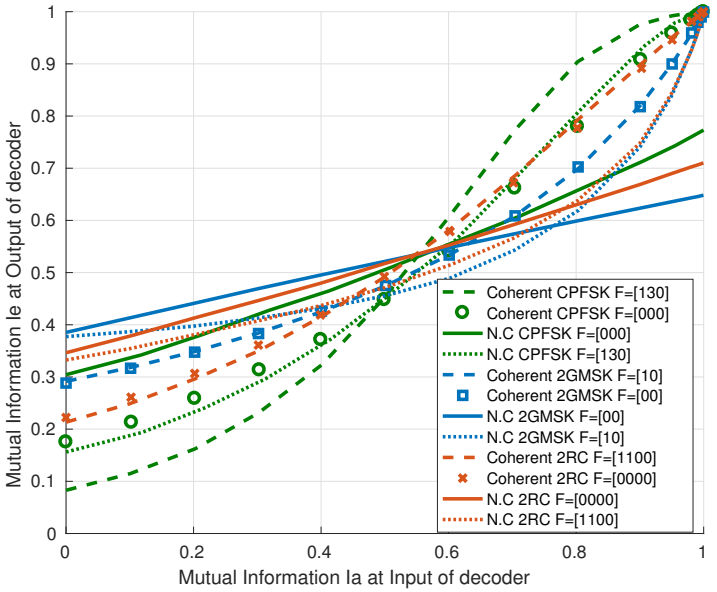
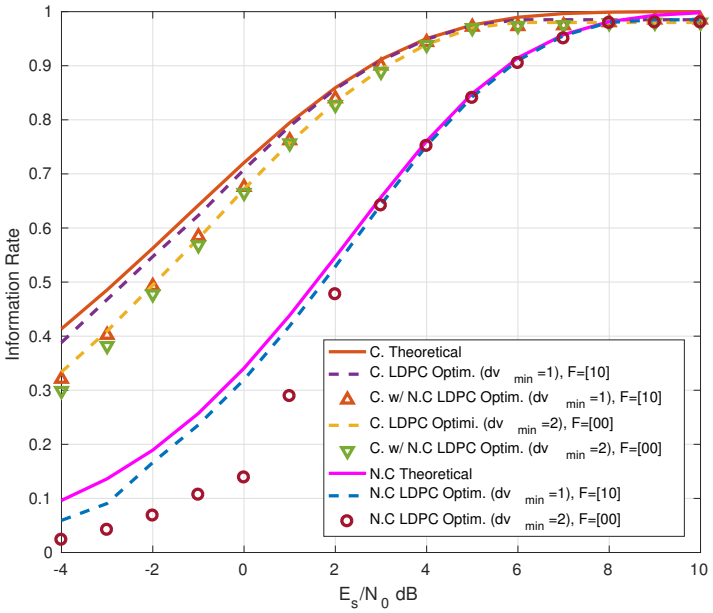


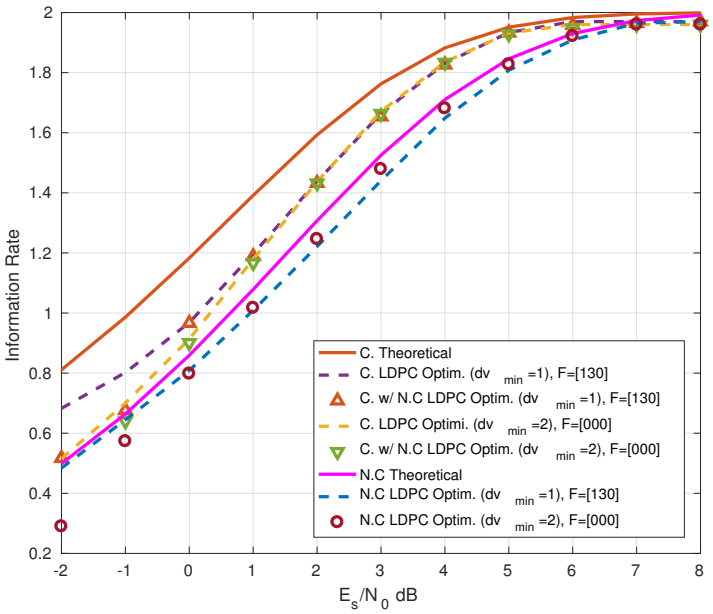

Figure 4.10: Binary EXIT charts: CPFSK $h = 5/7$, $M = 4$, 2GMSK $h = 1/2$, $M = 2$, $BT = 0.3$ and 2RC $h = 1/4$, $M = 4$

Table 4.4: Degree distribution binary LDPC ($R = 1/2$).

| | $d_c$ | $d_v$ | $\rho$ | $\lambda$ |
|---|---|---|---|---|
| ***2GMSK*** *($h = 1/2$, $BT = 0.25$, $M = 2$, $N = 2$)* | | | | |
| Coherent NCB-precoding ($F = [10]$) | $\{4, 5\}$ | $\{1, 2, 8\}$ | $\{0.39, 0.61\}$ | $\{0.12, 0.58, 0.3\}$ |
| Coherent ($F = [00]$) | $\{4, 5\}$ | $\{1, 2, 8\}$ | $\{0.39, 0.61\}$ | $\{0.12, 0.58, 0.3\}$ |
| N-C NCB-precoding ($F = [10]$) | $\{5, 6\}$ | $\{1, 2, 8\}$ | $\{0.44, 0.56\}$ | $\{\mathbf{0.06}, 0.53, 0.041\}$ |
| ***2RC*** *($h = 1/4$, $M = 4$, $N = 2$)* | | | | |
| Coherent NCB-precoding ($F = [1100]$) | $\{3, 4\}$ | $\{1, 2, 8\}$ | $\{0.14, 0.86\}$ | $\{0.26, 0.47, 0.27\}$ |
| Coherent ($F = [0000]$) | $\{3, 4\}$ | $\{1, 2, 8\}$ | $\{0.14, 0.86\}$ | $\{0.26, 0.47, 0.27\}$ |
| N-C NCB-precoding ($F = [1100]$) | $\{4, 5\}$ | $\{1, 2, 8\}$ | $\{0.04, 0.96\}$ | $\{\mathbf{0.06}, 0.6, 0.34\}$ |
| ***CPFSK*** *($h = 5/7$, $M = 4$, $N = 3$)* | | | | |
| Coherent NCB-precoding ($F = [130]$) | $\{3, 4\}$ | $\{1, 2, 8\}$ | $\{0.44, 0.56\}$ | $\{\mathbf{0.287}, 0.528, 0.185\}$ |
| Coherent ($F = [000]$) | $\{3, 4\}$ | $\{1, 2, 8\}$ | $\{0.19, 0.81\}$ | $\{0.26, 0.49, 0.25\}$ |
| Coherent ($F = [000]$ unconstrained) | $\{3, 4\}$ | $\{1, 2, 3, 8\}$ | $\{0.29, 0.71\}$ | $\{0.37, 0.18, 0.27, 0.18\}$ |
| N-C NCB-precoding ($F = [130]$) | $\{3, 4\}$ | $\{1, 2, 8\}$ | $\{0.29, 0.71\}$ | $\{\mathbf{0.27}, 0.5, 0.23\}$ |
| N-C NCB-precoding ($F = [130]$) | $\{3, 4\}$ | $\{1, 2, 3, 8\}$ | $\{0.54, 0.46\}$ | $\{0.39, 0.18, 0.25, 0.18\}$ |




(a.)




(b.)

Figure 4.11: Information Rate (a.) 2GMSK $h = 1/2$, $M = 2$, $BT = 0.3$ $N = 2$ (b.) CPFSK $h = 5/7$, $M = 4$, $N = 3$

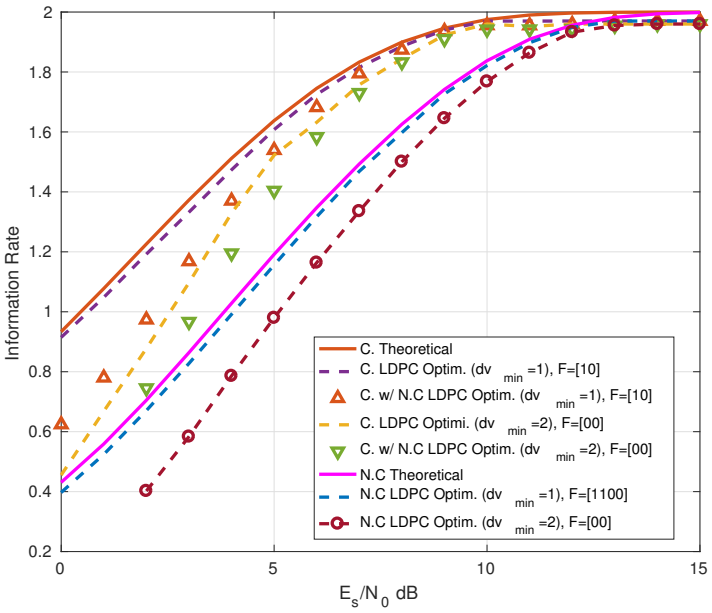



Figure 4.12: Information Rate 2RC $h = 1/4$, $M = 4$, $N = 2$

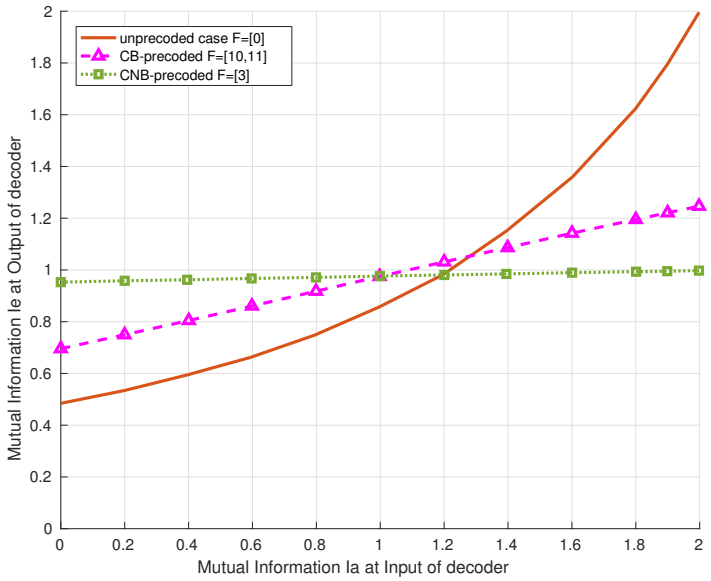
| Channel \Coding scheme | Coherent optimization | N-C w/ precoding optimization |
|---|---|---|
| Coherent regime | stable | stable |
| N-C regime w/ precoding | unstable | stable |

Table 4.5: Summation of coding scheme stability with precoding scheme

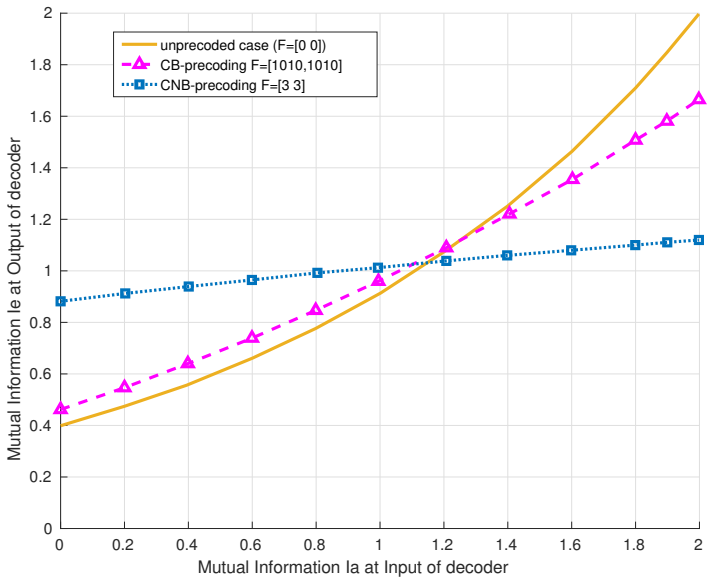
## 4.7 Capacity achieving without iterative decoding for M-ary CPM

For some CPM schemes, the BICM capacity can be (almost) equal to the information rate if some specific precodings are used, removing *de facto* the need for iterating. In [Ben+07], the authors designed a precoding leading to such features for binary CPMs. From EXIT charts perspective, their precoding was able to flatten the CPM EXIT trajectories while preserving the area under the curve. In that context, we designed section 3.5 a precoding scheme over $\mathbb{Z}_M$ called CNB-precoding allowing the $M$-ary CPM BICM capacity to reach the information rate for some specific parameters. *De facto* This section is dedicated to the evaluation of the CNB-precoding through simulations based on low-density parity-check codes.

First of all, we optimize for the 2REC and the RC modulation, a CNB-precoding scheme in order to flatten their respective EXIT trajectories. Fig. 4.13 displays the effect of the CNB-precoding schemes on the EXIT charts of those two modulations at rate one-half ($\simeq$ area under the curve). It appears that the CNB-precoding has flatten completely the RC EXIT curve and almost the 2REC's one. Based on the EXIT chart analysis, we optimize code profiles for each modulation and we considere two cases: the case without a precoding scheme and the case with the NCB-precoding. The profiles are given Table.4.6 and we can make two observations. First the code profiles for the precoded RC and 2REC are obviously the same since we have considered that both EXIT charts were flat (almost) at a rate one-half. Secondly, the code profiles of the precoded modulation has a $dv_{min}$ of 2 (i.e the smallest variable node degree is 2) whereas the unprecoded cases have a $dv_{min}$ of 1. Then we plotted the rate achievable by an optimized LDPC code Fig. 4.14. As expected, the case *RC+CNB-precoding+zero iteration* reaches the same rate than the case *RC+iterations*. Far more its seems that the CNB-precoding outperformed the case *RC+iterations* at low SNR. As well, regarding the case 2REC, the LDPC optimization does not change the gap between the case *CNB-precoding+zero iteration* and *no precoding+iterations* with respect to the theory (still around 1dB). Thus, we conclude by saying that the CNB-precoding might be a good trade off between complexity and performance.

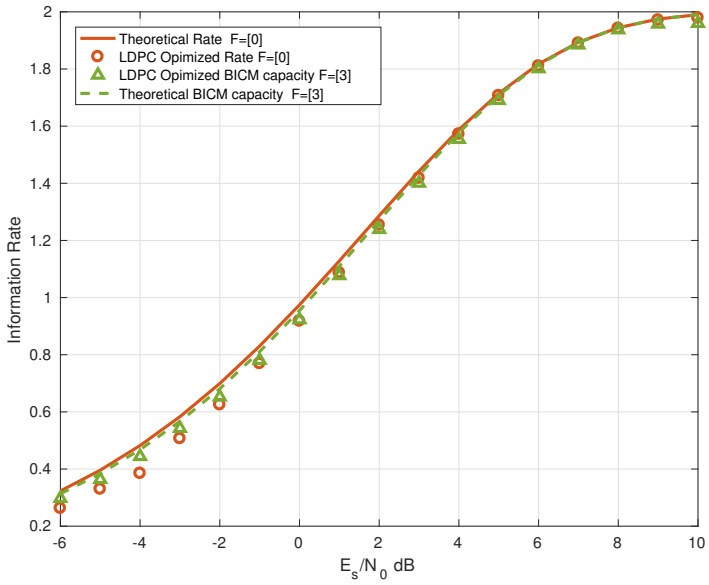



(a.)

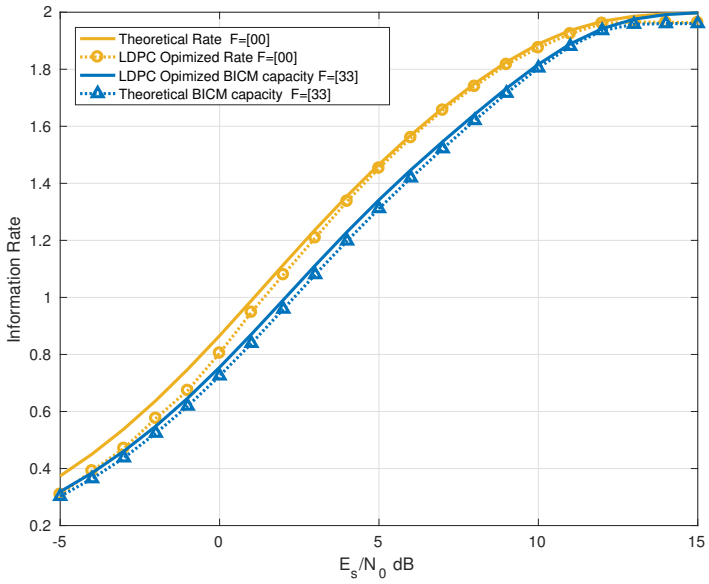



(b.)

Figure 4.13: Non-binary EXIT charts ($R = 1/2$) (a.) RC with $h = 1/4$ and $M = 4$ and (b.) 2REC with $h = 1/4$ and $M = 4$.




(a.)




(b.)

Figure 4.14: Information Rate ($R = 1/2$) (a.) RC with $h = 1/4$ and $M = 4$ and (b.) 2REC with $h = 1/4$ and $M = 4$.

Table 4.6: Degree distribution NB-LDPC ($R = 1/2$).

| | $d_c$ | $d_v$ | $\rho$ | $\lambda$ |
|---|---|---|---|---|
| ***2REC*** *($h = 1/4$, $M = 4$)* | | | | |
| Coherent ($F = [00]$) | $\{3, 4\}$ | $\{1, 2, 8\}$ | $\{0.19, 0.81\}$ | $\{0.2381, 0.5492, 0.2127\}$ |
| ***RC*** *($h = 1/4$, $M = 4$)* | | | | |
| Coherent ($F = [0]$) | $\{3, 4\}$ | $\{1, 2, 8\}$ | $\{0.69, 0.31\}$ | $\{0.12, 0.63, 0.25\}$ |
| ***CPM w/ flat EXIT charts*** | | | | |
| CNB-precoding (RC/2REC [3]/[33]) | $\{5, 6\}$ | $\{2, 3, 8\}$ | $\{0.59, 0.41\}$ | $\{0.33, 0.32, 0.35\}$ |

# Conclusions et perspectives (French)

## Conclusions

Dans cette thèse, on souhaitait améliorer les techniques de détection du signal de télémesure de la fusée en partant du constat que le canal de propagation entre le lanceur et les stations sol est très perturbé surtout pendant les phases de transition : allumage moteur, séparation des moteurs, de la coiffe... Ces phases génèrent des vibrations qui parcourent la fusée et dégradent le signal transmis. Un autre phénomène impactant le canal sont les *effets de flamme*. Ces effets provoquent une dérive de la phase voir même des sauts de la phase au cours de la transmission. Ces phénomènes accumulés peuvent conduire à une perte totale du lien de communication. Des méthodes de modulation appelées modulations à phase continue sont utilisées dans ce type de canal car elles sont robuste au déphasage et aux distorsions non-linéaires générées par l'amplificateur lorsqu'il est poussé à sa puissance maximale de sortie. La *continuous phase frequency shift keying* filtrée est la modulation la plus usitée dans les systèmes de télémesures. La chaîne de transmission se compose historiquement d'une modulation CPFSK associée à un code Reed-Solomon du côté de l'émetteur et d'un détecteur bloc ou de Viterbi côté récepteur. Cette solution n'est cependant pas satisfaisante par rapport à un système associant la CPFSK à des schémas de codage moderne capable d'atteindre la limite de Shannon (LDPC, Turbo-Codes). D'ailleurs [Ben15] on a montré que les codes de type LDPC étaient très adaptés aux modulations CPM. Cependant il ne suffit pas de remplacer dans la chaîne de transmission le code RS par un code LDPC, il faut également changer le décodeur *dur* par un décodeur *souple*. Ainsi l'objectif de cette thèse était d'élaborer côté émetteur la concaténation série d'une modulation CPM avec un code approchant la capacité et côté récepteur un détecteur souple non-cohérent basé sur un treillis.

Le chapitre 1 a présenté la modulation à phase continue et les méthodes de démodulation

existantes pour des canaux cohérent et non-cohérent. Cette partie a mis en lumière un désaccord entre les deux références majeures, [DS90] et [CFR00], au sujet de l'espace d'état utilisé pour démoduler des séquences CPM non-cohérentes. La cardinalité des états proposée pour la détection MAP [CFR00] était plus importante que celle introduite en [DS90] pour la détection ML. En fait l'auteur en [CFR00] ajoute à l'espace d'état fourni en [DS90] l'information de phase générant de surcroît un modèle étendu.

Dans ce contexte, on a étudié, dans le chapitre 2, le bénéfice et/ou la nécessité de cet écart en redérivant complètement les équations de la détection non-cohérente basées sur l'algorithme BCJR. Il est apparu que cet espace d'état redondant convenait à la démodulation dans les deux régimes cohérent et non-cohérent mais au prix d'une complexité accrue sans amélioration des performances. En fin de chapitre, on a tracé les courbes d'information mutuelle en sortie du détecteur souple pour les deux régimes. Il apparait que le comportement asymptotique des CPMs variaient avec avec la cohérence du canal. En fait, la plupart des CPMs utilisées ont une fonction de transfert d'information mutuelle convergeant vers le point de coordonnées $(1, 1)$ lors d'une démodulation cohérente alors que ce n'est plus du tout le cas en régime non-cohérent. Il en résulte un palier d'erreur en régime non-cohérent avec des schémas de codages convolutifs.

Afin de corriger cet effet néfaste sur le gain de codage, on a élaboré au chapitre 3 un pré-codage non-cohérent qui force la convergence des fonctions de transfert de l'information mutuelle vers le point $(1, 1)$. Le pré-codage consiste essentiellement à changer le mapping entre les symboles d'information et les trajectoires de la CPM. Ce *pré-traitement* préserve ***le débit maximal atteignable*** tout en permettant un décodage itératif efficace. L'élaboration des schémas de codage a été effectué au chapitre suivant.

Au chapitre 4, on a élaboré des schémas de codage LDPC concaténés en série avec des modulations CPMs qui donnent de bon gains pour les deux régimes. On a montré aux chapitres 2 et 3 que la trajectoire de l'information mutuelle est très différente suivant la cohérence du canal. Ainsi un schéma de codage optimisé pour le régime cohérent n'est pas adapté au régime non-cohérent et *vice versa*. Cependant la méthode de pré-codage proposé pour le chapitre 3 a largement résolue ce problème, on peut ainsi obtenir des profiles de code adaptés pour les

deux régimes avec pénalité de capacité réduite.

# Perspectives

## 1. Dérive de phase

Dans nos hypothèses de départ, on a simplifié le modèle de déphasage $\theta$ subit par le signal transmis. En effet, $\theta$ a été considéré dans la thèse comme constant au cours de la transmission d'une trame. On peut largement critiquer ce modèle puisqu'il est loin de la vérité physique du canal de propagation de la fusée. Il semblerait plus juste de dire que $\theta$ varie au cours de la transmission et donc que la nouvelle hypothèse qui doit être faite est $\theta(t)$. On a réalisé au chapitre 2 une brève évaluation de la robustesse du récepteur treillis non-cohérent à un déphasage temps-variant. On a suivi les recommandations de [DS90] pour modéliser la dérive de phase tel que $\theta_{k+1} = \theta_k + \Delta_k$ où $\Delta_k$ est une variable suivant une loi normale de moyenne nulle et de variance $\sigma^2_\Delta$. Il est apparu qu'un l'écart type de plus de 10° entre chaque symbole successif entraîne une forte dégradation ($> 3dB$) durant le processus de détection. Par conséquent, notre récepteur n'est pas efficace pour de telles contraintes. On recommande dans ce cas au lecteur d'utiliser une CPM à réponse complète et d'implémenter le récepteur par bloc avec une fenêtre de 1 symbole afin d'être le plus robuste possible à de large et rapide déphasage. Une autre voie à explorer pourrait être le suivi de la dérive de phase sur la fenêtre d'observation.

## 2. Précodage non-cohérent et détection cohérente

Au cours du chapitre 3, on a présenté le précodage non-coherent. Pour rappel, ce pré-traitement sur les symboles/bits était adapté aux deux modes de détection (cohérent et non-cohérent) contrairement à [Ben+07]. Cependant il est nécessaire d'utiliser l'espace d'état étendu introduit au chapitre 2 pour démoduler en régime cohérent les séquences précodées avec le précodage non-cohérent. Par conséquent la complexité du détecteur cohérent sera plus

importante qu'avec l'espace d'état minimal habituellement utilisé. Ainsi un axe d'amélioration serait de permettre la détection cohérent des séquences précodé non-cohérent avec l'espace d'état optimal.

### 3. Simulation à taille finie

On a évalué dans le dernier chapitre, les effets bénéfiques du précodage sur les performances d'un système concaténant une modulation CPM et un code LDPC. Des simulations asymptotiques (débit d'information et EXIT charts) ont été réalisées pour les modulations de type RC, GMSK et REC. Cependant, par manque de temps, nous avons pas mené de simulations à taille finie. De facto une amélioration possible serait de construire les matrices de parités à partir des profils de code données dans les tables du chapitre 4 et de lancer les simulations à taille finie.

### 4. Estimation du canal

On a considéré comme hypothèse de départ que le signal reçu était composé que d'un seul trajet. Or, il semble plus probable de considérer que le lien de communication soit affecté par du multi-trajet. En effet, en plus du trajet direct, d'autres trajets peuvent apparaître par réflexion sur la carlingue de la fusée ou sur le sol. Une estimation complète du canal de propagation devrait être réalisée. Dans le cas de systèmes conventionnels, le canal est estimé grâce à l'utilisation périodique de blocs de symboles connus positionnés entre les trames. Une estimation paramétrique par moindre carré est usuellement effectué à partir des échantillons reçus des symboles connus [Cha+17]. D'autres techniques d'estimation à l'aveugle,basées sur les propriétés statistiques du système, peuvent être mise en œuvre.

## 5. Égalisation

L'égalisation est une méthode efficace pour traiter le multi-trajet. La turbo-égalisation est un décodage itératif entre le canal de propagation vu comme un code convolutif non-redondant et le démodulateur (CPM). Une égalisation combinée avec un démodulateur cohérent fut initialement introduite par [Ozg08] pour les CPMs binaires puis étendue par la suite par [RBL05] et [Sch04] au cas M-aire. On pourrait adapter cette méthode au cas du récepteur non-cohérent basé sur un treillis afin de traiter le multi-trajet du canal de propagation de la fusée.

# Conclusions and perspectives

## Conclusions

In this thesis, we attempted to enhance the detection techniques of the launcher's telemetry signal. We have seen that the propagation channel between the launcher and the base stations is disturbed especially during the transitional phases as: engine-ignition, boosters separation, rocket's nose separation... Those phases generate a lot of vibration spreading over the launcher which cause impairments of the rocket's signal. Another phenomenon impacting the channel is called the *Flame Effects*. It leads to phase shift and/or phase hopping of the signal during the transmission. Those phenomenons may lead to the loss of the communication link. Modulations methods called *continuous phase modulations* are used in such transmissions since they are robust to channel phase shift and insensitive to nonlinear distortions generated by amplifiers at their maximal output power. The filtered continuous phase frequency shift keying was the most popular continuous phase modulation candidate for the telemetry system. The communication link is traditionally comprised of a CPFSK modulation associated with a Reed-Solomon code at the transmitter side, and a block or a Viterbi based detector at the receiver side. This solution was not satisfactory when directly compared to systems which join together the CPFSK modulation with modern coding schemes (LDPC, Turbo-codes) able to reach the Shannon capacity. Besides [Ben15] has shown that LDPC codes are well suited to CPM. However it is not enough to replace at the emitter/receiver the RS encoder/decoder for an LDPC transmitter/decoder, it is also necessary to change the hard demodulator to a soft one. In that respect, the aim of this thesis was to design, at the transmitter side, an efficient system composed of the serial concatenation of a capacity achieving code (as LDPC) with a CPM scheme and at the receiver side a non-coherent symbol/MAP receiver based on a trellis.

The first chapter aimed to introduce the continuous phase modulation and the demodulation's methods existing in the literature for the coherent and non-coherent channel. This

part brought to light a disagreement between the two major trellis-based detector references [DS90] and [CFR00] about the state space employed to demodulate non-coherent CPM sequences. The state cardinality proposed for the MAP detection in [CFR00] was greater than the one presented in [DS90] for the ML detection. Actually, the author in [CFR00] added to the state space provided in [DS90] the information phase generating an extended state space.

In that respect, we investigate in chapter 2, the benefit and/or the necessity of this discrepancy by rederiving completely the equations of the non-coherent detection based on the BCJR algorithm. It appeared that the *redundant* state space model offered the benefit to suit to coherent and non-coherent demodulation but at the expense of a greater complexity without performance enhancement. At the end of this chapter, we plotted the mutual information trajectories at the output of the soft detector in both the coherent and non-coherent regimes for various SNR. It appeared that the asymptotic behaviour of CPM varies with respect to the channel coherency. Actually, most CPMs in use have mutual information transfer function converging to the point $(1, 1)$ for coherent demodulation whereas it is not the case in non-coherent regime. As a result, convolutional coding schemes generate an error floor in non-coherent regime.

To solve this issue, we designed in chapter 3 a non-coherent precoding to force the convergence of the mutual information transfer functions to point $(1, 1)$. The precoding consists essentially in changing the mapping between the information symbols and the CPM waveforms. This preprocessing preserves ***the information rate*** (i.e the area under the mutual information chart) while enabling efficient iterative decoding. The design of coding schemes was carried out in the next chapter.

In chapter 4 we designed good LDPC schemes serially concatenated with CPM performing well for both the coherent and the non-coherent regimes. We pointed out along chapter 2 and 3 that the mutual information trajectories were very dissimilar depending on the coherency of the channel. Thus, a coding scheme optimized in coherent regime was not suited in non-coherent regime and *vice versa*. However, the precoded method proposed chapter 3 solved largely this issue, we obtained similar code profiles in both regimes with a reduced capacity penalty.

# Perspectives

## 1. Phase shift

In our hypothesis we simplified the model of the phase rotation $\theta$ undergone by the transmitted signal. Indeed in the thesis, $\theta$ was assumed constant during the transmission of a frame. This model can be widely criticized since it is physically far from the real launcher's propagation channel. It seems that $\theta$ varies along the transmission so the new assumption to be considered is $\theta(t)$. We provided in chapter 2 a brief evaluation of the robustness of the non-coherent trellis-based receiver to a timely varying phase deviation. We followed the recommendation of [DS90] to design the phase deviation such as $\theta_{k+1} = \theta_k + \Delta_k$ where $\Delta_k$ is an independent normal random variable with zero mean and a specified variance $\sigma^2_\Delta$. It appeared that a standard deviation of more than $10^\circ$ between each successive symbol generates strong degradation ($> 3dB$) during the detection process. Consequently, our receiver is no longer efficient against such constraints. We would advise the reader to consider a full response CPM and to implement the bloc receiver with an observation length of 1 symbol to be as robust as possible to such large uncommon deviation. Another path to explore could be the tracking of the phase shift inside the observation window.

## 2. Non-coherent precoding and coherent detection

Along chapter 3, we presented the non-coherent precoding (NC-precoding). This pre-processing on symbols/bits was adapted to both detection modes (coherent and non-coherent) unlike [Ben+07]. However it is necessary to use the extended state space model introduced chapter 2 to demodulate, in coherent regime, sequences precoded with the NC-precoding. Consequently, the complexity of the coherent detector will be greater than with the minimal state space traditionally used. An area for improvement is enable the detection of sequences precoded with the NC-precoding in coherent regime with the optimal state space.

### 3. Finite length simulation

We have evaluated in the last chapter, the benefits of precoding schemes in concatenated systems CPM/LDPC. In that respect, some asymptotic simulations were carried out (information rate/EXIT charts) for the RC, GMSK and REC modulations. Yet, for lack of time, we did not perform finite length simulations. De facto, one possible improvement should be to build the parity-check matrices from the code profiles left in chapter 4's tables and then to run the finite length simulations.

### 4. Channel estimation

We assumed in our hypothesis that the received signal was composed of solely one tape. However, it seems likely that the communication link is impacted by multipaths. Indeed, in addition to the line of sight, other tapes might come from the reflection on the launcher fuselage or on the ground. A complete channel estimation of the propagation channel of the launcher should be carried out. In the case of conventional systems, channels are modeled by using a periodic burst of known symbols located between frames. A parametric least square estimation is performed based on the received samples of the known symbols [Cha+17]. Other techniques might be explored as blind estimation which focuses on the statistic properties of the system.

### 5. Equalization

Equalization is an effective method to deal with multipath channels. A turbo equalization is an iterative decoding between the channel viewed as a non-redundant convolutional code and the (CPM) demodulator. A combined equalization and decoding were first performed by [Ozg08] for binary CPMs in coherent regime and were afterwards extended to the nonbinary case in [RBL05] and [Sch04]. We could suit this method to the non-coherent trellis based receiver to deal with the launcher channel multipaths.

Appendix A

# A detail proof of the link between coherent and non-coherent receivers

A short example is given Fig. A.1 to clarify the mathematical reduction of (A.1) and (A.2).

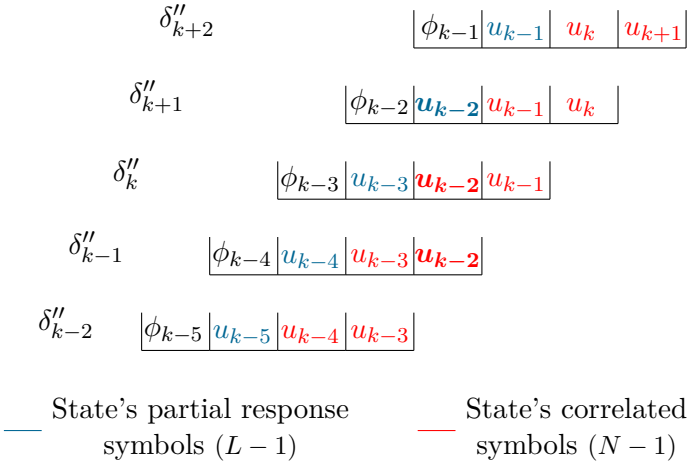



Figure A.1: TBR Extended State Space Model for N=3 and L=2

We can notice that $u_{k-2}$ is used in the following transition $\{\delta''_{k-2} \to \delta''_{k-1}, \delta''_{k-1} \to \delta''_k, \delta''_k \to \delta''_{k+1}, \delta''_{k+1} \to \delta''_{k+2}\}$. As well, we note that $\boldsymbol{r}_{k-2}$ is used only in transition implying $u_{k-2}$ as a correlated symbol (in red Fig. A.1) $\{\delta''_{k-2} \to \delta''_{k-1}, \delta''_{k-1} \to \delta''_k, \delta''_k \to \delta''_{k+1}\}$. Thus, in a more generic way, $u_{k-2}$ (resp. $\boldsymbol{r}_{k-2}$) intervenes only on states transition involving $\delta''^{k+L}_{k-N+1}$ (resp. $\delta''^{k+1}_{k-N+1}$). It comes out from the previous example,

$\forall\, k \in \{0, ..., N_s - 1\}$,

$$
\begin{aligned}
p(\boldsymbol{r}_{k-N+1}|\boldsymbol{r}_{L-1}^{N_s-1}, \delta''^{N_s-1}_{L-1}) &= p(\boldsymbol{r}_{k-N+1}|\boldsymbol{r}_{k-2N+2}^{k}, \delta''^{k+1}_{k-N+1}) \\
&= p(\boldsymbol{r}_{k-N+1}|\boldsymbol{r}_{k-2N+2}^{k}, u_{k-N+1}^{k}, \delta''_{k-N+1})
\end{aligned}
\tag{A.1}
$$

A much more generic expression of the hypothesis is given,

$\forall\,(i,j)\ \in\ \{0,...,N_s-1\}^2$ such as $i \leq j$

$$
\begin{aligned}
p(\boldsymbol{r}_{i}^{j}|\boldsymbol{r}_{L-1}^{N_s-1}, \delta''^{N_s-1}_{L-1}) &= p(\boldsymbol{r}_{i}^{j}|\boldsymbol{r}_{i-N+1}^{j+N-1}, \delta''^{j+N}_{i}) \\
&= p(\boldsymbol{r}_{i}^{j}|\boldsymbol{r}_{i-N+1}^{j+N-1}, u_{i}^{j+N-1}, \delta''_{i})
\end{aligned}
\tag{A.2}
$$

(A.1) and (A.2) will be helpful in the subsequent proof derivations.

### A.0.1 TBR Extended State Space: Mathematical Proof Appendix

This part is devoted to the proofs of equations referred to Section 2.4.1. Starting from (2.3), we expand $p(\boldsymbol{r}_{L-1}^{N_s-1}|u_k)$ in the subsequent manner.

$$
\begin{aligned}
p(\boldsymbol{r}_{L-1}^{N_s-1}|u_k) &= p(\boldsymbol{r}_{L-1}^{k-N}, \boldsymbol{r}_{k-N+1}^{k}, \boldsymbol{r}_{k+1}^{N_s-1}|u_k) \\
&= \sum_{\{\delta''_k\}} p(\boldsymbol{r}_{L-1}^{k-N}, \boldsymbol{r}_{k-N+1}^{k}, \boldsymbol{r}_{k+1}^{N_s-1}|\delta''_k, u_k) p(\delta''_k|u_k) \\
&= \sum_{\{\delta''_k\}} p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{L-1}^{k-N}, \boldsymbol{r}_{k-N+1}^{k}, \delta''_k, u_k) p(\delta''_k|u_k) p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k}, \delta''_k, u_k) p(\boldsymbol{r}_{k-N+1}^{k}|\delta''_k, u_k) \\
&= \sum_{\{\delta''_k\}} p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k}, \delta''_{k+1}) p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta''_k) p(\boldsymbol{r}_{k-N+1}^{k}|\delta''_k, u_k) p(\delta''_k)
\end{aligned}
\tag{A.3}
$$

Equation (2.5) that gives the forward probability $\alpha$ is obtained as follows:

$$
\begin{aligned}
\alpha_k(\delta''_k) &= p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta''_k)p(\delta''_k) \\
&= p(\boldsymbol{r}_{L-1}^{k-N-1},\boldsymbol{r}_{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta''_k)p(\delta''_k) \\
&= p(\delta''_k)\cdot\sum_{\{\delta''_{k-1}\}} p(\boldsymbol{r}_{L-1}^{k-N-1},\boldsymbol{r}_{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta''_k,\delta''_{k-1})p(\delta''_{k-1}|\boldsymbol{r}_{k-N+1}^{k-1},\delta''_k) \\
&= \sum_{\{\delta''_{k-1}\}} p(\boldsymbol{r}_{L-1}^{k-N-1}|\boldsymbol{r}_{k-N}^{k-2},\delta''_{k-1})p(\delta''_k)p(\delta''_{k-1}|\boldsymbol{r}_{k-N+1}^{k-1},\delta''_k)p(\boldsymbol{r}_{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta''_k,\delta''_{k-1})
\end{aligned}
\tag{A.4}
$$

Similarly, $\beta$ from (2.9) is calculated in the following manner:

$$
\begin{aligned}
\beta_{k+1}(\delta''_{k+1}) &= p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1}) \\
&= p(\boldsymbol{r}_{k+2}^{N_s-1},\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1}) \\
&= \sum_{\{\delta''_{k+2}\}} p(\boldsymbol{r}_{k+2}^{N_s-1},\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1},\delta''_{k+2})p(\delta''_{k+2}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1}) \\
&= \sum_{\{\delta''_{k+2}\}} p(\boldsymbol{r}_{k+2}^{N_s-1}|\boldsymbol{r}_{k-N+3}^{k+1},\delta''_{k+2})p(\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1},\delta''_{k+2})p(\delta''_{k+2}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1}) \\
&= \sum_{\{\delta''_{k+2}\}} \beta_{k+2}(\delta''_{k+2})p(\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1},\delta''_{k+2})p(\delta''_{k+2}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1})
\end{aligned}
\tag{A.5}
$$

### A.0.2 Non-coherent Detection Reduction Appendix

This part provides a detailed proof of the forward-backward and transition kernel probabilities presented in (2.19) in Section 2.4.2 devoted to the non-coherent regime. Based on those latter probabilities we derive the conditional probabilities given in (2.20) in the *non-coherent TBR state space*. Thus the relationship between $\alpha_k(\delta''_k)$ and $\alpha_k(\delta_k)$ is derived as follows.

$$
\begin{aligned}
\alpha_k(\delta_k) &= p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta_k)p(\delta_k) \\
&= p(\delta_k)\sum_{\phi_{k-N+1}} p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta_k,\phi_{k-N+1})p(\phi_{k-N+1}|\boldsymbol{r}_{k-N+1}^{k-1},\delta_k) \\
&= \sum_{\phi_{k-N+1}} p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta_k,\phi_{k-N+1})p(\phi_{k-N+1})p(\delta_k) \\
&= \sum_{\phi_{k-N+1}} p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta_k'')p(\delta_k'') \\
&= \sum_{\phi_{k-N+1}} \alpha(\delta_k'') \\
&= Q\cdot\alpha_k(\delta_k'')
\end{aligned}
\tag{A.6}
$$

Due to the marginalization step in the non-coherent regime, the observations are processed independently of the accumulated phase $\phi_{k-N+1}$. Therefore, this accumulated phase is not relevant in state $\delta_k$. This leads to the subsequent relation $p(\phi_{k-N+1}|\boldsymbol{r}_{k-N+1}^{k-1},\delta_k) = p(\phi_{k-N+1})$. Moreover we have seen in (2.17) that $\forall\phi_{k-N+1}\in\mathcal{Q}$, the $\alpha_k(\delta_k)$ are equals for an identical series of symbols $u_{k-N-L+2}^{k-1}$, that is why $\sum_{\phi_{k-N+1}}\alpha(\delta_k'') = Q\cdot\alpha_k(\delta_k'')$. Considering the same arguments, we can deduce the relationship between $\beta_{k+1}(\delta_{k+1}'')$ and $\beta_{k+1}(\delta_{k+1})$ in the following manner:

$$
\begin{aligned}
\beta_{k+1}(\delta_{k+1}) &= p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1}) \\
&= \sum_{\phi_{k-N+2}} p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1},\phi_{k-N+2})p(\phi_{k-N+2}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1}) \\
&= \sum_{\phi_{k-N+2}} p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1}'')p(\phi_{k-N+2}) \\
&= \tfrac{1}{Q}\cdot\sum_{\phi_{k-N+2}} \beta_{k+1}(\delta_{k+1}'') \\
&= \beta_{k+1}(\delta_{k+1}'')
\end{aligned}
\tag{A.7}
$$

We proceed analogously to the forward-backward probabilities to establish the relationship

between $\gamma(\delta''_k \to \delta''_{k+1}, \boldsymbol{r}^k_{k-N+1})$ and $\gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}^k_{k-N+1})$.

$$\begin{aligned}
\gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}^k_{k-N+1}) &= p(\boldsymbol{r}^k_{k-N+1}|\delta_k, u_k) \\
&= \sum_{\phi_{k-N+1}} p(\boldsymbol{r}^k_{k-N+1}|\delta_k, u_k, \phi_{k-N+1}) p(\phi_{k-N+1}|\delta_k, u_k) \\
&= \sum_{\phi_{k-N+1}} p(\boldsymbol{r}^k_{k-N+1}|\delta''_k, u_k) p(\phi_{k-N+1}) \\
&= \tfrac{1}{Q} \cdot \sum_{\phi_{k-N+1}} \gamma(\delta''_k \to \delta''_{k+1}, \boldsymbol{r}^k_{k-N+1}) \\
&= \gamma(\delta''_k \to \delta''_{k+1}, \boldsymbol{r}^k_{k-N+1})
\end{aligned} \tag{A.8}$$

We deduce the conditional probability in the *non-coherent TBR state space* from (A.6), (A.7) and (A.8).

$$\begin{aligned}
p(\boldsymbol{r}^{N_s-1}_{L-1}|u_k) =& \sum_{\{\delta''_k\}} \alpha_k(\delta''_k) \gamma(\delta''_k \to \delta''_{k+1}, \boldsymbol{r}^k_{k-N+1}) \beta_{k+1}(\delta''_{k+1}) \\
=& \sum_{\{\delta_k\}} \sum_{\{\phi_{k-N+1}\}} \alpha_k(\delta''_k) \gamma(\delta''_k \to \delta''_{k+1}, \boldsymbol{r}^k_{k-N+1}) \beta_{k+1}(\delta''_{k+1}) \\
=& \sum_{\{\delta_k\}} \gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}^k_{k-N+1}) \beta_{k+1}(\delta_{k+1}) \sum_{\{\phi_{k-N+1}\}} \alpha_k(\delta''_k) \\
=& \sum_{\{\delta_k\}} \gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}^k_{k-N+1}) \beta_{k+1}(\delta_{k+1}) \alpha_k(\delta_k)
\end{aligned} \tag{A.9}$$

### A.0.3 Coherent Detection Reduction Appendix

Similar to the non-coherent case, this part is dedicated to the computation of the forward-backward and kernel probabilities in the coherent regime presented in Section 2.4.3. The purpose will be the same, ie. to compute the conditional probability in the *coherent TBR state space*. Thus the relationship between $\alpha_k(\delta''_k)$ and $\alpha_k(\delta'_k)$ given in (2.25) is derived as follows :

$$
\begin{aligned}
\alpha_k(\delta'_k) &= p(\boldsymbol{r}_{L-1}^{k-1}|\delta'_k)p(\delta'_k) \\
&= p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta'_k)p(\boldsymbol{r}_{k-N+1}^{k-1}|\delta'_k)p(\delta'_k) \\
&= p(\boldsymbol{r}_{k-N+1}^{k-1},\delta'_k)\sum_{\{u_{k-N-L+2}^{k-L}\}} p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta'_k,u_{k-N-L+2}^{k-L})p(u_{k-N-L+2}^{k-L}|\boldsymbol{r}_{k-N+1}^{k-1},\delta'_k) \\
&= \sum_{\{u_{k-N-L+2}^{k-L}\}} p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta'_k,u_{k-N-L+2}^{k-L})p(\boldsymbol{r}_{k-N+1}^{k-1},\delta'_k,u_{k-N-L+2}^{k-L}) \\
&= \sum_{\{u_{k-N-L+2}^{k-L}\}} p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta''_k)p(\boldsymbol{r}_{k-N+1}^{k-1},\delta''_k) \\
&= \sum_{\{u_{k-N+1-L+1}^{k-L}\}} p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1},\delta''_k)p(\delta''_k)p(\boldsymbol{r}_{k-N+1}^{k-1}|\delta''_k) \\
&= \sum_{\{u_{k-N-L+2}^{k-L}\}} \alpha_k(\delta''_k)p(\boldsymbol{r}_{k-N+1}^{k-1}|\delta''_k)
\end{aligned}
\tag{A.10}
$$

In (A.10), $\{\phi_k, u_{k-N-L+2}^{k-L}\}$ is equivalent to $\{\phi_{k-N+1}, u_{k-N-L+2}^{k-L}\}$ since $\phi_k = \phi_{k-N+1} + 2\pi h \sum_{i=k-N-L+2}^{k-L} u_i$. The relationship between $\beta_{k+1}(\delta''_{k+1})$ and $\beta_{k+1}(\delta'_{k+1})$ introduced in (2.25) is easily established as follows :

$$
\begin{aligned}
\beta_{k+1}(\delta''_{k+1}) &= p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k},\delta''_{k+1}) \\
&= p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k},u_{k-N-L+2}^{k-L+1},\phi_k,u_{k-L+2}^{k}) \\
&= p(\boldsymbol{r}_{k+1}^{N_s-1}|\delta'_{k+1}) \\
&= \beta_{k+1}(\delta'_{k+1})
\end{aligned}
\tag{A.11}
$$

$r_{k+1}$ is fully described in coherent regime by $\{\phi_k, u_{k-L+2}^{k}\}$ since there is no phase shit corrupting the observations ($\theta$=0). Thus taking into account $r_{k+N+2}^{k}$ and $u_{k-N-L+2}^{k-L+1}$ is useless so $p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k},u_{k-N-L+2}^{k-L+1},\phi_k,u_{k-L+2}^{k}) = p(\boldsymbol{r}_{k+1}^{N_s-1}|\phi_k,u_{k-L+2}^{k})$. We proceed to establish

the relationship between $\gamma(\delta''_k \to \delta''_{k+1}, \boldsymbol{r}^k_{k-N+1})$ and $\gamma(\delta'_k \to \delta'_{k+1}, \boldsymbol{r}_k)$ given in (2.25).

$$\begin{aligned}
\gamma(\delta''_k \to \delta''_{k+1}, \boldsymbol{r}^k_{k-N+1}) &= p(\boldsymbol{r}^k_{k-N+1}|\delta''_k, u_k) \\
&= p(\boldsymbol{r}^{k-1}_{k-N+1}, r_k|\delta''_k, u_k) \\
&= p(\boldsymbol{r}^{k-1}_{k-N+1}|r_k, \delta''_k, u_k)p(\boldsymbol{r}_k|\delta''_k, u_k) \\
&= p(\boldsymbol{r}^{k-1}_{k-N+1}|\delta''_k)p(r_k|\delta'_k, u_k) \\
&= p(\boldsymbol{r}^{k-1}_{k-N+1}|\delta''_k)\gamma(\delta'_k \to \delta'_{k+1}, \boldsymbol{r}_k)
\end{aligned} \tag{A.12}$$

In the coherent regime, observations are processed independently from one to another, that is the reason why it stems from equation (2.14) (and based on reference [VCT10]) that $p(\boldsymbol{r}^k_{k-N+1}|\delta''_k, u_k) = \prod_{i=k-N+1}^{k} p(\boldsymbol{r}_i|\phi_i, u^i_{i-L+1})$. Finally, we deduce the conditional probability in the *coherent TBR state space* from equations (A.10), (A.11) and (A.12).

$$\begin{aligned}
p(\boldsymbol{r}^{N_s-1}_{L-1}|u_k) =& \sum_{\{\delta''_k\}} \alpha_k(\delta''_k)\gamma(\delta''_k \to \delta''_{k+1}, \boldsymbol{r}^k_{k-N+1})\beta_{k+1}(\delta''_{k+1}) \\
=& \sum_{\{\delta'_k\}} \sum_{\{\boldsymbol{u}^{k-L}_{k-N-L+2}\}} \alpha_k(\delta''_k)\gamma(\delta''_k \to \delta''_{k+1}, \boldsymbol{r}^k_{k-N+1})\beta_{k+1}(\delta''_{k+1}) \\
=& \sum_{\{\delta'_k\}} \gamma(\delta'_k \to \delta'_{k+1}, \boldsymbol{r}_k)\beta_{k+1}(\delta'_{k+1}) \sum_{\{\boldsymbol{u}^{k-L}_{k-N-L+2}\}} \alpha_k(\delta''_k)p(\boldsymbol{r}^{k-1}_{k-N+1}|\delta''_k) \\
=& \sum_{\{\delta'_k\}} \gamma(\delta'_k \to \delta'_{k+1}, \boldsymbol{r}_k)\beta_{k+1}(\delta'_{k+1})\alpha_k(\delta'_k)
\end{aligned} \tag{A.13}$$

### A.0.4 Mutual Information Rate Appendix

This part is dedicated to prove results set in section 2.7.1. Starting from equation (2.34).

$$
\begin{aligned}
I(u_0^{N_s-1}, \boldsymbol{r}_{L-1}^{N_s-1} | \delta''_{N+L-2}) &= E\left[ log_2 \left( \frac{p(u_0^{N_s-1} | \boldsymbol{r}_{L-1}^{N_s-1}, \delta''_{N+L-2})}{p(u_0^{N_s-1} | \delta''_{N+L-2})} \right) \right] \\
&= E\left[ log_2 \left( \frac{p(u_0^{N_s-1} | \boldsymbol{r}_{L-1}^{N_s-1}, \delta''_{N+L-2})}{\left(\frac{1}{M}\right)^{N_s-(N+L-2)}} \right) \right] \\
&= \left(N_s - (N+L-2)\right) log_2(M) \\
&\quad + E\left[ log_2 \left( p(u_0^{N_s-1} | \boldsymbol{r}_{L-1}^{N_s-1}, \delta''_{N+L-2}) \right) \right]
\end{aligned}
\tag{A.14}
$$

Relation (2.35) has been derived as follows :

$$
\begin{aligned}
p(u_{N+L-2}^{N_s-1} | \boldsymbol{r}_{L-1}^{N_s-1}, \delta''_{N+L-2}) &= p(u_{N_s-1}, ..., u_{N+L-2} | \boldsymbol{r}_{L-1}^{N_s-1}, \delta''_{N+L-2}) \\
&= \prod_{k=N+L-2}^{N_s-1} p(u_k | u_{N+L-2}^{k-1}, \boldsymbol{r}_{L-1}^{N_s-1}, \delta''_{N+L-2})
\end{aligned}
\tag{A.15}
$$

Appendix B

# A detail proof of the novel non-coherent MAP receivers

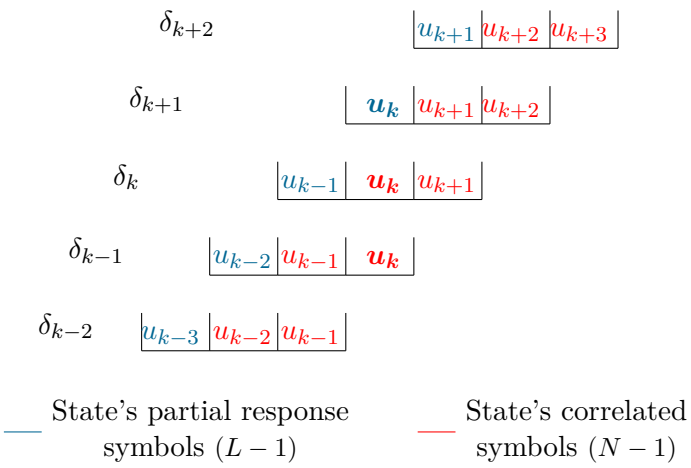



Figure B.1: Non-coherent TBR State Space Model for N=3 and L=2

A short example is given Fig. B.1 to clarify the mathematical reduction of (B.1) and (B.2). We can notice that $u_{k-2}$ is used in the following transition $\{\delta_{k-2} \rightarrow \delta_{k-1}, \delta_{k-1} \rightarrow \delta_k, \delta_k \rightarrow \delta_{k+1}, \delta_{k+1} \rightarrow \delta_{k+2}\}$. As well, we note that $\boldsymbol{r}_{k-2}$ is used only in transition implying $u_{k-2}$ as a correlated symbol (in red Fig. B.1) $\{\delta_{k-2} \rightarrow \delta_{k-1}, \delta_{k-1} \rightarrow \delta_k, \delta_k \rightarrow \delta_{k+1}\}$. Thus, in a more generic way, $u_{k-2}$ (resp. $\boldsymbol{r}_{k-2}$) intervenes only on states transition involving $\delta_{k-N+1}^{k+L}$ (resp. $\delta_{k-N+1}^{k+1}$). It comes out from the previous example,

$\forall\, k \in \{0, ..., N_s - 1\}$,

$$
\begin{aligned}
p(\boldsymbol{r}_{k-N+1}|\boldsymbol{r}_{L-1}^{N_s-1}, \delta_{L-1}^{N_s-N-L+2}) &= p(\boldsymbol{r}_{k-N+1}|\boldsymbol{r}_{k-2N+2}^{k}, \delta_{k-N+1}^{k+1}) \\
&= p(\boldsymbol{r}_{k-N+1}|\boldsymbol{r}_{k-2N+2}^{k}, u_{k-N+1}^{k}, \delta_{k-N+1})
\end{aligned}
\tag{B.1}
$$

A much more generic expression of the hypothesis is given,

$\forall\,(i,j)\ \in\ \{0,...,N_s-1\}^2$ such as $i \leq j$

$$
\begin{aligned}
p(\boldsymbol{r}_i^j|\boldsymbol{r}_{L-1}^{N_s-1}, \delta_{L-1}^{N_s-(N+L-2)}) &= p(\boldsymbol{r}_i^j|\boldsymbol{r}_{i-N+1}^{j+N-1}, \delta_i^{j+N}) \\
&= p(\boldsymbol{r}_i^j|\boldsymbol{r}_{i-N+1}^{j+N-1}, u_i^{j+N-1}, \delta_i)
\end{aligned}
\tag{B.2}
$$

(B.1) and (B.2) will be helpful in the subsequent proof derivations.

We start with the conditional probability equation $p(u_k|\boldsymbol{r}_{L-1}^{N_s-1})$ developed section 2.5.

$$
\begin{aligned}
p(\boldsymbol{r}_{L-1}^{N_s-1}|u_k) &= p(\boldsymbol{r}_{L-1}^{k-N}, \boldsymbol{r}_{k-N+1}^{k}, \boldsymbol{r}_{k+1}^{N_s-1}|u_k) \\
&= \sum_{\{\delta_k\}} p(\boldsymbol{r}_{L-1}^{k-N}, \boldsymbol{r}_{k-N+1}^{k}, \boldsymbol{r}_{k+1}^{N_s-1}|\delta_k, u_k)p(\delta_k|u_k) \\
&= \sum_{\{\delta_k\}} p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{L-1}^{k-N}, \boldsymbol{r}_{k-N+1}^{k}, \delta_k, u_k)p(\delta_k|u_k)p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k}, \delta_k, u_k)p(\boldsymbol{r}_{k-N+1}^{k}|\delta_k, u_k) \\
&= \sum_{\{\delta_k\}} p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k}, \delta_{k+1})p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta_k)p(\boldsymbol{r}_{k-N+1}^{k}|\delta_k, u_k)p(\delta_k)
\end{aligned}
\tag{B.3}
$$

The classical forward, backward and transition kernel probabilities (denoted $\alpha, \beta$ and $\gamma$ respectively) are given as follows.

$$\begin{aligned}
\gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}_{k-N+1}^{k}) &= p(\boldsymbol{r}_{k-N+1}^{k}|\delta_k, u_k) \\
\alpha_k(\delta_k) &= p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta_k)p(\delta_k) \\
\beta_{k+1}(\delta_{k+1}) &= p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k}, \delta_{k+1})
\end{aligned} \tag{B.4}$$

Classically with the BCJR algorithm, $\alpha_k$ can be calculated as

$$\begin{aligned}
\alpha_k(\delta_k) &= p(\boldsymbol{r}_{L-1}^{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta_k)p(\delta_k) \\
&= p(\boldsymbol{r}_{L-1}^{k-N-1}, \boldsymbol{r}_{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta_k)p(\delta_k) \\
&= p(\delta_k) \cdot \sum_{\{\delta_{k-1}\}} p(\boldsymbol{r}_{L-1}^{k-N-1}, \boldsymbol{r}_{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta_k, \delta_{k-1})p(\delta_{k-1}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta_k) \\
&= \sum_{\{\delta_{k-1}\}} p(\boldsymbol{r}_{L-1}^{k-N-1}|\boldsymbol{r}_{k-N}^{k-2}, \delta_{k-1})p(\delta_k)p(\delta_{k-1}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta_k)p(\boldsymbol{r}_{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta_k, \delta_{k-1})
\end{aligned} \tag{B.5}$$

where

$$\begin{aligned}
p(\boldsymbol{r}_{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta_k, \delta_{k-1}) &= p(\boldsymbol{r}_{k-N}|\boldsymbol{r}_{k-N+1}^{k-1}, \delta_{k-1}, u_{k-1}) \\
&= \frac{p(\boldsymbol{r}_{k-N}^{k-1}|\delta_{k-1}, u_{k-1})}{p(\boldsymbol{r}_{k-N+1}^{k-1}|\delta_{k-1}, u_{k-1})} \\
&= \frac{\gamma(\delta_{k-1} \to \delta_k, r_{k-N}^{k-1})}{p(\boldsymbol{r}_{k-N+1}^{k-1}|\delta_{k-1}, u_{k-1})}
\end{aligned} \tag{B.6}$$

Equation (B.6) is derived from Bayes' theorem and from equation (B.4). Moreover, we have

$$
\begin{aligned}
p(\delta_{k-1}|\boldsymbol{r}_{k-N+1}^{k-1},\delta_k)p(\delta_k) &= p(\delta_{k-1}|\delta_k)p(\delta_k) \\
&= p(\delta_k|\delta_{k-1})p(\delta_{k-1}) \\
&= p(u_{k-1})p(\delta_{k-1})
\end{aligned}
\tag{B.7}
$$

In equation (B.7), $p(\delta_{k-1}|\boldsymbol{r}_{k-N+1}^{k-1},\delta_k)$ is independent of observations $\boldsymbol{r}_{k-N+1}^{k-1}$. Finally, after all terms have been collected, a recursion of $\alpha$ is obtained as follows

$$
\alpha_k(\delta_k) = \sum_{\{\delta_{k-1}\}} \alpha_{k-1}(\delta_{k-1}) \frac{\gamma(\delta_{k-1} \to \delta_k, \boldsymbol{r}_{k-N}^{k-1})}{p(\boldsymbol{r}_{k-N+1}^{k-1}|\delta_{k-1}, u_{k-1})} p(u_{k-1}) \tag{B.8}
$$

Similarly, $\beta$ can be calculated using a backward recursion.

$$
\begin{aligned}
\beta_{k+1}(\delta_{k+1}) &= p(\boldsymbol{r}_{k+1}^{N_s-1}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1}) \\
&= p(\boldsymbol{r}_{k+2}^{N_s-1},\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1}) \\
&= \sum_{\{\delta_{k+2}\}} p(\boldsymbol{r}_{k+2}^{N_s-1},\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1},\delta_{k+2}) p(\delta_{k+2}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1}) \\
&= \sum_{\{\delta_{k+2}\}} p(\boldsymbol{r}_{k+2}^{N_s-1}|\boldsymbol{r}_{k-N+3}^{k+1},\delta_{k+2}) p(\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1},\delta_{k+2}) p(\delta_{k+2}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1}) \\
&= \sum_{\{\delta_{k+2}\}} \beta_{k+2}(\delta_{k+2}) p(\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1},\delta_{k+2}) p(\delta_{k+2}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1})
\end{aligned}
\tag{B.9}
$$

where

$$
\begin{aligned}
p(\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1},\delta_{k+2}) &= p(\boldsymbol{r}_{k+1}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1},u_{k+1}) \\
&= \frac{p(\boldsymbol{r}_{k-N+2}^{k+1}|\delta_{k+1},u_{k+1})}{p(\boldsymbol{r}_{k-N+2}^{k}|\delta_{k+1},u_{k+1})} \\
&= \frac{\gamma(\delta_{k+1}\rightarrow\delta_{k+2},\boldsymbol{r}_{k-N+2}^{k+1})}{p(\boldsymbol{r}_{k-N+2}^{k}|\delta_{k+1},u_{k+1})}
\end{aligned}
\tag{B.10}
$$

Equation (B.10) is obtained from Bayes' theorem and from equation (B.4). $p(\delta_{k+2}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1})$ being independent of observations $\boldsymbol{r}_{k-N+2}^{k}$, $p(\delta_{k+2}|\boldsymbol{r}_{k-N+2}^{k},\delta_{k+1}) = p(\delta_{k+2}|\delta_{k+1})$, which leads to $p(\delta_{k+2}|\delta_{k+1}) = p(u_{k+1})$. We get the recursion on $\beta$ as follows:

$$
\beta_{k+1}(\delta_{k+1}) = \sum_{\{\delta_{k+2}\}} \beta_{k+2}(\delta_{k+2}) \frac{\gamma(\delta_{k+1}\rightarrow\delta_{k+2},\boldsymbol{r}_{k-N+2}^{k+1})}{p(\boldsymbol{r}_{k-N+2}^{k}|\delta_{k+1},u_{k+1})} p(u_{k+1}) \tag{B.11}
$$

Finally gathering $\alpha, \beta$ and $\gamma$ leads to

$$
p(u_k|\boldsymbol{r}_{L-1}^{N_s-1}) \propto \sum_{\{\delta_k\}} \alpha_k(\delta_k)\gamma(\delta_k\rightarrow\delta_{k+1},\boldsymbol{r}_{k-N+1}^{k})\beta_{k+1}(\delta_{k+1})p(u_k) \tag{B.12}
$$

Using sufficient statistics at the output of the filter bank we have ([VCT10])

$$
p(\boldsymbol{r}_k|u_{k-L+1}^{k},\psi_k) \propto e^{\rho\cdot\Re\left(e^{-j\psi_k} r_{u_{k-L+1}^{k},k}\right)} \tag{B.13}
$$

where $\Re(.)$ stands for the real part, $\psi_k = \phi_k + \theta$ (with $\psi_0 = \theta$) and $\rho = 2\sqrt{Es}/N_0$. The branch metric associated to TBR state space model requires the computation of the conditional probability related to $\gamma$ and given for any CPM by

$$\begin{aligned}\gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}_{k-N+1}^k) &= p(\boldsymbol{r}_{k-N+1}^k|\delta_k, u_k) \\ p(\boldsymbol{r}_{k-N+1}^k|\delta_k, u_k, \psi_{k-N+1}) &\propto e^{\rho\cdot\Re\left(e^{-j\psi_{k-N+1}}\mu(u_{k-N-L+2}^k)\right)}\end{aligned} \tag{B.14}$$

where

$$\mu(u_{k-N-L+2}^k) = \sum_{i=k-N+1}^{k} r_{u_{i-L+1}^i,i}\cdot e^{-j2\pi h\sum_{n=k-N-L+2}^{i-L} u_n}$$

In non-coherent regime the phase induced by the channel is unknown. Averaging over the random phase $\psi$ removes the channel phase dependency from the branch metric and yields the well known modified zero order Bessel function of the first kind.

$$\begin{aligned}\gamma(\delta_k \to \delta_{k+1}, \boldsymbol{r}_{k-N+1}^k) &= p(\boldsymbol{r}_{k-N+1}^k|\delta_k, u_k) \\ &\propto I_0\left(\rho\cdot\left|\mu(u_{k-N-L+2}^k)\right|\right)\end{aligned} \tag{B.15}$$

Then equations (B.8) and (B.11) can be rewritten as

$$\begin{aligned}\alpha_k(\delta_k) &\propto \sum_{\{\delta_{k-1}\}} \alpha_{k-1}(\delta_{k-1})\frac{I_0\left(\rho\cdot\left|\mu(u_{k-N-L+1}^{k-1})\right|\right)}{I_0\left(\rho\cdot\left|\mu(u_{k-N-L+2}^{k-1})\right|\right)}p(u_{k-1}) \\ \beta_k(\delta_k) &\propto \sum_{\{\delta_{k+1}\}} \beta_{k+1}(\delta_{k+1})\frac{I_0\left(\rho\cdot\left|\mu(u_{k-N-L+2}^{k})\right|\right)}{I_0\left(\rho\cdot\left|\mu(u_{k-N-L+2}^{k-1})\right|\right)}p(u_{k})\end{aligned} \tag{B.16}$$

# Bibliography


[AAS13] John B Anderson, Tor Aulin, and Carl-Erik Sundberg. *Digital phase modulation*. Springer Science & Business Media, 2013 (cit. on pp. 5, 13, 22).

[AKB04] Alexei Ashikhmin, Gerhard Kramer, and Stephan ten Brink. "Extrinsic information transfer functions: model and erasure channel properties." In: *IEEE Transactions on Information Theory* 50.11 (2004), pp. 2657–2673 (cit. on pp. 64, 66, 118).

[Arn+06] Dieter-Michael Arnold et al. "Simulation-based computation of information rates for channels with memory." In: *IEEE Transactions on information theory* 52.8 (2006), pp. 3498–3508 (cit. on p. 64).

[ARS81a] Tor Aulin, Nils Rydbeck, and C-E Sundberg. "Continuous phase modulation–Part I: partial response signaling." In: *IEEE Transactions on Communications* 29.3 (1981), pp. 210–225 (cit. on pp. 5, 13, 20, 27).

[ARS81b] Tor Aulin, Nils Rydbeck, and C-E Sundberg. "Continuous phase modulation–Part I: partial response signaling." In: *IEEE Transactions on Communications* 29.3 (1981), pp. 196–209 (cit. on pp. 5, 13, 20, 27).

[Bah+74] Lalit Bahl et al. "Optimal decoding of linear codes for minimizing symbol error rate (corresp.)" In: *IEEE Transactions on information theory* 20.2 (1974), pp. 284–287 (cit. on pp. 20, 21, 34, 40, 47, 50).

[BB06] Amir Bennatan and David Burshtein. "Design and analysis of nonbinary LDPC codes for arbitrary discrete-memoryless channels." In: *IEEE Transactions on Information Theory* 52.2 (2006), pp. 549–583 (cit. on pp. 79, 81, 116, 117).

[Ben+07] Sergio Benedetto et al. "A pragmatic approach to coded continuous-phase modulation." In: *Information Theory and Applications Workshop, 2007*. IEEE. 2007, pp. 36–40 (cit. on pp. 4, 12, 74–76, 82, 87, 89, 91, 92, 94, 127, 133, 139).

[Ben+14a] Tarik Benaddi et al. “Design of unstructured and protograph-based ldpc coded continuous phase modulation.” In: *Information Theory (ISIT), 2014 IEEE International Symposium on*. IEEE. 2014, pp. 1982–1986 (cit. on pp. 103, 119).

[Ben+14b] Bouchra Benammar et al. “Asymptotic analysis and design of iterative receivers for non linear ISI channels.” In: *Turbo Codes and Iterative Information Processing (ISTC), 2014 8th International Symposium on*. IEEE. 2014, pp. 173–177 (cit. on pp. 119, 120).

[Ben15] Tarik Benaddi. “Sparse graph-based coding schemes for continuous phase modulations.” PhD thesis. 2015 (cit. on pp. 4, 12, 103, 116, 119, 120, 131, 137).

[BG96] Claude Berrou and Alain Glavieux. “Near optimum error correcting coding and decoding: Turbo-codes.” In: *IEEE Transactions on communications* 44.10 (1996), pp. 1261–1271 (cit. on pp. 4, 12, 103).

[Ccs] *Bandwidth-efficient modulations summary of definition, implementation, and performance*. 413.0-G-2. Consultative Committee for Space Data Systems. 2009 (cit. on p. 11).

[CFC10] Enrico Casini, Dario Fertonani, and Giulio Colavolpe. “Advanced CPM receiver for the NATO tactical narrowband waveform.” In: *2010-MILCOM 2010 MILITARY COMMUNICATIONS CONFERENCE*. IEEE. 2010, pp. 1725–1730 (cit. on pp. 20, 21, 25).

[CFR00] Giulio Colavolpe, Gianluigi Ferrari, and Riccardo Raheli. “Noncoherent iterative (turbo) decoding.” In: *IEEE Transactions on Communications* 48.9 (2000), pp. 1488–1498 (cit. on pp. 3, 5, 6, 11, 13, 14, 21, 34, 40, 41, 46, 47, 50, 56, 63, 67, 72, 132, 138).

[Cha+17] Romain Chayot et al. “Channel estimation and equalization for CPM with application for aeronautical communications via a satellite link.” In: *Military Communications Conference (MILCOM), MILCOM 2017-2017 IEEE*. IEEE. 2017, pp. 888–893 (cit. on pp. 134, 140).

[CR97] Giulio Colavolpe and Riccardo Raheli. “Reduced-complexity detection and phase synchronization of CPM signals.” In: *IEEE Transactions on Communications* 45.9 (1997), pp. 1070–1079 (cit. on pp. 21, 34).

[CT12] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012 (cit. on pp. 64, 77).

[CVT07] Shi Cheng, Matthew C Valenti, and Don Torrieri. “Coherent and multi-symbol noncoherent CPFSK: Capacity and code design.” In: *Military Communications Conference, 2007. MILCOM 2007. IEEE*. IEEE. 2007, pp. 1–7 (cit. on p. 119).

[CVT09] Shi Cheng, Matthew C Valenti, and Don Torrieri. “Coherent continuous-phase frequency-shift keying: parameter optimization and code design.” In: *IEEE Transactions on Wireless Communications* 8.4 (2009) (cit. on p. 119).

[Det11] Thomas Frederick Detwiler. “Continuous phase modulation for high speed fiber-optic links.” PhD thesis. Georgia Institute of Technology, 2011 (cit. on p. 25).

[DL12] Luc Deneire and Jerome Lebrun. “Continuous phase modulation and space-time coding: A candidate for wireless robotics.” In: *Wireless Personal Communications* 64.3 (2012), pp. 473–487 (cit. on p. 25).

[DS90] Dariush Divsalar and Marvin K Simon. “Multiple-symbol differential detection of MPSK.” In: *IEEE Transactions on Communications* 38.3 (1990), pp. 300–308 (cit. on pp. 3, 5, 6, 11, 13, 14, 20, 21, 38, 46, 47, 50, 56, 57, 59, 63, 67, 68, 132, 133, 138, 139).

[Gal63] Robert G Gallager. “Low-density parity-check codes.” PhD thesis. Cambridge, Massachusetts, 1963 (cit. on pp. 102, 107, 110).

[Geo00] Mark Geoghegan. “Description and performance results for a multi-h CPM telemetry waveform.” In: *MILCOM 2000. 21st Century Military Communications Conference Proceedings*. Vol. 1. IEEE. 2000, pp. 353–357 (cit. on pp. 20, 21, 25).

[Hag04] Joachim Hagenauer. “The EXIT chart-introduction to extrinsic information transfer in iterative processing.” In: *Signal Processing Conference, 2004 12th European*. IEEE. 2004, pp. 1541–1548 (cit. on pp. 64, 66, 76, 118).

[Ham86] Richard W Hamming. *Coding and Theory*. Englewood Cliffs, NJ, 1986 (cit. on p. 77).

[HEA05] Xiao-Yu Hu, Evangelos Eleftheriou, and Dieter-Michael Arnold. “Regular and irregular progressive edge-growth tanner graphs.” In: *IEEE Transactions on Information Theory* 51.1 (2005), pp. 386–398 (cit. on p. 106).

[Iri] *IRIG STANDARD 106-17*. Range Commanders Council. 2017, pp. 2–4 (cit. on pp. 2, 10).

[Kal89] Ghassan Kawas Kaleh. “Simple coherent receivers for partial response continuous phase modulation.” In: *IEEE Journal on Selected Areas in Communications* 7.9 (1989), pp. 1427–1436 (cit. on pp. 21, 34).

[KNH06] Jrg Kliewer, Soon Xin Ng, and Lajos Hanzo. “Efficient computation of EXIT functions for nonbinary iterative decoding.” In: *IEEE Transactions on Communications* 54.12 (2006), pp. 2133–2136 (cit. on pp. 79, 81).

[Lam+03] LH-J Lampe et al. “Noncoherent continuous-phase modulation for DS-CDMA.” In: *Communications, 2003. ICC’03. IEEE International Conference on*. Vol. 5. IEEE. 2003, pp. 3282–3286 (cit. on p. 25).

[Lau86] Pierre Laurent. “Exact and approximate construction of digital phase modulations by superposition of amplitude modulated pulses (AMP).” In: *IEEE transactions on communications* 34.2 (1986), pp. 150–160 (cit. on pp. 21, 34).

[LT06] Gee L Lui and Kuang Tsai. *Gaussian minimum shift keying (GMSK) precoding communication method*. US Patent 7,072,414. 2006 (cit. on p. 75).

[Mes+16] Malek Messai et al. “Binary CPMs with improved spectral efficiency.” In: *IEEE communications letters* 20.1 (2016), pp. 85–88 (cit. on p. 75).

[MKF14] Messai Malek, Amis Karine, and Guilloud Frédéric. “A low complexity iterative soft detection for bit interleaved coded CPM.” In: *Turbo Codes and Iterative Information Processing (ISTC), 2014 8th International Symposium on*. IEEE. 2014, pp. 12–16 (cit. on p. 21).

[MM95] Umberto Mengali and Michele Morelli. “Decomposition of M-ary CPM signals into PAM waveforms.” In: *IEEE transactions on information theory* 41.5 (1995), pp. 1265–1275 (cit. on pp. 21, 34).

[MMC98] Robert J. McEliece, David J. C. MacKay, and Jung-Fu Cheng. “Turbo decoding as an instance of Pearl’s" belief propagation" algorithm.” In: *IEEE Journal on selected areas in communications* 16.2 (1998), pp. 140–152 (cit. on p. 103).

[MN96] David JC MacKay and Radford M Neal. “Near Shannon limit performance of low density parity check codes.” In: *Electronics letters* 32.18 (1996), pp. 1645–1646 (cit. on p. 103).

[MPFBH92] Michel Mouly, Marie-Bernadette Pautet, and Thomas Foreword By-Haug. *The GSM system for mobile communications*. Telecom publishing, 1992 (cit. on p. 25).

[OSL17] Rami Othman, Alexandre Skrzypczak, and Yves Louët. “PAM Decomposition of Ternary CPM With Duobinary Encoding.” In: *IEEE Transactions on Communications* 65.10 (2017), pp. 4274–4284 (cit. on p. 75).

[Ozg08] Barıs Ozgül. “Advanced transceiver design for continuous phase modulation.” PhD thesis. Bogaziçi University, 2008 (cit. on pp. 135, 140).

[Pad+05] Krishnan Padmanabhan et al. “General CPM and its capacity.” In: *Information Theory, 2005. ISIT 2005. Proceedings. International Symposium on*. IEEE. 2005, pp. 750–754 (cit. on p. 65).

[Per+10] Alberto Perotti et al. “Capacity-achieving CPM schemes.” In: *IEEE Transactions on Information Theory* 56.4 (2010), pp. 1521–1541 (cit. on pp. 4, 12, 74–76).

[Pro] John G Proakis. *Digital Communications Fourth Edition, Chapter 11, 2001* (cit. on pp. 5, 13, 22).

[RBL05] Luca Rugini, Paolo Banelli, and Geert Leus. "Simple equalization of time-varying channels for OFDM." In: *IEEE communications letters* 9.7 (2005), pp. 619–621 (cit. on pp. 135, 140).

[RD99] Dan Raphaeli and Dariush Divsalar. "Multiple-symbol noncoherent decoding of uncoded and convolutionally coded continuous phase modulation." In: *Journal of Communications and Networks* 1.4 (1999), pp. 238–248 (cit. on pp. 3, 5, 11, 13, 20, 21, 34, 38–41, 43, 50, 72).

[Rim88] Bixio E Rimoldi. "A decomposition approach to CPM." In: *IEEE Transactions on Information Theory* 34.2 (1988), pp. 260–270 (cit. on pp. 5, 13, 20, 21, 29, 32, 34, 35, 58, 82).

[RS60] Irving S Reed and Gustave Solomon. "Polynomial codes over certain finite fields." In: *Journal of the society for industrial and applied mathematics* 8.2 (1960), pp. 300–304 (cit. on pp. 3, 11).

[RU01] Thomas J Richardson and Rüdiger L Urbanke. "The capacity of low-density parity-check codes under message-passing decoding." In: *IEEE Transactions on information theory* 47.2 (2001), pp. 599–618 (cit. on pp. 103, 114).

[Sch04] Philip Schniter. "Low-complexity equalization of OFDM in doubly selective channels." In: *IEEE Transactions on Signal processing* 52.4 (2004), pp. 1002–1011 (cit. on pp. 135, 140).

[Sco+10] Andrea Scorzolini et al. "European enhanced space-based AIS system study." In: *2010 5th Advanced Satellite Multimedia Systems Conference and the 11th Signal Processing for Space Communications Workshop*. IEEE. 2010, pp. 9–16 (cit. on p. 25).

[SG13] DVB Second Generation. "Digital Video Broadcasting (DVB); Second Generation DVB Interactive Satellite System." In: (2013) (cit. on pp. 20, 21, 23–25, 68, 69, 98).

[Sim05] Marvin K Simon. *Bandwidth-efficient digital modulation with application to deep space communications*. Vol. 2. John Wiley & Sons, 2005 (cit. on p. 25).

[Tan81] R Tanner. "A recursive approach to low complexity codes." In: *IEEE Transactions on information theory* 27.5 (1981), pp. 533–547 (cit. on p. 102).

[TB01] Stephan Ten Brink. "Convergence behavior of iteratively decoded parallel concatenated codes." In: *IEEE transactions on communications* 49.10 (2001), pp. 1727–1737 (cit. on pp. 66, 76, 103, 114).

[TBKA04] Stephan Ten Brink, Gerhard Kramer, and Alexei Ashikhmin. "Design of low-density parity-check codes for modulation and detection." In: *IEEE transactions on communications* 52.4 (2004), pp. 670–678 (cit. on pp. 78, 97, 119).

[VCT10] Matthew C Valenti, Shi Cheng, and Don Torrieri. "Iterative multisymbol noncoherent reception of coded CPFSK." In: *IEEE transactions on communications* 58.7 (2010), pp. 2046–2054 (cit. on pp. 3, 5, 11, 13, 20, 21, 32, 35, 38, 39, 54, 55, 66, 147, 153).

[Vit67] Andrew Viterbi. "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm." In: *IEEE transactions on Information Theory* 13.2 (1967), pp. 260–269 (cit. on pp. 20, 34, 40).

[Yan04] Shaohua Yang. "The capacity of communication channels with memory." PhD thesis. Citeseer, 2004 (cit. on p. 64).

[YKT05] Shaohua Yang, Aleksandar Kavcic, and Sekhar Tatikonda. "Feedback capacity of finite-state machine channels." In: *IEEE Transactions on Information Theory* 51.3 (2005), pp. 799–810 (cit. on p. 64).