Linear Precoding for Capacity maximization in MU-MIMO BC Channels

Amara Mustapha

DISSERTATION

In Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
from TELECOM ParisTech

Specialization: Communication and Electronics

Mustapha Amara

Linear Precoding for Capacity maximization in MU-MIMO BC Channels

Defense scheduled on the 28th of June 2011 before a committee composed of:

Reviewers		
Professor	Pascal Chevalier	Thales Communications & CNAM
Professor	Constantinos B. Papadias	Athens Information Technology
Examiners		
Professor	Luc Deniere	Laboratoire I3S (Université de Nice)
Professor	David Gesbert	EURECOM
Thesis Supervisors		
Professor	Dirk Slock	EURECOM
Doctor	Julie Yi Yuan	Orange Labs

THÈSE

présentée pour l'obtention du grade de

Docteur de TELECOM ParisTech

Spécialité: Communication et Electronique

Mustapha Amara

Optimisation d'Emetteurs-Récepteurs pour la liaison Descendante de systèmes MIMO Milti-Utilisateurs

Soutenance de thèse prévue le 28 Juin 2011 devant le jury composé de :

Rapporteurs		
Professeur	Pascal Chevalier	Thales Communications & CNAM
Professeur	Constantinos B. Papadias	Athens Information Technology
Examinateurs		
Professeur	Luc Deniere	Laboratoire I3S (Université de Nice)
Professeur	David Gesbert	EURECOM
Directeurs de thèse		
Professeur	Dirk Slock	EURECOM
Docteur	Julie Yi Yuan	Orange Labs

Abstract

Multiuser MIMO (MU-MIMO) downlink system known in the information theory as the broadcast channel system is one of the most importantly investigated subjects in wireless communications. This important interest comes from the high potential it offers in improving reliability and throughput of the system and the problem starts to be considered in recent systems such as LTE, WiFi, ...

In fact Information theory has shown that the capacity of BC could be achieved through dirty-paper coding (DPC). Considering the complexity and resource consumption of DPC, some suboptimal linear algorithms have been proposed. Among them we can find the iterative and the closed form solutions.

The practical precoder design techniques can also be differentiated according to the number of streams allowed per user. In fact, some of them can only support at maximum one stream per user even if the system is not fully charged. Some multi-stream precoding solutions have lately been proposed. Nevertheless, they present some drawbacks as some impose the number of streams per user and thus does not perform a stream selection to optimize the system sum-rate (SR) and some convergence problems towards some local maximum are limiting the throughput.

This thesis outlines our attempt to proposed some optimization algorithm for linear precoding in multiuser MIMO downlink scheme. It is essentially a collection of the following:

- Linear precoding for MU-MIMO BC with single stream support,
- linear precoding for MU-MIMO BC with multiple stream per user selection,
- solutions for global maximum tracking via double iterative and deterministic annealing algorithms,
- derivations of asymptotic analysis for zero forcing DPC stream selection algorithm.

Apart from the introduction and the conclusions, the document contains tree more chapters. The context of each one of these chapters is based on one or two publications, and some unpublished results can be found.

In general, precoder design methods are complicated processes and depends on the chosen criteria. In fact, the cost function that operators try to maximize is the total system throughput. This function is non convex and depends not only on the beam directions but also on the power distribution. Moreover, the optimization is done under some constraints. First of all, there is total power constraint that should be taken into account. The second constraint is the total maximal number of streams that can be served simultaneously $Q_{max} = \min\left(\sum_{k=1}^{K} N_{R_k}, N_T\right)$. Generally the second constraint will be, for a cellular system, on the base station (BS) side. Here we can clearly see that the problem involves many variables that are linked together in some non trivial manner. First of all streams should be correctly selected, then beamformers should be constructed and power allocated to each one.

Chapter 3, we deal with iterative solutions for precoder and decoder design for the multiuser MIMO BC. These algorithms are mainly based on some of the existing closed forms MISO solution presented in the literature. Some solutions for multiple stream per user are also presented. The chapter also presents and discusses solutions for the problem faced by these iterative solutions namely the local maximum convergence problem. The two proposed solutions are double iterative with flipping point and deterministic annealing tracking method.

Chapter 4 speaks about the second family of precoder namely the closed form (CF) precoding techniques. In the first part, we propose some enhancements to some of the existing closed form solutions namely the SJNR and PU-MMSE precoder offering low complexity solutions to elevate the transmit rates by means of wise user selection and power distribution. These algorithms are also adapted for MIMO configuration. In the second part, we present a new zero forcing DPC precoding technique based on successive SVD decompositions. This proposed algorithm constructs stream-wise the precoder in a recursive manner.

The main motivation behind chapter 5, is to investigate the performance achieved by some SDMA (Space Division Multiple Access) access precoding techniques in a more realistic Broadcast environment. Through the study a channel model has been proposed to point out some extreme situation translating the impact of line of sight (LOS) and spacial user distribution on the performances of the different algorithms. The considered techniques are

also compared to the conventional TDMA (Time Division Multiple Access) communication mode.

Acknowledgements

Contents

List of Figures

Acronyms

Here are the main acronyms used in this document. The meaning of an acronym is usually indicated once, when it first occurs in the text. The English acronyms are also used for the French summary.

AWGN	Additive White Gaussian Noise.
BC	Broadcast Channel.
BS	Base Station.
cdf	cumulative density function.
CF	Closed Form.
CSI	Channel State Information.
CSIT	Channel State Information at the Transmitter.
CSIR	Channel State Information at the Receiver.
DL	Downlink.
DPC	Dirty Paper Coding.
eq.	Equation.
FDD	Frequency Division Duplex.
i.i.d.	independent and identically distributed.
io	input-output.
l.h.s.	left hand side.
LOS	Line Of Sight.
LTE	Long Term Evolution.
MF	Matched Filter.
MIMO	Multiple Input Multiple Output.
MISO	Multiple Input Single Output.
ML	Maximum Likelihood.
MMSE	Minimum Mean Square Error.
MSE	Mean Square Error.
MSR	Maximum Sum Rate.
MS	Mobile Station.

MU	Multi User.
MU-MIMO	Multi User Multiple Input Multiple Output.
NLoS	Non Line of Sight.
NS	Null Space.
pdf	probability density function.
PD	Positive Definite.
PSD	Positive Semi-Definite.
PU-MMSE	Per User MMSE.
r.h.s.	right hand side.
Rx	Receiver(s).
SDMA	Space Division Multiple Access.
SIMO	Single Input Multiple Output.
SINR	Signal to Noise and Interference Ratio.
SNR	Signal-to-Noise Ratio.
SISO	Single-Input Single-Output.
SJNR	Signal to Jamming and Noise Ration.
SVH	Stojnic, Vikalo, Hassibi (precoding technique [29]).
SR	Sum Rate.
s.t.	such that.
STC	Space Time Coding.
SUS	Successive User Selection.
SVD	Singular Value Decomposition.
TDD	Time Division Duplex.
TDMA	Time Division Multiple Access.
Tx	Transmitter(s).
WF	Water Filling.
Wi-Fi	802.11.
WMMSE	Weighted Minimum Mean Square Error.
w.r.t.	with respect to.
WSR	Weighted Sum Rate.
ZF	Zero Forcing.
ZFBF	Zero Forcing BeamForming.
ZFDPC	Zero Forcing Dirty Paper Coding.

Chapter 1

Résumé en français

1.1 Introduction

Depuis le début du 21 ème siècle, les systèmes de communication mobile n'ont pas arrêté d'avoir une croissance de plus en plus rapide. Cette croissance a apporté avec une grande amélioration des débits et defficacité spectrale pour pouvoir subvenir aux besoins des applications de plus en plus gourmandes en bande passante. Cette amélioration est en grande en partie attribuée à lémergence des systèmes MIMO, qui en se basant sur les travaux de Shannon, permettent d'augmenter linéairement la capacité avec le nombre dantennes. Cependant, ce résultat suppose une connaissance parfaite du canal de transmission. Par ailleurs, cette asymptôte théorique établie par Shannon resta sans application pratique jusqu'à l'apparition du DPC (de l'anglais Dirty Paper Coding) publié par Costa dans [1]

Cette méthode optimale de précodage, reste cependant utopique. Quelques tentatives d'implémentations ont vu le jour permettant de valider la théorie en se rapprochant des performances promises. Mais, ces implémentations restent trop lourdes, présentent une énorme consommation des ressources de traitement et donc entre autres en termes d'énergie et de puissance de calcul requise. D'autre part, ces différentes techniques de précodage, qui sont des techniques non linéaires, se présentent comme des algorithmes itératifs non linéaires et demandent des temps de convergence irréaliste et surtout non réalisable à l'échelle des temps de latences imposés pour les systèmes de

communication.

Tous ces paramètres ont poussé les chercheurs à trouver des solutions plus simples à réaliser et se sont tournés vers les techniques de précodage linéaires. Ce techniques sont forcément des techniques sous optimales vu qu'il essayent de résoudre un problème d'optimisation non linéaire et non convexe avec une technique linéraire; mais permettent bien d'avoir des performances acceptables et couvrant les demandes de débits sans cesse en augmentation. Ces techniques peuvent être classées en deux grandes familles:

- les précodeurs non itératifs ou à forme approchée (Closed Form en anglais)
- les précodeurs itératifs (Iteratif algorithms en anglais)

Ces deux grandes familles donnent deux approches pour la proposition de solutions linéaires de précodage. La première famille se base essentiellement sur une solution dérivant une forme approchée des précodeurs selon un critère donné. Ce genre de solutions présentent des avantages évidentes en termes de vitesse de traitement vu que la solution est une formule approchée de la solution maximisant le débit en fonction du critère choisi. Par ailleurs, et comme il n'y a pas d'itérations ni de boucles, le système d'optimisation ne peut pas diverger et présentera toujours le même optimum. Par contre, du point de vue des performances obtenues ces techniques vu du fait qu'ils approchent une solution idéalement non linéaire se retrouvent avec des performances asséz éloignés de l'optimum.

La deuxième famille, celle des précodeurs itératifs, se basent comme leur nom l'indique sur des algorithmes d'optimisation itératifs tout en se basant sur des traitements linéaires. Cependant, ces rechniques vu leur évolution dans des espaces non convexes avec plein de maximas locaux n'assurent pas forcément une bonne convergence et risquent même de diverger en employant une technique d'optimisation conventionelle tel que le gradient.

Une autre classification des systèmes de communications et en particulier des précodeurs pour les systèmes mobiles commencent à apparaître et dominer les standards. Cette classification concerne le nombre de flux qui penveut être transmis à un utilisateur donné. En effet, depuis l'arirvée des systèmes MIMO avec des antennes mutiples à la réception, on a commencé à parler de compromis entre le nombre de flux et la diversité. En rélité, cette problématique est propre aux systèmes où les récepteurs aussi bien que les récepteurs présentent des antennes multiples. Ceci crée un certain nombre de degrès de liberté dans le système qui peuvent être exploités de deux manières. La première façon est de se dire que l'on a à disposition N_{R_k}

antennes de réceptions et de ce fait on a potentiellement N_{R_k} trajets décorrélés qui m'apportent de la diversité à la réception. Cette diversité peut être exploité par exemple à l'aide des récepteurs MRC (Maximum Ration Combining) pour trouver la meilleure puissance de réception pour le signal désiré et parvenir à l'extraire sans erreurs. La deuxième vision que l'on peut adopter consiste à dire: on a N_T antennes de transmission et N_{R_k} antennes de réception pour l'utilisateur k. De ce fait, on est capable de construire $\min(N_T, N_{R_k})$ directions indépendantes entre le transmetteur et le récepteur. Ceci permet donc de transmettre $\min(N_T, N_{R_k})$ flux indépendants et de ce fait augmenter le débit offert.

D'après ce qu'on vient de voir, ces solutions existantes, ne considèrent que les deux extrêmes et les quelques travaux qui existaient ne considéraient les compromis que pour un système avec un émetteur et un récepteur. Maintenant, si on va se mettre dans un système réel de communication mobile et en particulier les systèmes cellulaires, on va avoir des stations mobiles multiples qui vont communiquer avec une station de base BS comme illustré à la figure Figure 1.1. Ce système essayes de servir le maximum d'utilisateurs possible selon un certain critère préfixé. Ces critères peuvent être l'équité, assurer un débit minimum à tous les utilisateurs, s'adapter en fonction des applications et leur priorités, ou encore maximiser le débit total véhiculé par la cellule. Vu que cette thèse a été réalisée en collaboration avec un opérateur de télécommunication mobile, le critère adopté ici sera le dernier cité maximiser la somme des débits de la station de base noté SR et donné par

$$SR = \sum_{k=1}^{K} log_2 \det\left(\mathbf{I} + \mathbf{H}_k \mathbf{T}_k \mathbf{T}_k^H \mathbf{H}_k^H \left(\boldsymbol{\Upsilon}_k + N_0 \mathbf{I}\right)^{-1}\right) \tag{1.1}$$

où $\boldsymbol{\Upsilon}_k = \sum_{j=1, j\neq k}^{K} \mathbf{H}_k \mathbf{T}_j \mathbf{T}_j^H \mathbf{H}_k^H$ représente interférence subie par l'utilisateur k.

1.2 Optimisation dans un espace très hostile

Comme décrit plus tôt on se place dans une configuration avec une station de base avec N_T antennes d'émission et K utilisateurs où le $k$ième possède N_{R_k} antennes de réception. Pour simplifier la description de la problématique des maximas locaux, on suppose, sans atteindre à la généralité des observations, que tous les utilisateurs possédent le même nombre d'antennes de réceptions. Soit $N_R = N_{R_k} \forall k \in \{1, \dots K\}$.

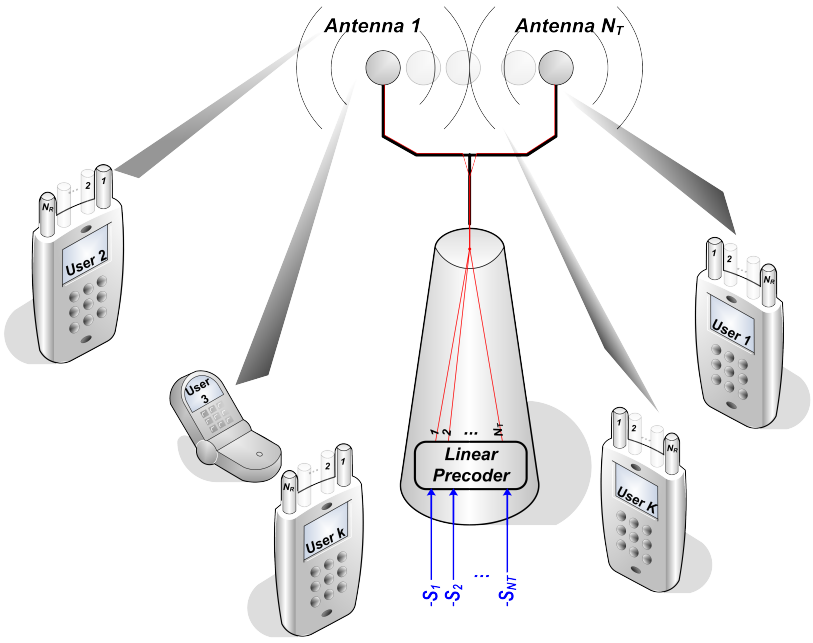


Figure 1.1: MU-MIMO system model.

Afin de toucher à l'origine de la difficulité rencontrée dans l'optimisation du SR dans les systèmes MU-MIMO, on présente ici quelques cas de figure. En fait, il est connu que le problème $\max_{\mathbf{T}_k} SR \quad s.c. \sum P_k \leqslant P_T$ est un problème d'optimisation non convexe vu la corrélation existante entre la direction des vecteurs de précodage et la distribution de puissance. Mais en réalité, la complexité est bien plus élevée. Cette fonction coût qu'on cherche à maximiser présente en plus de sa non convexité, un grand nombre de maximas locaux. Ces maximas locaux déjà observés pour les systèmes MISO, proviennent essentiellement de la selection des directions et de leur couplage avec les puissances. Cette problématique a été en partie résolue à l'aide de la subdivision des problèmes de puissance et de direction. La litérature propose un certain nombre de solution qui soit

optimisent chacune des deux quantités séparément et en déduit une solution globale, soit utilisent le couplage pour optimiser conjointement en alternant entre les deux problèmes.

Avec l'avènement des systèmes MIMO ayant des antennes multiples à la réception, le problème c'est bien plus compliqué. Et une seconde source de maximas bien plus importante est apparue. En fait, le problème devien maintenant comment choisir les directions de transmission pour être bien adaptés à cettes choisies au niveau de la réception en plus du problème de la distribution de puissance. De plus, des flux multiples peuvent être alloués à un même utilisateur ce qui requiert un traitement différent. Les figures

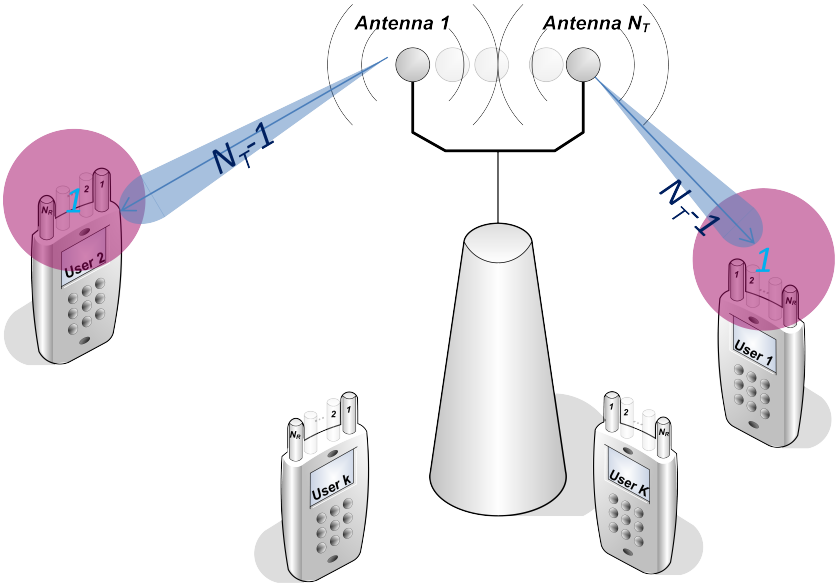


Figure 1.2: Multiuser MIMO optimization startegy 1.

Figure 1.2 et Figure 1.3 montrent deux solutions différentes pour partager les rôles entre le transmetteur (Station de base) et les récepteurs. Dans le premier cas de figure, 1.2, toute la fonction d'annulation d'interférence est assignée à la station de base qui est capable d'annuler $N_T - 1$ flux interférants. Les récepteurs quant à eux, ils n'ont rien à faire et reçoivent les signaux correctement. Cependant, dans le second cas de figure illustré par 1.3, le transmetteur émet dans un sous espace de dimension $N_T - N_R$ ne fait rien pour annuler toute l'interférences entre les flux. De ce fait, tout le travail doit se faire en amon au niveau des récepteurs. Ces derniers sont capables d'annuler $N_R - 1$ flux interférants.

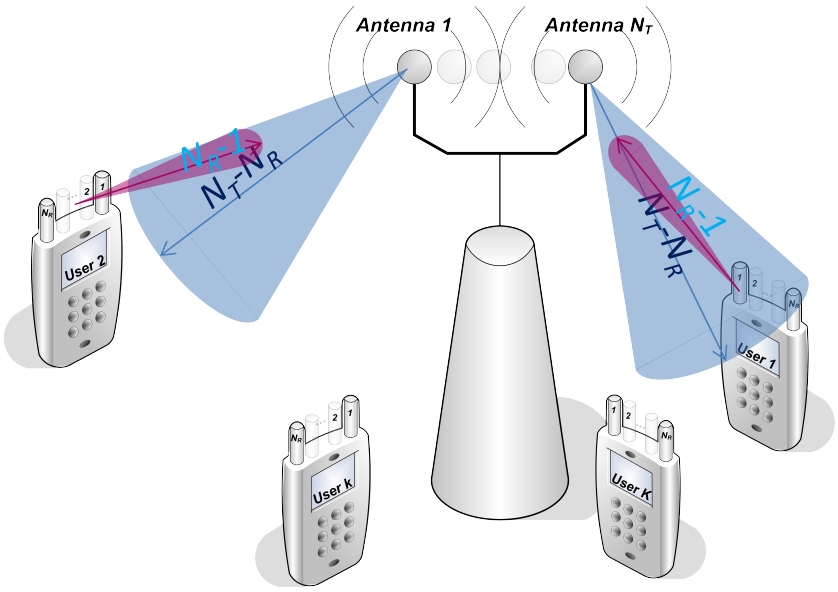
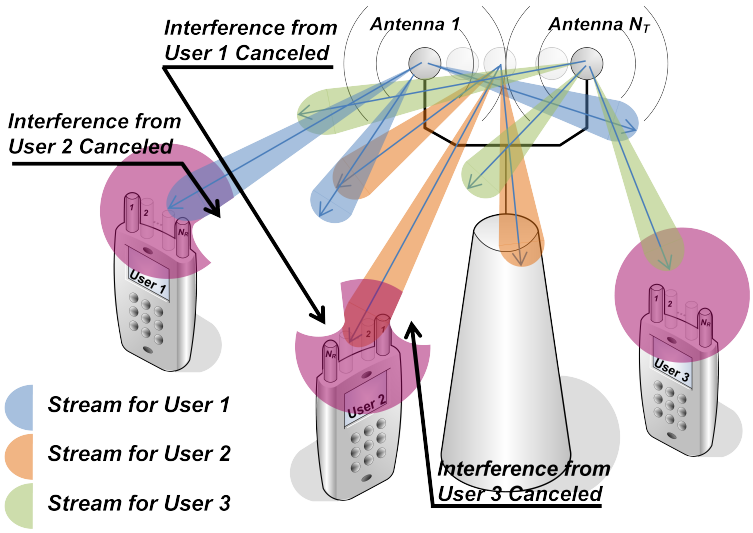


Figure 1.3: Multiuser MIMO optimization startegy 2.

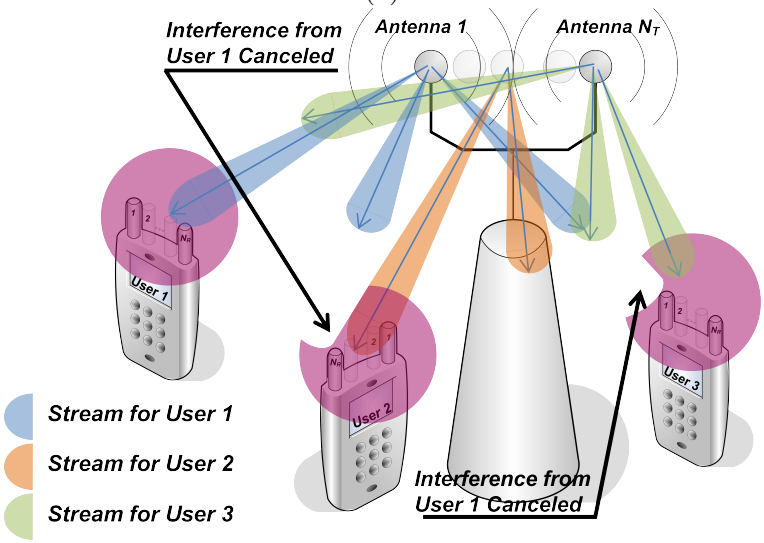
Un cas intermédiaire entre ces deux extrêmes, serait de partager les rôles entre transmetteur et récepteur. Dans ce cas supposons par exemple que le récepteur va essayer d'annuler l flux interférant $l \in \{1, \dots N_R\}$. Donc le transmetteur est obligé de considérer toute l'interférence restante et va de ce fait émettre dans un sous espace de dimension $N_T - l - 1$. En considérant déjà ce paramètre l traduisant le choix de l'affectation des rôles entre transmetteur et récepteurs, on voit bien que le nombre d'optimums explose rapidement et il peut être donné par:

$$\prod_{l=1}^{\mathrm{N}_T} \left(\sum_{i=0}^{N_R-1} \frac{(\mathrm{N}_T - 1)!}{k!(\mathrm{N}_T - 1 - l)!} \right) \tag{1.2}$$

Ici on vien d'éxposer une cause potentielle générant les maximas locaux multiples dans la fonction coût à optimiser. Mais les problèmes des maximums ne s'arrête pasà ce stade. En effet, même si on suppose avoir fixé les rôles des récepteurs et du transmetteur en termes du nombre de flux véhiculés et d'interférents bloqués, un second problème apparaît. Il s'agit de la manière avec laquelle ces rôles sont distrubués. La figure Figure 1.4 illustre ce phénomène. En effet, on voit ici que suivant les flux qu'on choisit à éliminer à la transmission, on obtien un ensemble différent de flux à éliminer à la réception ce qui conduit naturellement à une autre distribution des



(a)



(b)

Figure 1.4: Stream role affectations: (a) First case scenario; (b)Second case scenario.

précodeurs et des récepteurs d'où un nouveau maximum local.

A partir de ces observation et ayant aquis plus d'information sur les raisons qui bloquent les algorithmes de design des précodeurs, on a construit quelques solutions permettant de contourner cetains de ces obstacles ou du-

moins en minimiser l'impact pour améliorer les performances obtenues.

La section suivante présente quelsques solutions phares pour les algorithmes itératifs et surtout des solutions supportant l'optimisation avec flux multiples par utilisateur. La section 1.4 traite des solutions proposées avec des solutions non itératives avec support de flux multiples par utilisateur.

1.3 Algorithmes itératifs avec prise en compte de flux multiples

Dans cette section on s'intéresse à l'optimisation de précodeurs et de décodeurs pour un système MIMO multiutilisateur. Dans une première partie, on va se focaliser sur les systèmes simples avec un flux par utilisateur. On démontre l'importance d'une optimisation alternée et on propose un algorithme itératif basé sur des solutions préexistantes dans la litérature dans le cas de précodeurs linéaires en CF. Suite à ça, et comme indiqué dans la section précédente, on propose une solution permettant de minimiser l'impact des maximas locaux. La seconde partie, quant à elle, s'attaque aux systèmes MU-MIMO à flux multiples et une proposition de solution permettant de traiter le problème des maximas locaux est présentée.

1.3.1 Systèmes à flux unique

On se place donc dans un système constitué d'une station de base avec antennes de transmission et K mobiles avec N_{R_k} antennes de réception pour l'utilisateur k. Le canal de transmission est pour l'utilisateur modélisé par une matrice $\mathbf{H}_k$ de taille $N_{R_k} \times N_T$ représentant les liens entre les antennes à l'émission et à la réception. Chaque composante de la matrice est une variable aléatoire de moyenne nulle et de variance N_0 considérée égale à 1.

On se retrouve donc avec un canal de Rayleigh par utilisateur et on suppose qu'il y a indépendance totale entre les différents canaux des différents utilisateurs. On se propose d'étudier le système dans le cas de transmissions de flux multiples d'une manière simultanée de la station de base aux différents utilisateurs. Cette étude du lien descendant est connue en théorie de l'information par le canal de diffusion. Le nombre de flux est fixé à un flux par utilisateur même si le système peut supporter plus qu'un flux vu que l'on dispose d'antennes multiples à l'émission et à la réception. On suppose de plus que l'on a une connaissance parfaite du canal de transmission au niveau de l'émetteur ceci implique:

- L'invariance du canal de transmission entre l'estimation et la transmission.
- Une estimation parfaite de tous les composants des matrices de tous les utilisateurs puisqu'on est dans un canal de diffusion .

Le but que l'on recherche par cette étude est de maximiser le débit véhiculé par ce système MIMO multiutilisateur ainsi définit.

Etat de l'art et précodage linéaire

Le débit total maximal pouvant être véhiculé par le système vers tous les utilisateurs est une quantité très importante permettant le dimensionnement des systèmes de communication. Cette quantité a été définie par la théorie de l'information comme la capacité du système et est définie par

$$C^{erg} = \max_{p(X_1),...,p(X_K)} I(X_1, ..., X_K; Y) \tag{1.3}$$

Où I correspond à l'information mutuelle et $p(X_k), k \in \{1, \ldots, K\}$ sont les distributions des bits à l'entrée et destinés à l'utilisateur k.

La maximisation de cette quantité a été définie pour un système MIMO mono-utilisateur et cette quantité maximale a même été atteinte pratiquement par l'utilisation de l'algorithme de waterfilling défini en [2]. L'expression mathématique pour cette quantité est donnée par

$$C^{erg}_{\text{MIMO}} = \min(N_R, N_T) E_{\lambda_i} \log_2 \left(1 + \frac{P_T}{N_T} \lambda_i\right) \tag{1.4}$$

Dans cette expression E_x représente l'espérance mathématique sur x, P_T est la puissance totale à l'émission, et λ_i sont les valeurs propres de la matrice du canal. Si le cas mono-utilisateur a été résolu il n'en est pas de même pour le cas multiutilisateurs qui est le cas de notre étude et le cas le plus réaliste et le plus intéressant pour les systèmes de communications. En effet, la maximisation de l'information mutuelle pour un tel système aboutit à la maximisation d'une expression non linéaire sur les distributions des données à envoyer aux différents utilisateurs.

Cependant plusieurs propositions d'algorithmes permettant de tirer avantage des systèmes MIMO MU ont été proposées dans la littérature. Parmi ces solutions proposées, on retrouve les solutions linéaires et les solutions itératives. Les solutions linéaires sont des solutions basées sur des pré-codages et des récepteurs linéaires permettant de séparer les flux des différents utilisateurs afin de minimiser les interférences inter-utilisateurs et d'améliorer par

suite le débit véhiculé. Dans ce type de solution on en cite deux la première est une solution à base d'émetteurs MMSE et le second à base d'un émetteur basé sur le critère SJNR.

La première méthode [1] définit un émetteur MMSE pour chaque utilisateur en tenant en compte de l'interférence générée par tous les autres utilisateurs. Pour chaque utilisateur, la forme du précodeur est donnée par

$$\mathbf{t}_k = \sqrt{\frac{P_T}{tr\left(\sum_{j=1}^{K} \mathbf{t}_j \mathbf{t}_j^H\right)}} \left(\tilde{\mathbf{H}}_k^H \tilde{\mathbf{H}}_k + \frac{M_R}{P_T}\mathbf{I}\right)^{-1} \hat{\mathbf{H}}_k^H \tag{1.5}$$

où $\tilde{\mathbf{H}}_k^T = [\mathbf{H}_1 \cdots \mathbf{H}_{k-1}\mathbf{H}_{k+1} \cdots \mathbf{H}_K]$ et $\hat{\mathbf{H}}_k = \begin{bmatrix} \mathbf{H}_k \\ \tilde{\mathbf{H}}_k \end{bmatrix}$ Le récepteur quand lui est donné par le vecteur propre gauche le plus grand résultant de la décomposition en SVD du canal virtuel composé de la cascade .

Dans le second algorithme, la solution propose un pré-codeur défini comme la valeur propre généralisée de l'expression du SJNR ce pré-codeur pour l'utilisateur est donné en [27] par

$$\mathbf{t}_k = \xi_m \left[\sqrt{P_k}\left(\sum_{i=1, i\neq k}^{K} \mathbf{H}_i^H \mathbf{H}_i + \frac{1}{P_k}\mathbf{I}\right)^{-1} \mathbf{H}_k^H \mathbf{H}_k\right] \tag{1.6}$$

Où P_k est la puissance transmise à l'utilisateur k et $\xi_m(\mathbf{X})$ est la fonction qui renvoie la plus grande valeur propre de $\mathbf{X}$. Pour le récepteur ils ont proposé un simple récepteur MF .

Construction du décodeur optimal

Dans toutes ces propositions de solutions, les auteurs se sont basés sur des critères comme le MMSE et le SJNR. Or le but de tout système de communication est de maximiser le débit total transféré à travers le canal considéré. Pour cela nous considérons l'expression générale de l'expression du débit total d'un système MIMO multiutilisateurs. Elle est donnée par

$$SR_{MU-MIMO} = \sum_{k=1}^{K} \log_2 \left(\det\left(\mathbf{I}_{Q_k} + \mathbf{D}_k \mathbf{H}_k \mathbf{T}_k \mathbf{T}_k^H \mathbf{H}_k^H \mathbf{D}_k^H \left(\sum_{i=1, i\neq k}^{K} \mathbf{D}_k \mathbf{H}_k \mathbf{T}_i \mathbf{T}_i^H \mathbf{H}_k^H \mathbf{D}_k^H + N_0^2 \mathbf{I}\right)^{-1}\right)\right) \tag{1.7}$$

Où Q_k représente le nombre de flux attribués à l'utilisateur k, N_0 est la variance celle du bruit à la réception pour l'utilisateur k.

En considérant que les flux sont séparables ou dans le cas de l'allocation d'un seul flux par utilisateur, on peut réécrire cette quantité comme une somme des débits réalisables par flux. Le débit par flux est donné par

$$R_k = \log_2 \left(1 + \frac{\mathbf{d}_k \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H \mathbf{d}_k^H}{\sum\limits_{i=1, i \neq k}^{K} \mathbf{d}_k \mathbf{H}_k \mathbf{t}_i \mathbf{t}_i^H \mathbf{H}_k^H \mathbf{d}_k^H + N_0} \right) \tag{1.8}$$

La maximisation de cette quantité en fonction du récepteur revient à maximiser la quantité

$$\mathbf{d}_k \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H \mathbf{d}_k^H \left(\sum_{i=1, i \neq k}^{K} \mathbf{d}_k \mathbf{H}_k \mathbf{t}_i \mathbf{t}_i^H \mathbf{H}_k^H \mathbf{d}_k^H + N_0 \right)^{-1} \tag{1.9}$$

dont la solution n'est autre que (comme pour le cas SJNR) la plus grande valeur propre de la quantité

$$\psi = \left(\sum_{i=1, i \neq k}^{K} \mathbf{H}_k \mathbf{t}_i \mathbf{t}_i^H \mathbf{H}_k^H + N_0 \right)^{-1} \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H \tag{1.10}$$

Ainsi le récepteur optimal noté d_{MSR} qui maximise la somme des débits dans un tel système MIMO multiutilisateurs est donné par

$$\mathbf{d}_{MSR} = \xi_m \left(\left(\sum_{i=1, i \neq k}^{K} \mathbf{H}_k \mathbf{t}_i \mathbf{t}_i^H \mathbf{H}_k^H + N_0 \right)^{-1} \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H \right) \tag{1.11}$$

Un autre type de précodage linéaire présent dans la litérature est le précodage MMSE qui prend en considération l'effet du bruit et peut être donné par

$$\mathbf{t}_k = \sqrt{\frac{P_T}{tr\left(\sum\limits_{j=1}^{K} \mathbf{t}_j \mathbf{t}_j^H \right)}} \left(\sum_{i=1}^{K} \mathbf{H}_i^H \mathbf{d}_i^H \mathbf{d}_i \mathbf{H}_i + \frac{1}{P_T} \sum_{j=1}^{K} tr\left(\mathbf{d}_j^H \mathbf{d}_j \right) \mathbf{I} \right)^{-1} \mathbf{H}_k^H \mathbf{d}_k^H \tag{1.12}$$

Procédé itératif à flux unique

La solution que l'on prose alors pour optimiser la somme du débit total à travers le système ainsi considéré, va se baser sur ce récepteur au quel on rajoute une optimisation itérative. L'algorithme ainsi obtenu se présente comme suit.

Algorithme: Algorithme itératif à flux unique SJNR/MSR

---

1. On commence par faire une première itération en utilisant un pré-codeur linéaire $\mathbf{t}_k^0$ tel que le MMSE ou le SJNR et on détermine le récepteur optimal avec ce pré-codeur en utilisant (1.11).

2. On remplace le canal de transmission $\mathbf{H}_k$ par un canal virtuel équivalent donné par la cascade du canal et du récepteur noté $\mathbf{H}_k^{iter} = \mathbf{d}_k^{iter-1}\mathbf{H}_k$.

3. On recalcule le nouveau pré-codeur avec le nouveau canal ainsi obtenu en utilisant soit l'équation (1.12) pour le MMSE ou l'équation (1.6) pour un pré-codeur SJNR.

4. On recalcule le nouveau récepteur optimal $\mathbf{d}_k^{iter}$ en fonction de $\mathbf{t}_i^{iter}$ avec (1.11).

5. On calcule la somme des débits SR^{iter} obtenu en injectant $\mathbf{t}_k^{iter}$ et $\mathbf{d}_k^{iter}$ dans l'équation (1.7).

6. On répète les étapes 2/ à 5/ jusqu'à la convergence. La convergence est déterminée par la stabilisation de la valeur du débit total $\left|SR^{iter} - SR^{iter+1}\right| < \varepsilon$

Une représentation graphique de cet algorithme est donnée à la figure Figure 1.5 Cette méthode de transformation du canal virtuel peut être utilisée avec tous les algorithmes déjà connus dans le cas des systèmes MISO tel que l'algorithme SVH décrit sous méthode 2.1 dans [29]. En effet, la construction de canal virtuel permet de trabsformer le canal MU-MIMO en un canal MU-MISO et d'appliquer localement pour cette transformation les algorithmes d'optimisation connus pour les systèmes MISO où la complexité est moindre et où le découplage entre puissance et directions permettent de trouver de meilleures solutions.

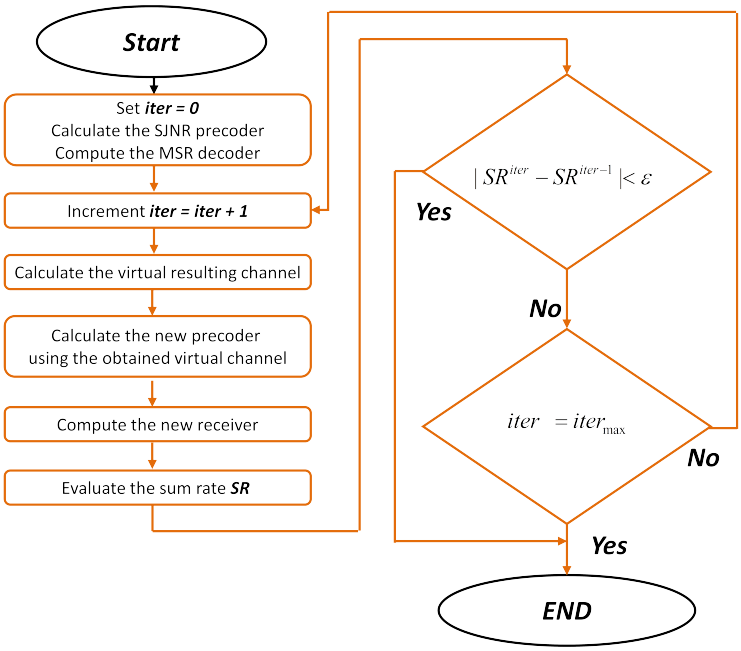


Figure 1.5: Algorithme itératif SJNR/MSR à flux unique.

La relation avec le vrai canal est par la suite rétablie à travers une boucle d'itération externe permettant de reprendre en considération la présence d'antennes multiples à la réception au niveau des terminaux mobiles. Cette méthode est bien évidemment sous optimale mais elle permet de mieux sonder l'espace des précodeurs possibles vu qu'on a une succession d'optimisation locale et une plus anarchique qui consiste à changer de canaux virtuels. Il faut aussi noter ici que la complexité de l'algorithme n'est pas plus important que d'autres algorithmes itératifs malgrè la présence de deux boucles itératives imbriquées. Ceci est en grande partie dû au fait que le canal virtuel choisi dans la boucle principale, n'est pas si loin d'une bonne solution et de ce fait constitue une bonne initialisation pour l'algorithme interne dont le rôle se réduit à un rafinage de la solution. Une représentation graphique de cette solution est données à la figure Figure 1.6

Procédé d'optimisation double itératif

Cette amélioration consiste à introduire un nouvel algorithme double itératif qui sans augmenter le nombre d'itérations nécessaires pour la convergence nous permet de diminuer l'impact des maximas locaux sur le processus de

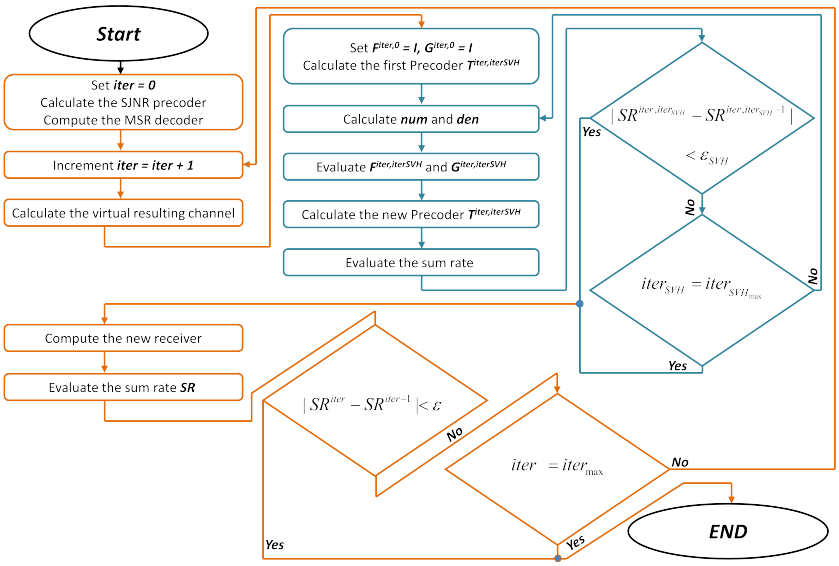


Figure 1.6: Algorithm itératif SVH avec flux unique et construction de flux virtuels.

convergence et par conséquent, nous permet d'améliorer la valeur du SR su système.

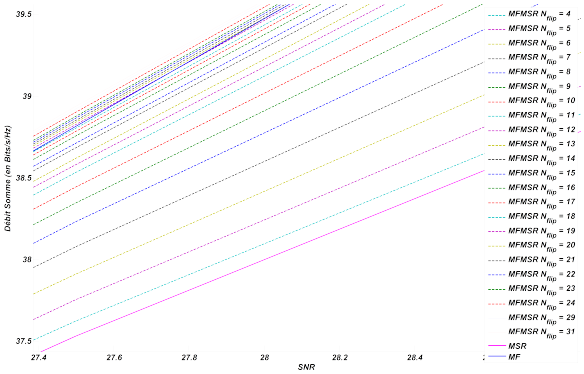
Cet algorithme se base sur le fait que le processus itératif décrit dans la section précédente qui, appliqué avec un précodeur SJNR et un récepteur MF, a la propriété de tenter de tirer le maximum de puissance pour le système MIMO-MU (Il s'agit là du rôle assuré par le récepteur MF qui vise à maximiser la puissance de réception en fonction du canal de transmission) mais présente, par ailleurs, une variation non linéaire pour le SR obtenu au cours des itérations. Un récepteur MF est donné par

$$\mathbf{d}_k = \frac{(\mathbf{H}_k \mathbf{t}_k)^H}{\left\| (\mathbf{H}_k \mathbf{t}_k) \right\|} \tag{1.13}$$

Par contre, le même algorithme appliqué avec un récepteur MSR, va avoir une évolution strictement croissante en fonction des itérations à partir du point d'initialisation mais peut converger vers un maxima local. Le but est donc de commencer par SJNR/MF pour bien se positionner et à partir de là de lancer un SJNR/MSR à partir du point de convergence de l'algorithme précédent.

Comme décrit précédemment, on va devoir à un certain moment changer d'algorithme dans notre processus itératif, seulement ce point de bascule-

ment ou de flip entre les algorithmes est crucial pour la convergence puisque changer du MF vers le MSR trop tôt peut nous mener à une mauvaise initialisation qui ne peut être corrigée par le MSR, et changer de récepteur trop tard ne nous permet pas de récupérer les avantages de MSR surtout dans la région à bas SNR. Or, la valeur optimale de la valeur de basculement est variable en fonction des réalisations du canal, du SNR du système et de la configuration utilisés (Nombre d'antennes à l'émission et à la réception).



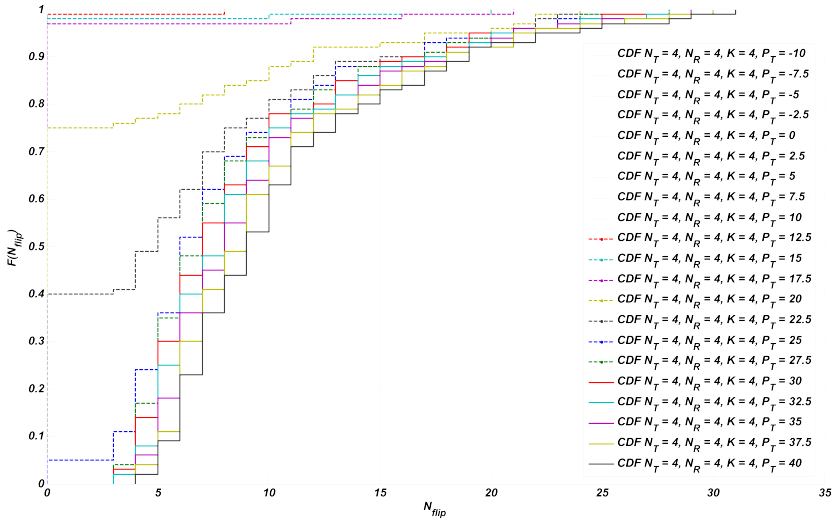
ITER 10000 444 20091222.png

Figure 1.7: Débit total en fonction de SNR et de N_{flip} pour N_T=N_R=K=4.

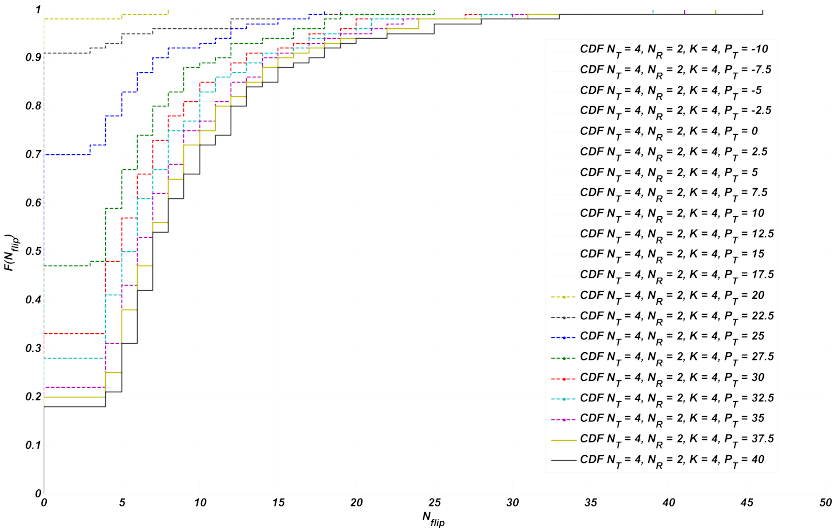
En effet, comme le montrent les résultats expérimentaux de la figure Figure 1.9 qui est le tracé du débit total du système en fonction du point de convergence pour un système avec $N_T = 4, N_R = 4, K = 4$ pour un flux par utilisateur et Figure 1.8 qui représente le tracées de la CDF de N_{flip} pour les cas où l'algorithme itératif utilisant MF dépasse celui utilisant MSR pour la même configuration du système.

Ceci dit, une idée serait d'appliquer une méthode simple de détermination du point optimal de basculement. Ceci a été fait dans la figure Figure 1.9 où l'on représente le point optimal pour deux critères le premier étant maximiser le débit total moyen alors que le second maximise le débit moyen instantanné. On voit clairement à travers ces courbes qu'il est difficile de trouver un point où les deux méthodes pourraient fonctionner correctement. Par ailleurs ces différences apparaissent aussi en fonction du nombre d'antennes aussi bien à l'émission qu'à la réception comme on peut le constater à travers les courbes de la figure Figure 1.8. Tout ces paramètres qui influencent la simplicité du choix du point optimal de basculement rendent impossible la construction de solution à base de tables de sélections ou autres.

Vu que la valeur de N_{flip} est très variable d'une configuration à une autre



(a)



(b)

Figure 1.8: CDF en fonction du point de basculement: (a) $N_T = N_R = K = 4$; (b) $N_T = 4, N_R = K = 2$.

et d'un état du système à un autre. Afin de proposer une solution générale qui puisses être appliquée tout le temps et à tout type de système, on a mis en place un algorithme dynamique de prise de décision pour déterminer la

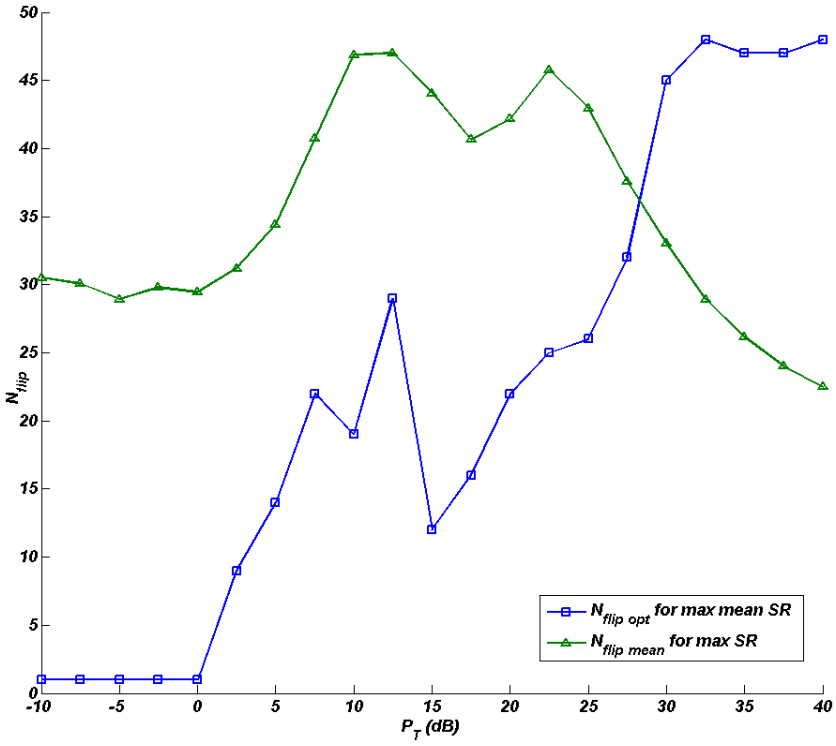


Figure 1.9: N_{flip} optimal en function of the SNR pour deux critères pour N_T=N_R=K=4.

valeur de N_{flip}.

Le procédé de prise de décision consiste à observer sur une fenêtre $MF_{fenetre}$ préfixée de nombre d'itérations l'évolution de l'algorithme en cours d'exécution (qui est l'algorithme SJNR/MSR avec un précodeur SJNR mais un récepteur MF). Une fois que cet algorithme se soit stabilisé, et donc qu'il ait convergé, on part de ce point et on exécute le second algorithme (celui de SJNR/MSR avec un précodeur SJNR et un récepteur MSR décrit précedemment). La fin de l'algorithme est définie soit par un nombre maximal d'itérations $N_{\max}^{iter}$ ou la stabilisation du SR du système. Donc l'algorithme itératif se présente comme suit:

Algorithme: Algorithme double itératif à flux unique par utilisateur

1. On initialise $N_{\max}^{iter}$, $MF_{fenetre}$, ε et ε_{MF}
2. On commence par faire une première itération, correspondant à l'initialisation de l'algorithme. Ce ci est fait en utilisant un précodeur linéaire en émission appelé $\mathbf{t}_k^0$ dans notre cas le SJNR. Et en utilisant l'équation (1.6), on détermine le récepteur $\mathbf{d}_k^0$ correspondant à ce précodeur et tenant compte de tous les autres précodeurs $\mathbf{t}_i^0, i = 1, \cdots, k-1, k+1, \cdots, K$ à l'aide de (1.13) représentant un récepteur MF.
3. On remplace le canal de transmission par un canal virtuel équivalent donné par la cascade du canal et du récepteur noté $\mathbf{H}_k^{iter} = \mathbf{d}_k^{iter-1}\mathbf{H}_k$.
4. On recalcule le nouveau le précodeur $\mathbf{t}_k^{iter}$ avec le nouveau canal ainsi obtenu en utilisant l'équation (1.6) pour un précodeur SJNR.
5. On recalcule le nouveau récepteur optimal $\mathbf{d}_k^{iter}$ en fonction de $\mathbf{t}_k^{iter}$ avec l'équation (1.13) vu que le récepteur est un filtre MF
6. On calcule la somme des débits SR^{iter} obtenu en injectant $\mathbf{t}_k^{iter}$ et $\mathbf{d}_k^{iter}$ dans l'équation (1.7).
7. On répète les étapes 3/ à 6/ $\mathrm{MF}_{\mathrm{fenetre}}$ fois en incrémentant $iter = iter + 1$.
8. Vérifier la convergence de l'algorithme MF en vérifiant la contrainte $Var\left(\left[SR^{iter-\mathrm{MF}_{\mathrm{fenetre}}} \cdots SR^{iter}\right]\right) \leqslant \varepsilon_{MF}$. Si la condition est vérifiée, on passe à 9/ en conserve le vecteur d_k^{iter} obtenu. Sinon on incrémentant $iter = iter + 1$ et on refait les étapes 3/, 4/, 5/, 6/et 8/ jusqu'à atteindre $iter = N_{\max}^{iter} - 1$ et on passe à 9/ en conserve le vecteur d_k^{iter} ainsi obtenu.

9. On incrémente le compteur $iter = iter + 1$ et on remplace le canal de transmission $\mathbf{H}_k$ par un canal virtuel équivalent donné par la cascade du canal et du récepteur noté $\mathbf{H}_k^{iter} = \mathbf{d}_k^{iter-1}\mathbf{H}_k$.

10. On recalcule le nouveau le précodeur $\mathbf{t}_k^{iter}$ avec le nouveau canal ainsi obtenu en utilisant l'équation (1.6) pour un précodeur SJNR.

11. On recalcule le nouveau récepteur optimal $\mathbf{d}_k^{iter}$ en fonction de $\mathbf{t}_k^{iter}$ avec (1.11) vu que le récepteur est un filtre MSR.

12. On calcule la somme des débits SR^{iter} obtenu en injectant $\mathbf{t}_k^{iter}$ et $\mathbf{d}_k^{iter}$ dans (1.7).

13. on répète les étapes 10/ à 12/ jusqu'à convergence de l'algorithme itératif. La convergence est déterminée par la stabilisation de la valeur du débit total $\left|SR^{iter} - SR^{iter+1}\right| < \varepsilon$; ou par le dépassement d'un nombre de itérations maximal $N_{\max}^{iter}$.

1.3.2 Double itératif en flux multiples

Dans cette partie, on s'intéresse à étendre le résltat précédent à un cas de système en supportant la possibilité de flux multiples pour un utilisateur donné. Cette configuration discutée dans l'introduction, génère d'avantage de maximas locaux et rend le processus d'optimisation encore plus difficile.

La litérature contient un algorithme similaire à celui qu'on a construit en simple flux appelé WMMSE et expliqué en [42]. Cet algorithme décrit un précodeur itératif à base d'un WMMSE (Weighted Minimum Mean Square Error). En considérant une matrice $\mathbf{W}$ de poids pour les différents flux de tous les utilisateurs (autrement dit $\mathbf{W} = \{\mathbf{W}_1, \ldots, \mathbf{W}_K\}$)et l'ensemble des récepteurs $\mathbf{D} = diag\{\mathbf{D}_1, \cdots, \mathbf{D}_K\}$ utilisés par les différents utilisateurs alors le précodeur WMMSE correspondant est donné par :

$$\mathbf{T} = diag\{\mathbf{T}_1, \cdots, \mathbf{T}_K\} = \beta\left(\mathbf{H}^H\mathbf{D}^H\mathbf{W}\mathbf{D}\mathbf{H} + \frac{tr(\mathbf{W}\mathbf{D}\mathbf{D}^H)}{P_T}\mathbf{I}_{N_T}\right)^{-1}\mathbf{H}^H\mathbf{D}^H\mathbf{W} \tag{1.14}$$

soit pour l'utilisateur k

$$\mathbf{T}_k = \beta\left(\sum_{j=1}^{K}\mathbf{H}_j^H\mathbf{D}_j^H\mathbf{W}_j\mathbf{D}_j\mathbf{H}_j + \frac{1}{P_T}tr(\mathbf{W}_j\mathbf{D}_j\mathbf{D}_j^H)\mathbf{I}_{N_T}\right)^{-1}\mathbf{H}_k^H\mathbf{D}_k^H\mathbf{W}_k \tag{1.15}$$

où $\beta = \sqrt{\frac{P_T}{\sum_{i=1}^{K} tr(\mathbf{T}_i \mathbf{T}_i^H)}}$ est un facteur scalaire pour respecter la contrainte de puissance.

Ce précodeur dépend des poids $\mathbf{W}$ affectés aux différents flux ainsi que des décodeurs $\mathbf{D}_k, k \in \{1, \cdots, K\}$ utilisés par les utilisateurs ce qui implique l'obligation d'utiliser un procédé itératif pour les optimiser. L'optimisation nécessite donc de définir les récepteurs. Parmi les différentes types de récepteurs qui existent, les auteurs proposent le récepteur MMSE. L'expression analytique du récepteur MMSE est

$$\mathbf{D}_k = \mathbf{T}_k^H \mathbf{H}_k^H \left(\mathbf{H}_k \mathbf{T}_k \mathbf{T}_k^H \mathbf{H}_k^H + \mathbf{I}_{N_{R_k}} + \sum_{i=1, i \neq k}^{K} \mathbf{H}_k \mathbf{T}_i \mathbf{T}_i^H \mathbf{H}_k^H \right)^{-1} \tag{1.16}$$

En suite, il faut définir les poids à affecter à chacun des flux. Cette distribution doit être inversement proportionnelle à l'erreur quadratique moyenne MSE (Mean Square Error). Elle est déterminée par l'expression

$$\mathbf{W}_k = \left(\mathbf{I}_k + \mathbf{T}_k^H \mathbf{H}_k^H \left(\mathbf{I}_{N_{R_k}} + \sum_{i=1, i \neq k}^{K} \mathbf{H}_k \mathbf{T}_i \mathbf{T}_i^H \mathbf{H}_k^H \right)^{-1} \mathbf{H}_k \mathbf{T}_k \right) \tag{1.17}$$

comme étant l'inverse de la matrice de l'erreur quadratique moyenne calculée pour chaque utilisateur.

Le procédé décrit dans est le suivant :

Algorithme: Algorithme WMMSE

1. Initialiser les précodeurs
2. Calculer selon la relation (1.16) les décodeurs MMSE correspondant aux précodeurs pour chaque utilisateur.
3. Calculer la nouvelle distribution des poids minimisant le MSE donnée par la formule (1.17)
4. Recalculer les nouveaux précodeurs en appliquant (1.14)
5. Reprendre les étapes 2. à 4. jusqu'à la convergence.

Pour analyser les performances du système ainsi obtenu, on se base sur la valeur du débit total du système qui est la somme des débits SR offerts aux différents utilisateurs :

$$SR = \sum_{k=1}^{K} \log_2 \left(\mathbf{I} + \frac{\mathbf{H}_k \mathbf{T}_k \mathbf{T}_k^H \mathbf{H}_k^H}{\mathbf{\Gamma}_k + N_0 \mathbf{I}} \right) \tag{1.18}$$

Où le terme $\mathbf{\Gamma}_k = \mathbf{H}_k \sum_{j=1, j\neq k}^{K} \mathbf{T}_j \mathbf{T}_j^H \mathbf{H}_k^H$ représente l'interférence générée par les autres utilisateurs et captés par l'utilisateur k.

Ce procédé présente une saturation à haut rapport signal à bruit SNR avec une variation de la pente de la courbe comme illustrée par la figure Figure 1.10. Ce problème de saturation se rajoute à un problème de

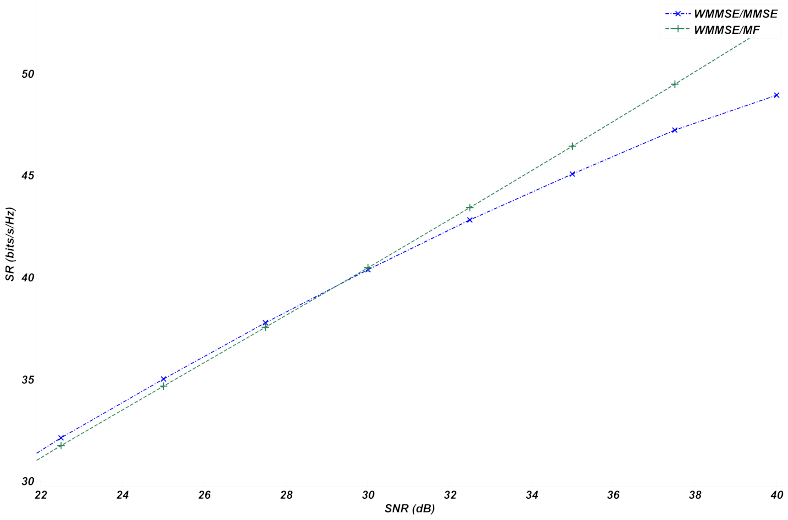


Figure 1.10: Débit en fonction du SNR pour le WMMSE/MF et le WMMSE/MSR N_T=N_R=K=4.

convergence connu pour les solutions itératives simples. La relation (1.14) pour le calcul du précodeur est une fonction directe des décodeurs utilisés par les différents utilisateurs. Le choix des décodeurs a de ce fait un impact direct sur les performances du système obtenu, mais il n'existe pas de décodeur qui permette d'obtenir des gains pour tout rapport signal à bruit SNR.

Une solution naïve consiste à appliquer plusieurs algorithmes avec différents décodeurs et de choisir le meilleur mais ceci a pour inconvénients

d'augmenter énormément la complexité et la consommation de ressources de calcul sans toutefois garantir d'obtention des meilleures performances.

Les algorithmes itératifs ne présentent pas tous les mêmes particularités. Les inventeurs ont observés que, contre toute attente, la cascade de deux algorithmes itératifs de type différent permet d'améliorer le débit total atteint comparativement à celui atteint avec un seul algorithme itératif sans pour autant augmenter globalement le nombre d'itérations. Cette cascade a la particularité de diminuer l'impact des maximas locaux sur le processus de convergence et d'augmenter par voie de conséquence la convergence du deuxième algorithme itératif vers un maximum absolu. En effet, la valeur d'initialisation du deuxième algorithme est obtenue à l'issue de plusieurs itérations du premier algorithme itératif et non plus de manière arbitraire. Ainsi, la détermination de la valeur d'initialisation prend en compte les conditions opérationnelles du système MIMO (canal de transmission, bruit (SNR), nombre d'antennes) contrairement à un choix arbitraire.

Le basculement du premier algorithme au deuxième algorithme est effectué de façon dynamique selon un test de sortie basé sur un seuil de variation du débit atteint sur une fenêtre donnée d'itérations. De manière alternative, il peut être fixé un nombre maximal d'itérations. Ce test assure que la valeur des matrices initiales utilisées par le deuxième algorithme prend suffisamment en compte les variations des conditions opérationnelles; le nombre d'itérations du premier algorithme est donc variable en fonction de ces conditions opérationnelles.

Typiquement, le premier algorithme itératif est choisi comme présentant une variation non stable du débit total et le deuxième algorithme est choisi comme présentant une variation strictement croissante du débit total.

Une solution serait de considérer le premier algorithme basé sur un récepteur MF et le deuxième algorithme prend en compte un deuxième type de récepteur dit MMSE. Ce choix est particulièrement avantageux car le récepteur de type MF (Matched Filter) permet d'obtenir des performances optimales à très hauts SNR alors que le récepteur de type MMSE possède de bon résultats à bas SNR, ce qui permet d'obtenir une solution optimisée en suivant le gradient.

L'algorithme itératif devient donc

Algorithme: Algorithme double itératif à flux multiples

1. Initialisation des paramètres,$iter = 0$,$N_{\max}^{iter}$ (nombre d'itérations maximale global), WIN_{MF}(fenêtre d'observation),ε ,ε_{MF} et Δ_{MMSE} (nombre minimal d'itérations du deuxième algorithme,$\Delta_{MMSE} \in \mathbb{N}^*$ et

$1 \leqslant \Delta_{MMSE} \leqslant N_{\max}^{iter} - WIN_{MF}$) et initialisation des précodeurs. L'initialisation des précodeurs est faite en utilisant des précodeurs linéaire appelé $\mathbf{T}_k^{iter}, k = 1, ..., K$ de type MF donné par $\mathbf{T}_k^{iter} = b\mathbf{H}_k^H$ où b est un facteur de normalisation permettant de respecter la contrainte de puissance $b = \sqrt{\frac{P_T}{\sum_{i=1}^{K} tr(\mathbf{T}_i \mathbf{T}_i^H)}}$. Après l'initialisation des précodeurs, le procédé incrémente l'indice d'itération : $iter = iter + 1$.

2. On calcule les décodeurs MF $\mathbf{D}_k^{iter}$ correspondant aux précodeurs $\mathbf{T}_k^{iter-1}$ pour chaque utilisateur. Ce décodeur MF est donné à la formule (**??**).

3. On recalcule les nouveaux poids $\mathbf{W}_k^{iter}$ pour tous les flux minimisant le MSE en appliquant la formule (1.17).

4. On recalcule les nouveaux précodeurs $\mathbf{T}_k^{iter}$ en fonction de $\mathbf{D}_k^{iter}$ et $\mathbf{W}_k^{iter}$ avec l'équation (1.14).

5. On calcule la somme des débits SR_{iter} obtenu en injectant $\mathbf{T}_k^{iter}$ dans l'équation (1.18).

6. On répète les étapes 2/ à 5/ WIN_{MF} fois en incrémentant à chaque fois $iter = iter + 1$.

7. Vérifier la convergence de l'algorithme MF en vérifiant la contrainte $Var\left(\left[SR_{iter-WIN_{\text{MF}}} \cdots SR_{iter}\right]\right)$ $= \sum_{i=0}^{WIN_{MF}} \left(SR_{iter-i} - \frac{1}{WIN_{MF}} \sum_{i=0}^{WIN_{MF}} SR_{iter-i} \right)^2 \leqslant \varepsilon_{MF}$. Si la condition est vérifiée, on passe à 8/ en conservant les matrices de précodage $\mathbf{T}_k^{iter}$ obtenu. Sinon on incrémentant $iter = iter + 1$ et on refait les étapes 2/, 3/, 4/, 5/ et 7/ jusqu'à atteindre $iter = N_{\max}^{iter} - \Delta$ et on passe à 8/

8. On prend le précodeur qui donne la meilleure performance dans la fenêtre WIN_{MF} considérée. Autrement dit, on considère $iter_{SR\max} = \angle \max_{i \in \{iter-WIN_{\text{MF}}, \cdots iter\}} SR_i$, $iter = iter_{SR\max}$, $\mathbf{T}_k^{iter} = \mathbf{T}_k^{iter_{SR\max}}$. Il est à noter ici qu'une sauvegarde des derniers précodeurs calculés est nécessaire.

9. On incrémente le compteur $iter = iter + 1$ et on calcule les décodeurs MMSE $\mathbf{D}_k^{iter}$ correspondant aux précodeurs $\mathbf{D}_k^{iter}$ pour chaque utilisateur. Ce précodeur MMSE est donné par (1.16).

10. On recalcule les nouveaux poids $\mathbf{W}_k^{iter}$ minimisant le MSE et donnée par la formule (1.17).

11. On recalcule les nouveaux précodeurs $\mathbf{T}_k^{iter}$ en appliquant (1.14) en fonction de $\mathbf{W}_k^{iter}$ et $\mathbf{D}_k^{iter}$.

12. On calcule la somme des débits SR_{iter} obtenu en injectant $\mathbf{T}_k^{iter}$ dans l'équation (1.18).

13. on répète les étapes 9/ à 12/ jusqu'à convergence de l'algorithme itératif. La convergence est déterminée par la stabilisation de la valeur du débit total $\left|SR_{iter} - SR_{iter-1}\right| < \varepsilon$; ou par le dépassement d'un nombre de itérations maximal $N_{\max}^{iter}$.

Une représentation graphique de l'algorithme est illustrée à la figure Figure 1.11. Il est important à noter ici que pour le décodeur MF, il faut utiliser la Frobenius norme au lieu de toute autre norme. Ceci est essentiellement dû au fait que cette norme considère une somme sur toutes les directions contenues dans la matrice et non seulement la plus importante ou les plus importantes. Le fait de considérer toutes les normes de toutes les directions permet de prévenir l'explosion de termes d'interférences et permet à l'algorithme d'optimisation de réagir avant qu'elles ne grandissent permettant une meilleure sélection des sous espaces pour diriger les flux.

1.3.3 Résultats expérimentaux

Afin d'illustrer les gains apportés par les nouveaux algorithmes et approaches proposés dans cette section, on considère des simulations avec des canaux de Rayleigh à composantes i.i.d. gaussiennes et circulantes. Les simulations sont moyennés sur 10^6 réalisations de canaux. et le nombre d'antennes de réception de tous les utilisateurs sont supposés égales sauf mentions contraires. Pour le précodage SJNR, une distribution uniforme de la puissance est prise et est donnée par $P_k = P_T/K$. Les trois paramètres de convergence pour les différents algorithmes ε_{SVH}, ε_{MF}, ε sont égales et fixés à 10^{-3}.
Dans toute la suite, N_{iter} represente le nombre d'itérations effectuées dans la boucle principale dont le maximum est donné par $iter_{max}$. Le nombre total des itérations des deux algorithme du procédé double itératif est défini comme la somme des nombre des itérations dans les deux boucles successives. Parailleurs, on fixe $WIN_{MF} = 5$ et $\Delta = 5$. On présente les courbes

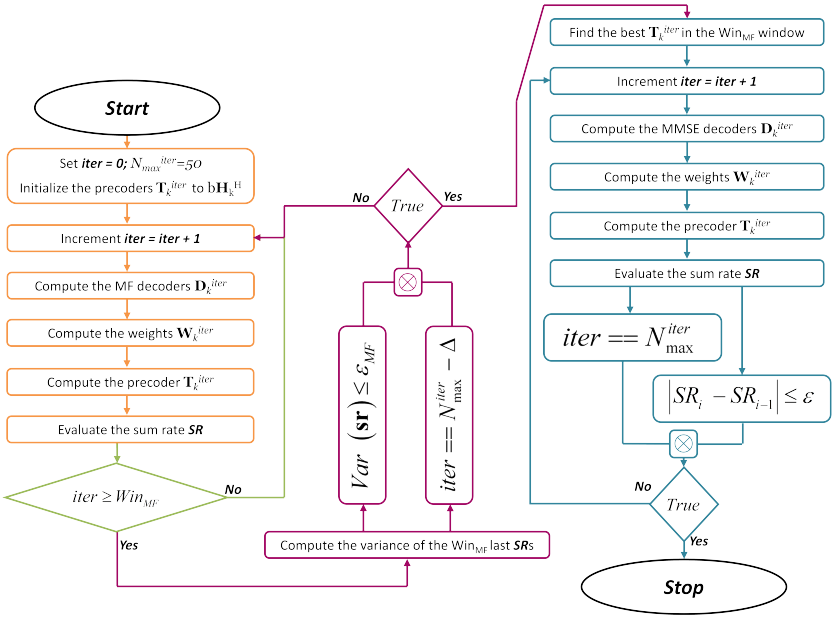


Figure 1.11: Algorithme double itératif.

du débit total obtenu par le système pour différentes configuration du canal de transmission et du nombre d'utilisateurs.

Solutions à flux unique par utilisateur

Dans le cas d'un flux unique par utilisateur, la Figure 1.12 représente pour un nombre fixe d'itérations totales $iter_{max} = N_{iter} = 50$ les performances de l'algorithme SVH/MSR avec différentes valeurs du nombre maximal d'itérations autorisées pour **Algorithm 3** (method 2.1). On considère $iter_{max}^{SVH} = N_{SVH} = 1, 2, 5$. La quatrième courbe décrit l'algorithme laissé libre jusqu'à convergence. Cette figure démontre que la convergence et les performances macximale de l'algorithme ne requièrent pas plus que deux itération de la boucle interne. Les 3 courbes "$TXSVH; RXMSR; N_{SVH} = 2; N_{iter} = 50$", "$TXSVH; RXMSR; N_{SVH} = 5; N_{iter} = 50$" et "$TXSVH; RXMSR; N_{SVH} = Conv; N_{iter} = 50$" sont presque confondues. Les gains de performances de la courbe représentant le système qui a convergé $iter_{max}^{SVH} = 5$ et $iter_{max}^{SVH} = 2$ est de l'ordre de 10^{-4} et 10^{-3} bits/Hz/s. Ce qui justifie l'utilisation de deux itérations pour l'algorithme d'optimisation MISO. D'un autre coté, et en observant les courbes "$TXSVH; RXMSR; N_{SVH} = 1; N_{iter} = 50$"

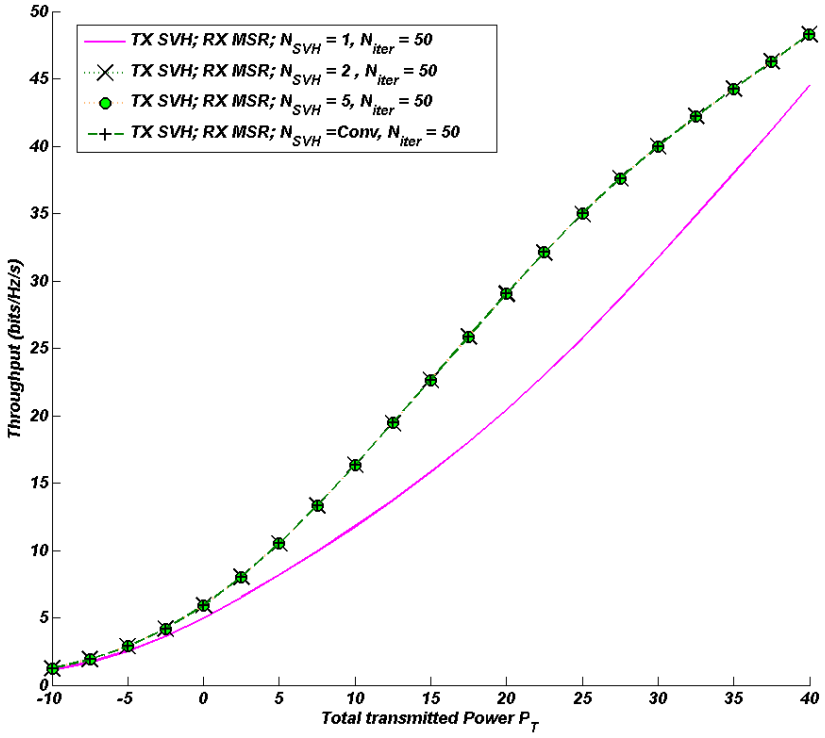


Figure 1.12: Débit en fonction de $iter_{max}^{SVH}$ pour $N_T = N_R = K = 4$.

avec un nobre d'itérations internes fixés à 1, démontre de faibles valeurs du débit offert. En effet, en analysant l'algorithme $iter_{max}^{SVH} = 1$, on voit clairemant qu'il représente un MMSE/MSR équivalent au MMSE/MMSE décrit dans [40] mais utilisant un récepteur normalisé. Le facteur de normalisation n'est autre qu'un facteur scalaire qui n'influence pas les performances dans une boucle vu qu'il est éliminé dans l'expression du débit. Mais, lors du processus itératif, il affecte la distribution des puissance d'une itération à l'autre et de ce fait change la solution d'optimisation obtenue. Ceci, démontre l'importance de la normalisation dans le processus d'optimisation dans les systèmes MU-MISO. Toutes ces observations démontres les gains significatifs qu'on peut obtenir juste en rajoutant une boucle d'obtimisation utilisant l'algorithme (method 2.1). Ces gains sont ainsi obtenus en introduisant peu de boucles supplémentaires et une faible complexité de traitement.

La Figure 1.13 montre l'influence du nobre total d'itérations pour l'algorithme SVH/MSR. un nombre fixe d'itérations $iter_{max}^{SVH} = 2$ est considéré vu qu'on

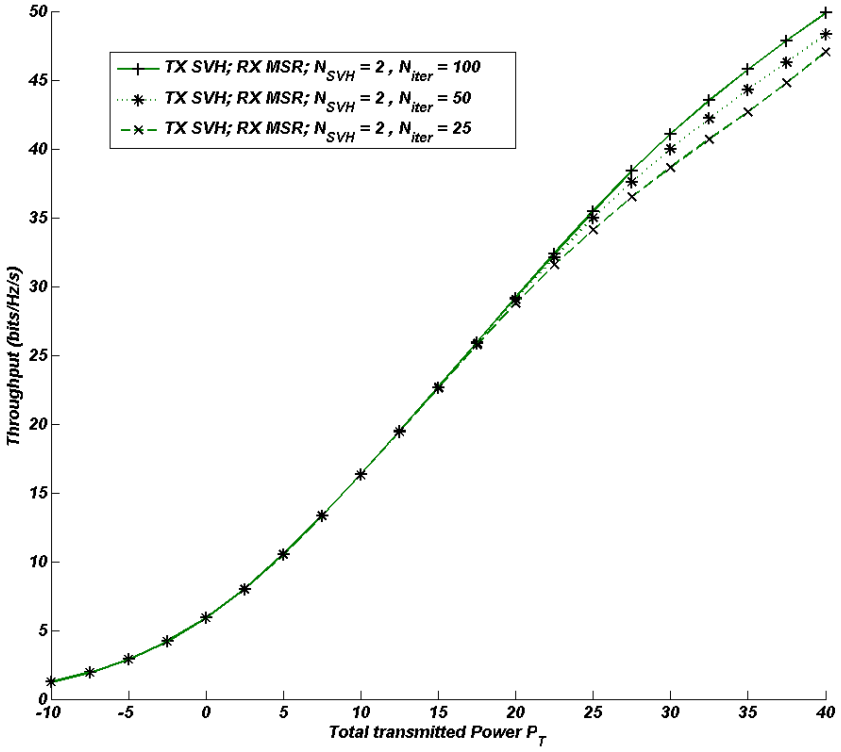


Figure 1.13: Débit en fonction de $iter_{max}$ pour $N_T = 4$, $N_R = 4$, $K = 4$.

a démontré précedemment que cela suffisait pour obtenir une bonne convergence. On varie alors le nobre d'itérations de la boucle externe. On remarque le débit obtenu augmente légèrement à bas SNR et un peux plus à haut SNR en fonction du nobre d'itérations. Ceci montre que malgrè le fait qu'un canal MU-MISO est considéré à chaque itération, l'algorithme SVH/MSR évolue grâce à la construction du canal virtuel vers une bonne solution exploitant la diversité du canal MIMO.

La Figure 1.14 represente une comparison des performances de SVH/MSR avec d'autres algorithmes existants. On considère un MMSE/MMSE iteratif présenté dans [40] appelé *original* et avec une version modifiée de ce dernier et appelée *normalized MMSE/MMSE*. Ces algorithmes utilisent un précodeur et un récepteur MMSE. L'initialisation est $\mathbf{d}_k^0 = \mathbf{I}_{1\times N_R}$,où $\mathbf{I}_{1\times N_R}$ possède un 1 dans la première position et des 0 ailleurs. La version *normalized*, introduit un facteur de normalisation au niveau du récepteur. Pour ces simulations, $N_{iter} = iter_{max}$ est contant et égal à 25 et 50. Le courbes représen-

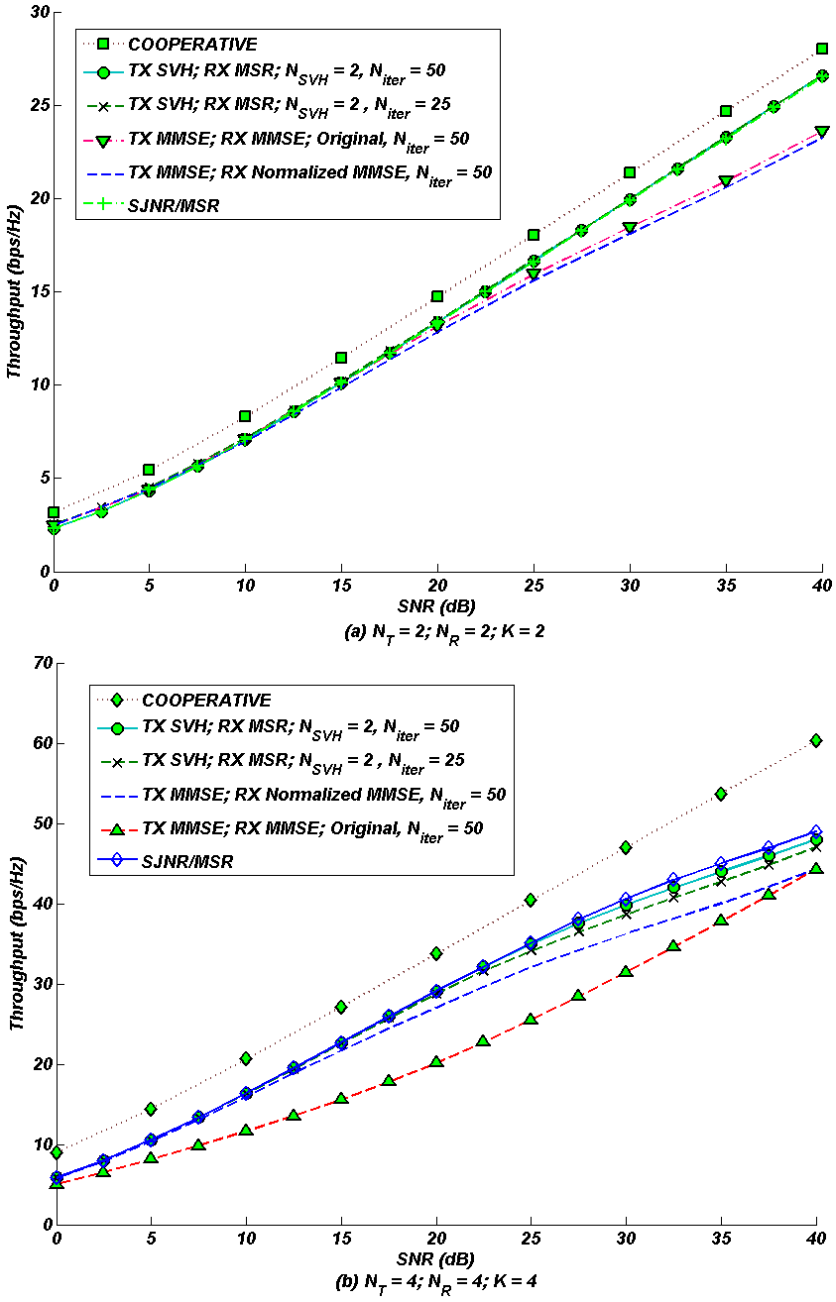


Figure 1.14: Débit pour $N_T = N_R = K = 2$ et $N_T = N_R = K = 4$.

tent les cas $N_T = 2$, $N_R = 2$, $K = 2$ et $N_T = 4$, $N_R = 4$, $K = 4$. L'analyse des courbes, démontrent que pour la même complexité et le même nombre d'itérations, l'algorithme proposé offre de bien meilleures performances comparé au MMSE/MMSE même celui modifié et utilisant un récepteur optimal. Par ailleurs, comparant les courbes "$TXSVH, RXMSR, N_{SVH} = 2, N_{iter} = 25$" et "$TXSVH, RXMSR, N_{SVH} = 2, N_{iter} = 50$" on remarque des performances similaires et une faible décroissance est notés malgrè une division du nombre d'itérations par 2. Ces rasiltats démontrent la stabilité et l'optimanité du SVH/MSR qui offre de meilleurs performances avec un très faible nombre d'itérations.

Pour comparer les performances aves une solution basée sur les valeurs propres des canaux, on rajoute les courbes du SJNR/MSR. Les simulations démontrent qu'à haut SNRs, le SJNR/MSR iteratif est meilleur que le SVH/MSR. Mais à bas SNRs, c'est le SVH/MSR qui l'emporte légèrement. Ces résultats peuvent être expliqués par le fait qu'à haut SNRs, sélectionner la plus grande valeur propre est plus efficace qu'une optimisation globale vu qu'on a assez de puissance. Par contre à bas SNRs, les valeurs propres sont toutes proches les unes des autres et de ce fait, il devient crucial de minimizer les superpositions entre les directions choisies.

On rajoute aussi ici la courbe de l'algorithme coopératif (i.e. un seul utilisateur MIMO comprenant tous les canaux $\mathbf{H}^T = \left[\mathbf{H}_1^T \cdots \mathbf{H}_K^T\right]$). Il s'agit de la borne supérieure maximale vu qu'elle considère la coopération totale entre les utilisateurs. Comparant le SVH/MSR iteratif avec le cooperatif démontre que le SVH/MSR reste parallèle à bas SNR mais sature rapidement à haut SNR et surtout quand les dimentions du système augmentent. Ceci montre que le SVH/MSR exploite correctement la diversité du système à bas SNRs.

Pour les figures suivantes, on considère les notations suivantes. $SJNR/MF$ est la courbe du SJNR/MF derivé de l'**Algorithm 2**. $SJNR/MSR$ décrit le SJNR/MSR donné en [48] et le $SJNR/MFMSR_D$ correspond à la courbe de l'algorithme double itératif proposé.

Les Figures 1.15 présentent un nomde d'itérations total fixe $N_{max}^{iter} = iter_{max} = 50$ le débit total pour les trois algorithmes $SJNR/MSR$, $SJNR/MF$ et le $SJNR/MFMSR_D$. la première configuration est $N_T = N_R = K = 4$, la seconde possède $N_R = 2$ antennes de réceptions par utilisateur. La dernière représente un système avec $N_T = N_R = K = 2$. Toutes ces configurations représentent des systèmes complètement chargés.

Ces courbes montrent que pour les deux premières configurations, le $SJNR/MF$ est meilleur que le $SJNR/MSR$ surtout à haut SNRs et que la courbe de débit augmente linéairement en fonction du débit. Ce com-

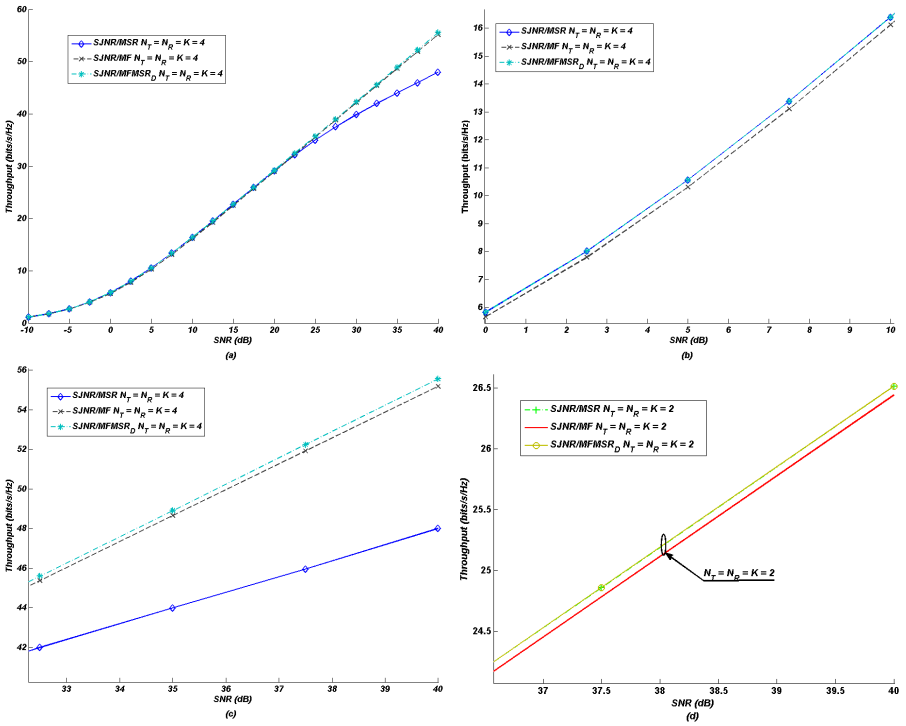


Figure 1.15: Débit en fonction du SNR P_T pour différentes configurations du système.

portement peut être expliqué par le fait qu'à haut SNRs, les différents plux peuvent être facilement séparés simplement par l'emploi de filtres MF et ainsi, à travers le processus itératif, les filtres sont calculés pour maximiser la puissance recue pour chaque utilisateur. A bas SNRs, le MF n'arrive pas à recouvrir les différents flux et de ce fait introduisent de la sous optimalité. Parailleurs, le MSR est capable de bien optimiser à bas SNRs. Ceci explique pourquoi l'algorithme $SJNR/MSR$ est meilleur que le $SJNR/MF$ dans cette region.

Basé sur ces observations, on propose une combination de ces deux algorithmes décrit dans ce chapitre: le $SJNR/MFMSR_D$. Comparant dette courbe avec celles du $SJNR/MF$ et $SJNR/MSR$ démontre de meilleurs débits dans toute la plage de SNRs. Les débits obtenus sont même plus élevés que le maximum des deux autres aproches. En effet, l'analyse de la Figure 1.15.c montre qu'à haut SNRs, les performances obtenus sont légèrement supérieures. A bas SNRs, comme le montre la Figure 1.15.b l'algorithme

proposé arrive à atteindre le meilleur des deux algorithmes simples.

Non seulement le $SJNR/MFMSR_D$ est capable de garantir les performances offertes par le meilleur des deux orpsédés mais en plus offre des améliorations de l'ordre de 10^{-4} comparé au $SJNR/MSR$ (qui est le meilleur algorithme dans toute la gamme de SNR). Ceci montre la stabilité de l'algorithme et ces bonne propriété de convergence pour toutes les configuration du système. Ces améliorations importantes de performances sont obtenues simplement par l'introduction de faibles complexité de calculs sans augmentation du délai de traitement ni du nombre d'itérations requises simplement par l'utilisation d'un bon critère de basculement entre algorithmes.

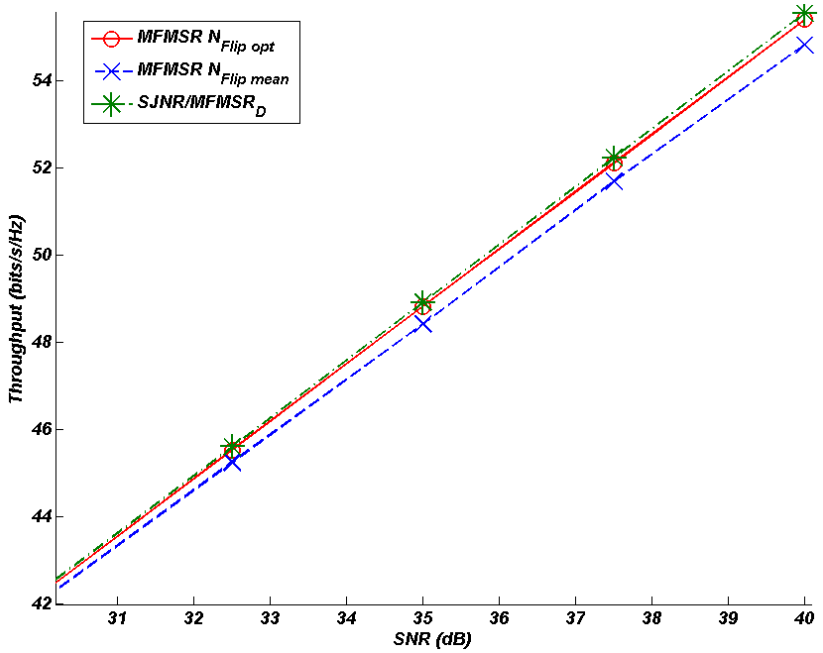


Figure 1.16: Débits pour $N_T = N_R = K = 4$.

Figure 1.16 represente le débit atteint par le système quand on utilise un point de basculement déduit de l'nalyse statistique donnée à la Figure 1.8. La première courbe représente un point de basculement ($N_{Flipopt}$) calculé pour maximiser la moyenne du débit total du système. La seconde courbe est le débit total obtenu en applicant avec le point de basculement ($N_{Flipmean}$) qui cherche à maximiser la moyenne instantannée du débit total. Il est à noter que ces deux courbes sont basée sur des tables de décisions

dérivées à partir des données statistiques. Ces performances sont comparées à celles obtenues par le procédé dynamique de sélection d'algorithme. Les résultats de simulation démontrent que l'algorithme proposé offre des performances les deux algorithmes séparément.

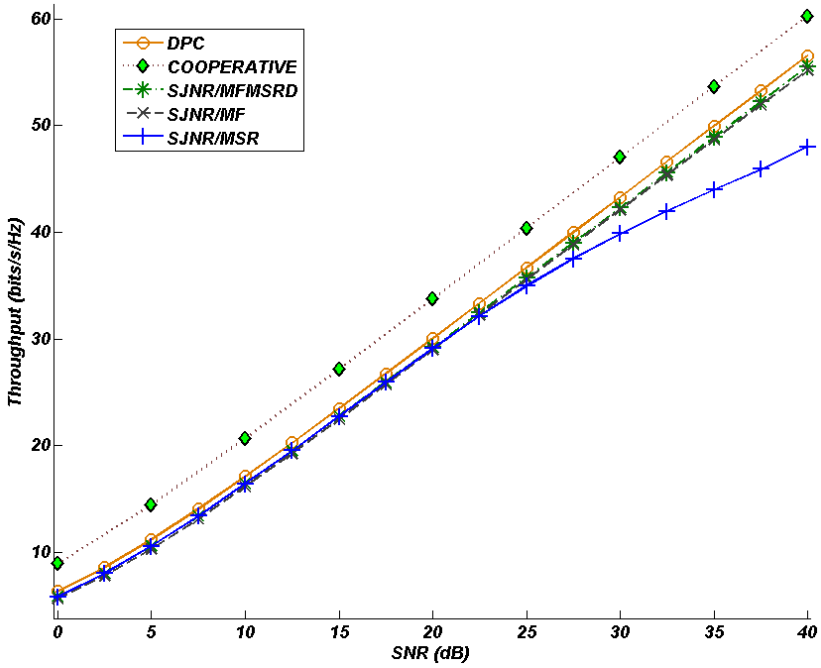


Figure 1.17: Débits pout le cooperatif, DPC et SJNR/MFMSRD pour $N_T = N_R = K = 4$.

Figure 1.17 effectue une comparaison des performances obtenues par l'algorithme double itératif présenté et les solutions préexistantes. On a, de ce fait, rajouté les courbes représentant l'approche coopérative et DPC. La courbe DPC a été générée en utilisant [25] et représente la borne supérieure pour les systèmes MU-MIMO à diffusion. Ces courbes montrent qu'avec la même complexité de calcul, l'algorithme proposé offre d'important gains. En plus, comparé avec les performances du système coopératif, le $SJNR/MFMSR_D$ offre des débits avec une pente qui tend vers celle du système parfait (celle du DPC).

Figure 1.18, affiche les performances obtenues par différents algorithmes basés sur l'algorithme SVH. Ces performances sont comparées à celles obtenues par le SJNR/MFMSRD. Les courbes montrent qu'à bas SNRs, les débits atteignables sont toujours meilleures pour les algorithmes impliquant une

optimisation basée sur le SVH après construction du canal virtuel MISO. Par ailleurs, à haut SNRs, le SVH perd de la performance. La courbe $SJNR/MF + SVH/MSR$ représente la cascade des deux algorithme combinés avec un algorithme de sélection dynamique de l'algorithme. La complexité totaleobtenue ainsi est la même que celle des algorithmes SJNR. En fait, comme on l'a montré plus tôt, La procédure d'optimisation interne SVH (**Algorithm 3**) ne requière que deux itérations pour atteindre des performances comparable à celles obtenues à la convergence. De ce fait, le nombre d'itéérations ont été réajustées en comptant deux fois ces itérations. Les résultats obtenus montrent des performances équivalentes à celles obtenues avec le procédé de saut dynamique double itératif SVH/MFMSRD. En comparant SJNR/MF + SVH/MSR avec $SJNR/MFMSR_D$, on remarque que la décision de changement d'algorithme doit être faite en fonction du SNR. Un bon compromis sans rajout de complexit"é consiste à introduite une troisième couche à l'optimisation double itérative en suivant le même procédé et critère de basculement en lançant le l'algorithme SVH/MSR en commençant du point de convergence obtenu précedemment.

La dernière figure pour le cas de simple flux par utilisateur, Figure 1.19 illustre le débit du système avec un algorithme $SJNR/MFMSR_D$ cascadé avec un algorithme SVH/MSR pour un système $N_T = N_R = K = 4$ qui utilise respectivement 2 et 6 itérations internes pour l'algorithme d'optimisation SVH. Les courbes obtenues montrent que l'utilisation du SVH après l'algorithme double itératif permet d'améliorer encore plus les performances à très bas SNRs.

Performance des algorithmes à flux multiples

Pour le cas de flux multiples par utilisateur, Figures 1.20 et 1.21 représentent les résultats de simulations pour un système MU-MIMO avec $N_T = 4$ antennes de transmissions, $N_R = 4$ antennes de réception par utilisateur et $K = 4$ utilisateurs. L'analyse effectuée ci-après reste vraie pour toutes les configurations des systèmes essentiellement celles définies par la norme LTE. Les courbes $WMMSE/MFMMSED$ obtenues par l'**Algorithm 7** sont comparé avec celles de $WMMSE/MMSE$ proposé dans [42] et $WMMSE/MF$ de l'**Algorithm 6** avec un récepteur MF. On rajoute aussi l'**Algorithm 5** pour la procédure double itérative à basculement dynamique en simple flux que l'on a précedemment présenté et appellé SJNR/MFMSRD.

La comparaison des courbes $WMMSE/MMSE$ et $WMMSE/MF$ montrent que le $WMMSE/MF$ offre de meilleures performances surtout à haut SNRs. Ce comportement peut être expliqué par le fait qu'à haut SNRs, les

flux peuvent être correctement séparés simplement par l'utilisation d'un filtre MF et que les itérations permettent ainsi d'optimiser afin de maximiser les puissances par flux. Cependant, le récepteur MMSE est capable d'offrir une meilleure séparation des utilisateurs et aidé par la matrice des poids $\mathbf{W}$ réoriente itérativement les recherches vers les utilisateurs les moins interférants.

L'algorithme proposé donne une courbe présentant de meilleures performances en termes de débit et ceci dans toute la plage des SNRs. Les débits obtenus sont même plus élevés que ce qu'on obtiendrais si on considérait le maximum des deux algorithmes instantannément. L'analyse de la Figure 1.21.a montre qu'à haut SNRs, L'algorithme proposé offre de meilleures performances. A bas SNRs, comme le montre la Figure 1.21.b, la procédure proposée est capable non seulement de récupérer le meilleur des deux algorithmes impliqués mais génère un gain supplémentaire. Ces résultats montrent la stabilité de l'algorithme en termes de convergence, performance et surtout face à toute configuration du système. Ces performances ont été obtenues juste en introduisant un point de basculement dynamique sans rajouter de complexité ni de délai de traitement supplémentaire ni en rajoutant d'itérations en offrant une convergence plus rapide.

De plus, en comparant la courbe de l'algorithme doube litératif avec la solution DPC, nous montre que la solution offre des performances très proches de l'optimal (DPC). En fait, les deux courbes sont parallèles pour toute la plage de SNR avec une différence de moins que 1 bit/s/Hz. Comparée au DPC (solution non linéaire), ce décalage en termes de performances est naturel et a été démontré par [36] qui a déterminé l'existance d'une constante entre les algorithmes linéaires et les algorithmes non linéaires à haut SNR; Ceci contre bien non seulement l'optimalité de la solution proposée mais aussi et surtout sa stabilité et sa bonne convergence basé sur une recherche rapide grâce au WMMSE/MF suivi par un rafinement de la solution par l'utilisation du WMMSE/MMSE. Finalement, en comparant les performances des algorithmes simple et multifplux, on remarque que dans le cas des systèmes à faible diversité ou dit aussi système à charge pleine les algorithmes sont équivalents. Cependant, en considérant un système où des degrès de liberté existent encore, on voir clairement la domination de la solution multiflux qui adopte la bonne pente exploitant pleinement le système comme le montre la Figure 1.22 qui représente les performances obtenues pour un système avec $N_T = N_R = 4$ et servant $K = 2$ utilisateurs..

La dernière figure pour cette partie, Figure 1.23 représente l'impact du choix de la norme pour l'algorithme WMMSE/MF. Ici on représente le débit obtenu pour trois cas de figure: La norme Frobenius, la norme Spectrale et

la courbe sans utilisation de normalisation pour le récepteur MF. Les résultats obtenue confirment les conclusions de la discussion précédente. En fait, vu que la norme Frobenius prend en compte la contribution des puissances reçues dans toutes les directions elle permet une meilleure optimisation de la recherche de l'allocation de puissance durant la première phase de l'algorithme. Cette optimisation permettant la sélection les meilleurs flux est cruciale pour la convergence du second algorithme qui est très sensible au point d'initialisation.

1.4 Optimisation de précodeurs en solution approchée pour MU-MIMO

Dans cette 1.4ème section, nous traitons une seconde famille de précodeurs. A savoir les précodeurs linéaires à forme approchée. Ce type de précodeurs sont généralement des approches analytiques où on se basent sur une formule pour calculer le précodeur et le décodeur requis.

Comme exposé dans la section 1.2, dans les systèmes MISO, on n'a pas besoin d'optimiser les récepteurs vu qu'ils n'ont aucune influence sur les performances des systèmes. Cependant, dans les systèmes MIMO les roles des récepteurs devient une composante primordiale dans le processus d'optimisation comme on l'avais vu à travers la section 1.3. Ces principes restent valables aussi pour le cas des précodage CF.

1.4.1 Importance du bon choix du récepteur

Dans cette partie, on va partir de deux solutions CF sur lesquelles on avait basé le premier algorithme itératif pour illustrer ce principe. En fait, les précodeurs SJNR proposé en [27] a été considéré avec un simple récepteur MF. Le précodeur PU-MMSE quant à lui, il a été proposé avec un décodeur qui considère le vecteur correspondant à la valeur propre la plus grande de la cascade canal-précodeur.

Il est clair dans ces deux cas que les décodeurs choisis ne sont pas optimisé pour les précodeurs considérés. De ce fait, on a proposé une solution générique pour la détermination du décodeur optimal. En fait, le décodeur optimal serait celui qui permetterait de maximiser le débit total du système qui n'est rien d'autre que la notre fonction coût à maximiser. Et en considérant que les flux sont séparables ou dans le cas de l'allocation d'un seul flux par utilisateur, on peut réécrire cette quantité comme une somme des

débits réalisables par flux. Le débit par flux est donné par

$$R_k = \log_2 \left(1 + \frac{\mathbf{d}_k \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H \mathbf{d}_k^H}{\sum\limits_{i=1, i \neq k}^{K} \mathbf{d}_k \mathbf{H}_k \mathbf{t}_i \mathbf{t}_i^H \mathbf{H}_k^H \mathbf{d}_k^H + N_0} \right) \tag{1.19}$$

La maximisation de cette quantité en fonction du récepteur revient à maximiser la quantité

$$\mathbf{d}_k \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H \mathbf{d}_k^H \left(\sum_{i=1, i \neq k}^{K} \mathbf{d}_k \mathbf{H}_k \mathbf{t}_i \mathbf{t}_i^H \mathbf{H}_k^H \mathbf{d}_k^H + N_0 \right)^{-1} \tag{1.20}$$

dont la solution n'est autre que (comme pour le cas SJNR) la plus grande valeur propre de la quantité

$$\psi = \left(\sum_{i=1, i \neq k}^{K} \mathbf{H}_k \mathbf{t}_i \mathbf{t}_i^H \mathbf{H}_k^H + N_0 \right)^{-1} \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H \tag{1.21}$$

Ainsi le récepteur optimal noté d_{MSR} qui maximise la somme des débits dans un tel système MIMO multiutilisateurs est donné par

$$\mathbf{d}_{MSR} = \xi_m \left(\left(\sum_{i=1, i \neq k}^{K} \mathbf{H}_k \mathbf{t}_i \mathbf{t}_i^H \mathbf{H}_k^H + N_0 \right)^{-1} \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H \right) \tag{1.22}$$

Une autre observation que nous avions vue dans la section précédente, et qui n'est pas évidente vu qu'elle est cachée dans les itérations, est que l'optimisation du SR ne peut pas se faire sans un passage par une bonne sélection d'utilisateurs et de flux et une bonne distribution de puissance surtout à bas SNRs.

En fait, il s'avère que ces deux procédés sont étroitement liés dans les canaux de diffusion vu qu'on dispose d'un budget total de puissance à distribuer sur un certain nombre d'utilisateurs. Si on considère une bonne sélection d'utilisateurs et ainsi de flux on aura besoin d'un minimum de puissance pour contrer les effets du canal et tout l'exès sera utiliser pour augmenter le débit. De ce fait, on a proposé une modification dans ces algorithmes en introduisant une sélection qui permette en éteignant un à un les flux des utilisateurs de reconnaître les mauvais utilisateurs à ne surtout pas servir. Cette procédure simple mais efficace permet des gains significatifs en termes de performances du système.

1.4.2 Précodage CF avec flux multiples par utilisateurs

Il est normal de commencer par étendre les algorithmes à notre disposition pour passer du simple flux par utilisateur à une configuration où l'on peut avoir des flux multiples. Ceci est surtout motivé par les systèmes où l'on a des antennes en transmission supérieur au nombre d'utilisateurs à servir $N_T \geq K$. Dans ce cas de figure, la diversité totale du système est sous exploités et donc le système ne fonctionne pas à débit maximal.

Afin de bien exploiter la totalité des ressources du système, une solution serait de permettre l'allouer de plus qu'un flux pour les utilisateurs gourmands en débits. Une solution qui a été proposé dnas la litérature est une extension du SJNR qui a été proposée dans [33]. SLNR est l'acronyme de Signal to Leackage and Noise Ration qui comme el SJNR considère les fuites de puissance des flux de l'utilisateur k et qui vont aller perturber les autres utilisateurs.

Le précodeur est donc l'ensemble des meilleures valeur propres existant dans le sous espace engendré par le signal utile et la fuite vers les autres utilisateurs. Il peut être donné par

$$\mathbf{T}_k = \angle \max_{\mathbf{T}_k \in \mathbb{C}^{N \times Q_k}} \frac{tr\left(\mathbf{T}_k^H \mathbf{H}_k^H \mathbf{H}_k \mathbf{T}_k\right)}{tr\left(\mathbf{T}_k^H \left(\sum\limits_{h=1, j\neq k}^{K} \widetilde{\mathbf{H}}_j^H \widetilde{\mathbf{H}}_j + N_0 \mathbf{I}\right) \mathbf{T}_k\right)} \tag{1.23}$$

où $\widetilde{\mathbf{H}}_j = \left[\mathbf{H}_1 \cdots \mathbf{H}_{j-1} \mathbf{H}_{j+1} \cdots \mathbf{H}_K\right]^T$.

Le précodeurs sont donc les matrices se trouvant dans l'espace propre généralisé de $\left(\sum\limits_{h=1, j\neq k}^{K} \widetilde{\mathbf{H}}_j^H \widetilde{\mathbf{H}}_j + N_0 \mathbf{I}, \mathbf{T}_k^H \mathbf{H}_k^H \mathbf{H}_k \mathbf{T}_k\right)$ et qui vérifient le système d'équation

$$\begin{cases} \mathbf{T}_k^H \mathbf{H}_k^H \mathbf{H}_k \mathbf{T}_k = \Lambda_k \\ \mathbf{T}_k^H \left(\sum_{h=1, j\neq k}^{K} \widetilde{\mathbf{H}}_j^H \widetilde{\mathbf{H}}_j + N_0 \mathbf{I}\right) \mathbf{T}_k = \mathbf{I} \end{cases} \tag{1.24}$$

avec Λ_k est une matrice diagonale positive.

Cette technique offre des avantages vu qu'on n'a besoin que de la matrice covariance pour les calculer au lieu de la connaissance de tout le canal. Cependant, les matrices de covariances ne sont pas simple à obtenir parfois même plus compliqués quand les dimension du système explosent. Par ailleurs, la solution proposée ne tiens pas dutout en compte une qualconque sélection d'utilisateurs et encore moins de flux ce qui détruit les performances de cette approache surtout avec un grand nombre d'utilisateur.

Tenant en compte toutes ces remarques, on propose dans la section suivante une solution récursive de construction de précodeurs linéaires extrapolés à partir d'une version relaxée d'une approche ZF-DPC (ZeroForcing Dirty Paper Coding).

1.4.3 Algorithme NS-ZF

Cette approche inspirée de l'algorithme ZFDPC-SUS proposé par Goldsmith en [30,49] permet de sélectionner les flux les plus orthogonaux entre eux pour minimiser les interférences.

Le but ici est de donner une expression simple des précodeurs en solution approchée afin de réaliser une sélection optimale des utilisateurs à servir, des flux à considérer et à affecter à chaque utilisateur ainsi que la distribution de puissance à affecter à ces flux choisis. La solution que nous proposons se base sur deux étapes qui peuvent être définis comme une première étape de sélection conjointe des flux et des utilisateurs; La deuxième étape est l'application du précodeur considéré pour ces flux et enfin l'affectation des puissances sur les précodeurs ainsi construits.

Description du processus de sélection des flux

Pour la sélection des flux on se base sur une approche récursive alternant un choix en fonction des valeurs propres des flux disponibles et projection orthogonale sur l'espace complémentaire aux directions préalablement choisis pour les autres flux déjà sélectionnés. L'algorithme s'articule de se fait comme suit:

Algorithme: Sélection récursive de flux

---

1. On calcule les SVDs des canaux de tous les utilisateurs de manière que $\mathbf{H}_k = \mathbf{U}_k \mathbf{S}_k \mathbf{V}_k^H$, $\forall i \in [1 \cdots K]$.

2. On commence par choisir le premier couple (utilisateur, flux) noté $\left(u_1, f_1\right)$ qui possède la valeur propre la plus grande donc le meilleur canal de transmission.

3. On reconstruit le canal virtuel correspondant à ce flux $\mathbf{h}_1 = \mathbf{u}_{u_1,f_1}^H \mathbf{H}_{u_1}$. Ce canal virtuel est alors rajouté dans une liste $\mathbf{H}_S = \mathbf{h}_1$. Sauvegarder le précodeur correspondant $\widetilde{\mathbf{t}_1} = \mathbf{v}_{u_1,f_1}$. Initialiser un compteur $i = 2$.

4. Construire la matrice de projection sur l'espace complémentaire aux canaux des flux sélectionnés $\mathbf{H}_S$. La matrice de projection est donnée par $\mathbf{P}_i^{\perp} = \mathbf{I} - \mathbf{H}_S^H \left(\mathbf{H}_S \mathbf{H}_S^H\right)^{-1} \mathbf{H}_S$.

5. Projeter tous les canaux de tous les utilisateurs sur cet espace complémentaire .

6. Faire la SVD de tous les canaux projetés $\mathbf{H}_k^{\perp} = \mathbf{H}_k \mathbf{P}_i^{\perp}$.

7. a. Parcourir tous les utilisateurs et tous les flux en calculant le niveau d'interférence qu'ils reçoivent des flux précédemment sélectionnés : $A_{j,l} = \sum_{m=1}^{i-1} \mathbf{u}_{l,j}^{\perp} \mathbf{H}_j \widetilde{\mathbf{t}_m}, \forall l \in [1 \cdots K], \forall j \in \left[1 \cdots \min\left(N_T, N_{R_l}\right)\right]$. Sélectionner parmi les flux ceux qui subissent un niveau d'interférence inférieur à un niveau δ prédéfini. Si aucun flux ne satisfait la contrainte d'interférence on sélectionne tous les utilisateurs. b. Sélectionner tous les utilisateurs

8. Prendre parmi les flux sélectionnés dans 7/ celui qui présente le meilleur canal (la plus grande valeur propre). Ce couple d'utilisateur flux sera noté par $\left(u_i, f_i\right)$.

9. Sauvegarder le précodeur $\widetilde{\mathbf{t}_i} = \mathbf{v}_{u_i,f_i}^{\perp}$, sauvegarder le canal virtuel correspondant à ce flux $\mathbf{h}_i = \left(\mathbf{u}_{u_i,f_i}^{\perp}\right)^H \mathbf{H}_{u_i}$. Mettre à jour la matrice des canaux $\mathbf{H}_S = \left[\mathbf{H}_S^T, \mathbf{h}_i^T\right]^T$. Incrémenter le compteur $i = i + 1$

10. Répéter les étapes 4/ à 10/ jusqu'à ce qu'on atteigne le nombre maximal de flux possible. C'est-à-dire $i > \min(\sum_{k=1}^{K} N_{R_k}, N_T)$.

11. Passer à l'étape suivante Construction des précodeurs.

Construction des précodeurs sur les flux sélectionnés

Pour la construction des précodeurs, pour les canaux virtuels des flux sélectionnés ($\mathbf{H}_S$) plusieurs alternatives sont possibles. En effet, on peut appliquer un précodeur qui inverse l'ensemble des canaux on applique alors un précodage Zéro Forcing (ZF) comme suit:

$$\mathbf{t}_{u_i,f_i} = \gamma \mathbf{h}_i^{\dagger}, \forall i \in [1 \cdots Q]$$

où $\mathbf{h}_i^\dagger$ est la $i$ème colonne de $\mathbf{H}_S^\dagger = \mathbf{H}_S^H \left(\mathbf{H}_S \mathbf{H}_S^H\right)^{-1}$. $\mathbf{t}_{u_i,f_i}$ est le vecteur de précodage pour le flux f_i envoyé à l'utilisateur u_i. où γ est la distribution de puissance qui peut être uniforme sur tous les flux ou défini par l'algorithme de distribution de puissance donné dans le paragraphe allocation de la puissance ci-dessous. On peut aussi imaginer un précodage MMSE donné par :

$$\mathbf{t}_{u_i,f_i} = \beta \mathbf{H}_S^H \left(\mathbf{H}_S \mathbf{H}_S^H + \sigma^2 \mathbf{I}\right)^{-1}, \forall i \in [1 \cdots Q] \tag{1.26}$$

Où est un facteur scalaire pour respecter la contrainte de puissance $\sum_{i=1}^{Q} tr\left(\mathbf{t}_{u_i,f_i} \mathbf{t}_{u_i,f_i}^H\right) = P_T$. Ou toute autre forme de précodage appliqué à la matrice $\mathbf{H}_S$.

Allocation de la puissance

Pour l'allocation de la puissance et dans le cas d'absence totale d'interférence (cas d'un précodage ZF) on considère les canaux virtuels construits et contenus dans $\mathbf{H}_S$. On calcule leur puissances respectives $\lambda_i = \left(\mathbf{h}_i \mathbf{h}_i^H\right), \forall i \in [1 \cdots Q]$ et on applique le waterfilling (WF) sur le vecteur $\Lambda = \left[\lambda_i \cdots \lambda_Q\right]$ ainsi construit en respectant la contrainte de puissance $P_T = \sum_{i=1}^{Q} P_i$ ce qui nous génère un vecteur de puissances $\left[P_1 \cdots P_Q\right]$. Les précodeurs à utiliser sont alors affectés par leurs puissances respectives

$$\mathbf{t}_{u_i,f_i} = \sqrt{P_i} \mathbf{h}_i^\dagger, \forall i \in [1 \cdots Q] \tag{1.27}$$

Améliorations de l'algorithme (Greedy Selection)

Pour améliorer les performances de l'algorithme et surtout dans le cas où on est obligé dans l'étape 8/ de sélectionner un utilisateur qui subit trop d'interférence on peut appliquer une sélection par étape des utilisateurs au cours de l'exécution de l'algorithme.

L'algorithme se base sur le gain en termes de SR grâce à un rajout d'un nouvel utilisateur. L'algorithme ainsi défini devient comme suit:

Algorithme: Sélection récursive de flux avec contrôlr d'admission

1. On calcule les SVDs des canaux de tous les utilisateurs de manière que $\mathbf{H}_k = \mathbf{U}_k \mathbf{S}_k \mathbf{V}_k^H$, $\forall i \in [1 \cdots K]$.

2. On commence par choisir le premier couple (utilisateur, flux) noté $\left(u_{1,}f_1\right)$ qui possède la valeur propre la plus grande donc le meilleur canal de transmission.

3. On reconstruit le canal virtuel correspondant à ce flux $\mathbf{h}_1 = \mathbf{u}_{u_1,f_1}^H \mathbf{H}_{u_1}$. Ce canal virtuel est alors rajouté dans une liste $\mathbf{H}_S = \mathbf{h}_1$. Sauvegarder le précodeur correspondant $\widetilde{\mathbf{t}_1} = \mathbf{v}_{u_1,f_1}$. Initialiser un compteur $i = 2$.

4. On évalue le SR ainsi obtenu $SR_1 = \log_2\left(1 + \frac{\mathbf{u}_{u_1,f_1}\mathbf{H}_{u_1}\mathbf{t}_{u_1f_1}\mathbf{t}_{u_1f_1}^H\mathbf{H}_{u_1}^H\mathbf{u}_{u_1,f_1}^H}{\mathbf{u}_{u_1,f_1}(\sigma^2\mathbf{I})\mathbf{u}_{u_1,f_1}^H}\right)$ où $\mathbf{t}_{u_1f_1}$ est donné par (1.27) Sauvegarder $\mathbf{t}_{u_1f_1}$. Initialiser un compteur $i = 2$.

5. Construire la matrice de projection sur l'espace complémentaire aux canaux des flux sélectionnés $\mathbf{H}_S$. La matrice de projection est donnée par $\mathbf{P}_i^{\perp} = \mathbf{I} - \mathbf{H}_S^H\left(\mathbf{H}_S\mathbf{H}_S^H\right)^{-1}\mathbf{H}_S$.

6. Projeter tous les canaux de tous les utilisateurs sur cet espace complémentaire .

7. Faire la SVD de tous les canaux projetés $\mathbf{H}_k^{\perp} = \mathbf{H}_k\mathbf{P}_i^{\perp}$.

8. a. Parcourir tous les utilisateurs et tous les flux en calculant le niveau d'interférence qu'ils reçoivent des flux précédemment sélectionnés : $A_{j,l} = \sum_{m=1}^{i-1}\mathbf{u}_{l,j}^{\perp}\mathbf{H}_j\widetilde{\mathbf{t}_m}, \forall l \in [1\cdots K], \forall j \in \left[1\cdots\min\left(N_T, N_{R_l}\right)\right]$. Sélectionner parmi les flux ceux qui subissent un niveau d'interférence inférieur à un niveau δ prédéfini. Si aucun flux ne satisfait la contrainte d'interférence on sélectionne tous les utilisateurs. b. Sélectionner tous les utilisateurs

9. Prendre parmi les flux sélectionnés dans 8/ celui qui présente le meilleur canal (la plus grande valeur propre). Ce couple d'utilisateur flux sera noté par $\left(u_{i,}f_i\right)$.

10. Sauvegarder le précodeur $\widetilde{\mathbf{t}_i} = \mathbf{v}_{u_i,f_i}^{\perp}$, sauvegarder le canal virtuel correspondant à ce flux $\mathbf{h}_i = \left(\mathbf{u}_{u_i,f_i}^{\perp}\right)^H\mathbf{H}_{u_i}$. Mettre à jour la matrice des canaux $\mathbf{H}_S = \left[\mathbf{H}_S^T, \mathbf{h}_i^T\right]^T$.

11. Passer à l'étape suivante Construction des précodeurs $\mathbf{t}_{u_j,f_j}; j \in [1,\cdots,i]$ en les sauvegardant.

12. Evaluer le nouveau SR:
$SR_i = \sum_{j=1}^{i} \log_2 \left(1 + \frac{\mathbf{u}_{u_j,f_j} \mathbf{H}_{u_j} \mathbf{t}_{u_j,f_j} \mathbf{t}_{u_j,f_j}^H \mathbf{H}_{u_j}^H \mathbf{u}_{u_j,f_j}^H}{\mathbf{u}_{u_j,f_j} \left(\mathbf{H}_{u_j} \sum_{m=1}^{i} \mathbf{t}_{u_m,f_m} \mathbf{t}_{u_m,f_m}^H \mathbf{H}_{u_j} + \sigma^2 \mathbf{I} \right) \mathbf{u}_{u_j,f_j}^H} \right)$.

13. Si $SR_i < SR_{i-1}$ Récupérer les précodeurs $\mathbf{t}_{u_j,f_j}$ avec $j = i-1$ et passer à l'étape 15/.

14. Si le nombre maximal de flux possible est atteint. C'est-à-dire $i = Q$ récupérer les précodeurs $\mathbf{t}_{u_j,f_j}$ avec $j = i$ et passer à 14/. Sinon incrémenter le compteur $i = i + 1$ et répéter les étapes 5/ à 14/

15. Arrêt

Le rajout de ce contrôle des flux permet d'une part d'améliorer le SR obtenu et d'autre part de réduire la complexité de l'algorithme quand le système ne supporte pas le nombre maximal de flux donné par Q.

Version DPC de la solution proposée

L'algorithme présenté ci-dessus représente une solution pratique pour générer les précodeurs maximisant le débit somme total du système tout en essayant de minimiser la partie interférence descendante (interférence subie pour les flux $\{f_i\}$ par tous les flux $\left\{f_j\right\}_{j<i}$). Cette solution, bien qu'elle élimine par construction toute interférence montante (subie pour les flux $\{f_i\}$ par tous les flux $\left\{f_j\right\}_{j>i}$), ne permet pas d'atteindre la capacité vu qu'elle est contrainte par la partie interférence résiduelle (interférence descendante). Cependant en appliquant un algorithme DPC sur ces flux sélectionnés (c'est-à-dire un algorithme qui parvient à éliminer d'une manière optimale les interférences descendantes) on est capable de déterminer les précodeurs optimaux. L'algorithme devient dans ce cas comme suit:

Algorithme: Sélection récursive ZFDPC

---

1. On calcule les SVDs des canaux de tous les utilisateurs de manière que $\mathbf{H}_k = \mathbf{U}_k \mathbf{S}_k \mathbf{V}_k^H$, $\forall i \in [1 \cdots K]$.

2. On commence par choisir le premier couple (utilisateur, flux) noté $\left(u_{1,}f_1\right)$ qui possède la valeur propre la plus grande donc le meilleur canal de transmission.

3. On reconstruit le canal virtuel correspondant à ce flux $\mathbf{h}_1 = \mathbf{u}_{u_1,f_1}^H \mathbf{H}_{u_1}$. Ce canal virtuel est alors rajouté dans une liste $\mathbf{H}_S = \mathbf{h}_1$. Sauvegarder le précodeur correspondant $\widetilde{\mathbf{t}_1} = \mathbf{v}_{u_1,f_1}$. Initialiser un compteur $i = 2$.

4. On évalue le SR ainsi obtenu $SR_1 = \log_2\left(1 + \frac{\mathbf{u}_{u_1,f_1}\mathbf{H}_{u_1}\mathbf{t}_{u_1f_1}\mathbf{t}_{u_1f_1}^H\mathbf{H}_{u_1}^H\mathbf{u}_{u_1,f_1}^H}{\mathbf{u}_{u_1,f_1}(\sigma^2\mathbf{I})\mathbf{u}_{u_1,f_1}^H}\right)$ où $\mathbf{t}_{u_1f_1}$ est donné par (1.27) Sauvegarder $\mathbf{t}_{u_1f_1}$. Initialiser un compteur $i = 2$.

5. Construire la matrice de projection sur l'espace complémentaire aux canaux des flux sélectionnés $\mathbf{H}_S$. La matrice de projection est donnée par $\mathbf{P}_i^{\perp} = \mathbf{I} - \mathbf{H}_S^H\left(\mathbf{H}_S\mathbf{H}_S^H\right)^{-1}\mathbf{H}_S$.

6. Projeter tous les canaux de tous les utilisateurs sur cet espace complémentaire .

7. Faire la SVD de tous les canaux projetés $\mathbf{H}_k^{\perp} = \mathbf{H}_k\mathbf{P}_i^{\perp}$.

8. a. Parcourir tous les utilisateurs et tous les flux en calculant le niveau d'interférence qu'ils reçoivent des flux précédemment sélectionnés : $A_{j,l} = \sum_{m=1}^{i-1}\mathbf{u}_{l,j}^{\perp}\mathbf{H}_j\widetilde{\mathbf{t}_m}, \forall l \in [1\cdots K], \forall j \in \left[1\cdots\min\left(N_T, N_{R_l}\right)\right]$. Sélectionner parmi les flux ceux qui subissent un niveau d'interférence inférieur à un niveau δ prédéfini. Si aucun flux ne satisfait la contrainte d'interférence on sélectionne tous les utilisateurs. b. Sélectionner tous les utilisateurs.

9. Prendre parmi les flux sélectionnés dans 7/ celui qui présente le meilleur canal (la plus grande valeur propre). Ce couple d'utilisateur flux sera noté par $\left(u_{i,}f_i\right)$.

10. Sauvegarder le précodeur $\widetilde{\mathbf{t}_i} = \mathbf{v}_{u_i,f_i}^{\perp}$, sauvegarder le canal virtuel correspondant à ce flux $\mathbf{h}_i = \left(\mathbf{u}_{u_i,f_i}^{\perp}\right)^H\mathbf{H}_{u_i}$. Mettre à jour la matrice des canaux $\mathbf{H}_S = \left[\mathbf{H}_S^T, \mathbf{h}_i^T\right]^T$.

11. Si le nombre maximal de flux possible est atteint. C'est-à-dire $i > \min(\sum_{k=1}^{K} N_{R_k}, N_T)$ passer à 12/. Sinon incrémenter le compteur $i = i+1$ et répéter les étapes 4/ à 10/

12. Appliquer l'algorithme DPC qui annule les interférences descendantes ce qui correspond au cas où chaque précodeur est tout seul et ne voit aucune interférence $\widehat{\mathbf{t}}_{u_i,f_i} = \widetilde{\mathbf{t}}_i, \forall i \in [1 \cdots Q]$.

13. Déterminer les puissances des canaux des flux sélectionnés. La puissance du $i$ème canal sélectionné est le terme de la diagonale de la matrice $\mathbf{\Lambda} = \mathbf{H}_S \mathbf{H}_S^H$ que l'on note λ_i. 13/ On applique un Water Filling sur ces valeurs λ_i pour avoir la distribution de puissance optimale des P_i et on les applique aux précodeurs $\mathbf{t}_{u_i,f_i} = \sqrt{P_i}\widehat{\mathbf{t}}_{u_i,f_i}, \forall i \in [1 \cdots Q]$

Afin d'évaluer les performances il suffit de calculer le débit somme total donné par $SR_i = \sum_{j=1}^{i} \log_2 \left(1 + \frac{1}{\sigma^2} \mathbf{H}_{u_j} \mathbf{t}_{u_j,f_j} \mathbf{t}_{u_j,f_j}^H \mathbf{H}_{u_j}^H\right)$.

La figure Figure 4.1 représente une représentation graphique de l'algorithme de sélection récursive se basant sur la construction de flux considérant une approche de zéro forçage DPC suivie par une seconde phase de construction de précodeurs linéaires. Figure 4.2 représente la version améliorée om l'on a intégré un contrôle d'admission des flux selon le critère du débit total maximal.

1.4.4 Simulations et résultats

Pour les simulations on considère un canal de Raleigh avec des matrices à composantes gaussiennes i.i.d. Le simulations sont réalisé sur 10^6 échantillons de canaux.

Amélioration des algorithmes CF

Dans la première figure Figure 1.26 est réalisé pour les algorithmes linéaires adaptés pour les systèmes MISO qu'on applique pour un système MIMO. Les algorithmes sont le PU-MMSE et le SJNR présentés au début de cette section. Les courbes obtenus confirment les observations théoriques et on retrouve bien des gains significatifs juste en optimisant le récepteur utilisé au niveau des terminaux. La seconde figure Figure 1.27 quant à elle,

représente l'effet d'une bonne procédure de sélection d'utilisateurs couplée avec une bonne gestion de la distribution de la puissance disponible. On voit bien à travers ces courbes qu'en cas d'absence de algorithme de sélection, le système commence à saturer à haut SNRs quand on augmente le nombre d'utilisateurs alors qu'on reste théoriquement toujours dans les limites que le système peut supporter. Les courbes avec sélection et optimisation de puissance quand à elles, offrent des débits strictements croissants en fonction du nombre d'utilisateurs et ceci pour tout SNR.

Algorithme de sélection récursif de flux

Pour illustrer les gains apportés par notre nouvel algorithme, on présente les courbes du débit total obtenu par le système pour une configuration présentant un système avec K utilisateurs ayant chacun $N_R = 4$ antennes de réception desservis par une station de base avec $N_T = 4$ antennes d'émission. Nous considérons un canal de Rayleigh complètement décorrélés pour les K utilisateurs. Nous traçons l'évolution de la somme du débit total du système à la figure Figure 1.28. Les courbes représentent les performances atteintes pour différentes valeurs de K en fonction du SNR. Les courbes "SR ZFBF NS WF" correspondant à notre algorithme sont comparées à celles obtenues par l'algorithme donné en [9] représentés par les courbes nommés "SR ZFBF SUS". La courbe "SR DPC" représente les performances obtenues par application de l'algorithme DPC donné dans [11]. Afin d'évaluer les gains que l'on peut avoir par application du DPC pour annuler les interférences descendante, on considère dans la figure Figure 1.29 un système avec $K = 100$ utilisateurs ayant chacun $N_R = 4$ antennes de réception desservis par une station de base avec $N_T = 4$ antennes d'émission. Nous considérons un canal de Rayleigh complètement décorrélés pour les K utilisateurs. Nous traçons l'évolution de la somme du débit total du système. La courbe "SR ZFDPC NS WF" correspondant à l'algorithme décrit avec une annulation DPC montre des gains supplémentaires par rapport à ceux de la figure Figure 1.28 obtenu avec un filtre ZF appliqué à l'émission. La figure Figure 1.30 représente les performances des algorithmes ZFBF SUS, ZFBF NS que nous avons présentés à la figure Figure 1.28 et on a aussi présenté les variantes avec précodage DPC des deux algorithmes. En effet, "SR ZFBF SUS" représente les performances de l'algorithme proposé par Goldsmith avec un précodage DPC où toutes les interférences sont annulées. Cet algorithme est appelé par les auteurs ZFDPC SUS. Cette courbe est à comparer avec notre variante DPC de l'algorithme DPC présentée cidessus dans la dernière partie. La courbe correspondant à cet algorithme

est notée "SR ZFDPC NS WF". Ces courbes confirment une fois encore la supériorité de l'algorithme faisant l'objet de ce chapitre, puisque pour tout SNR, le débit total moyen reste supérieur à celui obtenu par ceux de la litérature nous permettant de presque atteindre les performances annoncée par l'algorithme DPC. Par ailleurs, ces courbes montrent aussi la bonne convergence vers DPC avec de grandes valeurs de K. La figure Figure 1.31 représente l'impacte du choix de δ sur le SR de notre algorithme proposé dans ce chapitre pour un système avec $K = 100$ obtenus à 15 dB. Ceci montre qu'avec un choix judicieux de la valeur de δ, on peut améliorer encore un peu plus les performances du système. Seulement la valeur optimale du système est une fonction des paramètres du système ce qui rend difficile d'en déduire une expression analytique simple.

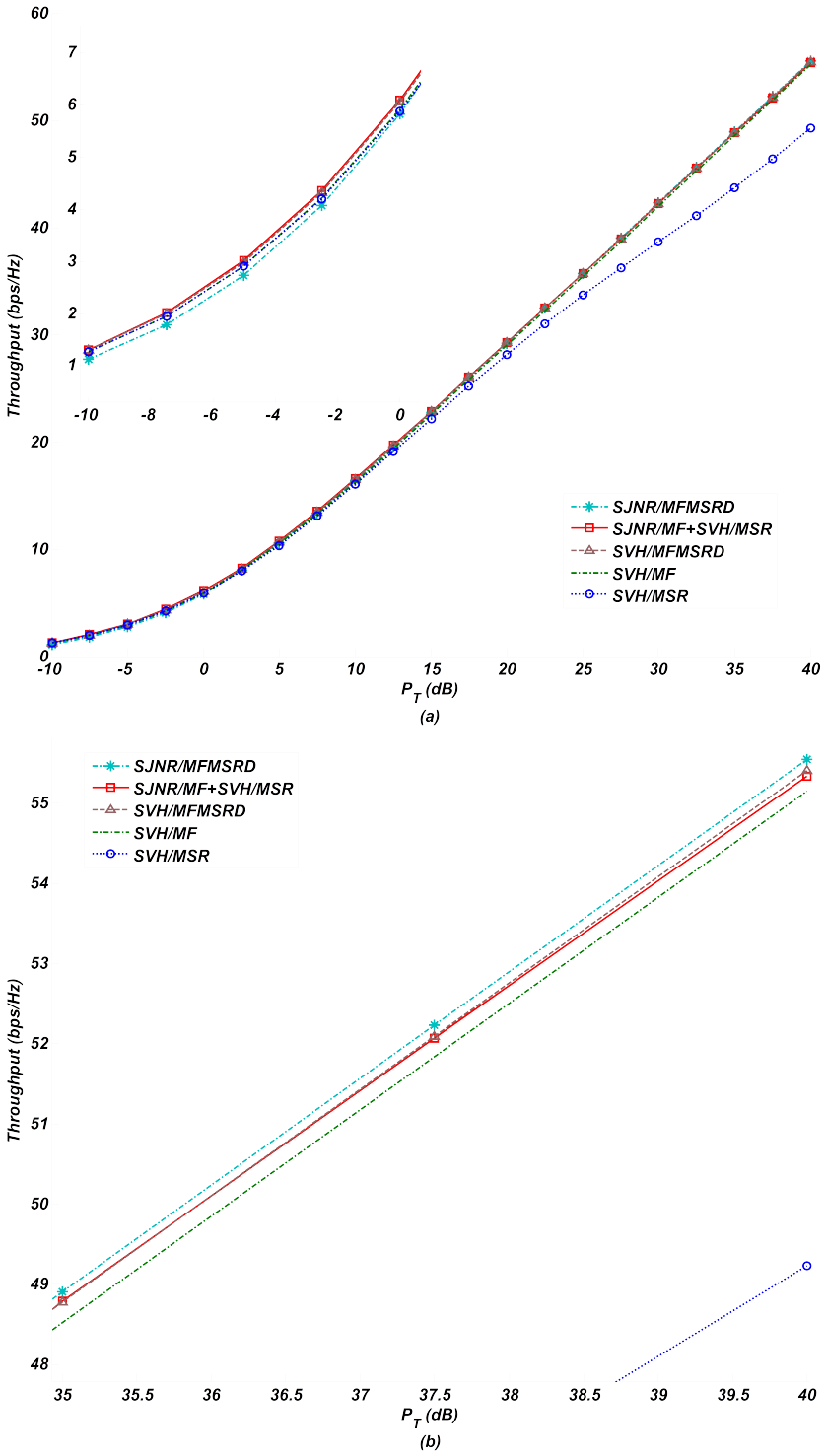


Figure 1.18: Débit pour $N_T = N_R = K = 4$.

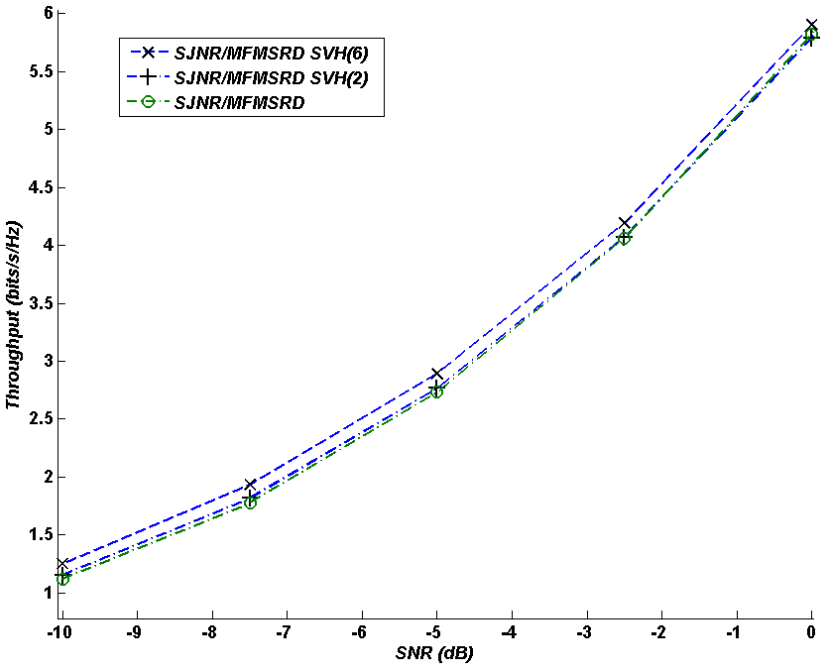


Figure 1.19: Débit pour $SJNR/MFMSR_D$ avec des itérations SVH supplémentaires

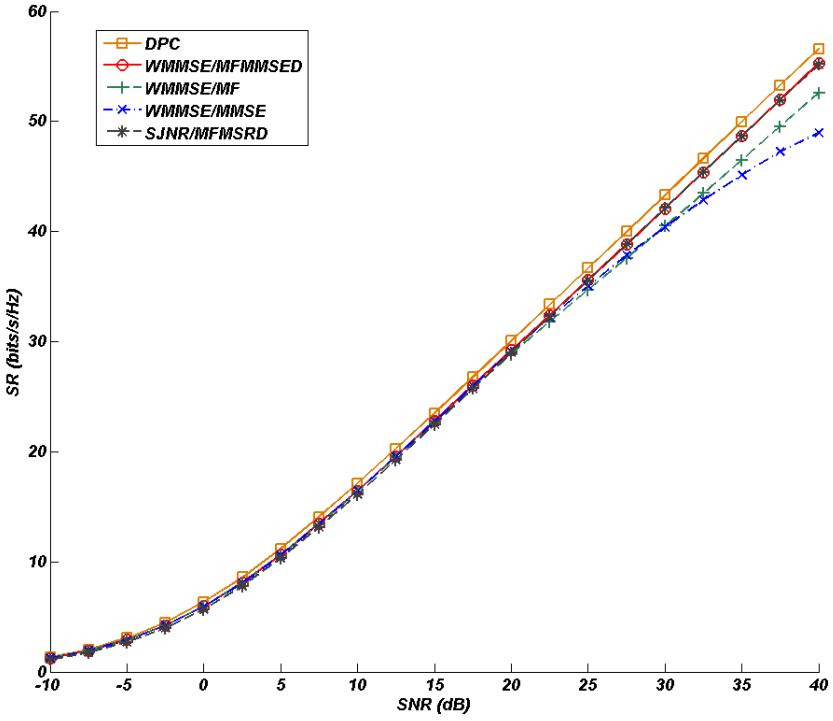


Figure 1.20: Débit pour $N_T = N_R = K = 4$.

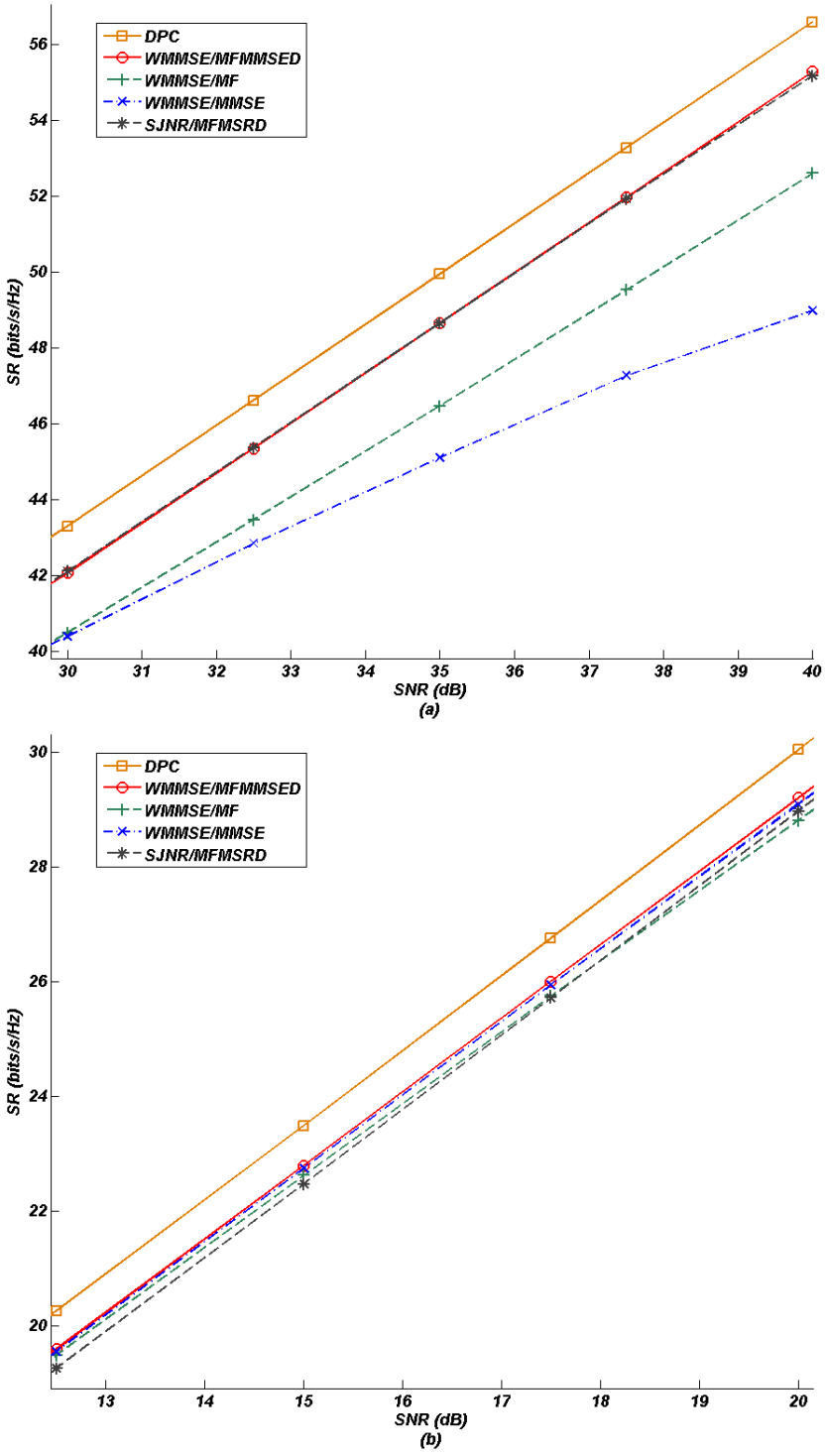


Figure 1.21: Débit pour $N_T = N_R = K = 4$.

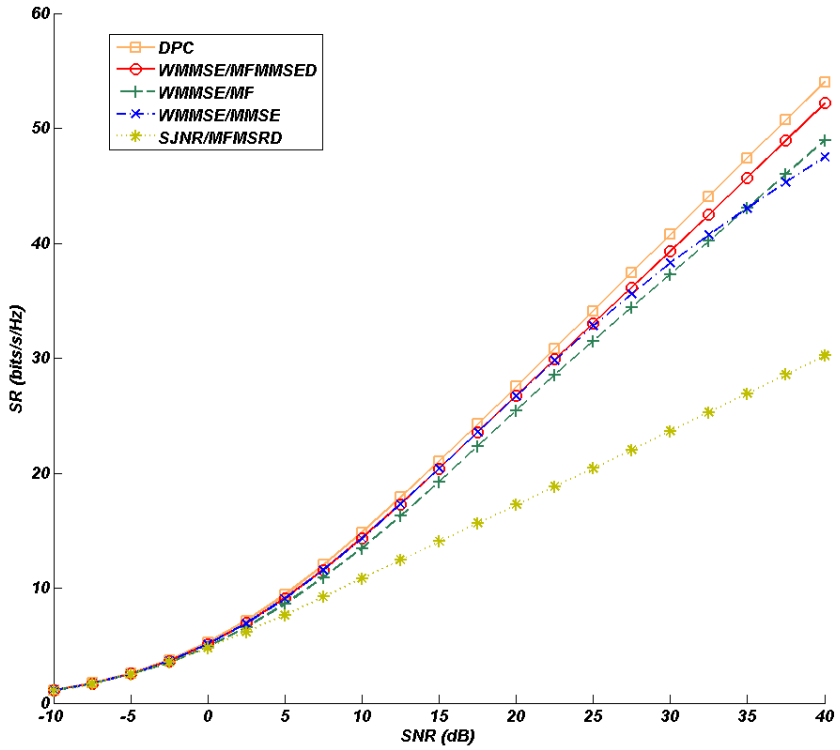


Figure 1.22: Débits pour $N_T = N_R = 4; K = 2$.

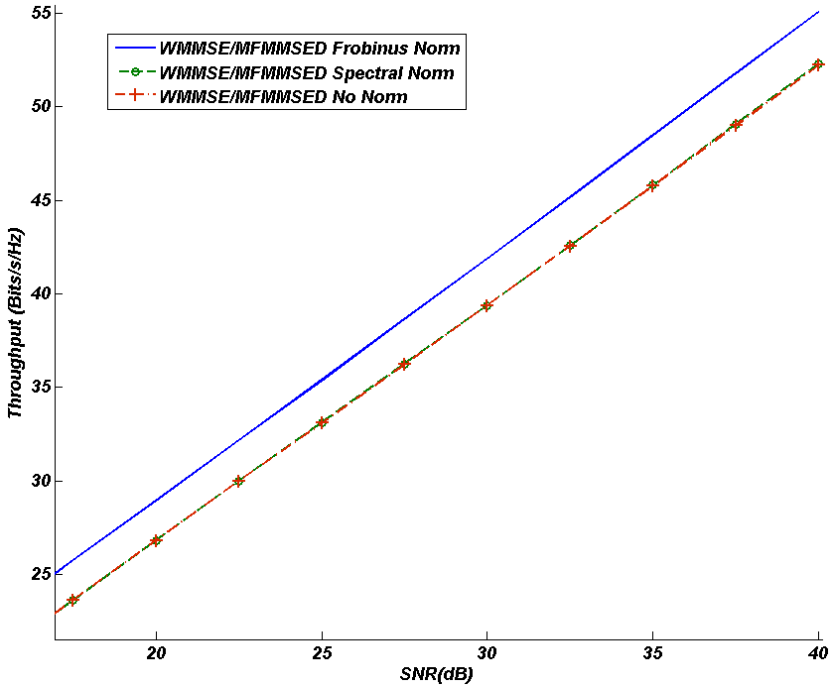


Figure 1.23: Débits pour $N_T = N_R = 4; K = 2$.

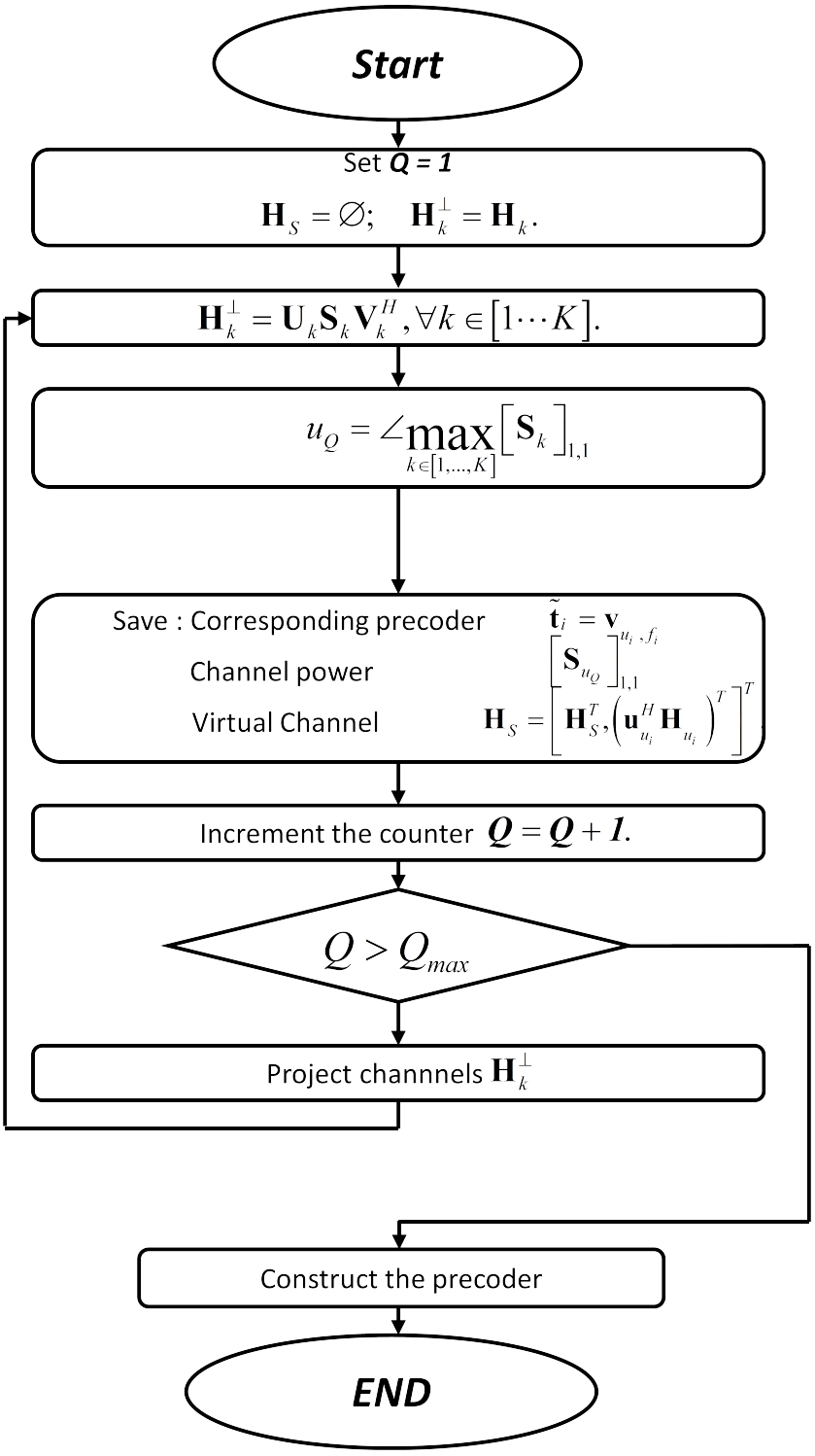


Figure 1.24: Algorithme de sélection récursif de flux.

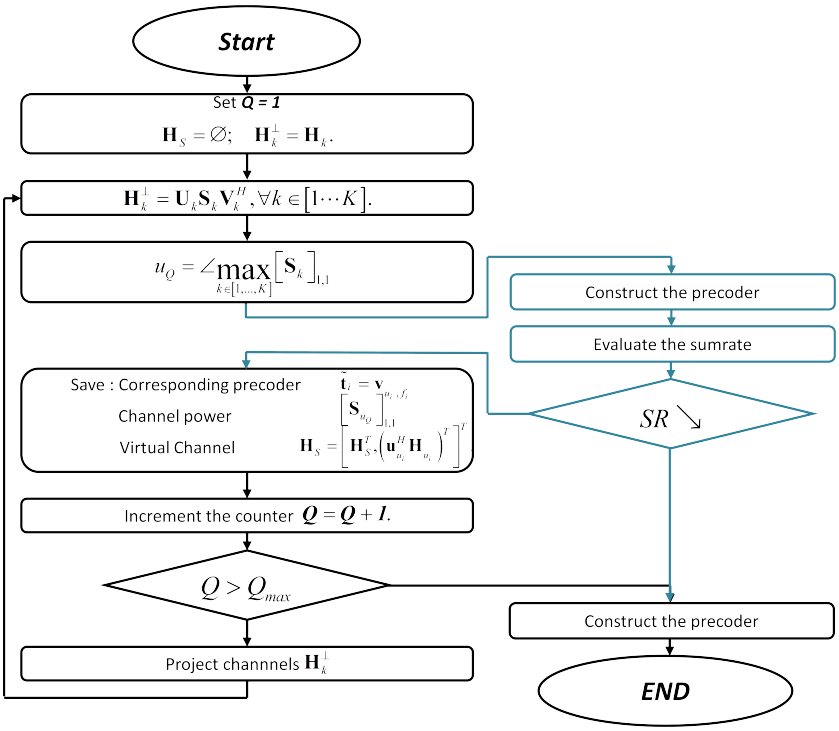


Figure 1.25: Algorithme de sélection récursif de flux avec contrôle d'admission.

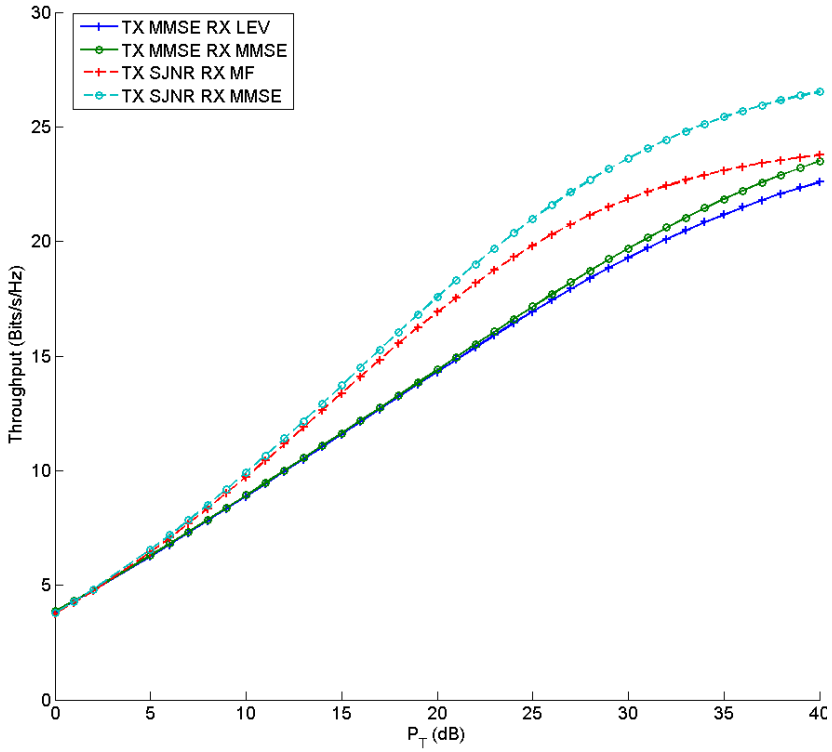


Figure 1.26: Impact du récepteur dans un système $N_T = 4$, $N_R = 2$ and $K = 3$.

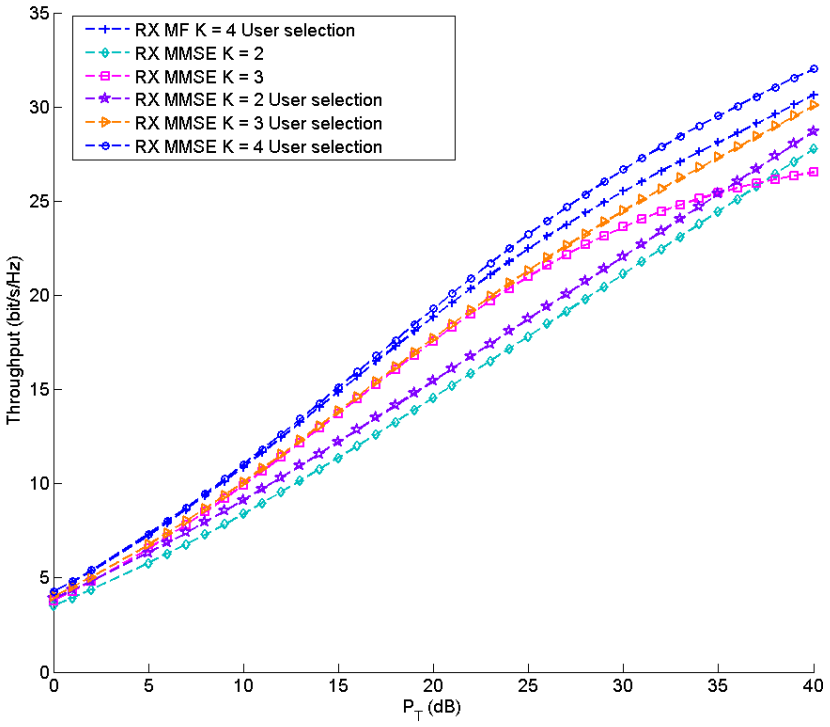


Figure 1.27: Débits avec et sans sélection d'utilisateurs avec un précodeur SJNR.

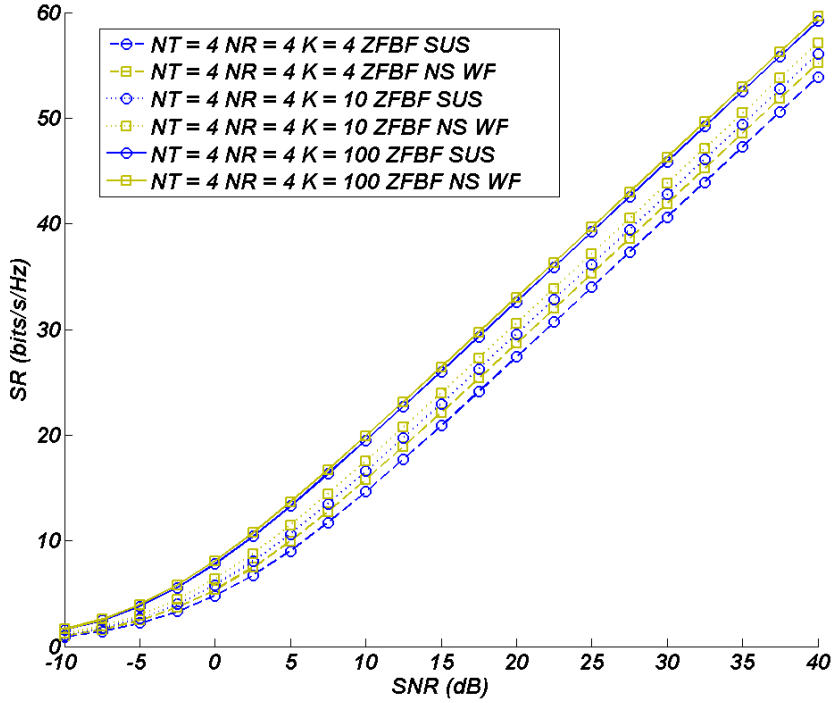


Figure 1.28: Débit total pour $K = 4, K = 10, K = 100$.

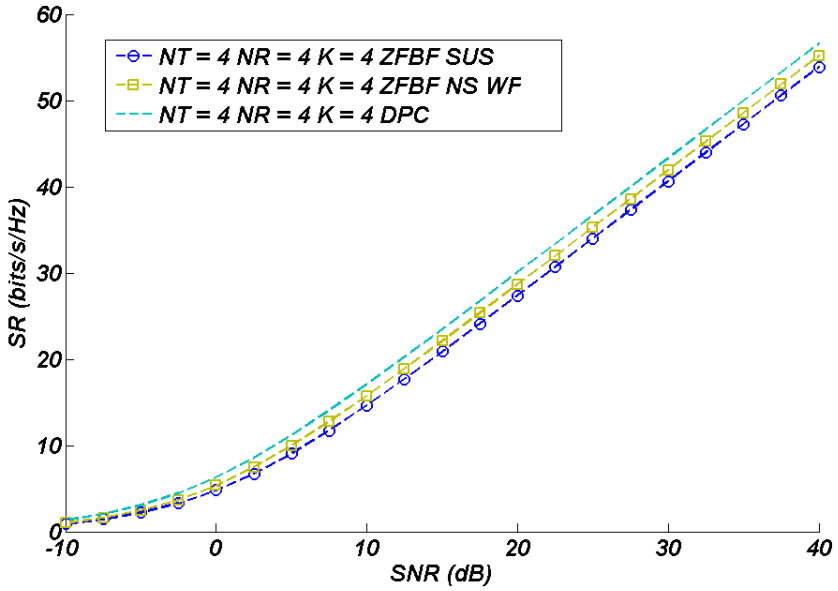


Figure 1.29: Débit total en fonction du SNR pour algorithme avec DPC.

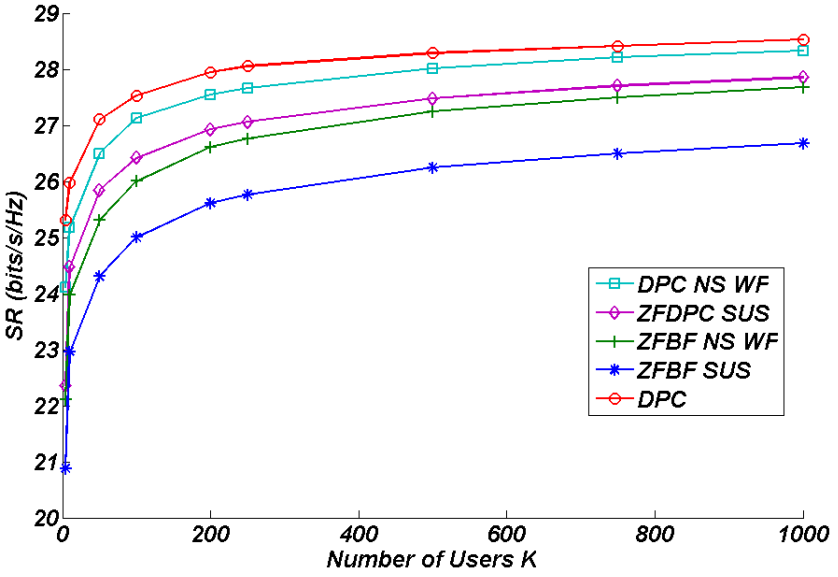


Figure 1.30: Débit total en fonction de K des algorithmes ZFDPC SUS et NS pour $SNR = 15dB$.

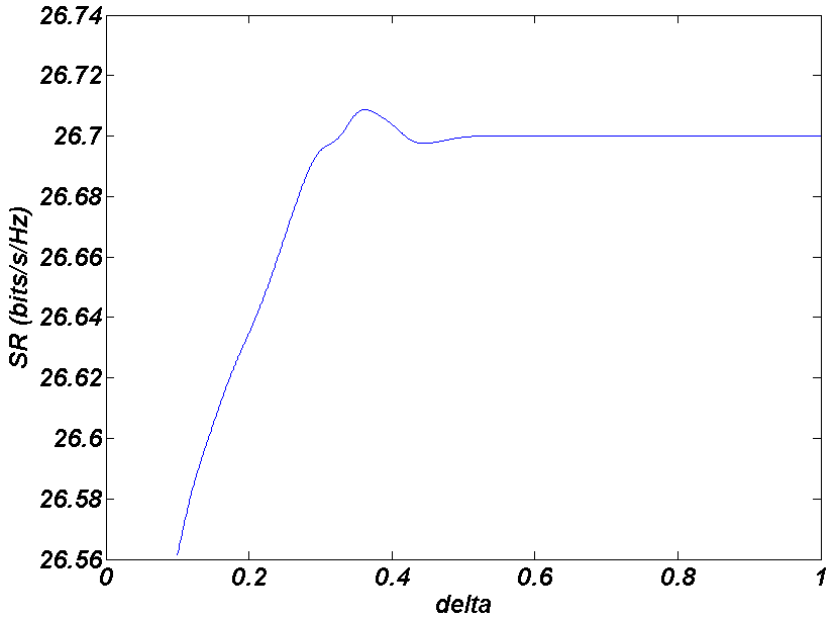


Figure 1.31: Débit total en fonction de δ pour $K = 100$

Chapter 2

Introduction

2.1 Introduction

The ever increasing growth of wireless-based services and the unstoppable demand of bandwidth, have dictated the evolution of mobile communication standards over the past years. In fact, new generation of wireless communication systems, i.e. 3^{rd} and 4^{th} generation, support packet-oriented services releasing time constraints compared to conventional voice services, but on the other hand require very high throughput. The spectral efficiency of wireless systems can be significantly improved by means of multiple-input multiple-output(MIMO) technologies, exploiting the spatial dimension in addition to the time and frequency dimensions, exploited in single-input single-output (SISO) systems. For these reasons, MIMO systems became the center of attention of communication society since they can provide reliable, high data rate communication. In point-to-point as well in point-to-multi-point MIMO systems, this spatio-temporal diversity created thanks to multiple antennas can be exploited by means of space-time coding (STC). This offers low error rates in the absence of channel state information at the transmitter (CSIT). Nevertheless, compared to the luck of CSIT where random precoding is precluded, an amount of CSIT offers extra knowledge on the channel organization and can thus be employed in order to enhance the data rates through spatial multiplexing. This is performed by transmitting independent data streams to the different users and/or to the same user.

This idea can be applied as well as in the downlink of a cellular system with multiple antennas at the base station. CSIT can be incorporated to serve users at different locations through space division multiple access (SDMA), for the purpose of system throughput maximization. But it can also be used in other systems, as the interference channel for interference alignment in order to improve the number of established links. In practical wireless communication systems, channel state information at the receiver (CSIR) can be easily obtained through training and is thus considered available for almost all communication systems. On the other hand, obtaining CSIT is a more complicated task as it requires either exploiting channel reciprocity especially for time division duplex systems (TDD) systems or using feedback transmissions from the receiver to the transmitter (generally the base station in our case). Some other techniques where the downlink CSI is extrapolated (or predicted) from the uplink CSI.

Several techniques for CSI extraction have been proposed these last few years as it has been proven that, by exploiting channel knowledge at the transmitter, CSI offers higher rates data transmission compared to the transmission using systems assuming channel knowledge available only at the receiver side. This is even more accentuated for MIMO systems as a better separation of users can be done via beam forming (spacial multiplexing). Nevertheless, to exploit this extra degree of diversity in MIMO communication systems, the usage of the precoding and decoding matrices is required and can be used either for inter-user interference elimination and thus increasing the number of served users. The second option is to exploit it for an increase of the inter-stream interference; this would allow multiple streams per user offering more flexibility to the system. In addition to the time, frequency and space resources, another important variable to be optimized for communication systems is the power distribution to be allocated to the different served users and/or distributed over the transmitted streams. Handling more intelligently the power allocation using the precoder matrices can help to improve further both efficiency and quality of the transmission.

2.2 Thesis Overview and Outline

This dissertation deals with the precoding and decoding design problem and is composed of two main parts. In the first part included in Chapter 2, iterative precoder design solutions are considered for the purpose of sum-rate maximization in multiuser MIMO scenarios. In the second part, comprising Chapter 3, the problem of maximizing the sum-rate in MIMO broadcast

channels using closed form linear precoding is addressed, providing practical designs for user scheduling and stream selection algorithms. Finally, Chapter 4 addresses the problem of ill conditioned channels. Thus a validation of some of the proposed algorithms is treated with a new proposed directionnal channel.

An abstract and introduction is provided at the beginning of each chapter.

2.2.1 MIMO Broadcast Channels

MIMO systems first appeared in point-to-point communication systems. These systems have been widely studied for example in [9, 14] showing that the performances increase linearly with the minimum of the number of transmit and receive antennas, supposing that the channel fading coefficients are statistically independent and known to the receiver. The spatial diversity offered by MIMO systems can be exploited either to reduce the error rate by transmitting the information signal over independently faded branches, or to increase the rate by transmitting independent data streams. It has been shown in [18] that there is a trade off between the spatial multiplexing gain and the diversity. This trade off is based on the existence of a number of degrees of freedom in the MIMO channel that can be exploited in order to achieve diversity or to increase the transmission rates. In the absence of CSIT, other techniques have been present in the literature capable of exploiting the spatial diversity of a MIMO system such as V-BLAST decoding technique proposed by Bell Labs in [41], D-Blast decoding approach proposed in [3, 8, 19, 23] or also the space-time precoding based on the spatial and temporal available dimensions. Several orthogonal space-time block code (O-STBC) have been proposed in the literature in order to apply this concept, as mentioned in [4, 6].

The interest in MIMO systems has naturally been oriented to more complex systems than to point-to-point MIMO communication system and applications to MIMO broadcast (BC) channels appeared during the last years. As shown in [11, 20], the capacity of BC channels can be boosted by exploiting the spatial multiplexing capability of transmit antennas allowing simultaneous transmissions to multiple users over the same bandwidth by means of SDMA. For these kinds of system, the aim is to maximize the total system throughput rather than trying to maximize the capacity of a single-user link. If a base station with N_T transmit antennas communicates with K receivers with N_R receiving antennas has perfect channel state in-

formation (CSIT), a multiplexing gain of $\min(N_T, \sum_{k=1}^{K} N_{R_k})$ can be achieved. In cellular systems, this is a setting that became of practical interest today, since multiple antennas are considered at both base station and receivers in new mobile communications standards like LTE Advanced and new WiFi generations.

It has recently been proven [22,35] that the capacity region of the MIMO broadcast channel is achieved by dirty paper coding (DPC) [1, 12, 25]. This technique is based on the fact that when the interference is known at the transmitter, it is possible to achieve the same capacity as if there were no interference. However, this technique presents two main disadvantages which limits its applicability: a high computational complexity unappropriated for real time communications and the fast changing nature of the communication channel and the need for full and perfect CSIT implying high sensitivity to channel coefficient estimation errors. In a multiuser MIMO environment, not only inter-user interference but also inter-stream interference must be taken into account for throughput maximization. In a system with N_T transmitting antennas and N_R receiving antennas for each user, the capacity region is characterized by a $\min(N_T, \sum_{k=1}^{K} N_{R_k})$ dimensional volume. The maximum achievable system throughput is the point in the capacity region that maximizes the sum of all users data rates (Sum Rate noted "SR" throughout this dissertation). Based on the available CSIT, our goal is to design systems that provide maximum sum-rates close to the sum capacity and that present reasonable complexity. A promising low complexity alternative to DPC for the downlink of MIMO single antenna systems is linear beamforming. Downlink linear beamforming, or precoding is nothing else but a linear transformation of the information stream vector $\mathbf{s}$ mapped into an N_T dimensional vector space, where N_T is the number of transmitting antennas at the base station. These techniques, although known to be suboptimal, have been shown [17, 24, 30, 34] to achieve a large portion of the DPC capacity, offering the best trade off between complexity and performance.

Suboptimal linear algorithms with lower implementation costs exist and can be divided into two main families:

1. The iterative [13, 28, 40, 45, 48].
2. The closed form solutions [21, 29, 32, 46].

Closed form solutions are known to be limited ones as the optimization process is in the best cases done in a recursive way and limits thus the

optimization process. On the other hand, iterative algorithms are able theoretically to get very close to the optimal solution; but they present two major problems, namely initialization and convergence. In fact, the initialization of an iterative algorithm is crucial for the quality of the obtained precoder and even for convergence sometimes. Solving the local convergence problem for iterative solutions remains still an open issue. Hence, how to avoid a local optimum which is not the global optimum remains an open topic in the iterative optimization procedure.

Another level of classification of precoder is the number of streams that might be offered to each user increasing by the way the offered throughput. In fact, there are precoder that can only support at maximum one stream per user even if the system is not fully charged. Such pecoder have been proposed and widely studied in [21, 27, 29, 32, 45–48]. Some multi-stream precoding solutions have nevertheless been proposed such as in [33, 49] for a closed form solution and [13, 27, 28, 40] for an iterative solution. This parameter is in reality very important in the optimization process. In fact, due to the non convexity of the optimization problem and also in some cases where the channel is very ill conditioned (such as two or multiple users with highly correlated channel matrices) it turns out to be much more efficient to neglect one of the two users and allocate more than one stream to the others increasing by the way the total SR of the system.

In the single stream per user case, only one stream can be allocated to each user and the algorithm chooses the best precoding vector for each user to maximize the total sum-rate of the system. In the multiple streams case, multiple streams can be allocated to the same user respecting two main constraints. The former is $Q_k \leq min\left(N_{R_k}, N_T\right)$ representing the number of streams allocated of user k and the latter is $Q = \sum_{k=1}^{K} Q_k \leq min\left(\sum_{k=1}^{K} N_{R_k}, N_T\right)$ representing the total number of streams allocated by the base station (BS). The allocation of these streams is done such as it maximizes the total sum-rate (SR). A further crucial point in SR maximization especially at moderate and low SNRs would be the optimization of the power distribution over the served streams. These two problems have been partially solved by applying SR-optimizing-weights for the multi-stream case [42] and by deriving the SR optimization precoder for the single stream case [29].

Throughout this dissertation, we are going to present some solutions provided for both cases of closed form and iterative precoding solutions. But let us first look at some basic techniques for linear beamforming. Linear precoding is nothing but a linear transformation of the data symbols mapped

on the different transmitting antennas through linear transformations i.e. post multiplied by matrices. The precoding can thus be presented as in figure 2.1 ,where **s** represents the data stream vector of size $Q \times 1$, **T** is the

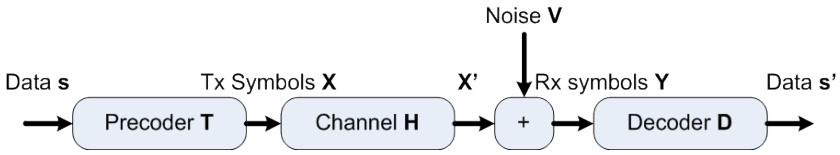


Figure 2.1: Linear precoding.

precoding matrix of size $Q \times N_T$, **X** represents the transmitted symbols, **H** is the channel matrix with $N_R \times N_T$ dimensions, **V** is the thermal noise at the receiver and **Y** is the received symbols to be decoded by decoder **D** of size $Q \times N_R$. The decoded symbols **s'** are thus expressed as:

$$\mathbf{s'} = \mathbf{D}\left(\mathbf{V} + \mathbf{HTs}\right) \tag{2.1}$$

Zero-forcing is nothing else but an inverse of the channel matrix introduced in [10] and has been extended to MISO multiuser systems in [30]. Considering a MIMO single user system as a cooperative extension of a MISO multiuser system, the expression of the zero forcing (ZF) transmitting filter is just the pseudo inverse of the channel, given by

$$\mathbf{T}_{ZF} = \sqrt{P_T}\mathbf{H}^H\left(\mathbf{H}\mathbf{H}^H\right)^{-1} \tag{2.2}$$

This simple approach could be easily extended to the multiuser MIMO system considering the overall channel matrix $\mathbf{H} = \left[\mathbf{H}_1^H, \ldots, \mathbf{H}_K^H\right]^H$ and applying the pseudo inverse to the overall matrix **H**. The sum-rate obtained by zero-forcing the channel can be expressed as [31]

$$R_{ZF} = \sum_{k=1}^{K} \det\left(\mathbf{I} + \frac{P_T}{N_T}\frac{\mathbf{H}_k\widetilde{\mathbf{H}}_k\widetilde{\mathbf{H}}_k^H\mathbf{H}_k^H}{N_0\mathbf{I} + \sum_{j=1,j\neq k}^{K}\mathbf{H}_k\widetilde{\mathbf{H}}_j\widetilde{\mathbf{H}}_j^H\mathbf{H}_k^H}\right) \tag{2.3}$$

with $\widetilde{\mathbf{H}}_j$ represents the **H** matrix where the channel matrix for user j is dropped: $\widetilde{\mathbf{H}}_j = [\mathbf{H}_1, \ldots, \mathbf{H}_{j-1}, \mathbf{H}_{j+1}, \ldots, \mathbf{H}_K]^T$.

However, a main problem with this technique is that the sum rate does not scale linearly with the number of antennas given that each stream is weighted by the corresponding singular value and that the singular values of the channel matrix are largely spread [26]. In order to overcome this

limitation, minimum mean squared error (MMSE) beamforming (also called regularized channel inversion) has been proposed. This regularizes the channel inversion and thus improves the condition of the inverse. The MMSE beamformer is given by

$$\mathbf{T}_{MMSE} = \beta \mathbf{H}^H \left(\alpha \mathbf{I} + \mathbf{H}\mathbf{H}^H \right)^{-1} \tag{2.4}$$

where β is chosen such that $tr(\mathbf{T}_{MMSE}\mathbf{T}_{MMSE}^H) = P_T$. The value of the coefficient α is $\frac{N_T N_0}{P_T}$ maximizing the SINR for each receiver when K becomes large enough, as shown in [26].

Another linear beamforming technique is block diagonalization ZF [21, 43]. This technique is based on null space beam extraction where, the SVDs of all users' channels is performed and a waterfilling is applied over the selected vectors corresponding to the selected singular values or nominal directions. This technique nevertheless presents a huge constraint as the number of receiving antennas $N_R = \sum_{k=1}^{K} N_{R_k}$ must be lower than the total number of allowable beams given by the number of transmitting antennas N_T. In this case, the recursive solution proposed in [26] is given in algorithm ***Algorithm 1***

Algorithm 1 MU-MIMO ZF Block Diagonalization

1. Compute the right null space $\widetilde{\mathbf{V}}_k^{\perp}$ of $\widetilde{\mathbf{H}}_k = \left[\mathbf{H}_1^T, \ldots, \mathbf{H}_{k-1}^T, \mathbf{H}_{k+1}^T, \ldots, \mathbf{H}_K^T\right]^T$ for $k = 1, ..., K$.
2. Compute the SVD of $\mathbf{H}_k \widetilde{\mathbf{V}}_k^{\perp}$
3. Apply waterfilling on the obtained singular values.
4. Apply the waterfilling result to the right hand singular vectors.

The obtained sum-rate that could be achieved with this scheme is

$$R_{BD} = \log_2 \det \left(\mathbf{I} + \frac{P_T}{N_T N_0 \left\| \mathbf{H}^{\dagger} \right\|^2} \mathbf{I} \right) \tag{2.5}$$

$$= Q \log_2 \left(1 + \frac{P_T}{Q N_0 \sum_{i=1}^{Q \leq N_T} \lambda_{i,\mathbf{H}^{\dagger}}^{-1}} \right) \tag{2.6}$$

where Q here denotes the maximal number of served streams representing the rank of matrix $\mathbf{H}$.

The last technique that we are going to introduce in this chapter is the time division multiple access (TDMA) approach for multiuser MIMO system. This approach consists in dividing the time slot among available users and allocating for each user an exclusive period. The distribution of time among users can be done in two different manners. The first possible way is to consider round-robin scheduling with a circulant constant distribution rotating from one user to the other one or any other fixed allocation. We call T_k the period allocated to user k representing a fraction of the system period T, i.e.

$$T = \sum_{k=1}^{K} T_k \tag{2.7}$$

This way of distributing time imposes a transmission for a user k even if he/she presents a bad channel. The obtained sum-rate can be expressed as the weighted sum rate of a single user MIMO system. The weights are computed as the inverse function of the time sharing. The achievable rates for a single user MIMO system with perfect CSIT that can practically be achieved using a waterfilling approach is given by the the sum-rate expression:

$$SR_{TDMA} = \sum_{k=1}^{K} W_k \log_2 \det\left(\mathbf{I} + \frac{1}{N_0}\mathbf{H}_k\mathbf{Q}_k\mathbf{H}_k^H\right) \tag{2.8}$$

$$= \sum_{k=1}^{K} \frac{T_k}{T} \log_2 \det\left(\mathbf{I} + \frac{1}{N_0}\mathbf{H}_k\mathbf{Q}_k\mathbf{H}_k^H\right) \tag{2.9}$$

where $\mathbf{Q}_k$ the covariance matrix being along the eigenvectors of the matrix $\mathbf{H}_k\mathbf{H}_k^H$ and the eigenvalues obtained by applying a water filling procedure [9]. The second manner to distribute time resources consists in selecting among the available users the one presenting the best channel matrix, i.e the one having the highest possible rate.

On average over the channel realizations, this would yield to a fair and equal average distribution. The sum rate that can be expected from this kind of approach converges to a single user MIMO system given by

$$C_{TDMA} = \max_{\mathbf{Q}\succcurlyeq 0, tr(\mathbf{Q})\leqslant P_T} \log_2 \det\left(\mathbf{I} + \frac{1}{N_0}\mathbf{H}\mathbf{Q}\mathbf{H}_k\right) \tag{2.10}$$

2.2.2 System Model and challenges

Let us consider a multi-user MIMO communication system with N_T transmitting antennas at the base station and K users with N_{R_k} receive antennas

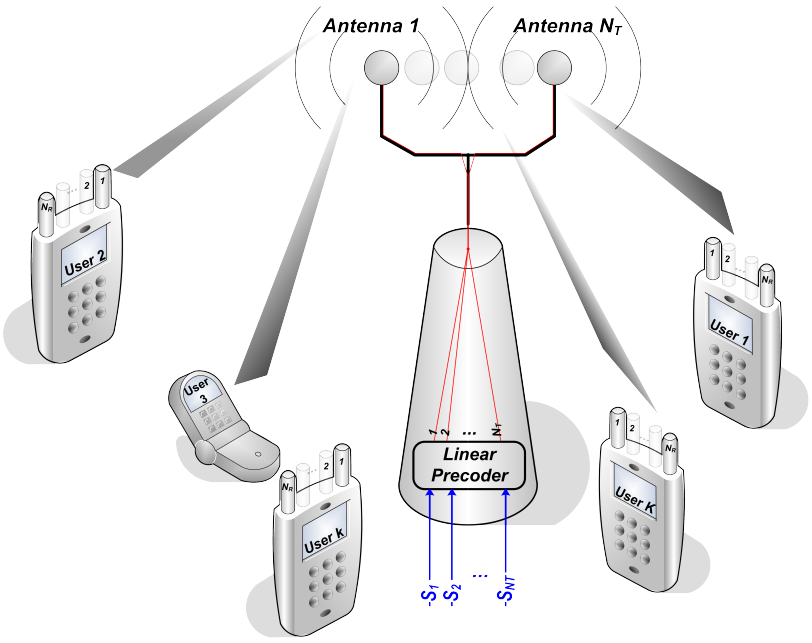


Figure 2.2: MU-MIMO system model.

for user k. Such a system is represented in Figure 2.2. We assume that the base station has a perfect knowledge of the channel state information (CSI) of all K users. Normally with perfect CSIT, the DPC algorithm becomes applicable and could be used for precoder design. Nevertheless, an extra time constraint separating the CSI acquisition and data transmission must be taken into consideration. Moreover, throughout this dissertation it will be shown that perfect SCIT is not really necessary as the linear proposed techniques are robust enough to support some channel estimation errors. Let s_k be a $Q_k \times 1$ vector representing the transmitted data symbols for user k where Q_k is the number of transmitted streams for the user k. This dissertation deals with both single and multiple streams per user. Thus, we have $Q_k = 1$ and $Q_k \leq min(N_{R_k}, N_T)$ respectively. The total number of streams must not exceed the maximum number that can be supported by the system, defined as $Q \leq min\left(\sum_{k=1}^{K} N_{R_k}, N_T\right)$

The total transmit power at the base station is assumed to be constant and equal to P_T . The noise variance is denoted as N_0. For the channel part, $\mathbf{H}_k$ denotes the MIMO channel for user k which is a $N_{R_k} \times N_T$ matrix. Each element composing the channel matrix is considered to be a complex Gaussian random variable with unit variance and zero mean. A detailed representation of the system is given in Figure 2.3
For user $k \in \{1, \dots K\}$, the received signal will be given by

$$\mathbf{Y}_k = \sum_{i=1}^{K} \mathbf{D}_k \mathbf{H}_k \mathbf{T}_i \mathbf{s}_i + \mathbf{V}_k \tag{2.11}$$

$$= \underbrace{\mathbf{D}_k \mathbf{H}_k \mathbf{T}_k \mathbf{s}_k}_{UsefulSignal} + \underbrace{\sum_{i=1, i \neq k}^{K} \mathbf{D}_k \mathbf{H}_k \mathbf{T}_i \mathbf{s}_i}_{Inter-UserInterference} + \underbrace{\mathbf{V}_k}_{Noise} \tag{2.12}$$

As we previously saw, intensive work has been done in the case of a multiuser MIMO system considering a single receiving antenna for each user. This approach simplifies the problem as everything becomes scalar at the reception and the optimization concentrates just on finding a precoding vector and a power allocation for each user i.e. stream.

Throughout this dissertation, we address the more realistic multiuser MIMO case as multiple antennas can also exist at the reception. This increases the complexity of the optimization procedure as we add extra dimensions. This opens among other things the opportunity for multiple streams per user. The different streams can not be treated in an equal way. In fact, streams aimed at the same user have fewer constraints among each other as cooperation is possible whereas streams distributed for different users should be treated as interference to each other (no cooperation is available). The second degree of difficulty introduced here is that the optimization process should not provide us precoding vectors, but precoding matrices. This means that we are going to deal with multidimensional optimization instead of a one dimensional, as it was the case.

The increase of the number of receiving antennas increases also the well known problem of local maximums. In fact, the previously centralized roles allocation for interference cancellation becomes now distributed among transmitter (Base Station) and receivers. The extra receiving dimension added at the reception can thus not only increase the multiplexing gain and data throughput rates, but also plays a fundamental role in the allocation of functionality. This problem is even more visible in the interference channel case. Therefore the questions that we tried to answer here are:

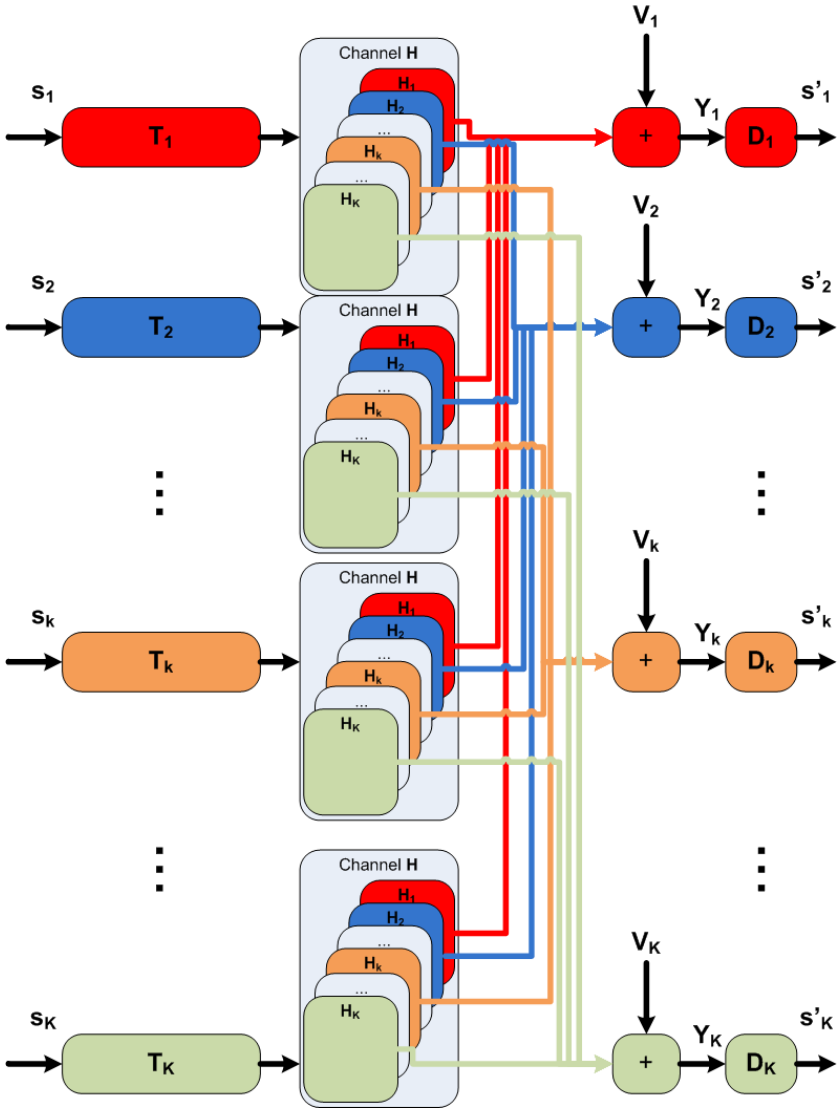


Figure 2.3: MU-MIMO system model.

- Does an increase in the number of receiving antennas increase the achievable sum rates ?

- How to optimize and determine the precoding matrices to maximize the system throughput ?

- How to select the streams to serve for sum rate maximization ?

- What power distribution should be adopted among selected streams ?

- Can we build low complexity algorithm for precoder and decoder design ?

- What would be the impact of channel configuration on the performances of the proposed algorithms and on the achievable sum rates?

2.3 Contributions

We propose some new algorithms for joint design of precoder and decoder in the case of broadcast multi user MIMO systems. Some of the algorithms are based on some existing ones and extended via some criteria to support multiuser MIMO channels with multiple receiving antennas.

We also propose some solutions for one of the main problems in iterative algorithms. Indeed, these algorithms are known to converge to some local optimum that could be very far from the maximum achievable rates. This problem can be explained by the non convexity of the optimized cost function and the role attribution to the precoder and decoders of different users. The optimization process is therefore sensitive to the initialization of the algorithm; for each starting point, extra constraints are added by imposing the convergence toward one of the local maximum defined by that starting point. The optimization procedure can in fact be seen as a tree decision process where, at each iteration, some directions are chosen and lead to one of the maximum. Another easy way of seeing the source of this problem would be to consider the cascade of the precoder, channel and decoder such

as :

$$\begin{pmatrix} \mathbf{D}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{D}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \mathbf{D}_K \end{pmatrix} \begin{pmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \\ \vdots \\ \mathbf{H}_K \end{pmatrix} \begin{pmatrix} \mathbf{T}_1 & \mathbf{T}_2 & \cdots & \mathbf{T}_K \end{pmatrix} \quad (2.13)$$

$$= \begin{pmatrix} \mathbf{D}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{D}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \mathbf{D}_K \end{pmatrix} \begin{pmatrix} \mathbf{H}_1\mathbf{T}_1 & \mathbf{H}_1\mathbf{T}_2 & \cdots & \mathbf{H}_1\mathbf{T}_K \\ \mathbf{H}_2\mathbf{T}_1 & \mathbf{H}_2\mathbf{T}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{H}_K\mathbf{T}_1 & \cdots & \cdots & \mathbf{H}_K\mathbf{T}_K \end{pmatrix} \quad (2.14)$$

In fact, for each selected stream, the interference that it receives can either be canceled out by the precoding vectors or by the receiving vectors. This offers a huge choice of precoder and decoder minimizing the interference and leads naturally to a local maximum. These observations demonstrate that the local optimum will depend directly on the first treated stream.

This dissertation presents two methods to minimize the impact of this problem in the iterative algorithms. The first one is what we call the double-iterative algorithm, based on the combination of two iterative algorithms in a wise manner. The second approach is the deterministic annealing, where, by means of iterative procedure, the starting point is considered as it represents a well known optimal solution. The optimization procedure is then launched by iteratively adding streams and optimizing precoding ad decoding matrices.

The second main contribution in this thesis is the proposal of new closed form recursive precoder optimization procedure based on recursive SVD decompositions. The method consists in a DPC zero forcing approach where the streams are selected recursively to optimize the system throughput. The obtained directions can then be used to construct the precoding and the corresponding decoding matrices. Although this approach is obviously suboptimal, the obtained performances are very close to those obtainable with a DPC precoding technique. The suboptimality comes from various parameters. In fact, the streams that are selected at recursion i depend entirely on those that have been chosen through all previous recursions and thus the starting point will be a determinant factor in the selection process. A second observation consists in an example illustrated in Figure 2.4 where the selection of the first stream V_1 induces discarding two streams V_3 and V_4 in favor of V_2 that would have been a better choice. Moreover, these determined directions can also be used to construct some beam forming pre-

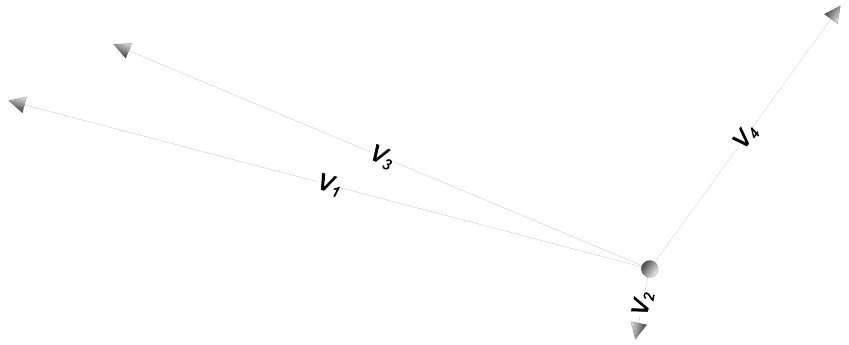


Figure 2.4: Recursive selection problem.

coding matrices, hence simplifying the normal iterative heavy approach.

In Chapter 3 we deal with precoder and decoder design for the multiuser MIMO broadcast channel. These algorithms are mainly iterative algorithms that have shown good performances for the multiuser MIMO case, where users dispose of one receiving antennas. Some existing iterative and closed form algorithms presented as a suboptimal solution, if compared to dirty-paper coding (DPC), have been proposed in the literature. In this chapter we present some new iterative precoder design algorithm based on some closed form existing solutions, like the Hassibi [29] algorithm, the SJNR [27] algorithm, and the PU-MMSE [32]. These iterative solutions are constructed on a sum rate maximization criteria. Moreover, we discuss the problem faced by iterative solutions namely the local maximum convergence problem. In fact, the proposed iterative algorithms, as well as the existing ones, are known to suffer from convergence problems [21]. The presented solution is based on a wise combination of the iterative algorithms and is shown to perform well for single stream and multi stream as well as for multi stream precoder optimization. The last part of Chapter 3 is based on a second approach known as deterministic annealing. The optimization starts from a well known optimal precoding at very low SNR. The SNR is then iteratively incremented through the optimization process. Streams are added one by one if enough power is available to switch them on. At each step, an iterative optimization procedure is performed to be able to track the global optimum.

In chapter 4 we present a second family of precoding technique i.e. the closed form precoding (CF) technique. The solutions designed in this case for sum rate maximization are generally based on a formula or on recursive algorithm. This means that the precoder and receiver are computed only once for each user. This makes these algorithms very fast and low resource

consuming. On the other hand, they offer poor performances.

Throughout Chapter 4, we propose some novel CF precoder design algorithms for the MU-MIMO BC. In the first part, we propose some enhancements to some of the existing closed form solutions, namely the SJNR and PU-MMSE precoder. The idea is essentially to optimize furthermore these algorithms, especially through user selection and power allocation methods. In the second part, we present a new zero forcing DPC precoding technique based on successive SVD decompositions. This proposed algorithm constructs stream-wise the precoder in a recursive manner. It contains two complementary steps: The first step consists in a recursive selection of the best available streams to minimize interference among users. This selection procedure is based on a null space criteria combined with the eigenvalues for each available stream. In the second step, the precoder are calculated according to the chosen precoding technique with eventually a power distribution optimization.

In the next chapter, Chapter 5 we investigate the performance achieved by some SDMA (Space Division Multiple Access) access techniques in a broadcast channel configuration. The study includes the two basic linear precoding schemes present in the literature, namely the CF (Closed Form) and the Iterative precoding techniques. These techniques have already been widely studied for the i.i.d Rayleigh channel. In fact, these precoding methods are essentially based on the exploitation of diversity created by multipaths present in such a channel to enhance the beamforming design and consequently increase the SINR (Signal to Interference and Noise Ratio) for each user. In this work, the impact of a reduction of these degrees of freedom has been investigated through simulations under other channel configurations. In other words we investigate the impact of channel diversity effect. The obtained results show that these precoding techniques remain efficient only for systems offering high number of degrees of freedom and fail when a direct LOS (Line Of Sight) is present, hence limiting the achievable Sum Rate (SR) of the system. The obtained results have also been compared to the TDMA (Time Division Multiple Access) communication mode.

The last chapter, Chapter 5 presents a summary of the subjects and results obtained in this work.

Chapter 3

Multiuser MIMO Iterative Precoding

3.1 Introduction

This chapter deals with precoder and decoder design for the multiuser MIMO broadcast channel. These algorithms are mainly iterative algorithms that have shown better performances for the multiuser MIMO case, where users dispose of one receiving antennas. The existing iterative algorithms presented as a suboptimal solution compared to dirty-paper coding (DPC) have been proposed in [13, 28, 29, 40]. Some other algorithms, called closed form solutions, have also been proposed as in [21, 32]. In this chapter we present some new iterative precoder design algorithm based on some closed form existing solutions like the Hassibi algorithm, the SJNR algorithm, and the PU-MMSE. These iterative solutions are constructed on a sum rate maximization criteria. In a second part of the chapter we discuss the problem of sub optimality due to the local maximum problem. In fact iterative algorithms are known to suffer from convergence problems [21]. The presented solution works fine for single stream and multi stream as well as for multi stream precoder optimization. The last part of this chapter is based on a second approach known as deterministic annealing, where the optimization starts from a well known optimal precoding at very low SNR. The optimization is then done iteratively by increasing the SNR and adding one by one

the streams when there is enough power to switch them on. At each step, an iterative optimization procedure is performed to be able to track the global optimum.

3.2 Lost in Iterations

To grab some insights in the difficulty of the challenge facing the MU-MIMO systems we consider two simple and extreme cases. In fact, it is well known from a mathematical point of view that the sum rate expression is non convex. The quantities to be optimized are the precoder and the power allocation over the served direction. To simplify, we must define the transmission directions and the weights that should be affected to those directions.

Nevertheless, these two parameters are correlated with each other and with those from the other users. This means that changing either the direction or the power of a stream is affecting all the others and may not only completely change both the directions and the powers for the other streams but even change the concerned/selected streams. These problems where already known in simple MISO communication systems where multiple terminals have just one receiving antenna. The solutions proposed in the literature are able to solve these problems either by splitting the problem into two and thus treat each parameter as an independent one or by performing joint optimization between them, optimizing sequentially these two parameters until convergence is achieved.

With the advances in signal processing hardware and antenna technology, multiple antennas have started to appear also on the mobile terminals. These antennas where at the beginning used just to increase the receiving diversity and algorithms combining different received signals on the different antennas like MRC (Maximum Ratio Combiner). A second important step was to consider the possibility of transmitting multiple streams for a considered user increasing the achievable data rates. These multiplexing gains where especially considered for single user MIMO systems.

Extending these systems to multi-user MIMO systems with multiple transmission and reception antennas, induced a higher level of complexity of the problem. In fact, the presence of multiple antennas N_{R_k} at the reception, introduces an N_{R_k} dimensional space that was for a long period neglected essentially due to the extension of MISO algorithms. The real increase in complexity appeared when algorithms maximizing the total sum rate started to appear. It is clear trough [29,42] that the optimization parameter became more correlated and espatially that a new optimization parameter appeared,

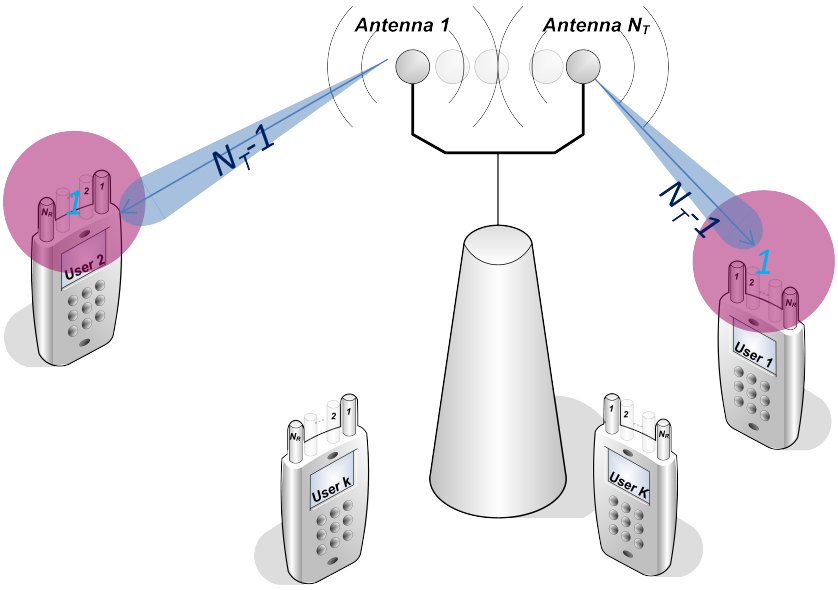


Figure 3.1: Multiuser MIMO optimization startegy 1.

namely the receiver.

As illustrated in Figures 3.1 and 3.2, there are now multiple manners to allocate the antennas at the transmission and reception. In fact, the first figure, Figure 3.1, shows a possible configuration where the receiver acts as single antenna receiver i.e. completely passive. This case is similar to the MISO case where all the interference management is done by the transmitter. So the transmitter or base station in the case of a broadcast channel should be able of canceling $N_T - 1$ interferer to be able transmit properly the data to the desired user. This case is among others presented in the recursive selection procedure that we propose in Chapter 5.

Figure 3.2 on the other hand, present the other extreme where the receiving antennas at the terminals are fully exploited to cancel out the interference generated in their direction. The maximum would be $N_R - 1$ streams as the last one constitutes the desired stream. In this case, the constraint on the transmitter (BS) is relaxed and only the remaining interference should be canceled while transmitting. This means that $N_T - N_R$ interference beams should still be killed to ensure an interference-free transmission.

$$\prod_{l=1}^{\mathrm{N}_T} \left(\sum_{i=0}^{N_R-1} \frac{(\mathrm{N}_T - 1)!}{k!(\mathrm{N}_T - 1 - l)!} \right) \tag{3.1}$$

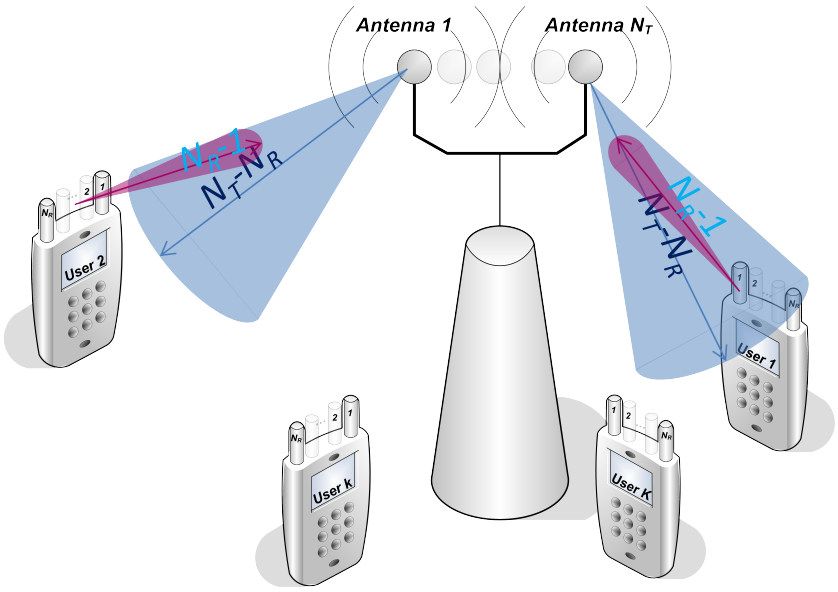
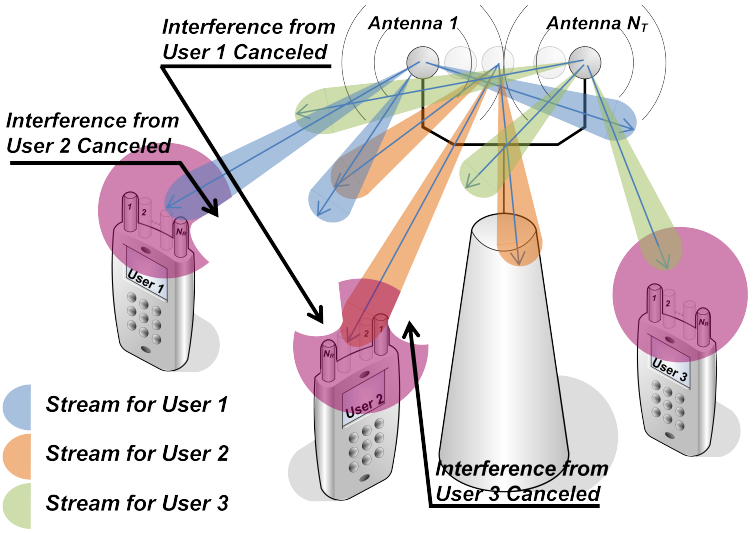


Figure 3.2: Multiuser MIMO optimization startegy 2.

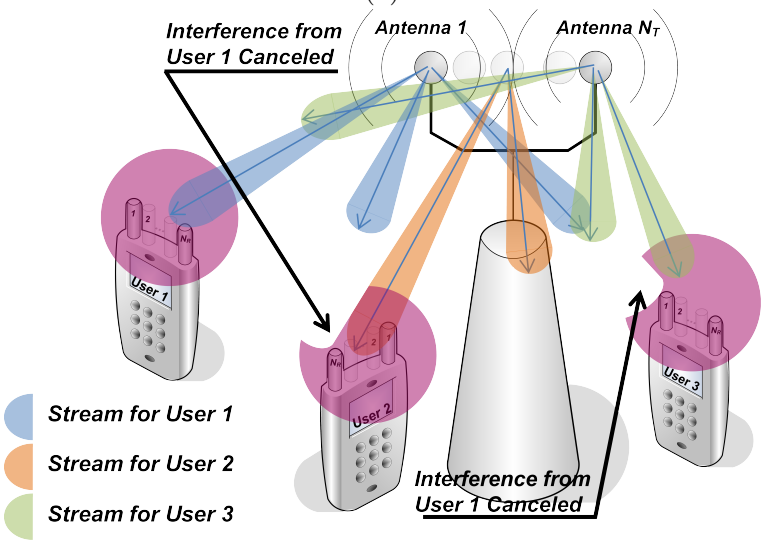
These two extreme cases present two optimal cases depending on the role that should be affected to each transmission/receiving unit. Some intermediate solutions could easily be imagined where the roles are more flexibly distributed to the different parts. For example if we consider that the receiver is going to zero force l interfering streams $l \in \{1, .., N_R - 1\}$ then the transmitter should take care of the $N_T - l - 1$ remaining interferer. The total number of possible combinations rapidly explodes with the system configuration as we obtain

This first illustrated problem obviously creates a huge number of potentially optimized precoder and power allocations, and thus explains the existence of a big number of local maximum. Unfortunately, this is not the only source of non convexity of the considered problem. In fact, during the optimization process, both receiver and precoder must be optimized to meet the fixed objectives. But, fixing for example just the number of directions or beams to be canceled out on each receiver, does not specify which ones should be dealt with at a given mobile station. In this increases even more the number of possible solutions depending on the starting point. In iterative or recursive solutions, it is obvious that the starting point should be fixed either randomly or considering some criteria. The problem is that,

once the starting point is set, the system is going to converge to a logical maximum but probably passing by a better maximum as illustrated in Figure 2.4. This phenomenon illustrated in Figure 3.3 is even more obvious in the interference channel case.



(a)



(b)

Figure 3.3: Stream role affectations: (a) First case scenario; (b)Second case scenario.

All these problems explain the difficulties encountered by the optimization of precoder in the MU-MIMO system and the important convergence issue that we try to minimize throughout this thesis.

3.3 Single Stream Iterative Optimization Algorithms

In this section we propose two iterative algorithms for the case of single stream per user. The first algorithm is based on SJNR and optimizes the sum-rate jointly with an MSR receiver. The second one, presented in subsection 3.3.4 is the so called SVH/MSR algorithm based on a reduction of the MIMO channel into a MISO channel via a virtual channel construction; then it applies an optimal MU-MISO optimization procedure proposed in [29]. This second optimization algorithm presents very good performances at low transmit powers.

3.3.1 SJNR Precoder

The objective is to design the transmit filters $\mathbf{t}_k$ under the total transmit power constraint $\sum_{k=1}^{K} P_k = P_T$. Here $P_k = tr\left(\mathbf{t}_k \mathbf{t}_k^H\right)$ denotes the transmitted power for user k.

We consider the Signal to Jamming plus Noise Ratio (SJNR) defined as the signal power of user k over the noise plus total power of interference caused by the user k and received by the other receivers. This has been introduced in [27] and is given by (3.2)

$$SJNR_k = \frac{\mathbf{t}_k^H \mathbf{H}_k^H \mathbf{H}_k \mathbf{t}_k}{\mathbf{t}_k^H \sum\limits_{j=1, j\neq k}^{K} \mathbf{H}_j^H \mathbf{H}_j \mathbf{t}_k + N_0 \mathbf{I}} \tag{3.2}$$

A solution to maximize the SJNR for the different users has been proposed in [27]. The authors demonstrate that the generalized eigenvalue of the SJNR expression is the optimal solution. The precoder for user k is therefore given by the expression of equation (3.3).

$$\mathbf{t}_k = \sqrt{P_k}\zeta_m \left[\left(\sum_{j=1, j\neq k}^{K} \mathbf{H}_j^H \mathbf{H}_j + \frac{N_0}{P_k}\mathbf{I} \right)^{-1} \mathbf{H}_k^H \mathbf{H}_k \right] \tag{3.3}$$

where $\zeta_m(\mathbf{X})$ represents the largest eigenvector of $\mathbf{X}$. The largest eigenvector is defined as the eigenvector corresponding to the largest eigenvalue of $\mathbf{X}$.

3.3.2 Receivers Design

Different structures have been proposed for the receiver design for MIMO systems. Among the existing proposed solutions there is the matched filter (MF) as proposed in [27] and given by

$$\mathbf{d}_{MF,k} = \frac{(\mathbf{H}_k \mathbf{t}_k)^H}{\|\mathbf{H}_k \mathbf{t}_k\|}, \tag{3.4}$$

where $\|\mathbf{x}\|$ is the norm of vector $\mathbf{x}$.

Another receiving structure that maximizes the total sum rate of the system is the one derived in [48] and noted $\mathbf{d}_{MSR,k}$.

Indeed, maximizing the total sum rate for a one stream per user system reduces the problem to jointly maximizing all the throughputs R_k for $k \in \{1, \ldots, K\}$

$$R_k = log_2 \left(1 + \frac{\mathbf{d}_k \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H \mathbf{d}_k^H}{\mathbf{d}_k \left(\boldsymbol{\Upsilon}_k + N_0 \mathbf{I} \right) \mathbf{d}_k^H} \right) \tag{3.5}$$

where $\boldsymbol{\Upsilon}_k = \mathbf{H}_k \sum_{j=1, j \neq k}^{K} \mathbf{t}_j \mathbf{t}_j^H \mathbf{H}_k^H$ represents the interference generated by the other users and collected by user k.
The solution of this problem is the generalized eigenvector of matrices

$$\mathbf{d}_{MSR,k}^H = \varsigma_m \left(\mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H, \mathbf{H}_k \sum_{j=1, j \neq k}^{K} \mathbf{t}_j \mathbf{t}_j^H \mathbf{H}_k^H + N_0 \mathbf{I} \right) \tag{3.6}$$

.

Finally, for the single stream case, the optimum receiver maximizing the system total sum-rate is given by

$$\mathbf{d}_{MSR,k} = \zeta_m \left(\boldsymbol{\Psi}_k \right)^H, \tag{3.7}$$

where $\boldsymbol{\Psi}_k = \left(\sum_{j=1, j \neq k}^{K} \mathbf{H}_k \mathbf{t}_j \mathbf{t}_j^H \mathbf{H}_k^H + N_0 \mathbf{I} \right)^{-1} \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H$ and $\zeta_m \left(\mathbf{X} \right)$ is, as previously defined, the largest singular vector of matrix $\mathbf{X}$.

It should be noted that in the case of a Hermitian semidefinite positive matrix the eigen decomposition is equivalent to an SVD (Singular Value Decomposition) and that the generated singular values are in decreasing order. Given the structure of $\boldsymbol{\Psi}_\mathbf{k}$, $\zeta_m \left(\boldsymbol{\Psi}_\mathbf{k} \right)$ is, in the case of single stream per user

considered in this section, of the form $\mathbf{y}^* = \frac{\mathbf{y}}{\|\mathbf{y}\|}$ where

$$\mathbf{y} = \mathbf{t}_k^H \mathbf{H}_k^H \left(\sum_{j=1}^{K} \mathbf{H}_k^H \mathbf{t}_j \mathbf{t}_j^H \mathbf{H}_k + N_0 \mathbf{I} \right)^{-1}$$

is the MMSE receiver and $\mathbf{y}^*$ becomes the normalized MMSE receiver.

3.3.3 SJNR/MSR Iterative Procedure

In order to propose an iterative version for precoder and decoder optimization, we have to link the introduced precoders and receivers through a common variable. This link must be evolving through iterations to to ensure convergence of the optimization process. Therefore, we propose a virtual channel $\mathbf{h}_k^{iter}$ for each user k and define it at each iteration by

$$\mathbf{h}_k^{iter} = \mathbf{d}_k^{iter-1} \mathbf{H}_k \tag{3.8}$$

where d_k^{iter-1} represents the decoder computed for user k at the previous loop; d_k can for example be the MSR decoder given in (3.7). The iterative versions of the precoder is obtained by injecting the iterative virtual channel (3.8) into expression (3.3). The obtained iterative precoder is

$$\mathbf{t}_k^{iter} = \sqrt{P_k} \zeta_m \left[\left(\sum_{j=1, j \neq k}^{K} (\mathbf{h}_j^{iter})^H \mathbf{h}_j^{iter} + \frac{N_0^{iter}}{P_k} \mathbf{I} \right)^{-1} (\mathbf{h}_k^{iter})^H \mathbf{h}_k^{iter} \right] \tag{3.9}$$

where $N_0^{iter} = N_0 \mathbf{d}_k^{iter-1} \left(\mathbf{d}_k^{iter-1} \right)^H$ and $\mathbf{d}_k^{iter-1}$ is the used receiver for user k and $\mathbf{h}^{iter}$ represents the virtual channel given by (3.8). The term $\zeta_m(.)$ of the precoder in equation (3.9) is equivalent to a normalized MMSE precoder for the same reasons previously stated for the receiver in Section 3.3.2.

We need at this point to generate an iterative version for the decoding structure to be used in the iterative process. The iterative proposed version of this receiver is then

$$(\mathbf{d}_{MSR,k}^{iter})^H = \zeta_m \left(\mathbf{\Psi}_k^{iter} \right) \tag{3.10}$$

where

$$\mathbf{\Psi}_k^{iter} = \left(\sum_{j=1, j \neq k}^{K} \mathbf{H}_k \mathbf{t}_j^{iter} (\mathbf{t}_j^{iter})^H (\mathbf{H}_k)^H + N_0 \mathbf{I} \right)^{-1} \mathbf{H}_k \mathbf{t}_k^{iter} (\mathbf{t}_k^{iter})^H (\mathbf{H}_k)^H \tag{3.11}$$

This receiver can be represented by the normalized version of the MMSE receiver as we are in a single stream configuration. The overall iterative algorithm is then described in ***Algorithm 2***.

We also denote the total sum-rate of the MU-MIMO system by SR. The expression of the throughput is the sum over the individual throughputs of users and is given by equation [16, 38, 39]:

$$SR = \sum_{k=1}^{K} \log_2 \left(1 + \frac{\mathbf{d}_k \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H \mathbf{d}_k^H}{\mathbf{d}_k \left(\boldsymbol{\Upsilon}_k + N_0 \mathbf{I} \right) \mathbf{d}_k^H} \right) \tag{3.12}$$

Algorithm 2 Iterative SJNR

1. Perform a first iteration using a linear precoder t_k^0 like an SJNR precoder according to equation (3.3) and we calculate the optimal receiver using (3.7).

2. Change the transmission channel $\mathbf{H}_k$ with a virtual one equivalent to the cascade of the true transmission channel and the calculated receiver. The new channel is given by (3.8).

3. The new precoder $\mathbf{t}_k^{iter}$ is calculated using the new channel $\mathbf{h}_k^{iter}$ according to (3.9).

4. Compute the new optimal receiver $\mathbf{d}_{MSR,k}^{iter}$ using (3.7) with the new precoder $\mathbf{t}_k^{iter}$.

5. Evaluate the total sum-rate for the obtained system $\mathbf{t}_k^{iter}$ and $\mathbf{d}_{MSR,k}^{iter}$ using equation (3.12). Repeat steps 2) to 5) until the algorithm converges. The convergence is determined either by the stabilization of the total sumrate obtained when $\left| SR^{iter} - SR^{iter-1} \right| < \varepsilon$ or when the predefined maximum number of iterations equal to $iter_{max}$ is attained.

3.3.4 SVH Algorithm

The SVH (standing for the initials of the authors Stojnic, Vikalo, Hassibi) algorithm presented in [29] under the reference method 2.1 proposes the optimal solution derived for the quasi-convex optimization problem. The

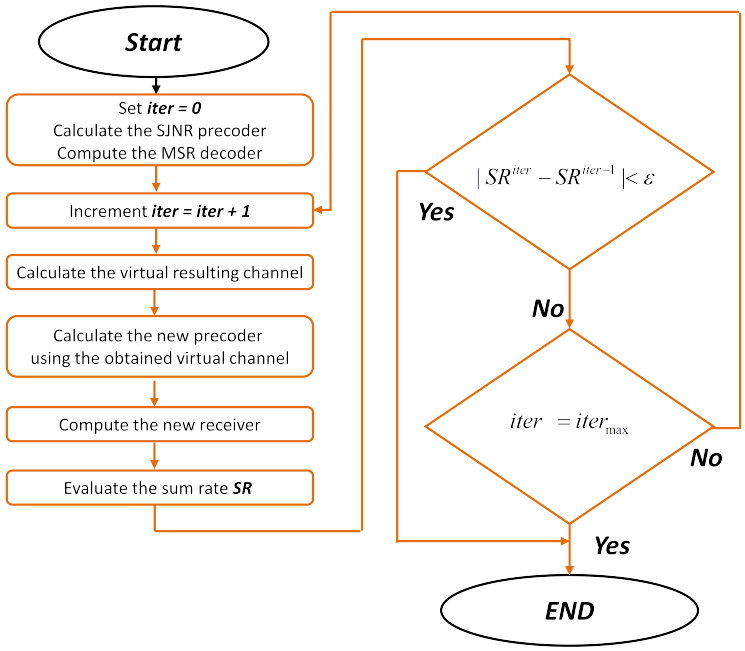


Figure 3.4: Generic SJNR single stream iterative algorithm flowchart.

paper solves it exactly using the bisection method and gives the mathematical expression for the precoder. The algorithm presented under the method 2.1 can be expressed for a MU-MISO system by system of equations (3.13).

$$\begin{cases} \mathbf{F}^{iter_{SVH}} & = diag\left(f_1, \cdots, f_K\right) & (3.13a) \\ \mathbf{G}^{iter_{SVH}} & = diag\left(g_1, \cdots, g_K\right) & (3.13b) \\ \mathbf{T}^{iter_{SVH}} & = \frac{\alpha \mathbf{H}^H \mathbf{G}^{iter_{SVH}}}{\left(\frac{N_0}{P_T} tr\left(\mathbf{F}^{iter_{SVH}}\right)\mathbf{I} + \mathbf{H}^H \mathbf{F}^{iter_{SVH}} \mathbf{H}\right)} & (3.13c) \end{cases}$$

with

$$\begin{cases} f_k & = \frac{num_k}{den_k\left(den_k + num_k\right)} & (3.14a) \\ g_k & = \frac{\left(\mathbf{H}\mathbf{T}^{iter_{SVH}-1}\right)_{kk}}{den_k} & (3.14b) \end{cases}$$

and α a normalization factor to respect the total power constraint

$$tr\left(\mathbf{T}^{iter_{SVH}}\left(\mathbf{T}^{iter_{SVH}}\right)^H\right) = P_T \tag{3.15}$$

. where

$$num_k = \left|\left(\mathbf{H}\mathbf{T}^{iter_{SVH}-1}\right)_{kk}\right|^2 \tag{3.16}$$

and

$$den_k = \sigma^2 tr\left(\mathbf{T}^{iter_{SVH}-1}\left(\mathbf{T}^{iter_{SVH}-1}\right)^H\right) + \sum_{n=1,n\neq k}^{K}\left|\left(\mathbf{H}\mathbf{T}^{iter_{SVH}-1}\right)_{kn}\right|^2 \tag{3.17}$$

Here $\mathbf{T}^{iter_{SVH}} = \left[\left(\mathbf{t}_1^{iter_{SVH}}\right), \cdots, \left(\mathbf{t}_K^{iter_{SVH}}\right)\right]$ and $\mathbf{H} = [\mathbf{h}_1^T, \cdots, \mathbf{h}_K^T]^T$.

The inner iterative algorithm consists in initializing the $\mathbf{F}^{iter_{SVH}}$ and $\mathbf{G}^{iter_{SVH}}$ matrices with the identity matrix $\mathbf{I}$ and computing the corresponding precoder $\mathbf{T}$. The algorithm then iterates by computing the new $\mathbf{F}$ and $\mathbf{G}$ corresponding to the previously computed precoder. The new precoder is then calculated in function of the obtained $\mathbf{F}$ and $\mathbf{G}$. The system converges when it is stabilized, meaning that the obtained value for the precoder no longer changes $|SR^{iter_{SVH}} - SR^{iter_{SVH}-1}| < \varepsilon_{SVH}$. The end of the algorithm can also be controlled by fixing the number of iterations. SR is calculated according to (3.12) with $\mathbf{d}_k = 1$.

The iterative MU-MISO optimization algorithm is described in ***Algorithm 3***.

Algorithm 3 Iterative SVH (method 2.1)

1. Initialize $\mathbf{F}^{iter,0}$ and $\mathbf{G}^{iter,0}$ matrices to $\mathbf{I}$ and compute the first precoder $\mathbf{T}_{SVH}^{iter,0}$ using equation (3.13c) and set $iter_{SVH}$ to 1.
2. Calculate the two vectors **num** and **den** with the elements given respectively by equation (3.16) and (3.17).
3. Compute matrices $\mathbf{F}^{iter,iter_{SVH}}$ and $\mathbf{G}^{iter,iter_{SVH}}$ using respectively equations (3.13a) and (3.13b).
4. Calculate the new precoding matrix $\mathbf{T}_{SVH}^{iter,iter_{SVH}}$ with (3.13c).
5. Repeat steps 2) and 4) until a maximal number of iterations $iter_{SVH} = iter_{max}^{SVH}$ is reached or the system is stabilized. The stability of the system is reached when $\left|SR^{iter,iter_{SVH}} - SR^{iter,iter_{SVH}-1}\right| < \varepsilon_{SVH}$. And save the last computed precoder $\mathbf{t}_{SVH,k}^{iter}$ for each user k.

3.3.5 SVH/MSR Iterative Algorithm

We define the SVH/MSR algorithm based on the association of the two previously described algorithms, namely the SVH precoder and the MSR

decoder [45]. These two algorithms are combined in an iterative way by means of a virtual channel given in (3.8).

The channel obtained through this transformation becomes a $1 \times N_T$ MISO channel. Under this condition, the SVH algorithm described earlier and denoted as ***Algorithm 3*** can be applied, and provides the optimal solution for each of the MU-MISO subproblems. The precoder is then calculated in an iterative way and used to compute the optimal MU-MIMO decoder for each user. Finally we get a general iterative procedure evolving through the virtual channel updates. It applies an internal iterative procedure, i.e. the SVH 2.1 algorithm, to compute the optimal MISO precoder for each user. The iterative SVH/MSR optimization procedure is defined in ***Algorithm 4***.

Algorithm 4 Iterative SVH/MSR

1. Initialization using a linear closed form precoder noted $\mathbf{t}_k^0 = \mathbf{t}_k$ using (3.3) for SJNR and compute the optimal receiver $\mathbf{d}_{MSR,k}^0$ using (3.7).

2. Exchange the transmission channel $\mathbf{H}_k$ with a virtual one equivalent to the cascade of the real transmission channel and the calculated receiver. The new channel is given by (3.8).

3. Launch the SVH optimization procedure over the virtual channels. The new precoder $\mathbf{t}_{SVH,k}^{iter}$ is calculated using the new channel $\mathbf{h}_k^{iter}$ according to the iterative process described in ***Algorithm 3***.

4. Compute the new optimal receiver $\mathbf{d}_{MSR,k}^{iter}$ using equation (3.10) with the new precoder $\mathbf{t}_{SVH,k}^{iter}$.

5. Evaluate the total sum-rate for the obtained system $\mathbf{t}_{SVH,k}^{iter}$ and $\mathbf{d}_{MSR,k}^{iter}$ using (3.12).

6. Repeat step 2) to step 5) until the algorithm converges. The convergence is determined either by the stabilization of the total sum-rate obtained when $\left|SR^{iter} - SR^{iter-1}\right| < \varepsilon$ or when the predefined maximum number of iterations of the external loop equal to $iter_{max}$ is attained. Here ε is a prefixed threshold.

The algorithm computes the SJNR precoder used to compute the initialization of the algorithm. Then, based on the obtained receivers, the algorithm computes the virtual channels to be considered and injected in

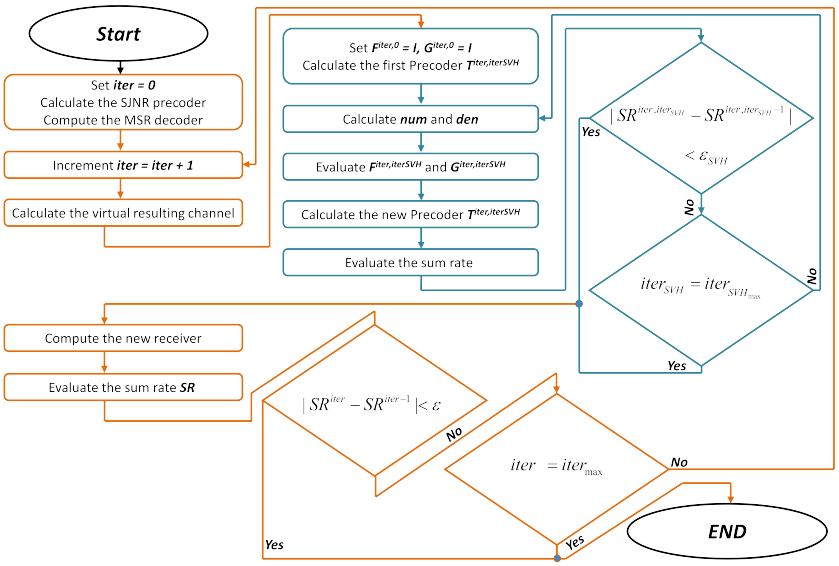


Figure 3.5: Virtual channel SVH single stream iterative algorithm flowchart.

the SVH algorithm (***Algorithm 3***). Once this algorithm has converged, the obtained precoder are used to compute the optimal corresponding receivers based on the MSR calculation.

Here we have two convergence control parameters. The first one is for the internal SVH algorithm given by ε_{SVH} or a maximal number of allowed iterations $iter_{max}^{SVH}$. An overall convergence control is based on the threshold ε or a maximal number of iterations $iter_{max}$. At first sight, it is obvious that this algorithm has a higher complexity and uses more iterations than the previously described algorithms. Nevertheless, we showed that the internal optimization procedure does not require more than one iteration ($iter_{SVH} = 2$) to converge to the best obtainable solution achievable with this algorithm and thus any extra iteration does not further improve the throughput. The fast convergence of ***Algorithm 3*** is due to the fact that the overall algorithm using the virtual channel selection is already optimizing the directions selection for the different streams by making them almost orthogonal. In addition to that, the initialization procedure is based on the receivers derived from the SJNR precoder. These precoder offer a good starting point for the algorithm and thus minimize the number of required iterations.

3.4 SJNR/MFMSR Dynamic Iterative Algorithm

3.4.1 Dynamic Flip Procedure

While analyzing the performances of the previously presented iterative algorithms, we observe that some perform better than others in function of the SNR region and the system configurations (number of transmitting, receiving antennas,...). Therefore, the main idea of the double iterative procedure is to combine two versions of the iterative algorithm derived from ***Algorithm 2*** using different receivers. This would enable the algorithm to cover the biggest part of the vector space containing the possible precoder. This minimizes the probability to get stuck into a local maximum.

It should be noted at this point that the use of the SJNR precoding technique is preferred to the SVH presented in ***Algorithm 3*** because although the SVH converges to the optimal solution,it is much more sensitive to the initialization point especially when K increases. Nonetheless, it can be used after a good initialization point has been found to refine the obtained precoder and decoders.

The two chosen algorithms, have been selected due to their behavior in function of the iterations. In fact, the SJNR/MSR-based algorithm has a linear convergence behavior but most of the time towards a local minimum. The second one, using an MF receiver in the optimization process, tries to maximize the transmitted power by inverting the equivalent channel and thus present an optimization with series of hops searching for the region optimizing the power distribution. Combining these two properties helps to get a better precoding solution.

Nevertheless, combining two versions of the algorithm implies fixing the number of iterations for both of them thus, a flipping point where the used algorithm is changed is required. Nonetheless, some statistical analysis of the throughputs given by the cascade of the two versions SJNR/MF and SJNR/MSR described in **Algorithm 2** demonstrated that the optimal flipping point is not only a function of the SNR (Signal to Noise Ratio) and of the system configuration (Number of transmitting and receiving antennas) but also of the channel realizations, namely the matrices $\mathbf{H}_k, k = 1..K$.

Figure 3.6 is an example of optimal flipping point in function of the total transmitted power through two different stochastic analysis. The simulation parameters that have been considered are the same as the ones taken in Section 3.8 with a constant total number of iterations $N_{max}^{iter} = 50$. The blue curve gives the optimal flipping point maximizing the mean sum rate of the system by averaging over the channel realizations. The second curve

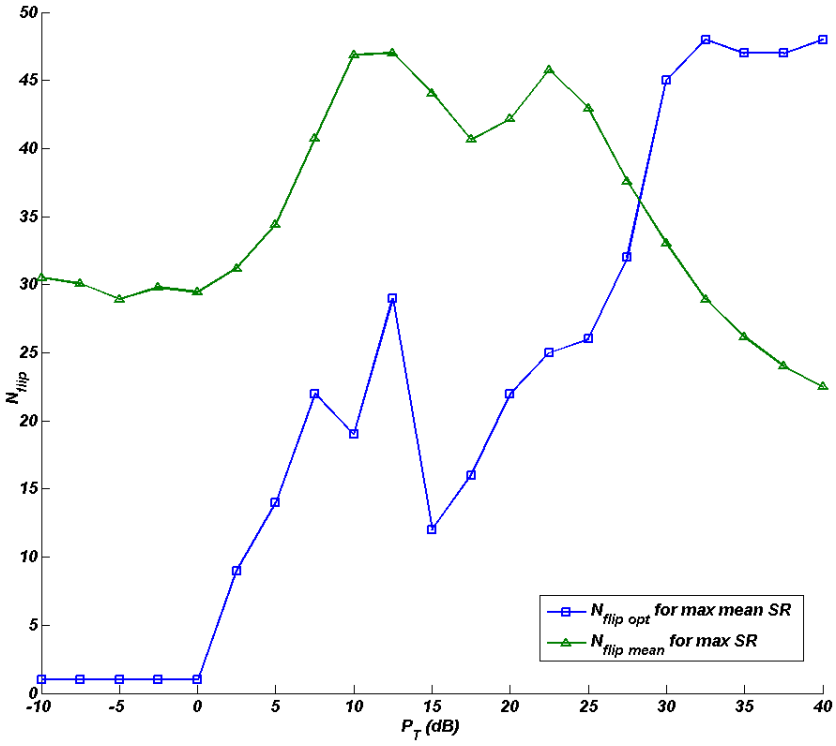
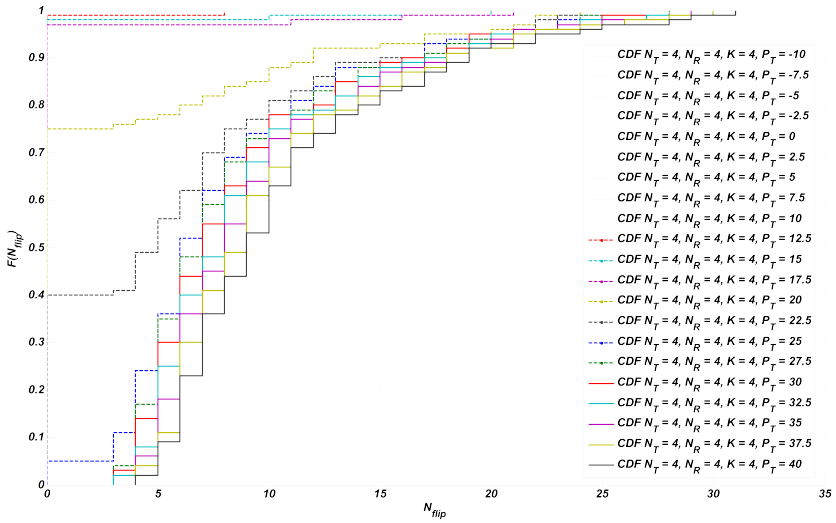
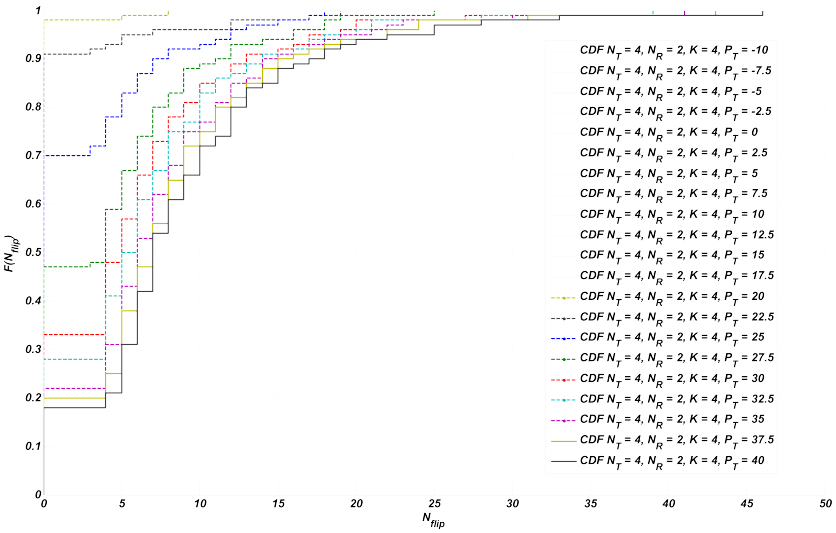


Figure 3.6: optimal N_{flip} in function of the SNR and two simple statistical criteria for N_T=N_R=K=4.

plots the mean flipping point maximizing the instantaneous sum rate of the system. Curves in Figure 3.7 presented some more statistical analysis



(a)



(b)

Figure 3.7: CDF of the throughput in function of the flipping point: (a) $N_T = N_R = K = 4$; (b) $N_T = 4, N_R = K = 2$.

demonstrate like those of Figure 3.6 that the best flipping point is a function

of all the parameters of the system. Therefore a standardized flipping point can not be defined. The solution would be to perform some lookup table in function of the system configuration. On the other hand, the dimensions of these tables can rapidly explode.

To solve this problem, a new dynamic selection procedure is proposed in this Section. It is based on the monitoring of the obtained throughput over a fixed number of iterations, called sliding window. The number of iterations in a window is denoted by WIN_{MF} as the first considered receiver is the MF one. The selection procedure is as follows.

During the first phase, the algorithm derived from ***Algorithm 2*** with an MF receiver is run for WIN_{MF} iterations. Afterwards, the monitoring procedure starts and at each iteration $iter \geq WIN_{MF}$, the variance of the obtained sum rate over the observed WIN_{MF} last iterations is computed according to equation (3.18).

$$\begin{aligned} SR_{var}^{iter} &= Var\left(\left[SR^{iter-WIN_{MF}+1}, \ldots, SR^{iter}\right]\right) \\ &= \frac{1}{WIN_{MF}} \sum_{i=0}^{WIN_{MF}-1} \left(SR^{iter-i} - \frac{1}{WIN_{MF}} \sum_{i=0}^{WIN_{MF}-1} SR^{iter-i} \right)^2 \end{aligned} \quad (3.18)$$

This variance is compared with a prefixed threshold ε_{MF} defining the convergence of the algorithm. If the SR_{var}^{iter} is stabilizing or if its increase is below the prefixed threshold, the MF receiver giving the best SR in this window is retained.

To avoid the divergence problems (divergence or non convergence), we introduce a last control parameter which limits the number of possible iterations for the SJNR/MF algorithm to $iter = N_{\max}^{iter} - \Delta$ where $\Delta \in \mathbb{N}^*$ and $1 \leq \Delta \leq N_{\max}^{iter} - WIN_{MF}$. Here $N_{\max}^{iter}$ is the total number of iterations allowed for the processing of a given transmission.

3.4.2 Double Iterative Procedure

In this subsection, the entire iterative procedure that we propose in [44] is presented. During the first phase, the SJNR/MF algorithm derived from ***Algorithm 2*** is computed, followed by the SJNR/MSR given also by ***Algorithm 2*** where the MF decoder is replaced by the MSR one defined in Section 3.3.2. The decision taken is based on the Dynamic decision procedure presented in the previous subsection. The overall iterative algorithm is then given by ***Algorithm 5***.

Despite the fact that the SJNR/MFMSRD iterative algorithm is a very good algorithm, it does not treat the power allocation issue and thus does

not fully optimize the system performances. On the other hand, the SVH algorithm performs optimal power allocation search along with the precoder design throughout the iterative optimization procedure. This suggests the idea of using it to optimize furthermore SNR regions that are more sensitive to power allocation variations (especially low SNR ones).

Algorithm 5 Double Iterative SJNR

1. Initialize $N_{\max}^{iter}$, WIN_{MF}, ε, ε_{MF} and Δ
2. A first iteration based on a closed form solution is done to initialize the algorithm.
   - 2.1. The SJNR linear precoder given by equation (3.3) is used. This first calculated precoder is called $\mathbf{t}_k^0$.
   - 2.2. Calculate the first optimal receiver $\mathbf{d}_k^0 = \mathbf{d}_{MF,k}$ for all users $k = 1, \ldots, K$ based on formula (3.4).
   - 2.3. Set $iter = 1$.
3. Transform the original transmission channel $\mathbf{H}_k$ to get the virtual channel linking the precoder and the decoder. The virtual channel is denoted by $\mathbf{h}_k^{iter}$ and is computed using (3.8).
4. The new precoder $\mathbf{t}_k^{iter}$ is computed using the new channel through (3.9).
5. Compute the new receiver $\mathbf{d}_k^{iter} = \mathbf{d}_{MF,k}$ in function of the precoder $\mathbf{t}_k^{iter}$ through (3.4).
6. Compute the total obtained throughput SR^{iter} by injecting $\mathbf{t}_k^{iter}$ and $\mathbf{d}_k^{iter}$ in (3.12).
7. **if** $iter < WIN_{MF}$ **then**
   $iter = iter + 1$ and jump to 3).
   end if

3.5 Multistream Hassibi Approach

It is straightforward to get to the multiple stream solution by extending the single stream optimal MISO solution proposed in [29] to the MU-MIMO

8. Test the convergence of the MF algorithm by verifying the flip point condition:
   if $Var\left(\left[SR^{iter-WIN_{MF}+1}, \ldots, SR^{iter}\right]\right) \leq \varepsilon_{MF}$ **then**
   jump to step 9).
   else
   if $iter = N_{\max}^{iter} - \Delta$ **then**
   jump to step 9).
   else
   $iter = iter + 1$ and jump to 3).
   end if
   end if

9. Consider the best set of decoding vectors $\mathbf{d}_k^{iter_{best}}, k = 1 \ldots K, iter_{best} \in \{iter - WIN_{MF} + 1, \ldots, iter\}$ that gives the highest SR among those present in the actual window and jump to 10).

10. Increment the counter $iter = iter + 1$ and compute the virtual channel given by the cascade on the channel and the receiver $\mathbf{h}_k^{iter} = \mathbf{d}_k^{iter-1}\mathbf{H}_k$. It must be noted that for the first time (when the algorithm comes from step 7), $\mathbf{d}_k^{iter-1}$ is an MF receiver such as calculated in step 5) in the iteration $iter_{best}$.

11. The new precoder $\mathbf{t}_k^{iter}$ is computed using (3.9) with the new obtained virtual channel.

12. Compute the new optimal receiver $\mathbf{d}_k^{iter} = \mathbf{d}_{MSR,k}$ in function of the precoder $\mathbf{t}_k^{iter}$ based on (3.10).

13. Evaluate the total sum rate SR^{iter} using $\mathbf{t}_k^{iter}$, $\mathbf{d}_k^{iter}$ and (3.12).

14. Repeat steps 10) to 13) until the algorithm converges. The convergence is either detected by the stabilization of the sum rate $\left|SR^{iter} - SR^{iter-1}\right| < \varepsilon$; or by the achievement of the maximal number of iterations $N_{\max}^{iter}$

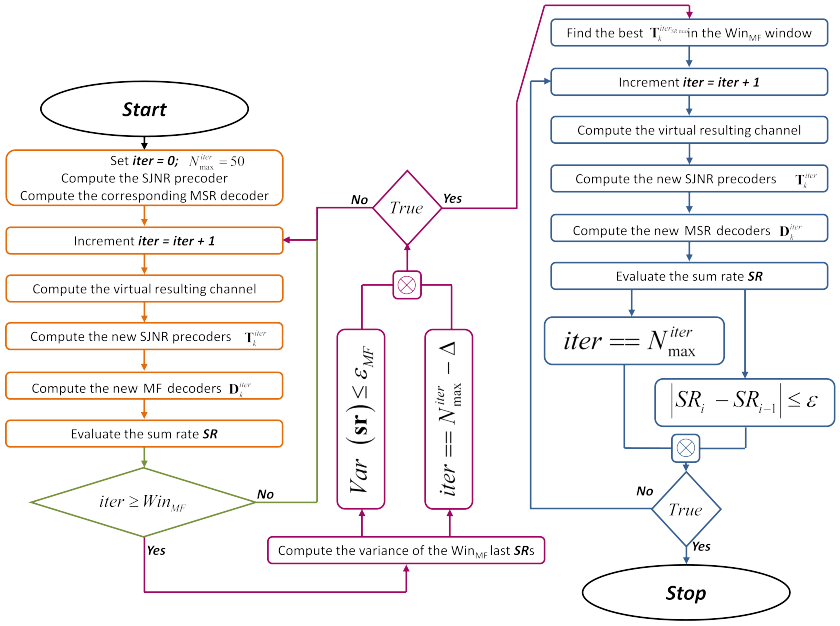


Figure 3.8: Double iterative single stream algorithm flowchart.

BC case. This part indeed presents an extension of the algorithm proposed in [29].

Let us consider the precoding matrix T_k of user k and, for total power constraint $\sqrt{\sum tr\left(\mathbf{T}_k\mathbf{T}_k^H\right)} = P_T$, the precoder can be written as

$$\widetilde{\mathbf{T}}_k = \frac{\sqrt{P_k}}{\sqrt{\sum_{j=1}^{K} tr\left(\mathbf{T}_j\mathbf{T}_j^H\right)}}\mathbf{T}_k = \sqrt{P_k}\beta_k\mathbf{T}_k \tag{3.19}$$

From this expression of the precoding matrices, we can rewrite the sum-rate expression as

$$SR = \sum_{k=1}^{K} \log\det\left(\mathbf{I}_k + P_k\beta_k^2\mathbf{H}_k\mathbf{T}_k\mathbf{T}_k^H\mathbf{H}_k^H\mathbf{R}_k^{-1}\right) \tag{3.20}$$

$$= \sum_{k=1}^{K} \log\det\left(\mathbf{I}_k + P_k\beta_k^2\mathbf{H}_k\mathbf{T}_k\mathbf{T}_k^H\mathbf{H}_k^H \left(\mathbf{R}_{n_k} + \sum_{j=1;j\neq k}^{K}\left(P_j\beta_j^2\mathbf{H}_k\mathbf{T}_j\mathbf{T}_j^H\mathbf{H}_k^H\right)\right)^{-1}\right) \tag{3.21}$$

where $\mathbf{R}_{n_k}$ is the noise covariance matrix for user k. Deriving the obtained sum rate expression according to the precoding matrix $\mathbf{T}_k$ gives

$$\frac{\partial SR}{\partial \mathbf{T}_k} = -\frac{1}{P_k}\sqrt{P_k}\beta_k\mathbf{T}_k tr\left(P_k\beta_k^2\mathbf{E}_k\mathbf{T}_k^H\mathbf{R}_k^{-1}\mathbf{H}_k\mathbf{T}_k\right) + \sqrt{P_k}\beta_k\mathbf{H}_k^H\mathbf{R}_k^{-1}\mathbf{H}_k\mathbf{T}_k^H\mathbf{E}_k + \sum_{j=1;j\neq k}^{K}\left(\frac{1}{P_k}\sqrt{P_k}\beta_k\mathbf{T}_k tr\left(P_j\beta_j^2P_k\beta_k^2\mathbf{E}_j\mathbf{T}_j^H\mathbf{H}_k^H\mathbf{R}_j^{-1}\mathbf{H}_k\mathbf{T}_k\mathbf{T}_k^H\mathbf{H}_k^H\mathbf{R}_j^{-1}\mathbf{H}_k\mathbf{T}_j\right)\right) - \sum_{j=1;j\neq k}^{K}\left(P_j\beta_j^2P_k\beta_k^2\mathbf{H}_k^H\mathbf{R}_j^{-1}\mathbf{H}_k\mathbf{T}_j\mathbf{E}_j\mathbf{T}_j^H\mathbf{H}_k^H\mathbf{R}_j^{-1}\mathbf{H}_k\mathbf{T}_k\right) \tag{3.22}$$

where $\mathbf{E}_k^{-1} = \mathbf{I}_k + P_k\beta_k^2\mathbf{T}_k^H\mathbf{H}_k^H\mathbf{R}_k^{-1}\mathbf{H}_k\mathbf{T}_k$ The optimal precoder is then obtained by setting $\frac{\partial SR}{\partial \mathbf{T}_k} = 0$. Compared to the Hassibi solution in [29], the constant factors are no more scalar factors but matrices. As well as the optimal solution for $\widetilde{\mathbf{T}}_k$ that becomes more complex to derive. Therefore, a more structured approach should be adopted. In fact in a MU-MIMO system the SR expression depends also on the receiver and thus it can be rewritten as

$$SR_2 = \sum_{k=1}^{K}\log\det\left(\mathbf{I}_k + P_k\beta_k^2\mathbf{D}_k\mathbf{H}_k\mathbf{T}_k\mathbf{T}_k^H\mathbf{H}_k^H\mathbf{D}_k^H\left(\mathbf{D}_k\mathbf{R}_k\mathbf{D}_k^H\right)^{-1}\right) \tag{3.23}$$

and in this case by deriving the SR expression as done in 3.22 and considering the virtual channel $\widetilde{\mathbf{H}}_k = \mathbf{D}_k\mathbf{H}_k$ given by the cascade of the channel and the receiver and the noise covariance matrix $\widetilde{\mathbf{R}}_k = \mathbf{D}_k\mathbf{R}_k\mathbf{D}_k^H$. The obtained

result shows clearly that the optimal precoder is a function not only of the channels and of the powers, but also of the receivers as

$$\begin{aligned}\frac{\partial SR_2}{\partial \mathbf{T}_k} &= -\frac{1}{P_k}\sqrt{P_k}\beta_k \mathbf{T}_k tr\left(P_k\beta_k^2\widetilde{\mathbf{E}}_k\mathbf{T}_k^H\widetilde{\mathbf{R}}_k^{-1}\widetilde{\mathbf{H}}_k\mathbf{T}_k\right)+\\ &\sqrt{P_k}\beta_k\widetilde{\mathbf{H}}_k^H\widetilde{\mathbf{R}}_k^{-1}\widetilde{\mathbf{H}}_k\mathbf{T}_k^H\widetilde{\mathbf{E}}_k+\sum_{j=1;j\neq k}^{K}\left(\frac{1}{P_k}\sqrt{P_k}\beta_k\mathbf{T}_k\right.\\ &\left.tr\left(P_j\beta_j^2P_k\beta_k^2\widetilde{\mathbf{E}}_j\mathbf{T}_j^H\widetilde{\mathbf{H}}_k^H\widetilde{\mathbf{R}}_j^{-1}\widetilde{\mathbf{H}}_k\mathbf{T}_k\mathbf{T}_k^H\widetilde{\mathbf{H}}_k^H\widetilde{\mathbf{R}}_j^{-1}\widetilde{\mathbf{H}}_k\mathbf{T}_j\right)\right)-\\ &\sum_{j=1;j\neq k}^{K}\left(P_j\beta_j^2P_k\beta_k^2\widetilde{\mathbf{H}}_k^H\widetilde{\mathbf{R}}_j^{-1}\widetilde{\mathbf{H}}_k\mathbf{T}_j\widetilde{\mathbf{E}}_j\mathbf{T}_j^H\widetilde{\mathbf{H}}_k^H\widetilde{\mathbf{R}}_j^{-1}\widetilde{\mathbf{H}}_k\mathbf{T}_k\right) \quad (3.24)\\ &= -\frac{1}{P_k}\sqrt{P_k}\beta_k\mathbf{T}_k tr\left(P_k\beta_k^2\left(\mathbf{I}_k+P_k\beta_k^2\mathbf{T}_k^H\mathbf{H}_k^H\mathbf{R}_k\mathbf{H}_k\mathbf{T}_k\right)\right.\\ &\left.\mathbf{T}_k^H\widetilde{\mathbf{R}}_k^{-1}\mathbf{D}_k\mathbf{H}_k\mathbf{T}_k\right)+\sqrt{P_k}\beta_k\mathbf{H}_k^H\mathbf{D}_k^H\widetilde{\mathbf{R}}_k^{-1}\mathbf{D}_k\mathbf{H}_k\mathbf{T}_k^H\\ &\left(\mathbf{I}_k+P_k\beta_k^2\mathbf{T}_k^H\mathbf{H}_k^H\widetilde{\mathbf{R}}_k\mathbf{H}_k\mathbf{T}_k\right)+\sum_{j=1;j\neq k}^{K}\left(\frac{1}{P_k}\sqrt{P_k}\beta_k\mathbf{T}_k\right.\\ &tr\left(P_j\beta_j^2P_k\beta_k^2\left(\mathbf{I}_j+P_k\beta_k^2\mathbf{T}_j^H\mathbf{H}_k^H\widetilde{\mathbf{R}}_j\mathbf{H}_k\mathbf{T}_j\right)\right.\\ &\left.\left.\mathbf{T}_j^H\mathbf{H}_k^H\mathbf{D}_k^H\widetilde{\mathbf{R}}_j^{-1}\mathbf{D}_k\mathbf{H}_k\mathbf{T}_k\mathbf{T}_k^H\mathbf{H}_k^H\mathbf{D}_k^H\widetilde{\mathbf{R}}_j^{-1}\mathbf{D}_k\mathbf{H}_k\mathbf{T}_j\right)\right)-\\ &\sum_{j=1;j\neq k}^{K}\left(P_j\beta_j^2P_k\beta_k^2\mathbf{H}_k^H\mathbf{D}_k^H\widetilde{\mathbf{R}}_j^{-1}\mathbf{D}_k\mathbf{H}_k\mathbf{T}_j\right.\\ &\left.\left(\mathbf{I}_j+P_k\beta_k^2\mathbf{T}_j^H\mathbf{H}_k^H\widetilde{\mathbf{R}}_j\mathbf{H}_k\mathbf{T}_j\right)\mathbf{T}_j^H\mathbf{H}_k^H\mathbf{D}_k^H\widetilde{\mathbf{R}}_j^{-1}\mathbf{D}_k\mathbf{H}_k\mathbf{T}_k\right) \quad (3.25)\end{aligned}$$

where $\widetilde{\mathbf{E}}_k^{-1}=\mathbf{I}_k+P_k\beta_k^2\mathbf{T}_k^H\widetilde{\mathbf{H}}_k^H\widetilde{\mathbf{R}}_k^{-1}\widetilde{\mathbf{H}}_k\mathbf{T}_k$

Considering this, we have also to derive the optimal receiver. The same derivation according to the $\mathbf{D}_k$ gives a similar expression depending on the precoders.

The obtained results show that the extension of the method in [29] converges to the same solution given in [42], derived based on the KKT condition. As shown, the two derivation methods are equivalent and thus present the same local maximum problem. Therefore, we extend the double iterative approach presented earlier for the MIMO single stream per user case to the multiple stream case based on the WMMSE/MFMMSE Iterative Algorithm from [42].

3.6 WMMSE/MFMMSE Double Iterative Algorithm

The objective for the multistream case, is to design the precoding matrices $\mathbf{T}_k$ under the total transmit power constraint $\sum_{k=1}^{K} P_k = P_T$. Here $P_k = tr\left(\mathbf{T}_k \mathbf{T}_k^H\right)$ denotes the transmitted power for user k. Therefore, inspired by the idea adopted for the single stream case, we will extend it to the multi-stream case. So we consider a multi-stream precoder design, namely the WMMSE precoder and two decoders; i.e. an MF and an MMSE receivers.

3.6.1 WMMSE Precoder

Let us consider the MMSE precoder minimizing the mean square error given by

$$\mathbf{T}_k = \alpha \left(\sum_{j=1}^{K} \mathbf{H}_j^H \mathbf{D}_j^H \mathbf{D}_j \mathbf{H}_j + \frac{tr(\mathbf{D}_j \mathbf{D}_j^H)}{P_T} \mathbf{I}_{N_T} \right)^{-1} \mathbf{H}_k^H \mathbf{D}_k^H \tag{3.26}$$

where α is a scalar factor. We apply a stream distribution among the available ones. The distribution of streams is done by assigning a weight matrix $\mathbf{W}_k$ to each user k.

Applying the distribution to the precoder gives the expression of the new MMSE precoder known as WMMSE (weighted MMSE), according to

$$\mathbf{T}_k = \beta \left(\sum_{j=1}^{K} \mathbf{H}_j^H \mathbf{D}_j^H \mathbf{W}_j \mathbf{D}_j \mathbf{H}_j + \frac{tr(\mathbf{W}_j \mathbf{D}_j \mathbf{D}_j^H)}{P_T} \mathbf{I}_{N_T} \right)^{-1} \tag{3.27}$$

where $\mathbf{W}_k$ is the weight matrix given to streams of user k and β is a scalar factor, introduced to respect the total power constraint $\sum_{i=1}^{K} tr\left(\mathbf{T}_i \mathbf{T}_i^H\right) = P_T$.

A condensed expression for all users is given by

$$\mathbf{T} = \beta \left(\mathbf{H}^H \mathbf{D}^H \mathbf{W} \mathbf{D} \mathbf{H} + \frac{tr(\mathbf{W}\mathbf{D}\mathbf{D}^H)}{P_T} \mathbf{I}_{N_T} \right)^{-1} \mathbf{H}^H \mathbf{D}^H \mathbf{W} \tag{3.28}$$

where

$$\begin{cases} \mathbf{T} = diag\left\{\mathbf{T}_1, \cdots, \mathbf{T}_K\right\} \\ \mathbf{D} = diag\left\{\mathbf{D}_1, \cdots, \mathbf{D}_K\right\} \\ \mathbf{W} = diag\left\{\mathbf{W}_1, \cdots, \mathbf{W}_K\right\} \end{cases}$$

3.6.2 Receiver Design

As receiving filter for the multistream optimization procedure we need to use a decoder capable of taking into account the power variations over the different streams of each user. Therefore, we propose the multi-stream matched filter of equation given by

$$\mathbf{D}_{MF,k} = \frac{\mathbf{T}_k^H \mathbf{H}_k^H}{\|\mathbf{H}_k \mathbf{T}_k\|} \tag{3.29}$$

where $\|\mathbf{X}\|$ is the Frobenius norm of matrix $\mathbf{X}$.

It must be noted at this point that it is very important to consider a Frobenius norm instead of, for example the spectral norm implemented in most of simulation programs such as Matlab. In fact, the Frobenius norm takes into consideration powers of the different eigenvalues, hence all the directions existing in the virtual transmission channel $\mathbf{H}_k\mathbf{T}_k$. On the other hand, the spectral norm for example considers only the biggest eigenvalue, thus the main considered direction. Considering that we are investigating the optimization of the transmit powers for all possible directions to find the optimal ones, all of them must be taken into consideration and therefore the Frobenius norm will give better convergence than other matrix norms.

Another receiving structure minimizing the mean square error is the MMSE receiver given by

$$\mathbf{D}_{MMSE,k} = \mathbf{T}_k^H \mathbf{H}_k^H \left(\mathbf{I}_{N_R} + \sum_{i=1}^{K} \mathbf{H}_k \mathbf{T}_i \mathbf{T}_i^H \mathbf{H}_k^H \right)^{-1} \tag{3.30}$$

3.6.3 Iterative Algorithm

Based on the previous precoder and decoders, an iterative algorithm can be defined to optimize the precoder and decoder design. The algorithm given in ***Algorithm 6*** has been originally proposed in [42] with an MMSE decoder. It can nevertheless be applied to different receiving structures.

The algorithm requires computing the weights for the different streams required for stream selection and power distribution. The authors in [42] propose $\mathbf{W_k}$ as weight matrix, where

$$\mathbf{W}_k = \mathbf{I}_k + \mathbf{T}_k^H \mathbf{H}_k^H \left(\mathbf{I}_{N_R} + \sum_{i=1, i\neq k}^{K} \mathbf{H}_k \mathbf{T}_i \mathbf{T}_i^H \mathbf{H}_k^H \right)^{-1} \mathbf{H}_k \mathbf{T}_k \tag{3.31}$$

Algorithm 6 Iterative WMMSE

1. Initialize the precoders $\mathbf{T}_k, k \in \{1, \cdots, K\}$.
2. Compute the selected type of receiver corresponding to the precoders $\mathbf{T}_k$ for all the users $k \in \{1, \cdots, K\}$.
3. Compute the weights according to (3.31).
4. Compute the new precoders $\mathbf{T}_k, k \in \{1, \cdots, K\}$ according to (3.28).
5. Repeat steps 2) to 4) until convergence.

corresponding to the inverse of the mean square error.

As a performance analysis we estimate the total sum-rate of the MU-MIMO system. The expression of the throughput is the sum over all selected streams of the individual achieved throughputs for each user and can be given by

$$SR = \sum_{k=1}^{K} log_2 \det \left(\mathbf{I} + \mathbf{H}_k \mathbf{T}_k \mathbf{T}_k^H \mathbf{H}_k^H \left(\mathbf{\Upsilon}_k + N_0 \mathbf{I} \right)^{-1} \right) \tag{3.32}$$

where, $\mathbf{\Upsilon}_k = \sum_{j=1, j \neq k}^{K} \mathbf{H}_k \mathbf{T}_j \mathbf{T}_j^H \mathbf{H}_k^H$ represents the interference received by user k.

3.6.4 Dynamic Flip Procedure

The analysis of the performances of the iterative algorithm using different decoders to optimize the precoder shows, like in the single stream case, various throughput levels. Some algorithms present higher sum-rates at high SNRs, like the WMMSE/MF proposed in Section 3.6.3. Some other have better performances at lower SNRs, like the WMMSE/MMSE proposed in [42].

One naive and direct solution to get high performances across the entire SNR range would be to run in parallel these two algorithms and then choose the best among them. This would solve the problem but requires twice the computational resources used by the iterative algorithms.

Moreover, also in the multiple stream case, iterative algorithms are sensitive to initialization and are known to suffer from convergence problems as mentioned in [21]. This major problem remains unsolved by adopting

this strategy. The main idea is to reuse the idea proposed earlier and to combine two versions of the iterative algorithm derived from ***Algorithm 6*** using different receiving structures to be able to cover the largest part of the space containing the possible transmitter. Therefore, a first algorithm WMMSE/MF sweeping through the space of possible precoders and trying to maximize the received powers is performed . The second algorithm WMMSE/MMSE presenting an increasing sum-rate behavior refines the solution towards the maximum. This will reduce the probability of entering a local maximum, which is one of the main limiting factor for iterative algorithms.

Combining two versions of the algorithm implies a flipping point where the used algorithm (receiver) is changed. Furthermore, some statistical analysis of the throughputs given by the cascade of the two versions WMMSE/MF and WMMSE/MMSE described in ***Algorithm 6*** demonstrated that here also the optimal flipping point is not only a function of the SNR (Signal to Noise Ratio) and of the system configuration (Number of transmitting and receiving antennas), but also of the channel realizations namely the matrices $\mathbf{H}_k, k = 1..K$.

A solution would be to perform some lookup tables in function of the system configuration. Unfortunately, the dimensions of these tables are exponential and can rapidly explode.

It turns out that the previously proposed flipping point selection procedures performs well in this case, too. The selection procedure is then based on the monitoring of the obtained throughput over a fixed number of WIN_{MF} iterations.

To be able to run the selection procedure, a minimum of WIN_{MF} observations of the iterative algorithm must be available. Therefore, in the first phase, ***Algorithm 6*** is run for WIN_{MF} iterations with a MF receiver. Starting from this point, the monitoring procedure is launched: at each iteration $iter \geq WIN_{MF}$ the variance of the obtained sum-rate over the last WIN_{MF} considered iterations is computed according to equation (3.18). This quantity is denoted by SR_{var}^{iter}.

This variance is compared to a prefixed threshold ε_{MF} which defines the convergence of the algorithm. So the average evolution of the SR over the last WINMF iterations is observed. If the SR_{var}^{iter} is stabilizing or if it increases below a prefixed threshold, the MF receiver giving the best SR in this window is retained. A last control parameter is introduced to avoid the divergence problem previously mentioned. It consists in limiting the number of possible iterations for the WMMSE/MF algorithm to $iter = N_{max}^{iter} - \Delta$ where $\Delta \in \mathbb{N}^*$ and $1 \leq \Delta \leq N_{max}^{iter} - WIN_{MF}$. Here N_{max}^{iter} is the total

number of iterations allowed for the processing of a given transmission and $\mathbb{N}^* = \{1, 2, \dots\}$. The goal of this limitation in the number of the total iterations is to avoid that the algorithm gets stuck in case of divergence or of non convergence.

3.6.5 Double Iterative Procedure

In this last subsection, the entire double iterative procedure is presented. In a first phase, the WMMSE/MF algorithm given in ***Algorithm 6*** with an MF decoder is executed, followed by the WMMSE/MMSE given in ***Algorithm 6*** with an MMSE decoder. The algorithm flipping decision is taken based on the DFP (dynamic flip procedure) presented in the previous subsection. The evolution of the receiver and the precoder is performed thanks to the weights distribution given by (3.31). The overall iterative algorithm is then described in ***Algorithm 7***.

Algorithm 7 Double Iterative WMMSE

1. Initialize $N_{\max}^{iter}$,WIN_{MF}, ε, ε_{MF}, Δ and $iter = 0$
2. Initialize $\mathbf{T}_k^{iter} = \beta \mathbf{H}_k^H, k \in \{1, \cdots, K\}$ where β is a scalar factor to respect the power constraint $\sum_{i=1}^{K} tr\left(\mathbf{T}_i \mathbf{T}_i^H\right) = P_T$.
3. $iter = iter + 1$. Compute $\mathbf{D}_k^{iter}, k \in \{1, \cdots, K\}$ using $\mathbf{T}^{iter-1}$ with (3.29), $\mathbf{W}_k^{iter}$ using $\mathbf{T}^{iter-1}$ as in (3.31) and $\mathbf{T}_k^{iter}$ using $\mathbf{D}_k^{iter}$ and $\mathbf{W}_k^{iter}$ with (3.28).

3.7 Deterministic Annealing

During the previous section we proposed a method to avoid, or at least to minimize, the impact of local maximums naturally present in the cost function (SR) to maximize. The previously presented method is less complex than a Markov-chain based solution for space sounding and offers important improvements thanks to the combination of two different optimizations.

Nevertheless, some other optimization solutions exist in the literature and are especially used in physics and thermodynamics. In fact the so called "'Heat"' algorithm is comes from the crystallography theories and applications where a system is optimized by heating it up and cooling it slowly

4. Compute SR^{iter} by using (3.32) injecting $\mathbf{T}_k^{iter}$.

   if $iter < WIN_{MF}$ **then**
   jump to step 3)
   end if

5. Verify convergence of ***Algorithm 6*** with the MF decoder:
   if $Var\left(\left[SR^{iter-WIN_{MF}} \cdots SR^{iter}\right]\right) \leq \varepsilon_{MF}$ **then**
   jump to 6)
   else if $iter < N_{\max}^{iter} - \Delta$ **then**
   jump to step 3)
   else
   jump to 6).
   end if

6. Consider the precoder giving the best SR over the last WIN_{MF} iterations.
   $iter_{SR\max} = \angle \max_{i \in \{iter-WIN_{MF}+1, \cdots, iter\}} SR^i$.
   $\mathbf{T}_k^{iter} = \mathbf{T}_k^{iter_{SR\max}}$.

7. $iter = iter + 1$, Compute $\mathbf{D}_k^{iter}$ using $\mathbf{T}^{iter-1}$ with (3.30), $\mathbf{W}_k^{iter}$ using $\mathbf{T}^{iter-1}$ using (3.31) and $\mathbf{T}_k^{iter}$ applying (3.28).

8. Compute SR^{iter} with (3.32) and verify convergence of ***Algorithm 6*** with the MMSE decoder:
   if $\left|SR^{iter} - SR^{iter-1}\right| < \varepsilon$ **then**
   jump to 9).
   else if $iter \leq N_{\max}^{iter}$ **then**
   jump to step 7)
   else
   jump to 9).
   end if

9. Stop the algorithm and consider the last computed precoders $\mathbf{T}_k^{iter}$ and decoders $\mathbf{D}_k^{iter}$, $k \in \{1, \cdots, K\}$.

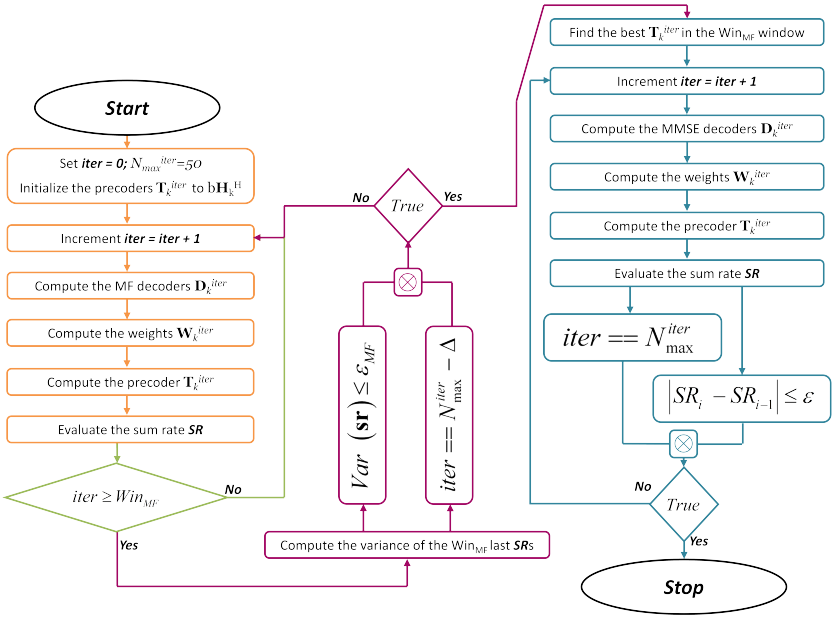


Figure 3.9: Double iterative multistream algorithm flowchart.

down. During this cooling process the molecules organize themselves in order to optimize the space occupation.

This concept appears also in mathematics under the name of Homotopy where an homotopic function is a function that can be continuously deformed to obtain a second continuous function. This property provides a powerful numerical tool for solving nonlinear problems in finite dimensions and nonlinear systems of equations via homotopy deformation methods [50].

This optimization approach has been also introduced in communication like in [5] for clustering problem solving.

In our case, we are going to use the fact that, based on the state of the system in condition C_1 with a given temperature P_1, the system status can be extrapolated at any other state C_i with a given temperature P_i simply by transforming the original solution.

Nevertheless, the solution for the first state must be exactly known, otherwise the error would be propagated through the optimization process as the temperature is progressively changed (in our case increased).

The temperature, for our optimization problem, will be the total power constraint. The initial state will be the case where the optimal solution is known. In fact, at very high transmit powers, the system can support the

maximum number of streams (i.e. $Q_{max} = \min(N_T, \sum_{k=1}^{K} N_{R_k})$). The problem here is that it would be difficult to know the best distribution of streams among users. A second problem that arises is the power distribution. On the other hand, at very low transmit powers, the system can not sustain more than one stream that obviously will be allocated to the user with the best channel, as our cost function is the maximization of the total sum rate. Moreover, the optimal precoder in this case is the Zero Forcing filter inverting the channel. Once the starting point set, the remaining problem is how to detect and select both stream and direction for the new added stream as power increases. That is what we call "decision at phase transition".

As seen in section 3.5, the optimal solution would result from the Weighted Sum Rate optimization algorithm proposed in [42] or from the extension of [29] to the MIMO system presented in section 3.5.

The resulting algorithm from this optimization procedure can be expressed as described in ***Algorithm 8***

Algorithm 8 Deterministic Annealing

1. Set the power to the minimum $P_T = P_{min}$.
   Set the precoding matrices to $\mathbf{0}_{N_{R_k} \times N_T}, \forall k \in [1, \ldots, K]$.
2. Increment the power $P_T = P_T + P_{Step}$.
3. Select the stream to add and increment Q.
4. Optimize the precoder and receiver.
5. **if** $Q > Q_{max}$ (i.e. the maximum number of streams is reached) **then**
   jump to step 9
   else
   jump to 2
   end if
6. End

As described in the ***Algorithm 8***, we need in step 3 a selection procedure to find the best stream to add if one stream has to be added and in step 4 an optimization procedure to refine the precoder and receiver to be used considering the perturbation introduced by the new selected stream.

In fact, as our target is the sum rate maximization, we should evaluate the impact of the newly introduced stream on the throughput. On the other

hand, the stream has just been introduced, introduces very little interference to the previously present streams. In this case the initialization vector for the new stream would be the corresponding maximal eigenvector of the cascade of the real channel $\mathbf{H}_i$ and of all the interference and noise part for the selected user i.

$$\mathbf{t}_i = \zeta_m \left[\mathbf{H}_i^H \left(N_0 \mathbf{I} + \sum_{j=1, j\neq i}^{K} \mathbf{H}_i \mathbf{T}_j \mathbf{T}_j^H \mathbf{H}_i^H \right)^{-1} \mathbf{H}_i \right] \tag{3.33}$$

Once the new direction for potential user i has been determined, the next step is to define the power that we are going to allocate to this new stream. But as the stream has just been introduced, its operating power is very low and a good approximation would be to consider the step power P_{Step} as an initialization. After augmenting the precoding matrix $\mathbf{T}_i$ we launch one of the optimization algorithms, for example ***Algorithm 6***. It must be noted here that the optimization process with a sum rate evaluation must be done for all potential available streams.

To summarize, the algorithm consists of an iterative procedure where the power is iteratively increased by the small quantity P_{Step}. At each iteration, the possibility of introducing a new stream is checked by computing the possible increase of sum rate. To compute this we use an iterative WMMSE algorithm. Whether a new stream has been selected after the power increase or not, the WMMSE optimization is run to ensure correct tracking of the maximum. The complexity of the algorithm is obviously important as two iterative algorithms are nested. Nevertheless, as the introduction of the new stream and/or the increase of the power is just a minor perturbation it would not dramatically change the previously obtained optimum. This is in fact one of the main ideas behind tracking the optimum. And thus, on the opposite to the external loop, little iteration in the internal loop is required for convergence.

The overall deterministic annealing for WMMSE is described in ***Algorithm 9***

3.8 Simulations And Results

In all our simulations, we consider for simplicity and without loss of generality that the number of receiving antennas is the same for all users, i.e. $N_{R_k} = N_R = 4$ or 2. We choose a Rayleigh fading channel $H_k = (h^k_{i,j})_{1\leq i\leq N_R, 1\leq j\leq N_T}$ such as $\mathbb{E}\|h^k_{i,j}\|^2 = 1$. The simulation generates 10^6 in-

Algorithm 9 WMMSE Deterministic Annealing

1. Set the power to the minimum $P_T = P_{min}$.
   Set the precoding matrices to $\mathbf{0}_{N_{R_k} \times N_T}, \forall k \in [1, \ldots, K]$.
2. Increment the power $P_T = P_T + P_{Step}$.
3. Compute $\mathbf{t}_i, i \in [1, \ldots, K]$ for potential streams using 3.33.
4. Optimize the precoder and receiver for all $\mathbf{t}_i, i \in [1, \ldots, K]$.
   Evaluate the sum rate $SR_k, k \in [1, \ldots, K]$.
5. $u_i = \arg \max_{k \in [1,\ldots,K]} \mathbf{SR}_k$.
6. **if** $u_i \neq \emptyset$ is selected **then**
   Update T_{u_i}.Increment $Q = Q + 1$.
   end if

7. Optimize the precoder and receiver for all $\mathbf{t}_i, i \in [1, \ldots, K]$.
8. **if** $Q > Q_{max}$ (i.e. the maximum number of streams is reached) **then**
   jump to step 9
   else
   jump to 2
   end if
9. End

dependent channel realizations for each user. To generate the total throughput of the system, we perform an average over all channel realizations of the sum rate SR given by (3.12) for the single stream case or the more general case with multiple streams per user given by (3.32). For the SJNR precoder, we distribute equally the energy over all the considered users according to $P_k = P_T/K$. The three convergence control parameters for the different algorithms ε_{SVH}, ε_{MF}, ε are fixed and equal to to 10^{-3}. In all the following, N_{iter} represents the temporary number of iterations in the external loop defined as $iter_{max}$. The number of iterations for the double iterative algorithms is defined as the sum of the iterations performed by each of the two iterative optimization procedures. Furthermore, we consider $WIN_{MF} = 5$ and $\Delta = 5$.

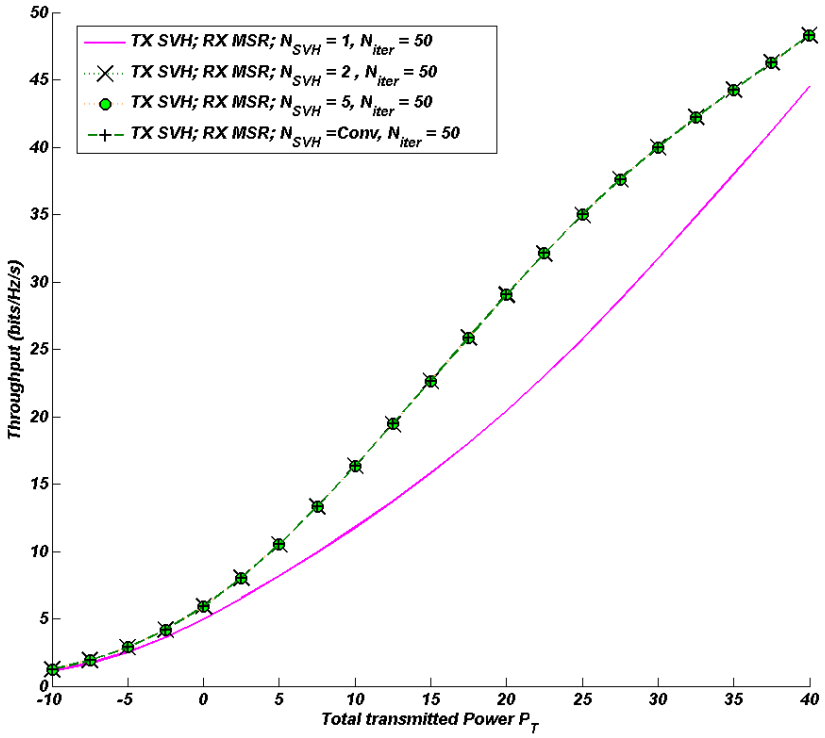


Figure 3.10: Throughput in function of $iter_{max}^{SVH}$ for $N_T = N_R = K = 4$.

In the case of a single stream per user, Figure 3.10 presents for a fixed number of total iterations $iter_{max} = N_{iter} = 50$ the SVH/MSR algorithm with different values of the maximal number of authorized iterations for **Al-**

gorithm 3 (method 2.1). We consider $iter_{max}^{SVH} = N_{SVH} = 1, 2, 5$. The fourth curve describes the algorithm when the number of iterations is unbounded. This means that the algorithm is run until the convergence of method 2.1 is achieved. This figure shows that the convergence and the performance evaluation of the system requires no more than two iterations. The 3 curves shown on the figure "$TXSVH; RXMSR; N_{SVH} = 2; N_{iter} = 50$", "$TXSVH; RXMSR; N_{SVH} = 5; N_{iter} = 50$" and "$TXSVH; RXMSR; N_{SVH} = Conv; N_{iter} = 50$" are almost the same. The performance improvement that can be achieved with no iteration limits compared to the algorithm with $iter_{max}^{SVH} = 5$ and $iter_{max}^{SVH} = 2$ is roughly around 10^{-4} and 10^{-3} bits/Hz/s, respectively. This demonstrates that no more than two iterations are required for the convergence of the internal MISO optimization method. On the other hand, looking at the curve named "$TXSVH; RXMSR; N_{SVH} = 1; N_{iter} = 50$" with a number of internal iterations equal to 1, shows a very low sum-rate for the system. As a matter of fact, if we analyze the algorithm for $iter_{max}^{SVH} = 1$, we can see that it represents indeed the MMSE/MSR algorithm equivalent to the MMSE/MMSE one described in [40] but with a normalized receiver. The normalization factor is just a scaling effect that does not affect the performances for a given loop as it is canceled out in the SR expression. But, when considering the iterative procedure, it changes the powers affected to the different streams from one loop to another and thus changes the optimization result. This shows the importance of performing the optimization of the sum-rate for the virtual MU-MISO system. These observations show that the proposed algorithm offers much better performances just by adding one extra loop for the transmitter optimization procedure (method 2.1). Compared to the existing iterative solutions, this increase in performances is achieved by introducing low extra complexity and very little computational delay.

Figure 3.11 shows the influence of the total number of iterations for the SVH/MSR algorithm. A fixed number of iterations $iter_{max}^{SVH} = 2$ is considered, we showed through the analysis of the previous figure that no further iterations are required. The external maximal number of iterations is changed. The obtained curves show that the total throughput of the system is slightly increasing at low SNRs and that it is increases faster as the SNR is increasing. This shows the importance of introducing the receiver structure in the optimization procedure and proves that joint optimization is required for MU-MIMO systems to be able to get the best out of it. It can also be concluded that although the optimization procedure considers, at each iteration, a MU-MISO channel optimization, the SVH/MSR solution evolves toward better performances by exploiting the diversity offered by the MIMO

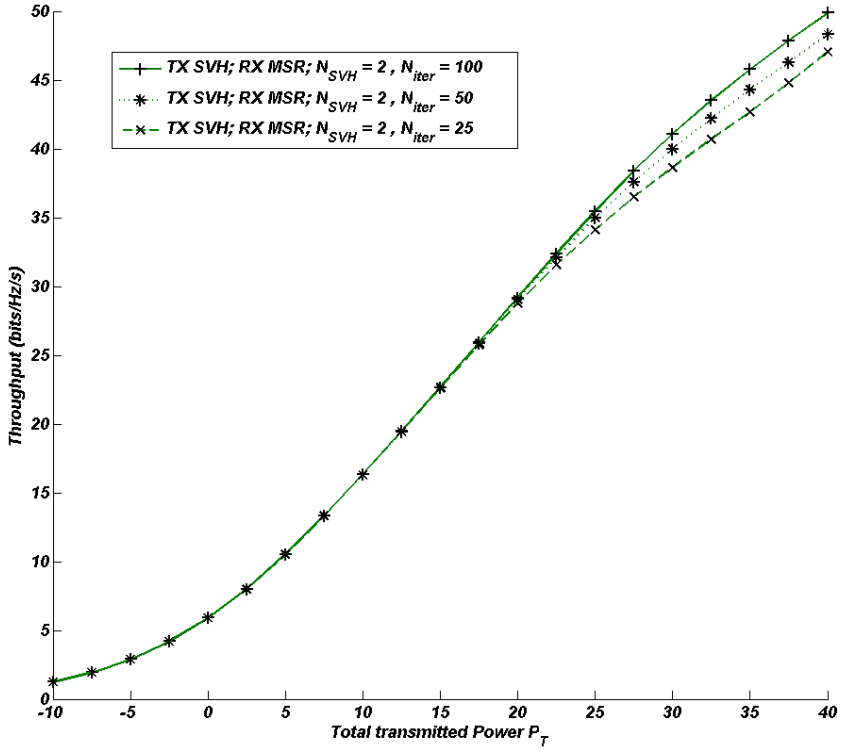


Figure 3.11: Throughput in function of $iter_{max}$ for $N_T = 4$, $N_R = 4$, $K = 4$.

channel. Moreover, the additional gain obtained is decreasing with the increasing number of iterations. This demonstrates the convergence property of the optimization process performed by the SVH/MSR algorithm.

Once we have shown that the algorithms works fine, we must look at its' performances compared to the existing iterative algorithms. In fact, Figure 3.12 represents a comparison of our iterative algorithm with a MMSE/MMSE iterative algorithm proposed in [40] referred to as the *original* one and with a modified version of it that is called *normalized MMSE/MMSE*. These algorithms use a MMSE precoder and a MMSE receiver at each iteration. The original paper proposes an initialization with $\mathbf{d}_k^0 = \mathbf{I}_{1\times N_R}$,where $\mathbf{I}_{1\times N_R}$ has only a one in the first position and zeros elsewhere. The modified version called *normalized*, is introducing a normalization factor applied at the receiver and has been introduced to compare it with the proposed MSR receiver that we showed in section III.B to be equivalent to a normalized MMSE. For these simulations, $N_{iter} = iter_{max}$ is considered to be constant

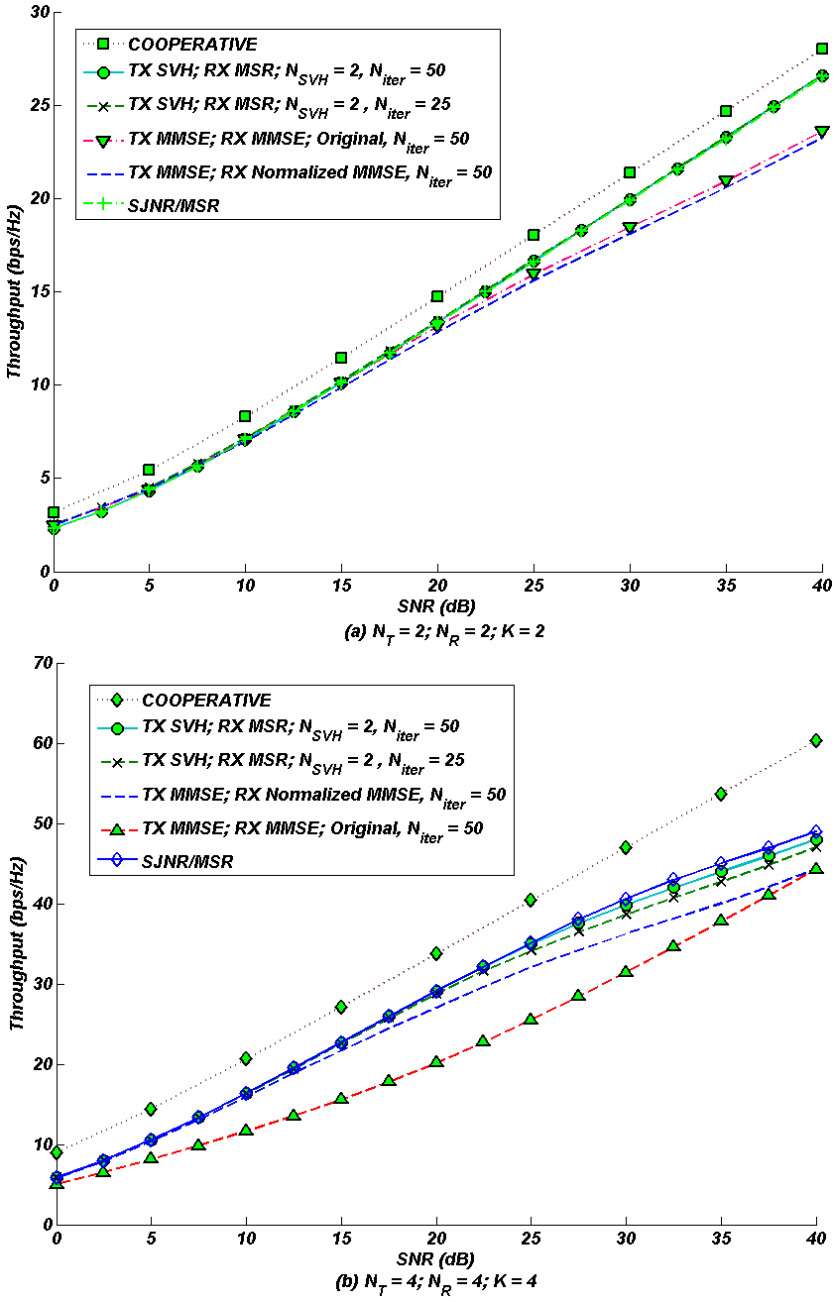


Figure 3.12: Throughput for $N_T = N_R = K = 2$ and $N_T = N_R = K = 4$.

and equals 25 and 50. The results are plotted for two cases. The first one is for $N_T = 2$, $N_R = 2$, $K = 2$ and the second for $N_T = 4$, $N_R = 4$, $K = 4$. Analysing the obtained curves, it can be seen that for the same number of iteration and an equivalent level of complexity, the new algorithm outperforms clearly the MMSE/MMSE one even by using the optimal receiver. Further more, by comparing curves "$TXSVH, RXMSR, N_{SVH} = 2, N_{iter} = 25$" and "$TXSVH, RXMSR, N_{SVH} = 2, N_{iter} = 50$" the obtained performances are very close as only a slight decrease of the sum-rate is noted despite a division by 2 of the number of external iterations $iter_{max}$ performed by **Algorithm 4**. In addition to that, these two curves remain always better than "$TXMMSE, RXMMSEOriginal, N_{iter} = 50$" and "$TXMMSE, RXNormalizedMMSE, N_{iter} = 50$" This result shows, the stability, and the strength of our SVH/MSR algorithm and demonstrates a much faster convergence speed.

To compare the performances of the iterative algorithm based on the eigen values, namely the SJNR precoder and the one based on the MU-MISO optimization, namely the SVH precoder, we plot in these figures the SJNR/MSR curves. Simulation results show that at high SNRs, the SJNR/MSR iterative procedure outperforms our SVH/MSR algorithm. On the other hand, at low SNRs, the SVH/MSR demonstrates sligtly better performances. these ersults can be explained by the fact that at high SNRs, selecting the best singular values for virtual channel construction is more efficient than a global optimization. Whereas at low SNRs, the singular values are close to each other and thus it becomes more crucial to minimize the overlapping of the selected directions.

We also present the cooperative (i.e. single user MIMO on the overall channel $\mathbf{H}^T = \left[\mathbf{H}_1^T \cdots \mathbf{H}_K^T\right]$) curves as a benchmark of the system. The cooperative curves are the highest upper bound of the considered system as it considers perfect cooperation between all users. Comparing the SVH/MSR iterative algorithm with the cooperative curves demonstrate that the SVH/MSR curves remain parallel at low transmit powers but saturates at high SNR when the system dimensions grows. This shows that our SVH/MSR algorithm is able to exploits better the diversity offered by the system especially at low SNRs.

For the following figures, we are going to use the same following notations for the curve names. The $SJNR/MF$ denotes the throughput curve for the SJNR/MF algorithm derived from **Algorithm 2**. The $SJNR/MSR$ describes the SJNR/MSR algorithm from [48] and the $SJNR/MFMSR_D$ corresponds to the curve for this proposed double iterative algorithm.

Figures 3.13 present for a fixed number of total iterations namely $N_{max}^{iter} =$

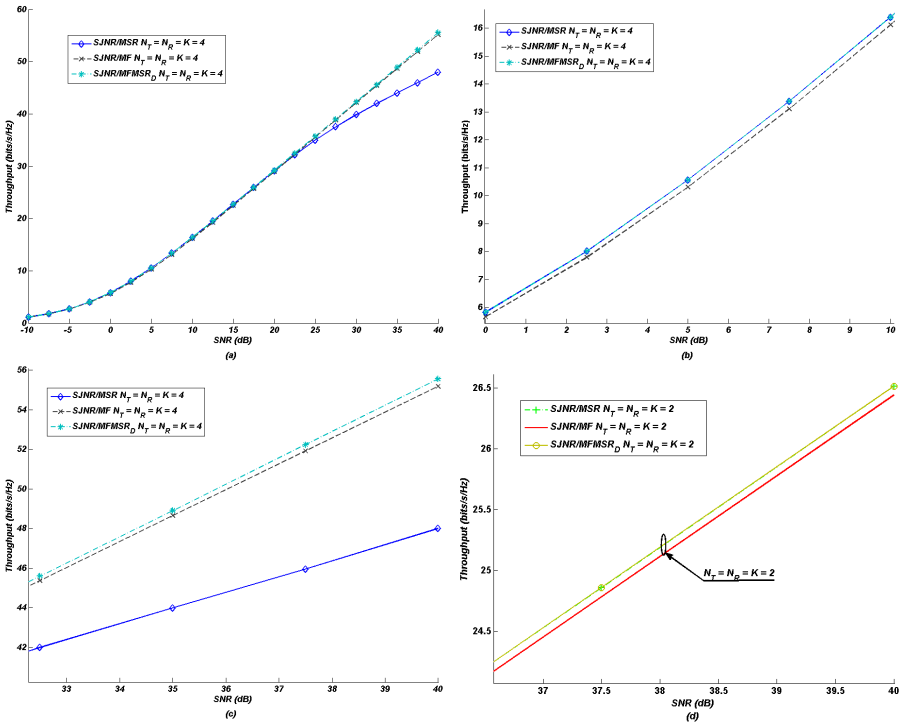


Figure 3.13: Throughput in function of total transmit power P_T for various system configurations.

$iter_{max} = 50$ the total throughput for three configurations of the systems $SJNR/MSR$, $SJNR/MF$ and the proposed $SJNR/MFMSR_D$. The first configuration is a system with $N_T = N_R = K = 4$, the second one has $N_R = 2$ receiving antennas per user. The last represented configuration is for $N_T = N_R = K = 2$. All these configurations present full loaded systems.

These figures show that for the two first configurations, the $SJNR/MF$ algorithm outperforms the $SJNR/MSR$ especially at high SNRs and the obtained throughput curve increases linearly in function of the total transmitted power. This behaviour can be explained by the fact that at high SNRs, the streams can be well separated just by using a matched filter at the reception and through the iterative procedure, the optimal precoder is calculated to maximize the received power for each user. At low SNRs, on the other hand, the MF fails to recover the streams in an optimal way and thus induces suboptimal precoders derivations. But, the MSR receiver is capable of providing a better separation of the users and used with the iterative

procedure to generate better precoders. This explains why the $SJNR/MSR$ algorithm outperforms the $SJNR/MF$ in the low SNRs region.

Based on these observation, we proposed a combination of the two algorithms described in this chapter. The proposed algorithms gives the curve $SJNR/MFMSR_D$. Comparing this curve to the $SJNR/MF$ and $SJNR/MSR$ ones demonstrates a better throughput in all the considered SNR range. The obtained throughputs are even higher than the maximum obtainable by selecting the best among the two considered algorithms. As a matter of fact, analysing Figure 3.13.c shows that at high SNRs, the proposed algorithm gives slightly better throughput performances. At low SNRs, as shown on the curves of Figure 3.13.b the proposed procedure is capable of recovering the best of the two used algorithms offering the best obtainable throughputs.

To verify the stability of our proposed algorithm we have conducted simulations for various system configurations (especially the LTE defined ones). The curves of the third configuration show indeed that the $SJNR/MF$ can no more follow and resolve the best directions for the streams and is therefore worse than the $SJNR/MSR$ one even at very high SNRs.

On the other hand, the proposed $SJNR/MFMSR_D$ algorithm is fully capable of getting the best out of the system based on the two previous algorithms. It even gives some slight ameliorations (in the order of 10^{-4}) compared to the $SJNR/MSR$ (The best algorithms for all considered SNR range). This shows the stability of our algorithm and its convergence for any system configuration. These performances are obtained just by introducing a dynamic flipping procedure that do not introduce any supplementary computational complexity and any extra delay or increase in the number of iterations.

Figure 3.14 represents the throughput achieved by the system when we use the flipping point calculated through statistical analysis given in Figure 3.7. The first curve corresponds to a flipping point ($N_{Flipopt}$) calculated to maximize the mean sum rate of the system. The second curve is the total throughput that we get by applying at a given transmit power level the mean ($N_{Flipmean}$) of the obtained flipping point maximizing instantaneous throughput. It is to be noted that for these two curves we used a lookup table obtained from the statistical analysis to determine the flipping point for each transmit power. These performance curves are confronted to the results obtained with our proposed double iterative algorithm using a dynamic flipping point. The results demonstrate that the proposed algorithm used to switch from one iterative procedure to another gives higher mean sum rates.

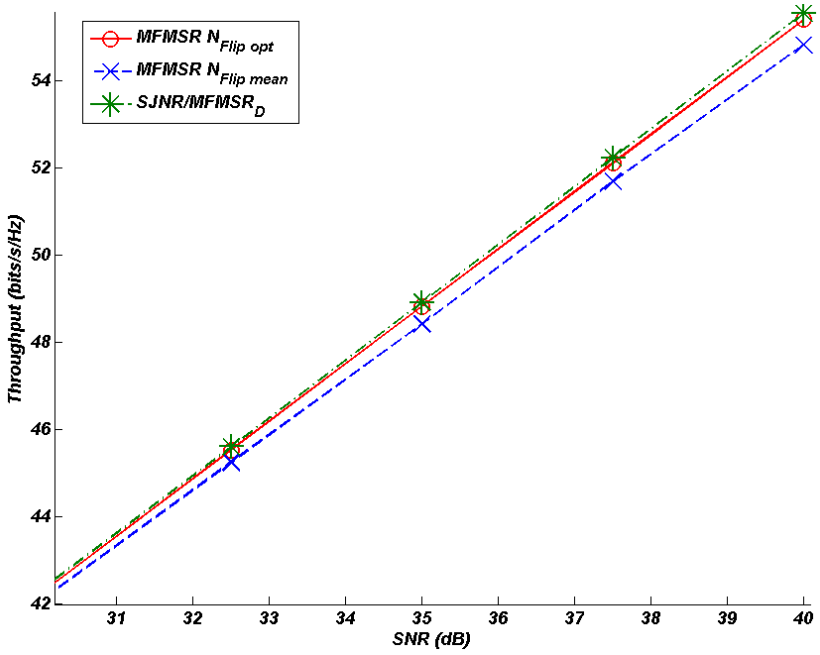


Figure 3.14: Throughput for $N_T = N_R = K = 4$.

Figure 3.15 gives a comparison of the performances obtained by the proposed algorithm and the existing ones in the literature. We added also the curves representing the cooperative algorithm. The DPC curve has been generated using the algorithm proposed in [25] and represents the upper bound for MU-MIMO broadcast systems. These curves show that for the same computational complexity, the proposed solution offers an important gain. Furthermore, compared to the performances given by a cooperative system, the $SJNR/MFMSR_D$ algorithm gives a throughput curve with a slope tending towards the cooperative performances (Ideal system).

Figure 3.16, represents the performances obtained by different SVH-based algorithms. These performances are compared to those achievable with the SJNR/MFMSRD. The curves show that at low SNRs, the achievable rates are always better for the algorithms involving an SVH optimization procedure for the virtual MISO constructed channel. On the other hand, at high SNRs, the SVH algorithms loses some performances. The curve entitled $SJNR/MF + SVH/MSR$ represents the cascade of the two algorithms with the dynamic flipping procedure described in Section IV.A. The overall complexity is the same as the SJNR ones. In fact, as we showed

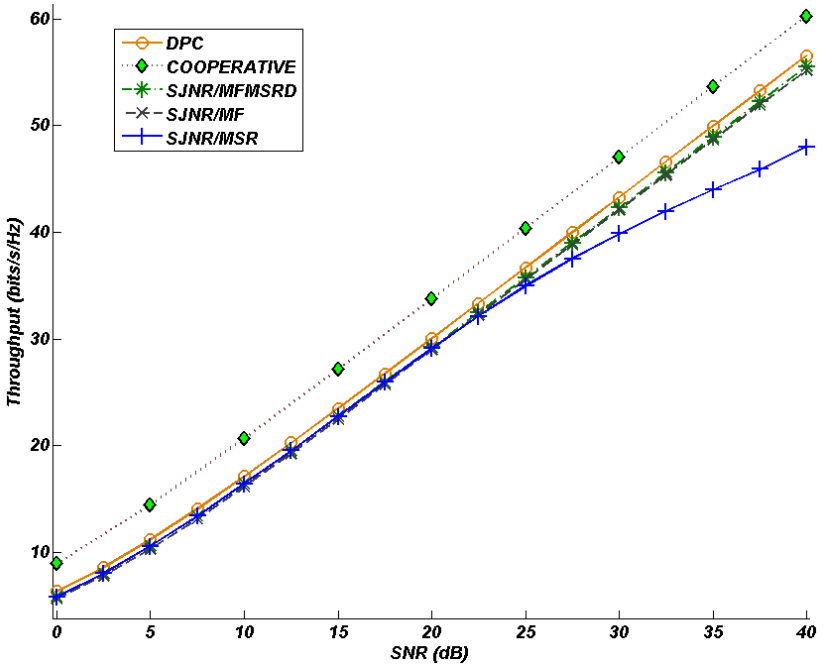


Figure 3.15: Throughput for cooperative, DPC and proposed SJNR/MFMSRD iterative algorithm in case $N_T = N_R = K = 4$.

earlier, the SVH internal optimization procedure (**Algorithm 3**) requires two iterations for convergence. Therefore the number of iterations have been adjusted by counting twice these iterations. The overall results shows almost the sames performances like those obtained with the dynamic double iterative SVH/MFMSRD algorithm. Comparing SJNR/MF + SVH/MSR with $SJNR/MFMSR_D$ shows that a decision must be done in function of the SNR on the choice of the algorithm to apply for precoder optimization. Trying to get a good compromise without to be oblige to establish a second selection procedure, we propose to add a third optimization to the double iterative algorithm with the same criteria and lunching the SVH/MSR algorithm starting from the obtained convergence point.

The last figure for the single stream case, Figure 3.17 represents the system throughput with an $SJNR/MFMSR_D$ algorithm cascaded by an SVH algorithm as a precoder and an MSR receiver for $N_T = N_R = K = 4$ using respectively 2 and 6 internal iterations for the SVH algorithm. The obtained curves demonstrate that the use of an SVH after our double iterative algorithm can improve further the performances at low transmit powers.

For the multi-stream case, Figures 3.18 and 3.19 represents simulation results for a MU-MIMO system with $N_T = 4$ transmitting antennas, $N_R = 4$ receiving antennas per user and $K = 4$ users. The analysis done below remains true for all system configurations (especially the LTE defined ones). The curves $WMMSE/MFMMSED$ obtained with **Algorithm 7** is compared to the $WMMSE/MMSE$ proposed in [42] and $WMMSE/MF$ of **Algorithm 6** with an MF decoder. We add also the single stream dynamic algorithm of **Algorithm 5** denoted as previously SJNR/MFMSRD.

Comparing $WMMSE/MMSE$ and the $WMMSE/MF$ curves shows that the $WMMSE/MF$ gives better performances especially at high SNRs. This behavior can be explained by the fact that at high SNRs, the streams can be well separated just by using a matched filter at the reception and through the iterative procedure, the optimal precoder is calculated to maximize the received power for each user. At low SNRs, on the other hand, the MF filter fails to recover the streams in an optimal way and thus induces suboptimal precoders derivations. But, the MMSE receiver is capable of providing a better separation of the users and reorients iteratively aided by **W** the search towards the least interfering users.

The proposed algorithm gives a curve presenting better throughput in all the considered SNR range. The obtained throughputs are even higher than the maximum obtainable by selecting the best among the two considered algorithms. Analyzing Figures 3.19.a shows that at high SNRs, the proposed algorithm gives better throughput performances. At low SNRs, as shown on the curves in 3.19.b the proposed procedure is capable not only of recovering the best of the two used algorithms but to generate even a slightly better throughput. These results show the stability of our algorithm and its convergence. These performances are obtained just by introducing a dynamic flipping procedure that does not introduce any supplementary computational complexity, any extra delay or any increase in the number of iterations thus no extra processing latency.

Moreover, comparing the curve of the proposed double iterative solution with dynamic selection procedure with the DPC curve shows that the solution is getting very close to the optimal precoding. The two curves are parallel for all SNR range presenting an offset of less than 1 bit/s/Hz. Compared to a DPC (non linear precoders), the proposed double iterative solution present the same slope with a constant offset for all SNR range. This result demonstrates that the obtained rates are the best that can be obtained with a linear proding algorithm according to [36]. This shows the stability and the good convergence of the algorithm containing a quick search using the WMMSE/MF algorithm followed by a refinement procedure with

the WMMSE/MMSE. Finally, comparing the performances to the single stream dynamic case shows equivalent performances. This is an expected result as the system is fully charged and thus the number of streams is equal to the maximal number of transmitting antennas in both cases.

Figure 3.20 gives the performances obtained in a system with $N_T = N_R = 4$ and serving $K = 2$ users. In this case the single stream solutions are under exploiting the system capabilities as only 2 streams are scheduled whereas the multi stream tries to exploit the full diversity by distributing the four available streams on the $K = 2$ users. This explains the difference in slope of the two solutions. This figure also confirms the previous comments.

The last figure, Figure 3.21 represents the impact of the choice of the norm used for the WMMSE/MF. Here we represent the obtained throughputs for the three cases: the Frobenius, the Spectral norm and without the use of a normalization for the MF receiver. The results obtained confirm the previous discussion as the Frobenius norm takes into account the contribution of the received powers in all the directions and so allows a much better optimization for the power allocation research in the first iterative algorithm. This optimization of selection of the best streams to consider is crucial for the final obtained performances as it is used as an initialization point for the second algorithm.

3.9 Conclusion

In this chapter, we considered linear precoding optimization for sum rate maximization. We reviewed some of the solutions presented in the literature and proposed a novel iterative joint optimization procedure for the single and multistream cases. The proposed double iterative procedure combines two iterative sum-rate maximization algorithms based on joint precoder and receiver optimization. For the single stream case the considered algorithms are the SJNR/MF derived from the algorithm in [48] and the SJNR/MSR proposed in [48]. Further improvements at low SNRs are possible by introducing one extra iteration with the SVH/MSR algorithm described in **Algorithm 4** and starting from step 2). Further comparisons done with some existing MMSE/MMSE iterative solution given in [13, 40], showed much better performances with the same complexity levels. For the multi-stream case we proposed the WMMSE/MF described in **Algorithm 6** and the WMMSE/MMSE proposed in [42]. A dynamic switching solution has been proposed to cascade the considered algorithms which allows us to extract

the best of them without introducing further complexity. This solved the burden of finding the optimal flipping point. We also showed throughout the realized simulations that the presented algorithm is not only achieving the best throughput given by the two used algorithms but even gives further gains and thus offers rates closer to the system capacity represented by the DPC as the obtained slope is the same as the one obtained by using a DPC, which is the best that can be done with linear precoding according to [36].

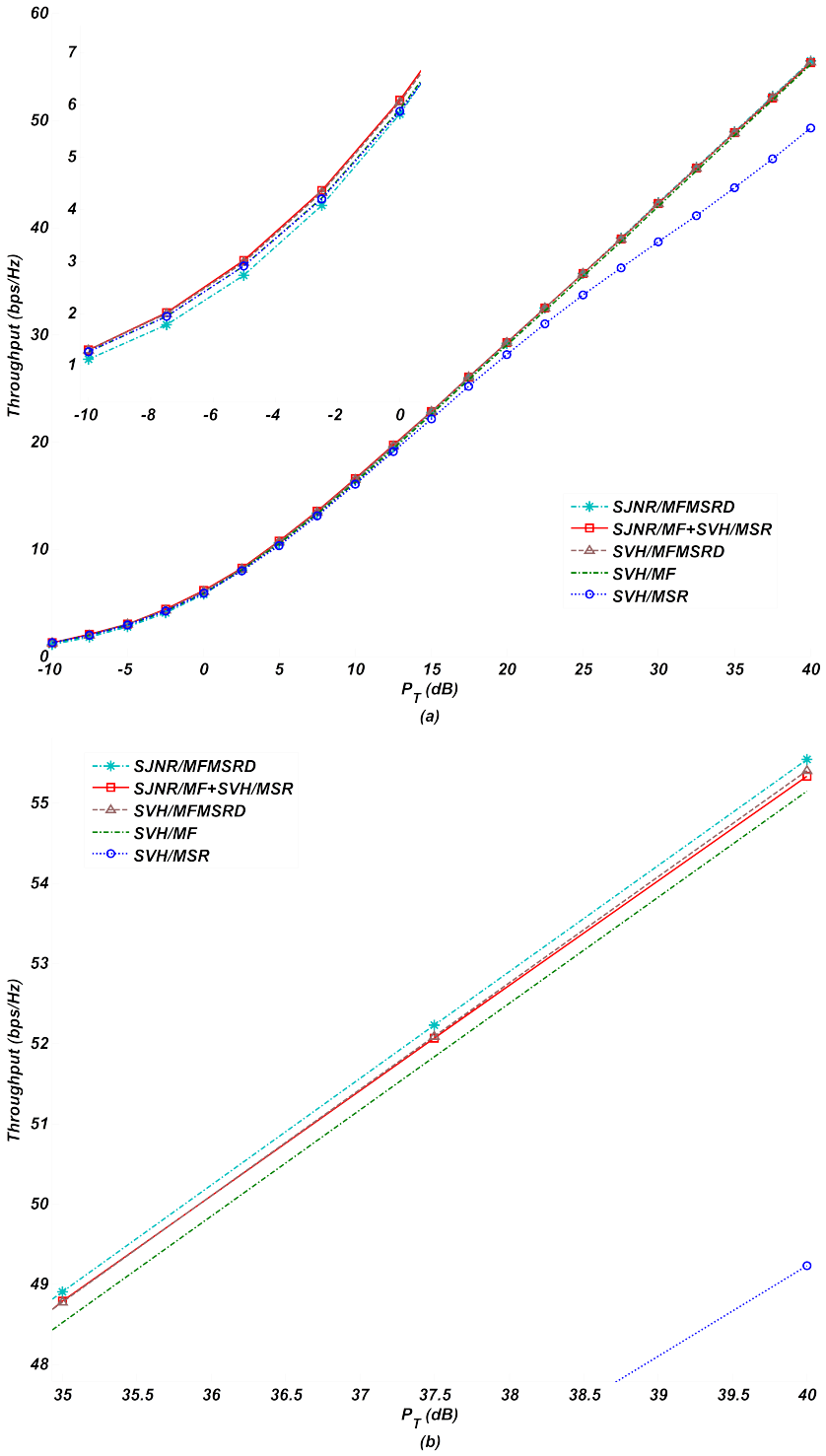


Figure 3.16: Throughput for $N_T = N_R = K = 4$.

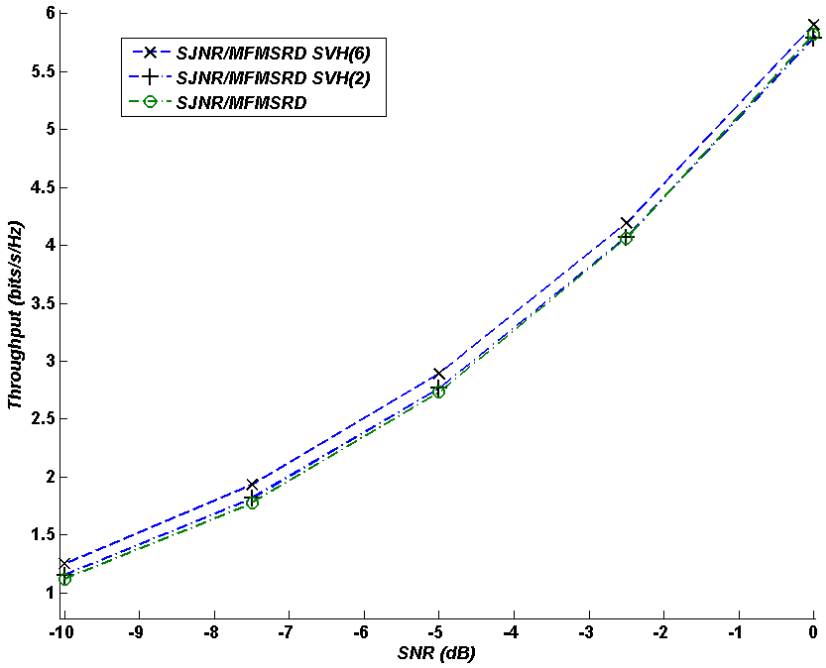


Figure 3.17: Throughput of $SJNR/MFMSR_D$ with extra SVH iterations

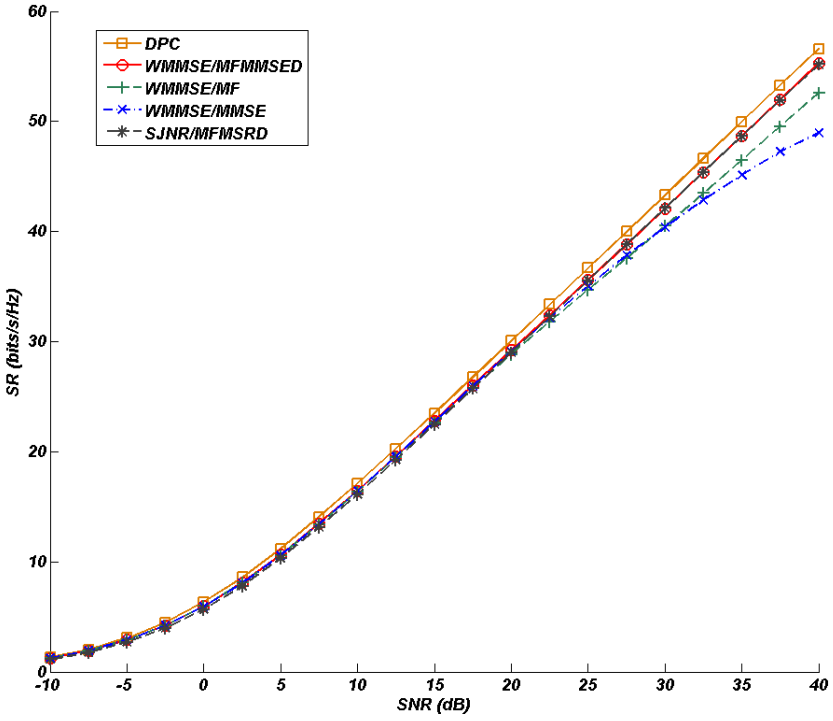


Figure 3.18: Throughput for $N_T = N_R = K = 4$.

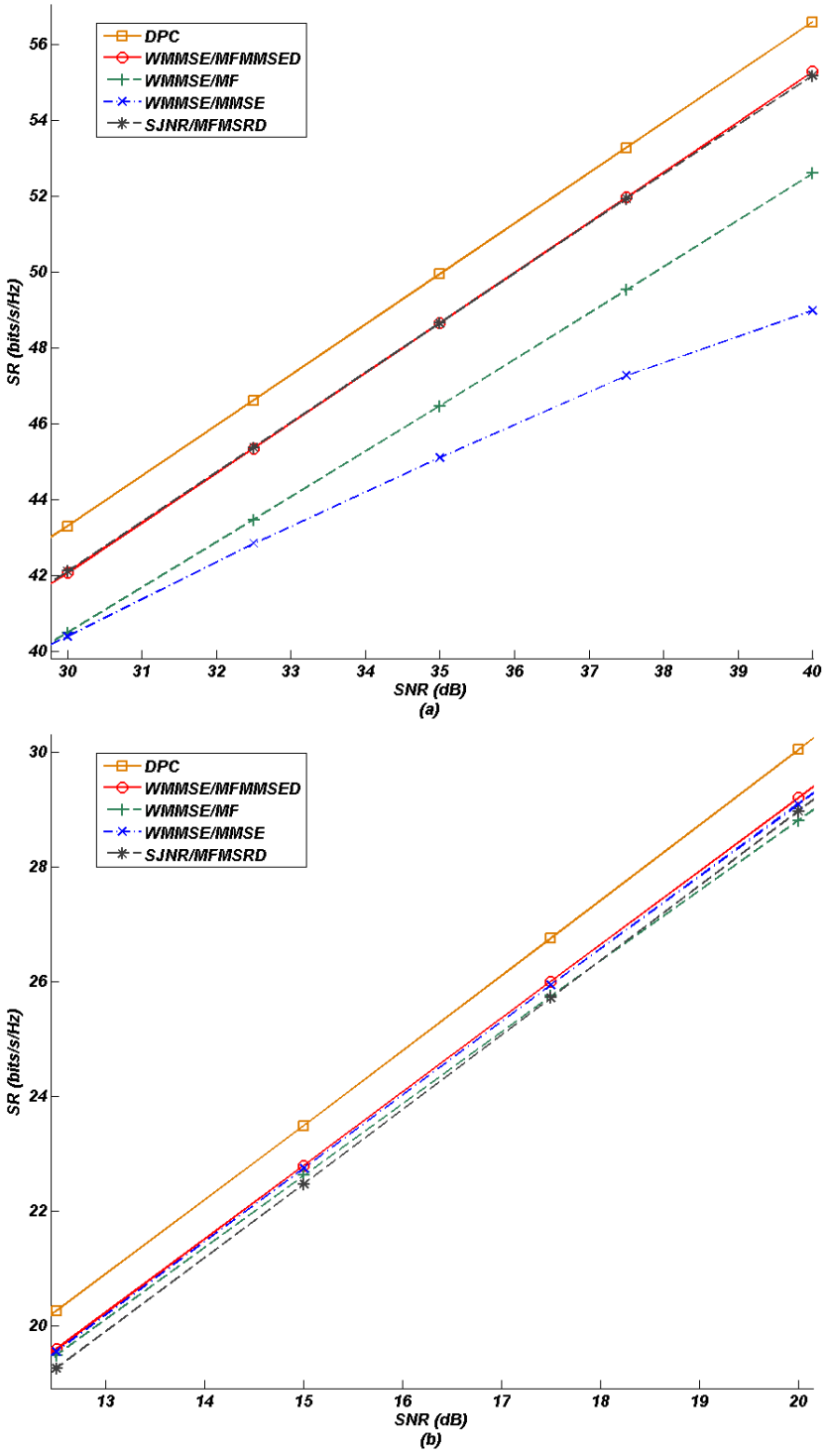


Figure 3.19: Throughput for $N_T = N_R = K = 4$.

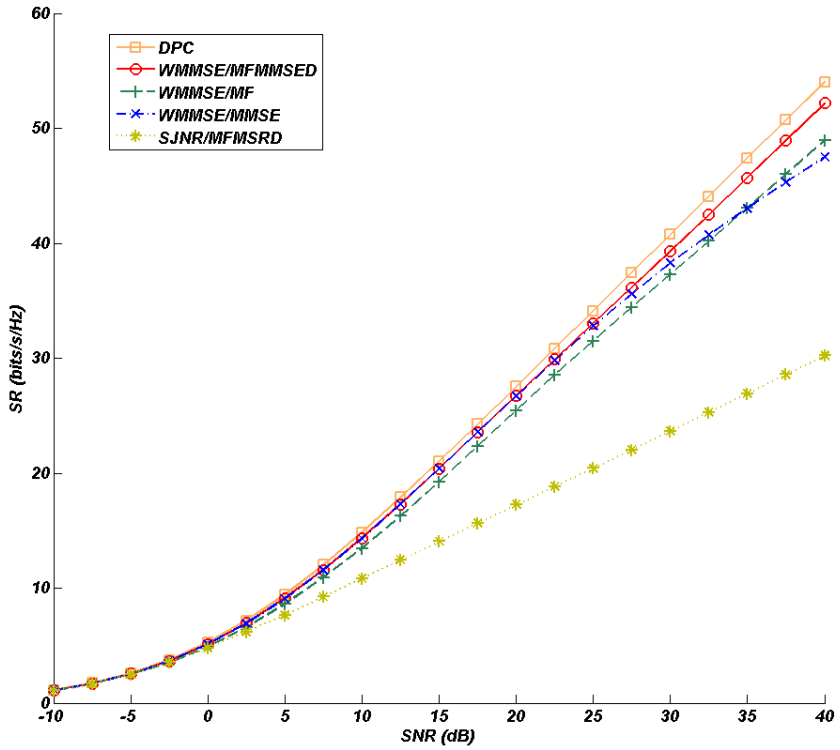


Figure 3.20: Throughput for $N_T = N_R = 4; K = 2$.

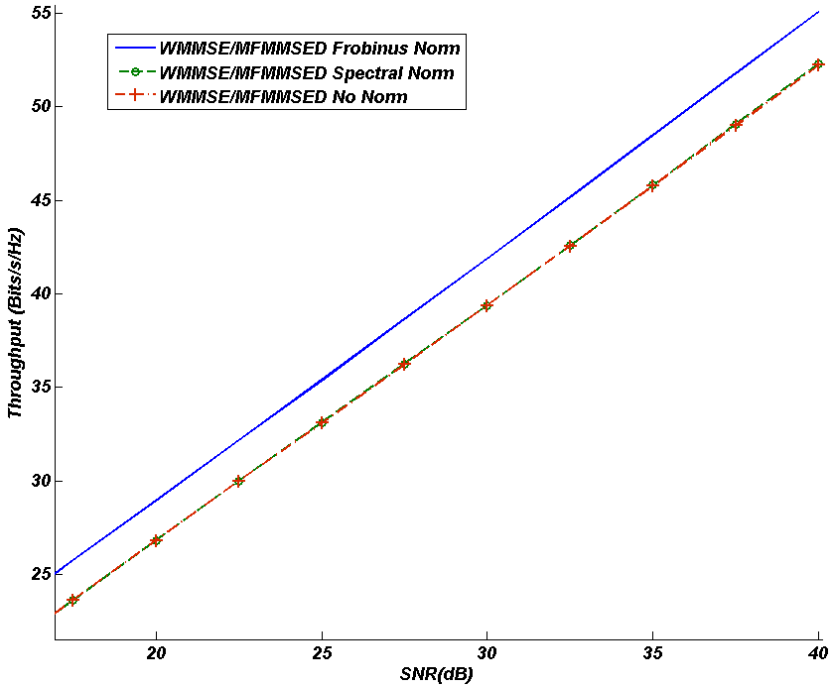


Figure 3.21: Throughput for $N_T = N_R = 4; K = 2$.

Chapter 4

Multiuser MIMO Closed Form Precoding

4.1 Introduction

In the previous chapter, we presented some new iterative solutions for precoder design. We presented some methods to solve one of the main problems faced by these algorithms namely the local optimum convergence. We also proposed extension to support multiple streams per user.

In this chapter we are going to speak about a second family of precoder namely the closed form precoding (CF) techniques. These solutions designed for sum rate maximization are generally a one-formula based solutions or are based on recursive algorithms. This means that the precoder and receiver are computed only once for each user. This makes these algorithms very fast and low resource consuming. On the other hand, they offer poor performances.

Through this chapter, we propose some novel CF precoder design algorithms for the MU-MIMO BC. In the first part, we propose some enhancements to some of the existing closed form solutions namely the SJNR and PU-MMSE precoder. The idea is essentially to further more optimize these algorithms especially through user/stream selection and power allocation methods. In the second part, we present a new zero forcing DPC precoding technique based on successive SVD decompositions. This pro-

posed algorithm constructs stream-wise the precoder in a recursive manner. It contains two complementary steps:

1. The first step consists in a recursive selection of the best available streams to minimize interference between streams. This selection procedure is based on a null space criteria combined with the eigenvalues for each available stream.

2. In the second step, the precoder are computed according to the chosen precoding technique eventually followed by a power distribution optimization.

From this ZF-DPC precoder, we derive a ZF beamforming construction algorithm. The proposed solution constructs based on the selected directions performed by the ZF-DPC algorithm some linear beamformer based on the LU decomposition of the virtual channels generated for each selected stream.

In the next section, a description of the best existing CF single and multi-stream precoding algorithm is provided. Section 4.3 presents some improvements for existing single stream solutions. In Section 4.4, the new precoding algorithm is explained and different forms are exposed and an asymptotic analysis when $K \to \infty$ is derived. The last section contains some simulation results, and demonstrates the performances obtained by the proposed algorithms.

4.2 State of the art

4.2.1 PU-MMSE

The PU-MMSE algorithm proposed in [32] is a single stream per user precoding technique proposing a simplification of the successive MMSE (SMMSE). Here, a precoder is designed for each user taking into consideration all signals transmitted to the other ones as interference. Equation (4.1) is used to compute the precoder.

$$\hat{\mathbf{T}}_k = \left(\tilde{\mathbf{H}}_k^H \tilde{\mathbf{H}}_k + \frac{N_{R_k}}{P_T} N_0 \mathbf{I} \right)^{-1} \hat{\mathbf{H}}_k^H \tag{4.1}$$

Where

$$\tilde{\mathbf{H}}_k^T = \left[\mathbf{H}_1^T \cdots \mathbf{H}_{k-1}^T \mathbf{H}_{k+1}^T \cdots \mathbf{H}_K^T \right] \tag{4.2}$$

and

$$\hat{\mathbf{H}}_k^T = \left[\mathbf{H}_k^T \tilde{\mathbf{H}}_k^T \right] \tag{4.3}$$

To maximize the transmitted power in the best direction, an SVD decomposition is applied to the virtual channel which is composed of the cascade of the channel and the transmitter $\mathbf{H}_k\hat{\mathbf{t}}_k$. And the eigenvector $\mathbf{v}_k^{(1)}$ corresponding to the largest singular value is considered. Finally the obtained precoder is given by

$$\mathbf{t}_k = \beta\hat{\mathbf{T}}_k\mathbf{v}_k^{(1)} \tag{4.4}$$

where

$$\mathbf{H}_k\hat{\mathbf{T}}_k = \mathbf{V}_k\mathbf{\Sigma_k}\mathbf{V_k^H} \tag{4.5}$$

and

$$\beta = \sqrt{\frac{P_T}{tr\left(\sum_{j=1}^{K}\hat{\mathbf{t}}_j\mathbf{v}_j^{(1)}\left(\hat{\mathbf{t}}_j\mathbf{v}_j^{(1)}\right)^H\right)}} \tag{4.6}$$

As a decoder for this system, the left hand eigenvector (LSV) matching the largest singular value of the virtual channel decomposition is used. This ensures that the maximum possible power is localized in the best direction.

$$\mathbf{d}_k = \left(\mathbf{u}_k^{(1)}\right)^H \tag{4.7}$$

4.2.2 SJNR algorithm

The SJNR is defined as the ratio between the useful power for user k over the jamming and noise powers. The jamming power is the total power of the interference caused by the beam aimed at user k and received by the other mobiles. This algorithm was proposed in [27], and considers the generalized singular value of the SJNR expression given in

$$SJNR_k = \frac{\mathbf{t}_k^H\mathbf{H}_k^H\mathbf{H}_k\mathbf{t}_k}{\mathbf{t}_k^H\sum\limits_{j=1,j\neq k}^{K}\mathbf{H}_j^H\mathbf{H}_j\mathbf{t}_k + N_0\mathbf{I}} \tag{4.8}$$

The corresponding precoder for user k is then given by

$$\mathbf{t}_k = \sqrt{P_k}\zeta_m\left[\left(\sum_{j=1,j\neq k}^{K}\mathbf{H}_j^H\mathbf{H}_j + \frac{N_0}{P_k}\mathbf{I}\right)^{-1}\mathbf{H}_k^H\mathbf{H}_k\right] \tag{4.9}$$

Where P_k denotes the transmitted power for user k with $\sum\limits_{k=1}^{K}P_k = P_T$ and $\zeta_m[\mathbf{X}]$ a function computing the largest eigenvector of $\mathbf{X}$. The largest

eigenvector is defined as the eigenvector corresponding to the largest singular value of $\mathbf{X}$. It must be noted that in the case of an hermitian semidefinite positive matrix the singular decomposition is equivalent to an SVD(Singular Value Decomposition) and that the generated singular values are in decreasing order.

As a receiver [27] proposes a matched filter (MF) given by

$$\mathbf{d}_k = \frac{(\mathbf{H}_k \mathbf{t}_k)^H}{\|\mathbf{H}_k \mathbf{t}_k\|} \tag{4.10}$$

4.2.3 SLNR Precoder

The two previously presented closed form solutions, are single-stream-oriented solutions and thus become suboptimal in systems presenting a higher diversity order. Here we call diversity order the number of possible streams for a given user k that can be upper bounded by $\min\left(\sum_{k=1}^{K} N_{R_k}, N_T\right)$. One extension of the SJNR solution has been presented under the name of SLNR. This CF precoding technique is presented in [33] and consists in an extension of the SJNR (Signal to Jamming and Noise Ration) proposed in [27] to support multiple streams per user. Similarly to the SJNR, the authors base their algorithm on the generalized eigen space of the interference part and the desired signals. In fact, this corresponds to the maximization of the signal to leakage and noise ratio. The leakage is defined as the power transmitted over one stream but received by the other ones. The expression for a stream aimed at user k is given by

$$\mathbf{T}_k = \angle \max_{\mathbf{T}_k \in \mathbb{C}^{N \times Q_k}} \frac{tr\left(\mathbf{T}_k^H \mathbf{H}_k^H \mathbf{H}_k \mathbf{T}_k\right)}{tr\left(\mathbf{T}_k^H \left(\sum\limits_{h=1, j \neq k}^{K} \widetilde{\mathbf{H}}_j^H \widetilde{\mathbf{H}}_j + N_0 \mathbf{I}\right) \mathbf{T}_k\right)} \tag{4.11}$$

where $\widetilde{\mathbf{H}}_j = \left[\mathbf{H}_1 \cdots \mathbf{H}_{j-1} \mathbf{H}_{j+1} \cdots \mathbf{H}_K\right]^T$.

The precoders to be considered here are the matrices in the generalized eigenspace of the couple of matrices $\left(\sum\limits_{h=1, j \neq k}^{K} \widetilde{\mathbf{H}}_j^H \widetilde{\mathbf{H}}_j + N_0 \mathbf{I}, \mathbf{T}_k^H \mathbf{H}_k^H \mathbf{H}_k \mathbf{T}_k\right)$ verifying this system of constraints

$$\begin{cases} \mathbf{T}_k^H \mathbf{H}_k^H \mathbf{H}_k \mathbf{T}_k = \Lambda_k \\ \mathbf{T}_k^H \left(\sum_{h=1, j \neq k}^{K} \widetilde{\mathbf{H}}_j^H \widetilde{\mathbf{H}}_j + N_0 \mathbf{I}\right) \mathbf{T}_k = \mathbf{I} \end{cases} \tag{4.12}$$

with Λ_k is a diagonal positive matrice.

This technique offers some advantages as only the covariance matrices are

required and not the entire channel knowledge. Nevertheless, the solution described in [33] imposes the number of stream per user and thus underestimates the real achievable sum-rates and limits the possible applications to real scenarios with greater number of users.

4.2.4 ZFBF-SUS Precoder

To the best of our knowledge, the best linear CF precoder present in the literature is the so called ZFBF-SUS (zero forcing beam forming with successive user selection) that has been proposed in [30, 49]. This recursive precoder design technique is based on the selection of semi-orthogonal users. In the case where each user has more than one receive antennas, the selection is not done on users but on their available streams. To select the best candidate, the SVD of all users are considered. Based on these, the best users are then determined.

The selection process proposed in [49] is a recursive solution where at each recursion the user that generates the least interference to the previously selected ones are chosen. Once the selection procedure accomplished, the authors apply either a ZFBF (zero forcing beam forming) or a DPC precoding on the selected streams. This eliminates the residual part of the interference.

To evaluate the performances of the system, they take the achieved sum-rate (SR) given by formulas [49, (22)] and [49, (9)] respectively for ZFBF-SUS and ZFDPC-SUS.

The overall algorithm is summarized in ***Algorithm 10***.

4.3 Improved Single stream CF precoder

In our study in this section, we base our work on the two first algorithms explained in Section 4.2, and propose two major modifications. The first modification is made at the receiver side; we propose an alternate receiver to optimize the sum rate SR of the system given by

$$SR = \sum_{k=1}^{K} \log_2 \left(1 + \mathbf{d}_k \mathbf{H}_k \mathbf{t}_k \mathbf{t}_k^H \mathbf{H}_k^H \mathbf{d}_k^H \left(\mathbf{d}_k \left(\mathbf{H}_k \sum_{j=1, j\neq k}^{K} \mathbf{t}_j \mathbf{t}_j^H \mathbf{H}_k^H + N_0 \mathbf{I} \right) \mathbf{d}_k^H \right)^{-1} \right) \quad (4.13)$$

Algorithm 10 ZFBF-SUS

1. Selecting the required streams
   - 1.1. Perform an SVD (Singular Value Decomposition) over the channel matrices of all users.
   - 1.2. Select the best available stream. Hence the channel corresponding to the highest singular value.
   - 1.3. Construct a set of streams minimizing the interference part such as: $Interference < \delta$ where δ is a prefixed threshold taken equal to $\frac{1}{log(K)}$. Select among the set the stream with the biggest singular value.
   - 1.4. Repeat recursively step c) until the maximum number of allowed streams $Q = \min(\sum_{k=1}^{K} N_{R_k}, N_T)$ is reached.
2. Construct interferenceless precoders by applying a Zero Forcing Beam Forming over the selected streams: $\mathbf{C}^{\dagger} = \mathbf{C}^H \left(\mathbf{C}\mathbf{C}^H\right)^{-1}$
3. Perform a power distribution using a Waterfilling algorithm (WF).

4.3.1 Receiver design

The proposed receivers in the considered solutions are suboptimal receivers. In fact, the precoding procedure produces one stream per user by performing a beamforming at the transmission minimizing the inter-user interference. However, not all the interference coming from other users is completely eliminated. This shows that the different virtual streams can not be considered as orthogonal and thus makes these receivers suboptimal.

However, seen from the users' part, the system is as a one stream signal part with multiple receive antennas and some interference. To maximize the Signal to Iterference and Noise Ratio of such a system the optimal receiver [3, 10, 11] is an MMSE one given by

$$\mathbf{d}_k = \mathbf{t}_k^H \mathbf{H}_k^H \left(\mathbf{H}_k \left(\sum_{i=1}^{K} \mathbf{t}_i \mathbf{t}_i^H \right) \mathbf{H}_k^H + N_0 \mathbf{I} \right)^{-1} \tag{4.14}$$

That is why we propose a MMSE receiver to increase the total system performance by optimally exploiting the receiving diversity.

4.3.2 User selection procedure

Another way to deal with the inter-user interference is to apply power allocation algorithms. But the design of the system makes it difficult to perform a power allocation to optimize the system performance. Moreover, as the different streams are not orthogonal, the famous Waterfilling technique used in the case of single user MIMO can not be applied in this case. Therefore, we are considering a scheduling procedure to indirectly optimize the power allocation and minimize the impact of "bad" users.

So the second amelioration consists in introducing a power allocation optimization among all users. We select some streams among all the streams of our users, give these selected streams all the transmit power, neglect the non selected users and try to find the best set of streams that maximizes the sum rate of the system.

The algorithm turns on and off one by one the users and evaluates the sum rate obtained. The calculation of the precoders is then performed only on the basis of the selected users using formula (4.4) (respectively formula (4.9)) for the PU-MMSE pre-coder (respectively the SJNR pre-coder). This user selection procedure is feasible thanks to the low computational complexity of the closed form solutions.

4.4 Null Space Zero Forcing

The objective is to maximize the total sum-rate under power constraint (4.15) of a MU-MIMO broadcast channel allowing users to get more than one stream without the use of an iterative solution. Iterative solutions are known to have some problems for convergence [28]. We propose a recursive algorithm by designing the precoder to be used at the base station.

$$P_T \leq \sum_{i=1}^{Q} P_i = \sum_{i=1}^{Q} tr\left(\mathbf{t}_{u_i,f_i}\mathbf{t}_{u_i,f_i}^H\right) \tag{4.15}$$

4.4.1 Main Algorithm

Stream Selection

The proposed algorithm is based on the selection of streams from the null spaces (NS) generated from the previously selected ones by means of orthogonal projections. The algorithm is based on SVD computation of the projected channels at each recursion. This implies, at each step i, to compute the projected channels on the orthogonal space of all the $i-1$ previously

selected precoder

$$\mathbf{H}_i^{\perp} = \mathbf{H}_{i-1}^{\perp}\mathbf{P}_{\mathbf{H}_{S,i-1}}^{\perp} \tag{4.16}$$

where $\mathbf{H}_i^{\perp}$ is the channel of different users at recursion i.
Nevertheless, the important part of the SVD here is the largest singular value and it's corresponding right singular vector. This transforms the SVD computation to a simple largest singular value computation. Moreover, the computational complexity can be further reduced as for each user k

$$\begin{aligned}
\mathbf{H}_{i,k}^{\perp} &= \mathbf{H}_{i-1,k}^{\perp}\mathbf{P}_{\mathbf{H}_{S,i-1}}^{\perp} && (4.17)\\
&= \mathbf{H}_{i-1,k}^{\perp}\left(\mathbf{I} - \mathbf{H}_{S,i-1}^{H}\left(\mathbf{H}_{S,i-1}\mathbf{H}_{S,i-1}^{H}\right)^{-1}\mathbf{H}_{S,i-1}\right) && (4.18)\\
&= \mathbf{H}_{i-1,k}^{\perp} - \mathbf{H}_{i-1,k}^{\perp}\widetilde{\mathbf{t}}_{i-1}\widetilde{\mathbf{t}}_{i-1}^{H} && (4.19)
\end{aligned}$$

where $\mathbf{H}_{S,i-1} = \left[[\mathbf{S}_{u_1}]_1\widetilde{\mathbf{t}}_1, \ldots, [\mathbf{S}_{u_{i-1}}]_1\widetilde{\mathbf{t}}_{i-1}\right]$ is the list of the $i-1$ selected precoder. This constructs a set of beamformers for streams that does not generate any interference with the previously selected ones. These streams can be considered to find the new stream orthogonal to the previously selected directions (i.e. streams). The selection process from this set is done based on the quality of the directions (i.e. their respective singular values). The selection algorithm is detailed in ***Algorithm** 11* steps 0 to 6.

A variant of the algorithm necessary to avoid getting empty selection is to accept some remaining interference after the projection process. For that step 2 of ***Algorithm** 11* can be replaced by where the interference part $A_{i,k}$ is computed according to (4.22)

$$A_{i,k} = \sum_{m=1}^{i-1}[\mathbf{S}_{u_k}]_1\,\mathbf{v}_{u_k,1}^{H}\widetilde{\mathbf{t}}_m \qquad \forall k \in [1\cdots K] \tag{4.22}$$

So the selection procedure proposed here cancels out all the Upper Interference (UI) part defined as the interferences generated by $\{f_j\}_{j>i}$ to the stream $\{f_i\}$.

Precoder construction

The aim of the precoder construction part is to apply a power distributions and to get rid of the remaining parts of the interference that we call lower interference (LI): interferences generated by the streams $\{f_j\}_{j<i}$ to the stream $\{f_i\}$.
To construct the precoder corresponding to the selected streams, different

Algorithm 11 Null Space ZF

0. Initializing $\mathbf{H}_S = \emptyset$; $i = 1$; Set $\mathbf{H}_k^{\perp} = \mathbf{H}_k$.

1. Compute the SVDs for all channels
   $\mathbf{H}_k^{\perp} = \mathbf{U}_k \mathbf{S}_k \mathbf{V}_k^H, \forall k \in [1 \cdots K]$.

2. Create a set with all available streams.
   Select
   $u_i = \arg \max_{k \in [1,\ldots,K]} [\mathbf{S}_k]_1$.
   Increment f_{u_i}.

3. Save the corresponding precoder $\widetilde{\mathbf{t}}_i = \mathbf{v}_{u_i,1}$.
   Update the matrix of selected streams
   $\mathbf{H}_S = \left[\mathbf{H}_S, [\mathbf{S}_{u_i}]_1 \widetilde{\mathbf{t}}_i\right]$

4. Compute the orthogonal projection matrix to the space generated by $\mathbf{H}_S$. This matrix is given by

$$\mathbf{P}_i^{\perp} = \mathbf{I} - \mathbf{H}_S^H \left(\mathbf{H}_S \mathbf{H}_S^H\right)^{-1} \mathbf{H}_S. \tag{4.20}$$

5. Project all channels of all users over the null space

$$\mathbf{H}_k^{\perp} = \mathbf{H}_k \mathbf{P}_i^{\perp} \tag{4.21}$$

6. Increment the counter $i = i + 1$.

   if $i > Q_{max}$ (i.e. the maximum number of streams is reached) **then**
   jump to step 7
   else
   jump to 1
   end if

7. Construct the precoders and define the powers allocation for the streams as described in Section 4.4.1.

2. Compute the interference levels received by all the remaining streams $A_{i,k}$ using (4.22). Create a set containing the streams getting interference less than a desired threshold δ. If the set is empty then select all the streams.
   Select the stream corresponding to the highest singular value.

approaches can be considered. In fact, the easiest way would be to use a DPC precoder that will efficiently eliminate all the Lower Interference part. In this case the equivalent precoders are taken such as (4.23)

$$\mathbf{t}_{u_i,f_i} = \sqrt{P_i}\widetilde{\mathbf{t}}_i, \forall i \in [1 \cdots Q] \tag{4.23}$$

The power distribution can be taken uniformly distributed or allocated according to a Water Filling (WF) algorithms as no interference remains. The WF algorithm is applied on the powers obtained by (4.24) from the computed virtual channels $\mathbf{h}_i$ given in $\mathbf{H}_S$.

$$\lambda_i = \left(\mathbf{h}_i \mathbf{h}_i^H\right), \forall i \in [1 \cdots Q] \tag{4.24}$$

After that, a WF is applied over the λ_i. This generates the power distribution $\{P_i\}_{i\in[1\cdots Q]}$ for the considered streams respecting the total power constraint (4.15). Another possible precoder is ZF precoding that inverts the matrice $\mathbf{H}_S$ containing the selected streams. The precoder are then given by (4.25)

$$\mathbf{t}_{u_i,f_i} = \sqrt{P_i}\mathbf{h}_i^\dagger / \left\|\mathbf{h}_i^\dagger\right\|, \forall i \in [1 \cdots Q] \tag{4.25}$$

where $\mathbf{h}_i^\dagger$ is the i^{th} column of $\mathbf{H}_S^\dagger = \mathbf{H}_S^H \left(\mathbf{H}_S \mathbf{H}_S^H\right)^{-1}$.
$\mathbf{t}_{u_i,f_i}$ represents the precoding vector for stream f_i sent to user u_i and P_i is the power allocated to this stream. The P_i are generated according to the same procedure used for the DPC as no interference remains after ZF.
The overall algorithm is given in ***Algorithm** 11* and a graphical flowchart is given in figure Figure 4.1.

4.4.2 ZFBF-NS with Greedy Selection

In this part, an improvement of the algorithm described above is introduced. The idea here is to optimize the performances obtained by applying a ZF precoding to the selected streams. In fact, in some cases and especially at low SNRs where a low amount of power is available, applying a ZF over Q

selected streams may degrade the performances. This is due to the high level of constraints imposed through the ZF process. In fact, performing a zero-forcing on the maximum number of streams, namly $\min(N_T, \sum_{k=1}^{K} N_{R_k})$, could in some cases try to force the last streams by degrading the performances of the others. In this case, it would be more beneficial from a system point of view to eject the most constraining streams and to redistribute the ressources to a smaller set of streams.

Taking into consideration this observation, we propose solution to solve this problem consists in relaxing some of the constraints by applying a greedy selection procedure while selecting the different streams. The algorithm evaluates at each recursion the impact of introducing the new selected stream. And according to the result the algorithm decides weather to accept the new stream and carry on with the algorithm (in case there are no losses) or to reject it and stop the algorithm.

Taking into account these observations, the new obtained algorithm is described in ***Algorithm*** *12*.

Algorithm 12 ZFBF-NS-G

0. Set $SR_0 = 0$
1. Execute ***Algorithm 11*** until step 3.
2. Evaluate the SR_i using expression (4.26).
3. **if** $SR_i < SR_{i-1}$ **then**
   get $\mathbf{t}_{u_j,f_j}$, $j = i - 1$ jump to 5
   end if
4. **if** $i == Q_{max}$ (i.e. the maximum number of streams is reached) **then**
   get precoders $\mathbf{t}_{u_j,f_j}$, $j = i$ jump to 5
   else
   jump to 2
   end if
5. **END**

$$SR_i = \sum_{j=1}^{i} \log_2 \left(1 + \frac{\mathbf{u}_{u_j,f_j} \mathbf{H}_{u_j} \mathbf{t}_{u_j,f_j} \mathbf{t}_{u_j,f_j}^H \mathbf{H}_{u_j}^H \mathbf{u}_{u_j,f_j}^H}{\mathbf{u}_{u_j,f_j} \left(\mathbf{\Gamma}_j + N_0 \mathbf{I} \right) \mathbf{u}_{u_j,f_j}^H} \right) \tag{4.26}$$

where $\mathbf{\Gamma}_j = \mathbf{H}_{u_j} \sum_{m=1,m\neq j}^{i} \mathbf{t}_{u_m,f_m} \mathbf{t}_{u_m,f_m}^H \mathbf{H}_{u_j}$ is the remaining interference.

In addition to the improvements in the obtained sum-rate, these modifications give us the possibility to reduce the complexity of the system by avoiding extra useless computations. A flowchart of the algorithm is given in figure Figure 4.2

4.4.3 Large System Analysis $K \to \infty$

Fist we are going to make an analysis with a growing number of users to confirm the convergence of the algorithm towards the DPC performances. For that, we base our analysis on the eigenvalue distribution. In fact according to [36] the linear precoder and the DPC performances have the same slope in function for the transmission power. They differ on the other hand with a constant term given by $\min(\sum_{k=1}^{K} N_{R_k}, N_T) \log_2(\min(\sum_{k=1}^{K} N_{R_k}, N_T))$. As we can see from this expression, the difference depends on the number of independent eigenvalues and thus ZF-DPC converges to the perfect DPC case when K grows.

Therefore, by increasing the number of users K and keeping a constant number of receiving N_R and transmitting antennas N_T we increase the probability of finding a good solution minimizing the ZFBF and the DPC gap.

To perform the large system analysis, we are going to start by computing the gap between the DPC and the DPC version of the null space proposed solution in our algorithm by supposing all lower interference equal to 0 thanks to the use of DPC. We are also going to consider ***Algorithm 11*** with the second option for step 2 and remind the property given for the largest eigenvalue distribution known to be a complex Wishart distribution. This can be given according to [53] by (4.27)

$$f_{\lambda_{\max}}(i,x) = \frac{\sum_{j=1}^{N_T-i+1} \sum_{m=|N_T-N_{RT}|}^{(N_T+N_{RT}-2i-j+1)j} c_{j,m} e^{-jx} x^m}{\prod_{j=1}^{N_T-i+1} (N_{RT}-j)!\,(N_T-j)!} \tag{4.27}$$

$$= \sum_{j=1}^{N_T-i+1} \sum_{m=|N_T-N_{RT}|}^{(N_T+N_{RT}-2i-j+1)j} d_{j,m} e^{-jx} x^m \tag{4.28}$$

where $c_{j,m}$ is a constant independent of the size of the channel matrices and $N_{RT} = \sum_{k=1}^{K} N_{R_k}$.

And the corresponding CDF can be written as (4.29).

$$F_{\lambda_{\max}}(i,x) = \sum_{j=1}^{N_T-i+1} \sum_{m=|N_T-N_{RT}|}^{(N_T+N_{RT}-2i-j+1)j} \frac{d_{j,m}}{j^{m+1}} \left(\Gamma\left(m+1\right) - \Gamma\left(m+1, jx\right)\right) \quad (4.29)$$

here $\Gamma\left(m,x\right)$ is the gamma function and the term $\Gamma\left(m+1\right) - \Gamma\left(m+1,jx\right)$ can be replaced by the lower incomplete gamma function $\gamma(m+1,jx)$.

The eigen value distribution is only one part of the SINR component present in the capacity function. The decrease of the eigen value due to successive projection should also be taken into account. This quantity will be called at each recursion $F_{Weight}\left(i,x\right)$.

$$\begin{aligned} &F_{Weight}\left(i,x\right) = \\ &= \Pr\left\{A^2_{u_i,f_i} \leqslant x | A^2_{u_1,f_1} \leqslant \delta, ..., A^2_{u_{i-1},f_{i-1}} \leqslant \delta\right\} \quad (4.30) \\ &= \frac{\Pr\left\{A^2_{u_1,f_1} \leqslant \delta, ..., A^2_{u_{i-1},f_{i-1}} \leqslant \delta, A^2_{u_i,f_i} \leqslant x\right\}}{\Pr\left\{A^2_{u_1,f_1} \leqslant \delta, ..., A^2_{u_{i-1},f_{i-1}} \leqslant \delta\right\}} \quad (4.31) \end{aligned}$$

The weights are obviously decreasing with the number of recursions as at each new selected stream, a projection is done according to (4.20) and thus the eigenvalues obtained from the new reconstructed channel (4.21) are lower than the previous ones. In addition to that power allocation should be taken into account as we perform water filling after eliminating all possibly remaining lower interference terms. The distribution of the SINR is then given by (4.32)

$$F_{SINR}\left(i,x\right) = \int_0^{+\infty} P_i f_{\lambda_{\max}}\left(i,z\right) F_{Weight}\left(i,\frac{x}{z}\right) dz \quad (4.32)$$

The power distribution $\{P_i\}_{1\leq i\leq N_T}$ are just scalar factors and can be introduced at the end of the derivations.

Nevertheless, the distribution of $F_{Weight}\left(i,x\right)$ is difficult to compute and the rigorous expression is not really required for convergence analysis. All we need is a lower bound to demonstrate the large K convergence. If we look closer at this term, it represents the probability of getting a weight larger than x conditioned by having stream i in the set constructed through option a) of ***Algorithm** 11* with the second option for step 2 and having previously successfully selected $i-1$ streams i.e. all the interferences $A_{u_i,f_i} \leq \delta, i \in [\![1,..,i-1]\!]$.

The probability of selecting the $i-1$ first streams i.e.

$\Pr\left\{A^2_{u_1,f_1} \leqslant \delta, ..., A^2_{u_{i-1},f_{i-1}} \leqslant \delta\right\}$
is the probability of $i-1$ random variables that are bounded such as

$$\left\|\left(A^2_{u_1,f_1}, ..., A^2_{u_{i-1},f_{i-1}}\right)\right\| \leqslant \sqrt{\sum_{j=1}^{i-1} \delta^2} \tag{4.33}$$

Following the method given in [52, page 34], the corresponding distribution can be computed starting from the joint p.d.f. of the vector $\left(|z_1|^2, ..., |z_{N_T}|^2\right)$ with z_i bounded complex random variables. Without loss of genarality, we can take them such as $\|(z_i)_{1\leq i\leq N_T}\| \leq 1$.

This distribution corresponds to the one given by an N_T dimensional hypersphere and so we get :

$$f\left(z_1, ..., z_{N_T}\right) = \Gamma\left(N_T\right)/2\pi^{N_T} \tag{4.34}$$

Performing now a variable change such as $x_{2i-1} + jx_{2i} = z_i$,where j is the imaginary number such as $j^2 = -1$ gives us the joint probability of the $2(i-1)$ terms as

$$\begin{aligned} f\left(x_1, ..., x_{2i-2}\right) \quad &= \Gamma\left(N_T\right)/2\pi^{N_T} \\ &\int ... \int_{\sum_{j=2i-1}^{2N_T} x_j^2 = \sum_{j=1}^{2i-2} x_j^2} dx_{2i-1} dx_{2N_T} \end{aligned} \tag{4.35}$$

This integral can be transformed using [52, page 35] to get

$$f\left(x_1, ..., x_{2i-2}\right) = \frac{\Gamma\left(N_T\right)\pi^{1-i}\left(1 - \sum_{j=1}^{2i-2} x_j^2\right)^{N_T-i}}{\Gamma\left(N_T - i + 1\right)} \tag{4.36}$$

A polar transformation to recombine the elements such as :

$$\begin{cases} x_{2j-1} &= r_j \cos\left(\theta_j\right) \\ x_{2j} &= r_j \sin\left(\theta_j\right) \end{cases} \tag{4.37}$$

transforms the p.d.f. such as

$$f\left(r_1, ..., r_{i-1}\right) = \frac{2^{i-1}\Gamma\left(N_T\right)\left(1 - \sum_{j=1}^{i-1} r_j^2\right)^{N_T-i} \prod_{j=1}^{i-1} r_j}{\Gamma\left(N_T - i + 1\right)} \tag{4.38}$$

as the Jacobian is given by $\frac{2^{i-1}}{\pi^{1-i}} \prod_{j=1}^{i-1} r_j$ Finally to get the desired joint probability $f\left(|z_1|^2, ..., |z_{i-1}|^2\right)$ we pose $\left|z_j\right|^2 = r_j^2$ as the Jacobian is given by $2^{1-i}\left(\prod_{j=1}^{i-1} r_j^2\right)^{-1/2}$ is

$$f\left(|z_1|^2, ..., |z_{i-1}|^2\right) = \frac{\Gamma\left(N_T\right)\left(1 - \sum_{j=1}^{i-1}\left|z_j\right|^2\right)^{N_T - i}}{\Gamma\left(N_T - i + 1\right)} \tag{4.39}$$

From this expression and following a recurrence approach it is simple to express it in function of the previous selection step as equations (4.40)-(4.49).

$$\begin{aligned}
&\Pr\left(\left|z_1^2\right| \leqslant \delta, ..., \left|z_{i-1}^2\right| \leqslant \delta\right) \\
&\quad = \frac{\Gamma\left(N_T\right)}{\Gamma\left(N_T - i + 1\right)} \\
&\qquad \int ... \int_{\left|z_1^2\right| \leqslant \delta, ..., \left|z_{i-1}^2\right| \leqslant \delta} \left(1 - \sum_{j=1}^{i} \left|z_j\right|^2\right)^{N_T - i} d\left|z_1^2\right| ... d\left|z_{i-1}^2\right| \quad (4.40) \\
&\quad = \frac{\left(N_T - i + 1\right)^{-1}\Gamma\left(N_T\right)}{\Gamma\left(N_T - i + 1\right)} \\
&\qquad \left[\left(\int ... \int_{\left|z_1^2\right| \leqslant \delta, ..., \left|z_{i-2}^2\right| \leqslant \delta} \left(1 - \sum_{j=1}^{i-2} \left|z_j\right|^2\right)^{N_T - i + 1} d\left|z_1^2\right| ... d\left|z_{i-2}^2\right|\right) - \right. \\
&\quad \left.\left(\int ... \int_{\left|z_1^2\right| \leqslant \delta, ..., \left|z_{i-2}^2\right| \leqslant \delta} \left(1 - \delta - \sum_{j=1}^{i-2} \left|z_j\right|^2\right)^{N_T - i + 1} d\left|z_1^2\right| ... d\left|z_{i-2}^2\right|\right)\right] \quad (4.41)
\end{aligned}$$

$$= ....$$

$$= \frac{\Gamma(N_T)\sum_{j=0}^{i-2}(-1)^j C_{i-2}^j \int_{|z_1^2|\leqslant\delta}\left(1-j\delta-|z_1|^2\right)^{N_T-3} d|z_1^2|}{\Gamma(N_T-i+1)\prod_{j=1}^{i-2}(N_T-i+j)} \tag{4.42}$$

$$= \frac{\Gamma(N_T)\sum_{j=0}^{i-2}(-1)^j C_{i-2}^j \int_0^{\delta}(1-j\delta-x)^{N_T-2} dx}{\Gamma(N_T-i+1)\prod_{j=1}^{i-2}(N_T-i+j)} \tag{4.43}$$

$$= \frac{\Gamma(N_T)\sum_{j=0}^{i-2}(-1)^j C_{i-2}^j \frac{(1-j\delta)^{N_T-1}-(1-(j+1)\delta)^{N_T-1}}{N_T-1}}{\Gamma(N_T-i+1)\prod_{j=1}^{i-2}(N_T-i+j)} \tag{4.44}$$

$$= \frac{\Gamma(N_T)\sum_{j=0}^{i-2}(-1)^j C_{i-2}^j \left[(1-j\delta)^{N_T-1}-(1-(j+1)\delta)^{N_T-1}\right]}{\Gamma(N_T-i+1)\prod_{j=1}^{i-1}(N_T-i+j)} \tag{4.45}$$

as $\prod_{j=0}^{i-2} N_T-i+j = \frac{\Gamma(N_T)}{\Gamma(N_T-i+1)}$ this expression becomes

$$\Pr\left(|z_1^2|\leq\delta,...,|z_{i-1}^2|\leq\delta\right)$$

$$= \sum_{j=0}^{i-2}(-1)^j\left(C_{i-1}^j - C_{i-2}^{j-1}\right)\left[(1-j\delta)^{N_T-1}-(1-(j+1)\delta)^{N_T-1}\right] \tag{4.46}$$

$$= \left[\sum_{j=0}^{i-2}(-1)^j C_{i-1}^j(1-j\delta)^{N_T-1} + \sum_{j=1}^{i-1}(-1)^j C_{i-1}^{j-1}(1-j\delta)^{N_T-1} - \right.$$
$$\sum_{j=0}^{i-2}(-1)^j C_{i-2}^{j-1}(1-j\delta)^{N_T-1} - \sum_{j=1}^{i-1}(-1)^j C_{i-1}^{j-1}(1-j\delta)^{N_T-1} +$$
$$\left.\sum_{j=1}^{i-1}(-1)^j C_{i-2}^{j-1}(1-j\delta)^{N_T-1}\right] \tag{4.47}$$

$$= \sum_{j=0}^{i-1}(-1)^j C_{i-1}^j(1-j\delta)^{N_T-1} \tag{4.48}$$

$$\Pr\left(\left|z_1^2\right| \leq \delta, ..., \left|z_{i-1}^2\right| \leq \delta\right) = \sum_{j=0}^{i-1} (-1)^j C_{i-1}^j \sum_{k=0}^{N_T-1} (-1)^k C_{N_T-1}^k j^k \delta^k \quad (4.49)$$

Going back to the $F_{Weight}\left(i, \frac{x}{z}\right)$ distribution, and considering the same way the probability that the power of element i is $A_{i,k} \leq \delta$ we obtain the following c.d.f. equations (4.50)-(4.4.3).

$$\begin{aligned} F_{Weight}(i,x) &= 1 - \frac{\Gamma(N_T)}{\Gamma\left(N_T - i + 1\right) \sum_{j=0}^{i-1} (-1)^j C_{i-1}^j \sum_{k=0}^{N_T-1} (-1)^k C_{N_T-1}^k j^k \delta^k} \\ &\int ... \int_{|z_1^2| \leqslant 1-x-\sum_{k=2}^{i-1}|z_k^2|, ..., |z_{i-2}^2| \leqslant 1-x-|z_{i-1}^2|, |z_{i-1}^2| \leqslant 1-x} \left(1 - \sum_{j=1}^{i-1} \left|z_j^2\right|\right)^{N_T-i} d\left|z_1^2\right| ... d\left|z_{i-1}^2\right| \quad (4.50) \\ &\geq 1 - \frac{\Gamma(N_T)}{\Gamma\left(N_T - i + 1\right) \sum_{j=0}^{i-1} (-1)^j C_{i-1}^j \sum_{k=0}^{N_T-1} (-1)^k C_{N_T-1}^k j^k \delta^k} \\ &\int ... \int_{\sum_{k=1}^{i-1}|z_k^2| \leq 1-x} \left(1 - \sum_{j=1}^{i-1} \left|z_j^2\right|\right)^{N_T-i} d\left|z_1^2\right| ... d\left|z_{i-1}^2\right| \quad (4.51) \end{aligned}$$

$$\begin{aligned} F_{Weight}^L(i,x) &= 1 - \frac{\Gamma(N_T) \int_{Z=0}^{1-x} (1-Z)^{N_T-i} Z^{i-2} dZ}{(i-2)!\Gamma\left(N_T - i + 1\right) \sum_{j=0}^{i-1} (-1)^j C_{i-1}^j \sum_{k=0}^{N_T-1} (-1)^k C_{N_T-1}^k j^k \delta^k} \quad (4.52) \\ &= 1 - \frac{\Gamma(N_T)(1-x)^{i-1} {}_2F_1(i-1, i-N_T, i, 1-x)}{(i-1)!\Gamma\left(N_T - i + 1\right) \sum_{j=0}^{i-1} (-1)^j C_{i-1}^j \sum_{k=0}^{N_T-1} (-1)^k C_{N_T-1}^k j^k \delta^k} \quad (4.53) \end{aligned}$$

is obtained by using [51, Page 320]. And finally the obtained lower bound for the SINR distribution $F^L_{SINR}(i,x)$ is given by (4.55)

$$F^L_{SINR}(i,x) = \int_0^{+\infty} f_{\lambda_{\max}}(i,z)\, F^L_{Weight}\left(i,\frac{x}{z}\right) dz \tag{4.54}$$

$$\begin{aligned} &= \int_0^{x} f_{\lambda_{\max}}(i,z)\, F^L_{Weight}\left(i,\frac{x}{z}\right) dz + \\ &\int_x^{x/(1-i\delta+\delta)} f_{\lambda_{\max}}(i,z)\, F^L_{Weight}\left(i,\frac{x}{z}\right) dz + \\ &\int_{x/(1-i\delta+\delta)}^{+\infty} f_{\lambda_{\max}}(i,z)\, F^L_{Weight}\left(i,\frac{x}{z}\right) dz \end{aligned} \tag{4.55}$$

We can clearly see from equation (4.51) that for $x \leq 1-i\delta+\delta \Leftrightarrow z \geq \frac{x}{1-i\delta+\delta}$ the weight distribution is equal to 0 as we are integrating on an empty set (condition (4.33) is not satisfied). On the other hand, if $x > 1 \Leftrightarrow z \leq x$ then the condition (4.51) is always achieved and the c.d.f. becomes equal to 1.

And due to the distribution of the previously obtained $F^L_{Weight}\left(i,\frac{x}{z}\right)$ the last integral is equal to 0 and the first one gives $F_{\lambda_{\max}}(i,x)$. Integrating by parts the second integral leads to (4.56).

$$\begin{aligned} F^L_{SINR}(i,x) \quad = \quad & \\ & F_{\lambda_{\max}}\left(\frac{x}{1-i\delta+\delta}\right) - \frac{\sum_{k=i-1}^{N_T-1} C^k_{N_T-1}(1-i)^{-k}\left(\sum_{j=0}^{i-1}(-1)^j C^j_{i-1} j^k\right)}{\sum_{k=i-1}^{N_T-1}(-1)^k C^k_{N_T-1}\sum_{j=0}^{i-1}(-1)^j C^j_{i-1}(j\delta)^k} \\ & \sum_{l=1}^{N_T-i-1}\sum_{m=|N_T-N_{RT}|}^{(N_T+N_{RT}-2i-l+1)l}\left(d_{l,m}\sum_{n=0}^{k}(-x)^{k-n}C^n_k l^{k-n-m-1}\right. \\ & \left.\left(\Gamma(n-k+m+1,lx) - \Gamma\left(n-k+m+1,\frac{lx}{1-i\delta+\delta}\right)\right)\right) \end{aligned} \tag{4.56}$$

This lower bound expression remains nevertheless complex. One way of simplifying it is to use the incomplete gamma function development in the c.d.f. expression of the eigenvalues (4.29). In fact according to [51, Page 899],

$$\Gamma(m+1,ix) = m!e^{-ix}\left(\sum_{k=0}^{m}\frac{i^k x^k}{k!}\right) \tag{4.57}$$

By injecting this development into the eigenvalues c.d.f. and using $\Gamma(m+1) = m!$ we get

$$F_{\lambda_{\max}}(i,x) = 1 - \frac{e^{-x}x^{N_T-i+N_{RT}-i}\left(1+O(e^{-x}x^{-1})\right)}{(N_T-i-1)!(N_{RT}-i-1)!} \tag{4.58}$$

and

$$f_{\lambda_{\max}}(i,x) = \frac{e^{-x}x^{N_T-i+N_{RT}-i}\left(1+O(e^{-x}x^{-1})\right)}{(N_T-i-1)!(N_{RT}-i-1)!} \tag{4.59}$$

Injecting this result into equation (4.56) and considering from (4.59) that $d_{l,m} = \frac{1}{(N_T-i-1)!(N_{RT}-i-1)!}$ we can obtain an approximation (4.61) of the lower bound for the c.d.f..

$$\begin{aligned} F^L_{SINR}(i,x) \quad &= 1 - \frac{e^{-\frac{x}{1-i\delta+\delta}}x^{N_T-i+N_{RT}-i}(1-i\delta+\delta)^{2i-N_T-N_{RT}}}{(N_T-i-1)!(N_{RT}-i-1)!} - \\ &\frac{e^{-lx}\sum_{k=i-1}^{N_T-1} C^k_{N_T-1}(1-i)^{-k}\left(\sum_{j=0}^{i-1}(-1)^j C^j_{i-1} j^k\right)}{(N_T-i-1)!(N_{RT}-i-1)!\sum_{k=i-1}^{N_T-1}(-1)^k C^k_{N_T-1}\sum_{j=0}^{i-1}(-1)^j C^j_{i-1}(j\delta)^k} \\ &\sum_{l=0}^{N_T-i+1}\sum_{m=|N_T-N_{RT}|}^{(N_T+N_{RT}-2i-l+1)l}(n-k+m)!\left(\sum_{n=0}^{k}(-x)^{k-n}C^n_k l^{k-n-m-1}\right. \\ &\left.\left(\sum_{a=0}^{n-k+m}\frac{l^a x^a}{a!}\left(1-(1-i\delta+\delta)^{-a}e^{1-i\delta+\delta}\right)\right)\right) - O(e^{-x}x^{N_T-i+N_{RT}-2i-1}) \end{aligned} \tag{4.60}$$

$$\begin{aligned} &\approx 1 - \frac{\Gamma(i)\,e^{-x}x^{N_T+N_{RT}-2i}}{\sum_{k=i-1}^{N_T-1}(-1)^k C^k_{N_T-1}\sum_{j=0}^{i-1}(-1)^j C^j_{i-1}(j\delta)^k\Gamma(N_T-i+1)\,\Gamma(i-1)(i-1)^{i-2}} \\ &\qquad + O\left(e^{-x}x^{N_T+N_{RT}-2i-1}\right) \end{aligned} \tag{4.61}$$

As $F^L_{SINR}(i,x)$ represents a lower bound this property, gives us that:

$$Pr\left\{u(i) - \log\log\sqrt{K} \leqslant F_{SINR}(i,x)\right\} \geqslant 1 - O\left(\log(K)^{-}1\right) \tag{4.62}$$

where

$$\begin{aligned} u(i) = \log&\left(\frac{\Gamma(i)\,K(i-1)^{1-i}}{\Gamma(N_T-i+1)\,\Gamma(i-1)}\right) + \\ &\log\log\left(\frac{\Gamma(i)\,K(i-1)^{1-i}}{\Gamma(N_T-i+1)\,\Gamma(i-1)}\right)^{(N_T+N_{RT}-2i)} \end{aligned} \tag{4.63}$$

Now we can derive the throughput applying the power distribution. Here, to be able to derive the equations, we consider a uniform power distribution for the N_T selected streams. It should be mentionned that this assumption is realistic in the high SNR regime as first of all, the power distribution converges toward the uniform distribution and that secondly at high SNRs, DPC and linear precoding techniques become comparable with a constant offset as demonstrated by [36]. So Considering the obtained bound and the uniform power distribution, the sum-rate expressions becomes

$$Pr\left\{u(i) - \log\log\sqrt{K} \leqslant F_{SINR}(i,x)\right\} \geqslant 1 - O\left(\log(K)^{-1}\right) \tag{4.64}$$

$$\Rightarrow Pr\left\{1 + \frac{P_T}{N_T}u(i) - \frac{P_T}{N_T}\log\log\sqrt{K} \leqslant 1 + \frac{P_T}{N_T}F_{SINR}(i,x)\right\} \geqslant 1 - O\left(\log(K)^{-1}\right) \tag{4.65}$$

$$\Rightarrow Pr\left\{\log_2\left(1 + \frac{P_T}{N_T}u(i) - \frac{P_T}{N_T}\log\log\sqrt{K}\right) \leqslant \log_2\left(1 + \frac{P_T}{N_T}F_{SINR}(i,x)\right)\right\} \geqslant 1 - O\left(\log(K)^{-1}\right) \tag{4.66}$$

Starting from (4.66), and considering the approximation of the DPC sum-rate is upper bounded by (4.67) according to [15]

$$SR_{DPC} \leq N_T\log_2\left(1 + \frac{P_T}{N_T}\left(\log(K) + O\left(\log\left(\log(K)\right)\right)\right)\right) \tag{4.67}$$

we can compute the difference between DPC and the DPC version of the proposed algorithm where all the lower interference has been perfectly canceled

out for large K as (4.77)

$$SR_{DPC} \leqslant N_T \log_2 \left(1 + \frac{P_T}{N_T} \left(\log\left(K\right) + O\left(\log\left(\log\left(K\right)\right)\right)\right)\right) \tag{4.68}$$

$$\begin{aligned}
&\Rightarrow SR_{\text{DPC}} - SR_{\text{ZFBFNS - DPC}} \\
&\quad \leqslant N_T \log_2 \left(1 + \frac{P_T}{N_T} \left(\log\left(K\right) + O\left(\log\log\left(K\right)\right)\right)\right) - \\
&\quad \sum_{i=1}^{N_T} \log_2 \left(1 + \frac{P_T}{N_T} u(i) - \frac{P_T}{N_T} \log\log\sqrt{K}\right) \left(1 - O\left(\log^{-1}(K)\right)\right) \qquad (4.69) \\
&\quad = N_T \log_2 \left(1 + \frac{P_T}{N_T} \left(\log\left(K\right) + O\left(\log\log\left(K\right)\right)\right)\right) \\
&\qquad\qquad - \sum_{i=1}^{N_T} \log_2 \left(1 + \frac{P_T}{N_T} u(i) - \frac{P_T}{N_T} \log\log\sqrt{K}\right) + \\
&\qquad \sum_{i=1}^{N_T} \log_2 \left(1 + \frac{P_T}{N_T} u(i) - \frac{P_T}{N_T} \log\log\sqrt{K}\right) O\left(\log^{-1}(K)\right) \qquad (4.70) \\
&\quad = \sum_{i=1}^{N_T} \log_2 \left(\frac{1 + \frac{P_T}{N_T} \left(\log\left(K\right) + O\left(\log\log\left(K\right)\right)\right)}{1 + \frac{P_T}{N_T} u(i) - \frac{P_T}{N_T} \log\log\sqrt{K}}\right) + \\
&\qquad \sum_{i=1}^{N_T} \log_2 \left(1 + \frac{P_T}{N_T} u(i) - \frac{P_T}{N_T} \log\log\sqrt{K}\right) O\left(\log^{-1}(K)\right) \qquad (4.71) \\
&\quad = \sum_{i=1}^{N_T} \log_2 \left(1 - 1 + \frac{1 + \frac{P_T}{N_T} \left(\log\left(K\right) + O\left(\log\log\left(K\right)\right)\right)}{1 + \frac{P_T}{N_T} u(i) - \frac{P_T}{N_T} \log\log\sqrt{K}}\right) + \\
&\qquad \sum_{i=1}^{N_T} \log_2 \left(1 + \frac{P_T}{N_T} u(i) - \frac{P_T}{N_T} \log\log\sqrt{K}\right) O\left(\log^{-1}(K)\right) \qquad (4.72) \\
&\quad = \sum_{i=1}^{N_T} \log_2 \left(1 + \frac{\frac{P_T}{N_T} \left(\log\left(K\right) - u(i) + \log\log\sqrt{K} + O\left(\log\log\left(K\right)\right)\right)}{1 + \frac{P_T}{N_T} u(i) - \frac{P_T}{N_T} \log\log\sqrt{K}}\right) + \\
&\qquad \sum_{i=1}^{N_T} \log_2 \left(1 + \frac{P_T}{N_T} u(i) - \frac{P_T}{N_T} \log\log\sqrt{K}\right) O\left(\log^{-1}(K)\right) \qquad (4.73) \\
&\quad = \sum_{i=1}^{N_T} \log_2 \left(1 + \frac{\frac{P_T}{N_T} \left(\log\left(K\right) - \log\left(K\right) - \log\left(\frac{(i-1)^{2-i}}{\Gamma(N_T - i + 1)}\right)\right)}{1 + \frac{P_T}{N_T} u(i) - \frac{P_T}{N_T} \log\log\sqrt{K}} -\right.
\end{aligned}$$

$$\frac{\frac{P_T}{N_T}\left(N_T+N_{RT}-2i\right)\log\left(\log\left(K\right)+\log\left(\frac{(i-1)^{2-i}}{\Gamma(N_T-i+1)}\right)\right)}{1+\frac{P_T}{N_T}u(i)-\frac{P_T}{N_T}\log\log\sqrt{K}}+$$
$$\frac{\left(N_T+N_{RT}-2i\right)\log\left(\log\left(K\right)+\log\left(\frac{(i-1)^{2-i}}{\Gamma(N_T-i+1)}\right)\right)}{1+\frac{P_T}{N_T}u(i)-\frac{P_T}{N_T}\log\log\sqrt{K}}+$$
$$\left.\frac{\frac{P_T}{N_T}\left(\log\log\sqrt{K}+O\left(\log\log\left(K\right)\right)\right)}{1+\frac{P_T}{N_T}u(i)-\frac{P_T}{N_T}\log\log\sqrt{K}}\right)+$$
$$\sum_{i=1}^{N_T}\log_2\left(1+\frac{P_T}{N_T}u(i)-\frac{P_T}{N_T}\log\log\sqrt{K}\right)O\left(\log^{-1}(K)\right) \tag{4.74}$$

$$=\sum_{i=1}^{N_T}\log_2\left(1-\frac{\frac{P_T}{N_T}\log\left(\frac{(i-1)^{2-i}}{\Gamma(N_T-i+1)}\right)}{1+\frac{P_T}{N_T}u(i)-\frac{P_T}{N_T}\log\log\sqrt{K}}-\right.$$
$$\frac{\frac{P_T}{N_T}\left(N_T+N_{RT}-2i\right)\log\left(\log\left(K\right)+\log\left(\frac{(i-1)^{2-i}}{\Gamma(N_T-i+1)}\right)\right)}{1+\frac{P_T}{N_T}u(i)-\frac{P_T}{N_T}\log\log\sqrt{K}}-$$
$$\left.\frac{\frac{P_T}{N_T}\left(\log\left(2\right)+\log\log\left(K\right)+O\left(\log\log\left(K\right)\right)\right)}{1+\frac{P_T}{N_T}u(i)-\frac{P_T}{N_T}\log\log\sqrt{K}}\right)+$$
$$\sum_{i=1}^{N_T}\log_2\left(1+\frac{P_T}{N_T}\left(\log\left(K\right)+\log\left(\frac{(i-1)^{2-i}}{\Gamma\left(N_T-i+1\right)}\right)\right)-\right.$$
$$\left.\frac{P_T}{N_T}\log\log\sqrt{K}\right)O\left(\log^{-1}(K)\right) \tag{4.75}$$

$$As\log\left(\frac{(i-1)^{2-i}}{\Gamma\left(N_T-i+1\right)}\right)=\left(2-i\right)\log(i-1)-\sum_{j=0}^{N_T-i}\log\left(N_T-i-j\right)$$

$$\approx\sum_{i=1}^{N_T}\log_2\left(1+\frac{O\left(\log\log\left(K\right)\right)}{1+\frac{P_T}{N_T}u(i)-\frac{P_T}{N_T}\log\log\sqrt{K}}\right)$$
$$+N_T\log_2\left(\frac{P_T}{N_T}\left(\log\left(K\right)-\log\log\left(K\right)\right)\right)O\left(\log^{-1}(K)\right) \tag{4.76}$$

$$\leqslant 2N_TO\left(\frac{\log\log\left(K\right)}{\log\left(K\right)}\right) \tag{4.77}$$

Taking now into account the remaining interference from the ZF solution

it has been proven in [30, 37] that the zero forcing solution is, for large K, almost equivalent to the DPC version. And thus considering the same method it can be easily demonstrated that for optimal values of δ the zero forcing strategy converges towards the DPC.

4.5 Simulations and Results

4.5.1 Single stream algorithms

In all our simulations, we consider that we have only one stream per user $Q_k = 1$ and that the number of receiving antennas is the same for all users $N_{R_k} = N_R$ and in most cases $N_R = 2$. We choose a Rayleigh fading channel $H_k = (h^k_{i,j})_{1 \leq i \leq N_R, 1 \leq j \leq N_T}$ such as $\mathbb{E}\|h^k_{i,j}\|^2 = 1$. The simulation generates 10^6 independent channel realizations for each user. To generate the total throughput of the system, we perform an average over all channel realizations on the quantity SR given in equation (4.13). For the SJNR pre-coder, we distribute the energy equally over all considered users according to $P_k = P_T / K$.

For the figures presented in this chapter, the *"TX MMSE RX LEV"* represents the MU-MIMO system with the PU-MMSE pre-coder and left hand eigen vector corresponding to formula (4.7); the *"TX MMSE RX MMSE"* is for the system with the PU-MMSE pre-coder and an MMSE receiver given in formula (4.14); *"TX SJNR RX MF"* represents the curves of the system with an SJNR pre-coder and a matched filter receiver given by (4.10); *"TX SJNR RX MMSE"* are the curves for an SJNR pre-coder and an MMSE receiver given in formula (4.14).

Figures 4.3 and 4.4 compare the MMSE receiver with the proposed receivers in [5] and [6] for the case of $N_T = 4$ and demonstrate that MMSE receiver gives improvements on the two precoding procedures with different user parameters $K = 2$ and 3. This gain is very low for the case of two users as the interference is very low. In fact in a 2 user system with one stream per user using 4 transmitting antennas leaves 2 degrees of freedom. These degrees of freedom give the pre-coder the possibility to reduce further the interference part. In the case of 3 users, the gain introduced by our receiver is higher as the interference increases and the degrees of freedom are decreased by 1. Nevertheless, the gain for the PU-MMSE pre-coder case remains low. We can also see that the sum rate of the SJNR solution is starting to saturate at very high transmit power although it remains better than the PU-MMSE pre-coder. This demonstrates the limits for using such

a linear closed form pre-coder.

Figure 4.5 illustrates the performance of the system for $N_T = 2$. The improvement introduced by the use of an MMSE receiver with a PU-MMSE precoder is low as it is the case in Figure 4.4 and can only be perceived at low transmit power. On the other hand for the SJNR precoding, we note a very important gain by introducing the MMSE receiver. This gain is getting bigger at high transmit power.

In Figure 4.7, we consider the system after introduction of the user selection algorithm. The curves represented are for the SJNR precoder and for the case where $N_T = 4$ and $K = \{3, 4\}$. For comparison, we also report on the same figure the case $K = 2$ shown in Figure 4.3 and $K = 3$ of Figure 4.4. Comparing curves with and without user selection procedure for $K = 3$ we conclude that the selection procedure improves the system throughput.

Analyzing curves that applies the user selection procedure for $K = \{2, 3, 4\}$, shows that the throughput improves with an increasing number of users. This gain can be explained by the smart power allocation corresponding to shutting off the most interfering users.

The next figure, Figure 4.7, represents a comparison of the PU-MMSE and the SJNR precoders for the same receiving structure that is an MMSE receiver. Simulation results demonstrate that in all simulated cases, the SJNR precoder outperforms the PU-MMSE. This observation has been verified to be true, as long as an MMSE receiver is used at the mobile side, for all simulated system parameters respecting the LTE standard [12] namely for $N_R \in \{1, 2, 4\}$ and $N_T \in \{1, 2, 4\}$. Moreover, the choice of the MMSE receiver has not only been motivated by the fact that it offers the best total throughput, but also because it was not possible to make a fair comparison between the considered precoding schemes with other types of receivers. Indeed, by using another type of receivers (MF or LEV), the two capacity curves of the SJNR and the PU-MMSE precoders intersect, i.e., no curve is always better than the other one for all transmit power region; this makes it impossible to draw conclusions.

4.5.2 Multistream Case

In all our simulations, we consider that the number of receiving antennas is the same for all users N_{R_k}=N_R.

The simulation generates 10^6 independent channel realizations for each user. To generate the total throughput of the system, we perform an average over all channel realizations on the quantity SR. The channel coefficients $(h_{i,j}^k)_{1\leq i\leq N_R, 1\leq j\leq N_T}$ are generated such as $\mathbb{E}\|h_{i,j}^k\|^2 = 1$. In the presented plots, the SNR is taken as P_T/N_0.

Simulation results are presented and performances of the proposed solution are compared to some existing techniques.

Unless otherwise mentioned, the system configuration considered here is a base station with $N_T = 4$ transmitting antennas and K users with $N_R = 4$ receiving antennas for each user.

Figure 4.8 represents a comparison of some existing precoding technique. We represented here the SLNR algorithm from [33] for different stream configurations as well as an exhaustive search over the possible stream sets noted SLNR Max. These curves are compared to the ZFBF-SUS from [49] and the ZF version from [21]. The number of receiving antennas is $N_R = 3$ for $K = 2$ users to respect the constraints for the ZF algorithm. Through these curves we can clearly see the dominance of the ZFBF-SUS algorithm. This algorithm is therefore serve as a comparison base for our proposed algorithm.

Curves in Figures 4.9 represent the performances obtained with $K = 4$ users. Simulations results show that the proposed algorithm is getting closer to the channel capacity (represented by the DPC curve) and remains better than the ZFBF-SUS curves for all SNRs. We also plot the proposed algorithm with an MMSE precoder. Nevertheless, simulation results demonstrate slightly lower performances as some interference are still remaining after applying the MMSE precoders.

The next Figure 4.10 confirms the superiority of ZFBF-NS compared to the ZFBF-SUS even for a growing number of considered users. We can also note here that the difference is getting smaller with a growing number of users K. This confirms the results of the asymptotic study performed for both algorithms demonstrating a convergence towards DPC performances with a scale of $(\log(\log(K))/\log(K))$.

The next curves represented in Figure 4.11 show the gain that can be obtained for a $K = 4$ system by applying the greedy selection, and thus confirm the expected gains and resources savings. What is important here is the relative gains.

The curves given in Figure 4.12 compares the DPC version of [49] and ***Algorithm 11***. The obtained results show that applying DPC as a precoder to cancel out the LI gives the same gain for both selection algorithms. This result has been verified to be true for all system configurations and demon-

strates the real gain obtained thanks to the precoders as in both cases the LI interference is completly removed. This shows that selecting recursively streams present in the null space is much more efficient than selecting the streams based on the SVD decomposition of the original channels. The next figure, Figure 4.13 represents the performances for the considered algorithms with DPC and beam forming (BF) precoding represented in function of the number of users K at $SNR = 15dB$. Comparing the algorithms with each other, these curves confirm the previously stated results. Using the same precoding technique, the proposed selection algorithm outperforms the successive selection algorithm presented in [49].

For all the previously presented results, the step 7) of ***Algorithm 11*** and step 3) of ***Algorithm 12*** have been executed with option b). Here we present, in Figure 4.14 the impact of $\widetilde{\delta}$ for option a). This figure is presented for $SNR = 15dB$ and $K = 100$.

Analyzing this last curve, we observe that at very low values of $\widetilde{\delta}$ the limited number of streams in the selected set decreases the probability of getting some 'good' virtual projected channels (large eigenvalues for the projected channel and moderate interference). On the other hand, at high values of $\widetilde{\delta}$, the selected set contains almost all available streams (in particular those with high interference); this increases the probability of selecting 'bad' streams and decreases the system performances. Therefore, there is an optimal intermediate value of $\widetilde{\delta}$ maximizing the SR.

4.6 Conclusion

In this chapter, we put forward improvements on two closed form per-user linear transmitter precoding vectors design methods, namely PU-MMSE and SJNR. The chapter describes the two proposed modifications which are the user power distribution on the selected users and the receiver structure optimization where an MMSE receiver is used.

These improvements have been validated by means of simulations. Important gains have been observed. This highlights the importance of the receiver structure optimization when multi-antennas are present at the receiver side. This principle can be generalized to all pre-coders as long as inter-user interference exists. This chapter also points out that the SJNR precoder outperforms the PU-MMSE precoder when the optimum MMSE receiver is used.

We also presented a new CF precoding technique based on a recursive selection of the streams to serve. The selection process is done according to

a null space criteria. In fact, the selected streams are those with the best channel located in the null space of the previously selected ones. A further improvement has been introduced by performing a greedy selection. Thus extra unnecessary computations are reduced and further improvements of the total sum-rate are obtained. These selection algorithms (with and without greedy selection) have been compared to the existing successive selection procedure proposed in [49]. The comparison demonstrates that our algorithms outperform the existing one and get closer to DPC.

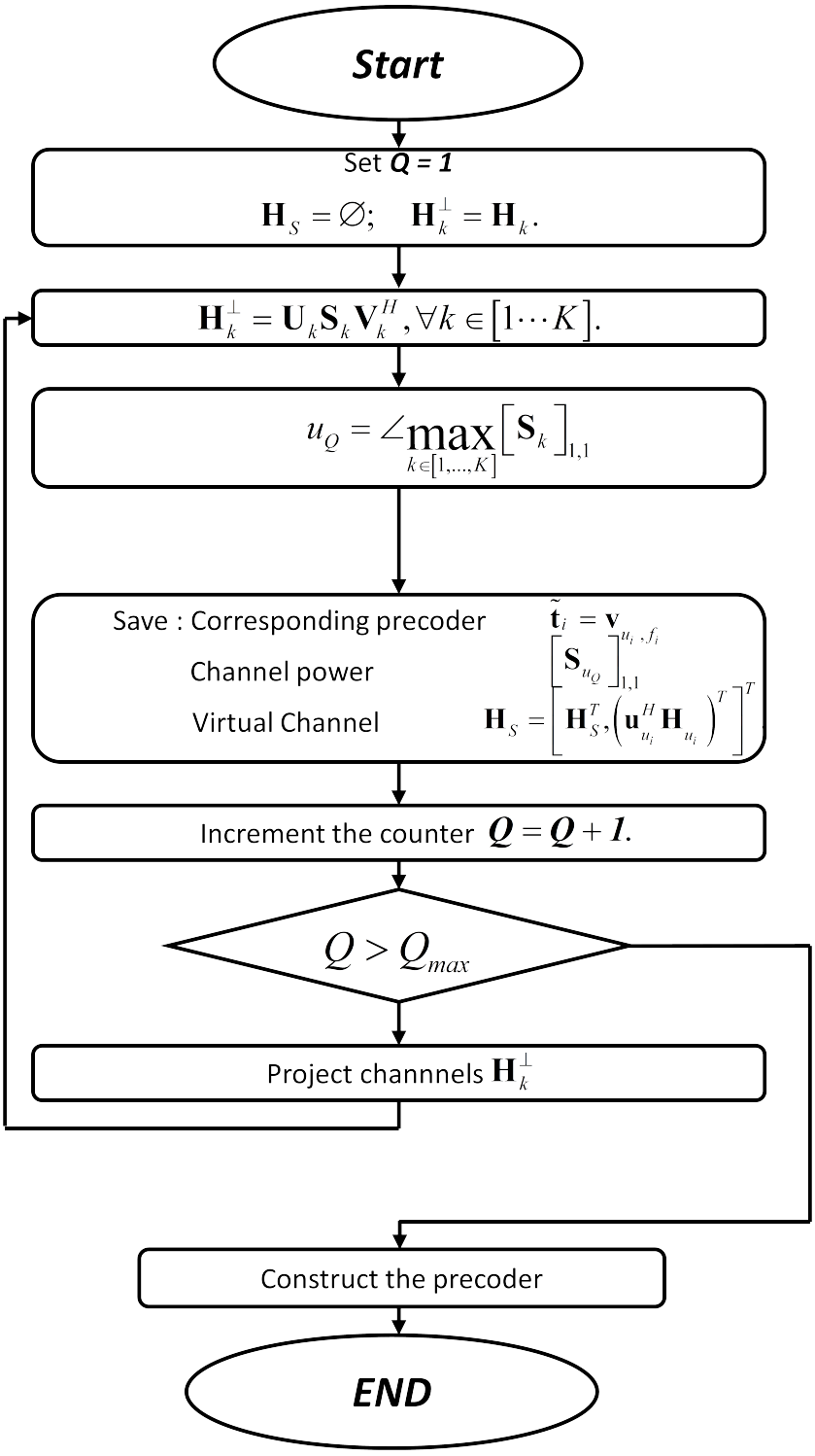


Figure 4.1: Récursive stream selection (main algorithm).

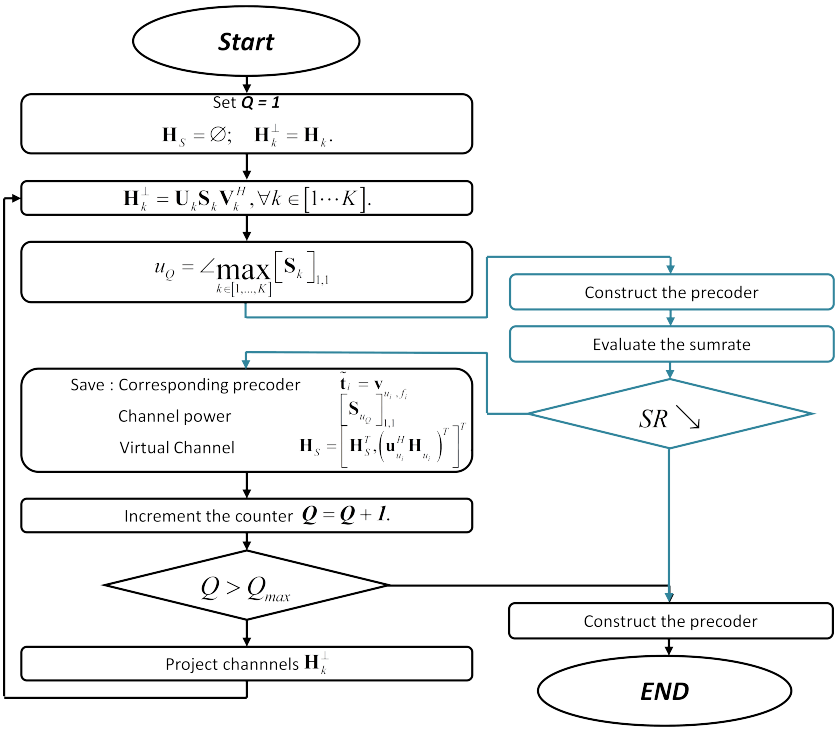


Figure 4.2: Algorithme de sélection récursif de flux avec contrôle d'admission.

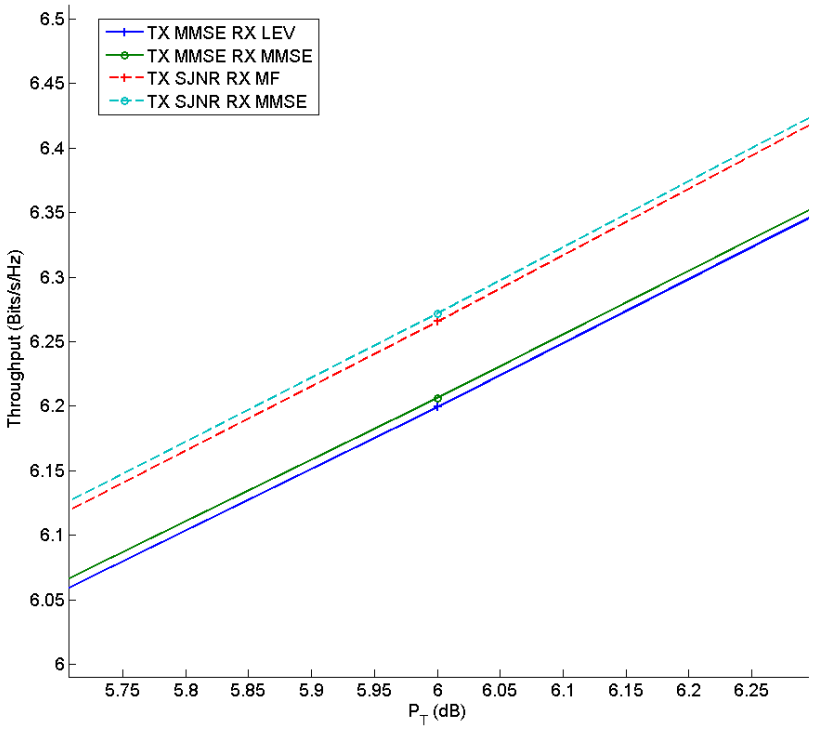


Figure 4.3: Throughput in function of transmit power P_T for $N_T = 4$, $N_R = 2$ and $K = 2$.

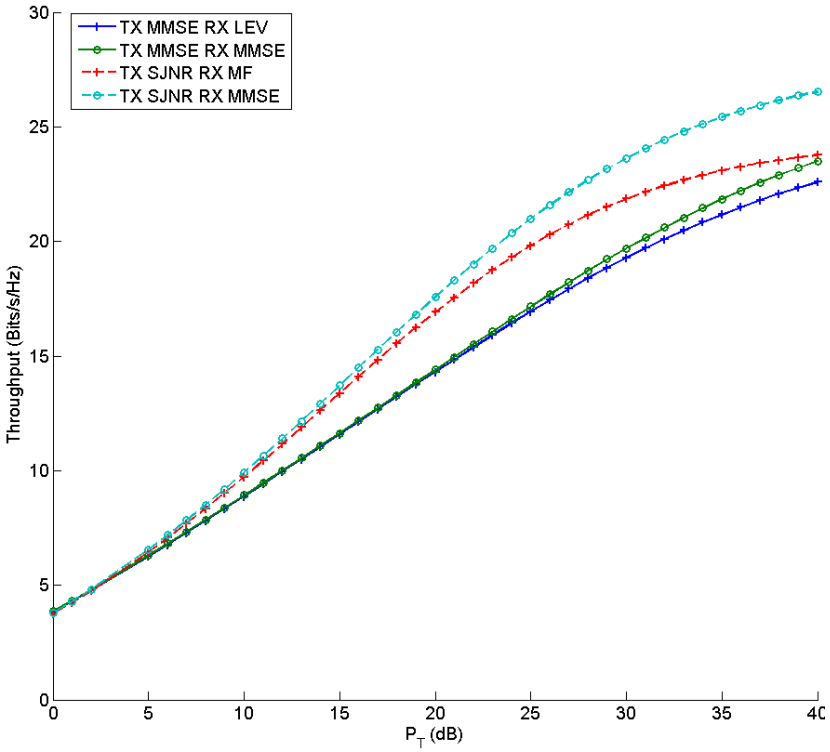


Figure 4.4: Throughput for $N_T = 4$, $N_R = 2$ and $K = 3$.

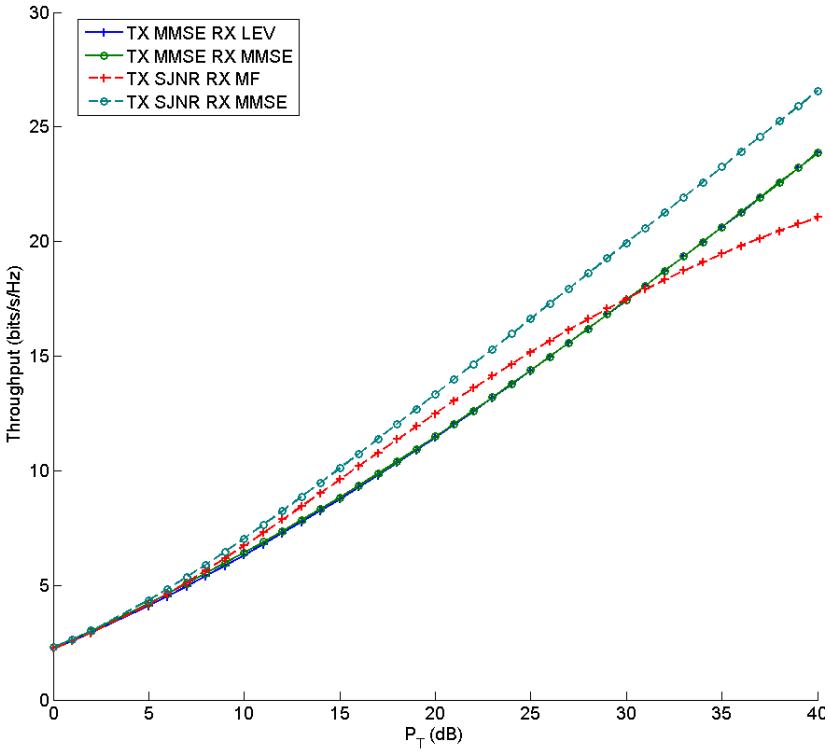


Figure 4.5: Throughput for $N_T = 2$, $N_R = 2$ and $K = 2$.

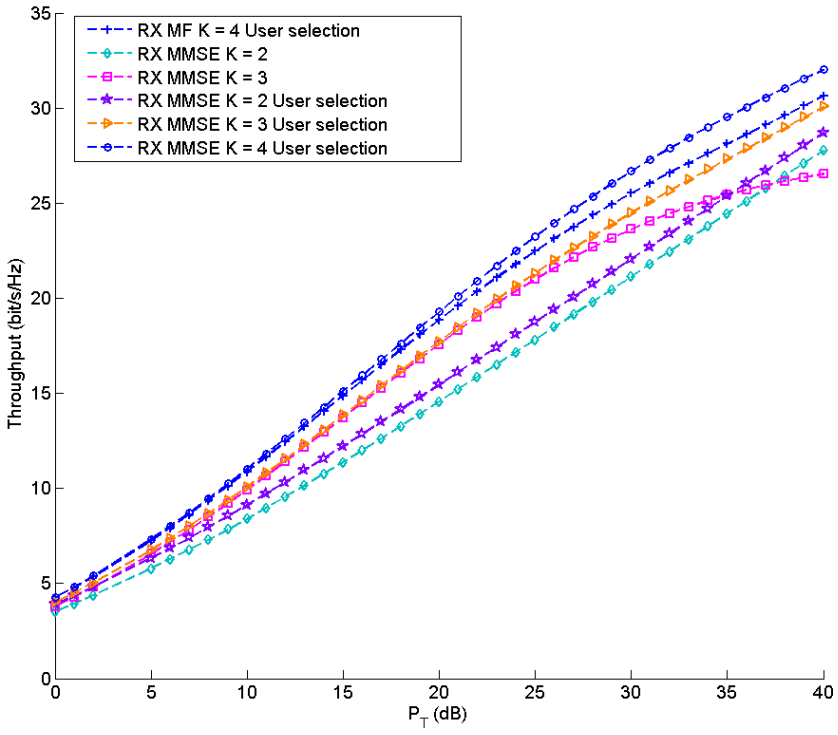


Figure 4.6: Throughput with and without user selection optimization for SJNR precoder.

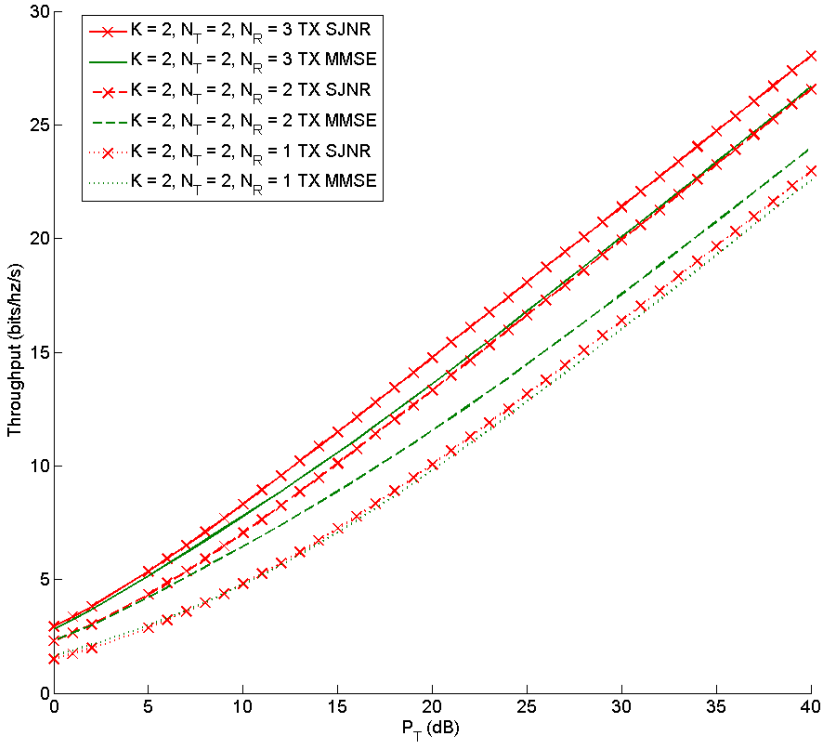


Figure 4.7: Throughput for an MMSE receiver, $N_T = 2$, $N_R = \{1, 2, 3\}$ and $K = 2$.

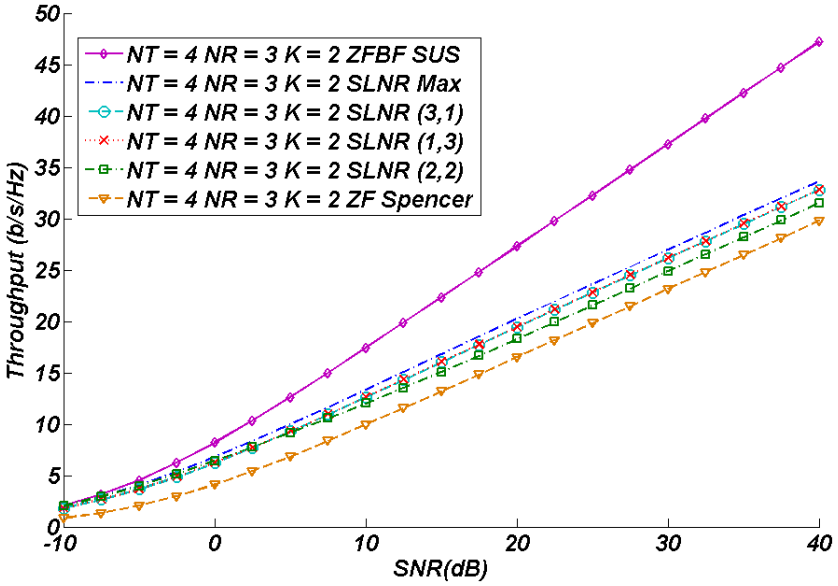


Figure 4.8: Comparing existing algorithms

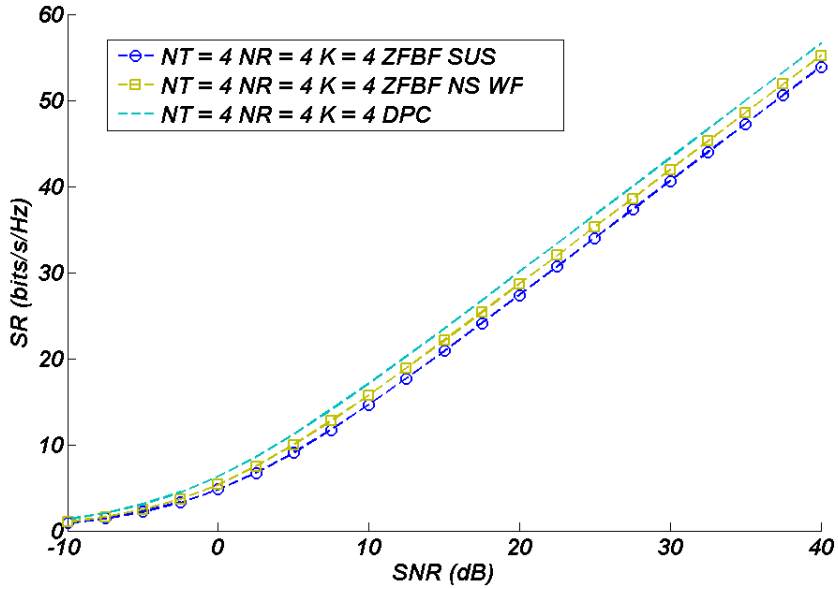


Figure 4.9: Throughput as a function of SNR

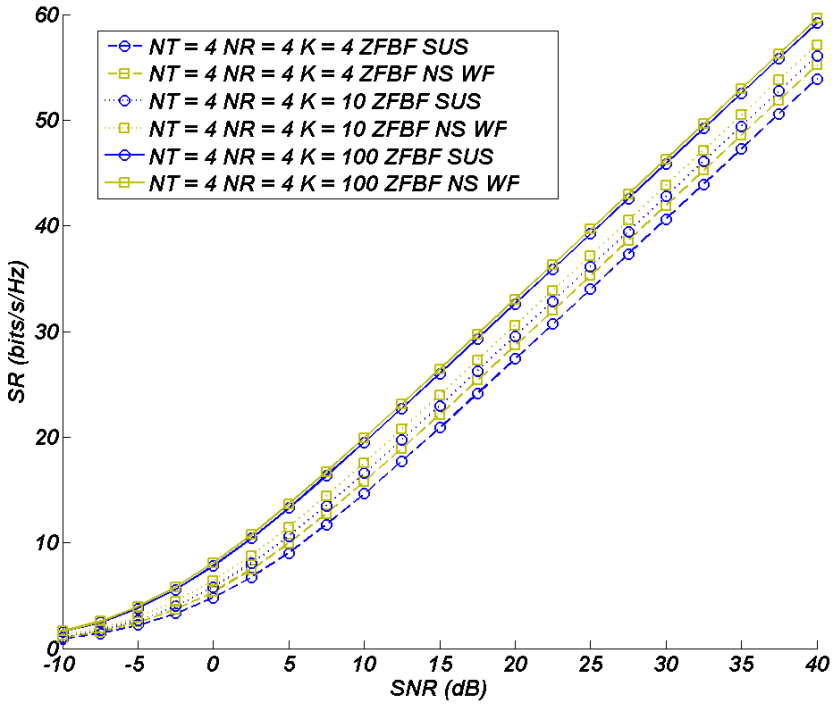


Figure 4.10: Throughput for $K = 4, K = 10, K = 100$

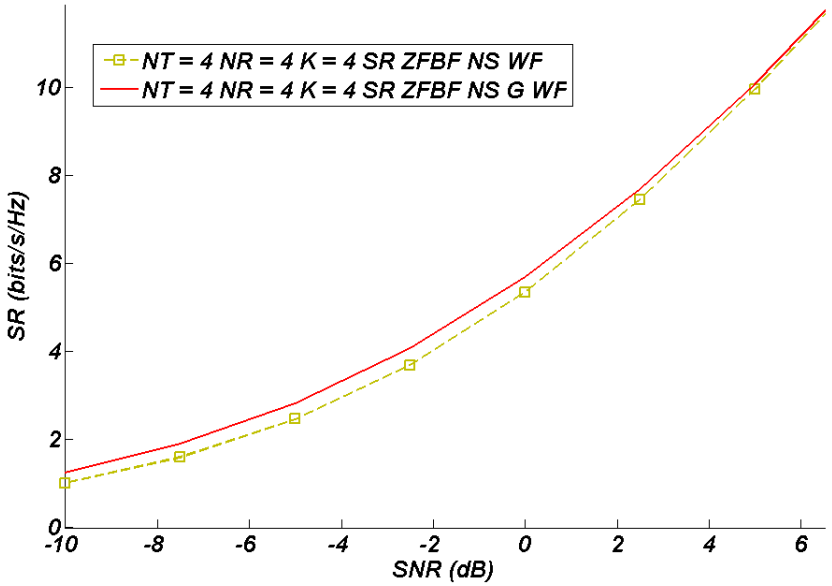


Figure 4.11: Gains with Greedy Selection for $K = 4$

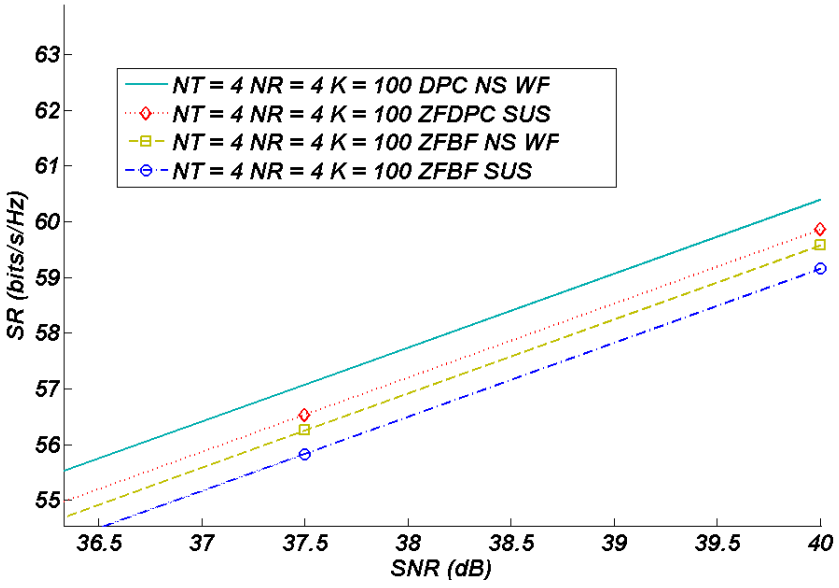


Figure 4.12: DPC-NS-WF and ZFDPC-SUS for $K = 100$

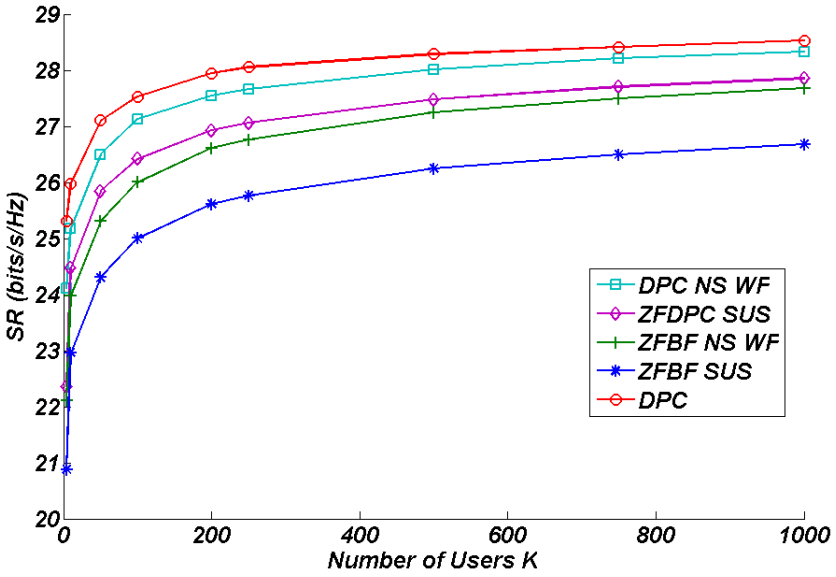


Figure 4.13: Performances as a function of K for $SNR = 15dB$

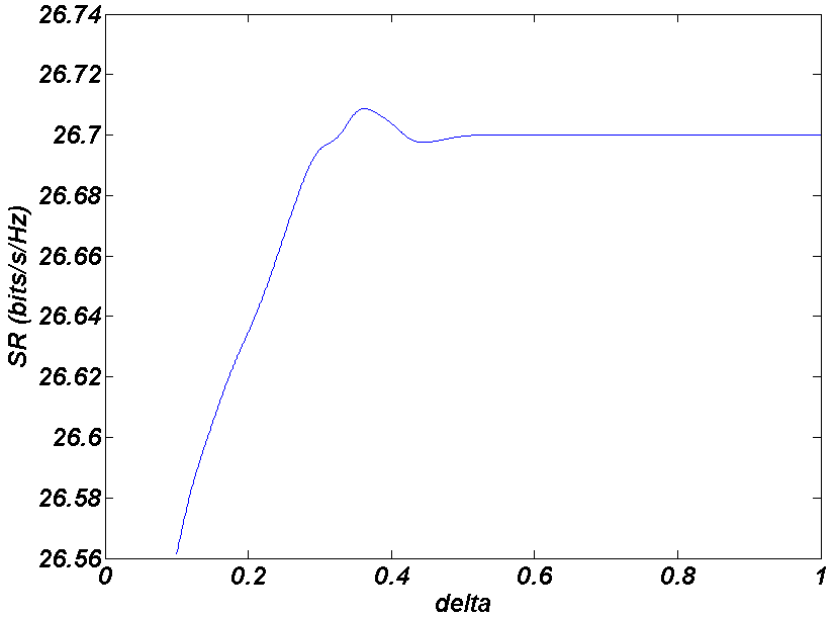


Figure 4.14: Performances as a function of $\widetilde{\delta}$ for $SNR = 15dB$ and $K = 100$

Chapter 5

Impact of channel configuration on performances

5.1 Introduction

We investigate in this chapter the performance achieved by some SDMA (Space Division Multiple Access) access techniques in a Broadcast channel configuration. The study includes the two basic linear precoding schemes present in the literature namely the CF (Closed Form) and the Iterative precoding techniques. These techniques have already been widely studied for the i.i.d Rayleigh channel. In fact, these precoding methods are essentially based on the exploitation of diversity created by multipaths present in such a channel to enhance the beamforming design and consequently increase the SINR (Signal to Interference and Noise Ratio) for each user. In this work the impact of a reduction of these degrees of freedom has been investigated through simulations under other channel configurations. The obtained results demonstrate that these precoding techniques remain efficient only for systems offering high number of degrees of freedom and fail when a direct LOS (Line Of Sight) is present limiting the achievable Sum Rate (SR) of the system. The obtained results have also been compared to the TDMA (Time Division Multiple Access) communication mode.

In the next section, the new proposed system model is detailed and in the last part results are presented and discussed.

The study in this part is limited to the single stream per user configuration. This choice has been done as for Rican channels the appearance of a direct path completely destroys the diversity offered by the MIMO channel. Thus multiple streams per user can not be supported by the system.

5.2 System Model

We assume in this chapter that the base station has a perfect knowledge of the channel state information (CSI) of all K users. Let S_k a $Q_k \times 1$ vector representing the transmitted data symbols for user k where Q_k is the number of transmission streams for the same user. In our study we are interested in the case of one stream per user Q_k=1.

The total transmit power at the base station is supposed to be constant and equal to P_T. The noise variance is noted N_0. For the channel part, H_k denotes the MIMO channel for user k which is a $N_{R_k} \times N_T$ matrix. Different channel configurations are taken into consideration. The first channel is a Rician channel. The expression of the channel coefficients are given by the expression

$$\begin{aligned} \mathrm{H_k} = & \sqrt{\frac{\mathrm{K_{R_k}}}{\mathrm{K_{R_k}+1}}}\underline{1}_{N_R \times N_T} + \\ & \sqrt{\frac{1}{\mathrm{K_{R_k}+1}}} \begin{bmatrix} \mathrm{h}^{\mathrm{k}}_{1,1} & \cdots & \mathrm{h}^{k}_{1,\mathrm{N_T}} \\ \vdots & \ddots & \vdots \\ \mathrm{h}^{k}_{\mathrm{N_R},1} & \cdots & \mathrm{h}^{k}_{\mathrm{N_R},\mathrm{N_T}} \end{bmatrix}_{N_R \times N_T} \end{aligned} \tag{5.1}$$

In this case the terms $\mathrm{h}^{\mathrm{k}}_{\mathrm{i,j}}$ of the matrix represent i.i.d random Gaussian complex variable and K_{R_k} is the Rician factor. So it turns out that for $K_{R_k} = 0$ the Rayleigh channel is obtained.

The second considered channel is a channel that we constructed in such a manner to be able to study the impact of spacially separated users (This can be the case if a good scheduler is employed and that the number of users is important). Inspired from [2, 7], the expression of such a channel is given by

$$\begin{aligned} \mathbf{H_k} = & \sqrt{\frac{\mathrm{K_{R_k}}}{\mathrm{K_{R_k}+1}}}\mathbf{1}_{N_R \times N_T} \exp\left(\tfrac{j2\pi d}{\lambda} \sin(\theta)\, diag\left(\mathbf{a}_{1\times N_T}\right)\right) + \\ & \sqrt{\frac{1}{\mathrm{K_{R_k}+1}}}\mathbf{P}_k^{1/2} \odot \exp\left(\tfrac{j2\pi d}{\lambda} \sin\left(\theta + \boldsymbol{\varphi}_{N_R\times 1}\right)\mathbf{a}_{1\times N_T}\right) \end{aligned} \tag{5.2}$$

where

$$\boldsymbol{\varphi} = \left[\begin{array}{cccc} \boldsymbol{\varphi_1} & \boldsymbol{\varphi_2} & \cdots & \boldsymbol{\varphi_{N_R}} \end{array} \right]^T_{1\times N_R} \tag{5.3}$$

represent a random vector with variance V_θ. $\mathbf{a}$ is a vector such as $\mathbf{a}_{1\times N_T} = [1 \dots N_T - 1]$ representing the contribution of each element of the antenna array and $\mathbf{P}^{1/2}$ is the matrix representing the powers for each fading link ans is given by

$$\mathbf{P}_k^{1/2} = \left\{ p_{i,j}^k \right\}_{1\leq i\leq N_R, 1\leq j\leq N_T} \text{ where } p_{i,j}^k = {\|v\|}/{\sqrt{2}} \tag{5.4}$$

and v is a Gaussian complex random variable.

5.3 Linear Precoder

In this section we consider some of the previously presented linear precoding techniques presented in the two previous chapters. We briefly remind the precoding techniques that are going to be used for simulations in this chapter.

5.3.1 MMSE Precoder

We use the MU CF given in [32] which is a simplification of the successive MMSE (SMMSE) . The precoder of user k is constructed by

$$\hat{\mathbf{t}}_k = \beta \left(\tilde{\mathbf{H}}_k^H \tilde{\mathbf{H}}_k + \frac{N_{R_k}}{P_T} N_0 \mathbf{I} \right)^{-1} \hat{\mathbf{H}}_k^H \left[\mathbf{v}_k \right]_1 \tag{5.5}$$

Where $\tilde{\mathbf{H}}_k^T = \left[\mathbf{H}_1^T \cdots \mathbf{H}_{k-1}^T \mathbf{H}_{k+1}^T \cdots \mathbf{H}_K^T \right]$ and $\hat{\mathbf{H}}_k^T = \left[\mathbf{H}_k^T \tilde{\mathbf{H}}_k^T \right] \left[\mathbf{v}_k \right]_1$ is the largest singular vector of the virtual channel $\mathbf{H}_k \hat{\mathbf{t}}_k$ and β a scalar factor to respect the total power constraint. The iterative version of MMSE precoding scheme is the oneproposed in [13,40]. It is based on successive optimization of the MMSE receiver $\mathbf{d}_k$ based on the obtained MMSE precoder $\mathbf{t}_k$ and vice versa. The MMSE iterative precoder is given by

$$\mathbf{T}_k = \beta \left(\sum_{i=1}^{K} \mathbf{H}_i^H \mathbf{D}_i^H \mathbf{D}_i \mathbf{H}_i + \frac{1}{P_T} \sum_{j=1}^{K} N_0 tr \left(\mathbf{D}_j^H \mathbf{D}_j \right) \mathbf{I} \right)^{-1} \mathbf{H}_k^H \mathbf{D}_k^H \tag{5.6}$$

where β is a normalization factor to respect the total transmit power constraint.and the MMSE iterative receiver is given by

$$\mathbf{D}_k = \mathbf{T}_k^H \mathbf{H}_k^H \left(\mathbf{H}_k \sum_{i=1}^{K} \mathbf{T}_i^H \mathbf{T}_i \mathbf{H}_k^H + N_0 \mathbf{I} \right)^{-1} \tag{5.7}$$

5.3.2 SJNR Precoder

This precoder is designed to increase the SJNR ratio. The SJNR for user k is defined as the total transmitted signal aimed to user k over the noise and the extra transmitted power received by the other users generated by the considered user k. This principle has been introduced in [27] and the CF precoder maximizing this quantity is given by

$$\mathbf{t}_k = \sqrt{P_k}\zeta_m \left[\left(\sum_{j=1, j\neq k}^{K} \mathbf{H}_j^H \mathbf{H}_j + \frac{N_0}{P_k}\mathbf{I} \right)^{-1} \mathbf{H}_k^H \mathbf{H}_k \right] \tag{5.8}$$

The iterative versions of the precoder is obtained by injecting the iterative virtual channel

$$\mathbf{h}_k^{iter} = \mathbf{d}_k^{iter-1} \mathbf{H}_k \tag{5.9}$$

into expression (5.8). The obtained iterative precoder becomes

$$\mathbf{t}_k^{iter} = \sqrt{P_k}\zeta_m \left[\left(\sum_{j=1, j\neq k}^{K} (\mathbf{h}_j^{iter})^H \mathbf{h}_j^{iter} + \frac{N_0^{iter}}{P_k}\mathbf{I} \right)^{-1} (\mathbf{h}_k^{iter})^H \mathbf{h}_k^{iter} \right] \tag{5.10}$$

where $N_0^{iter} = N_0 \mathbf{D}_k^{iter-1} \left(\mathbf{D}_k^{iter-1} \right)^H$ and $\mathbf{D}_k^{iter}$ is the used receiver for user k and H^{iter} represents the virtual channel given by (5.9).

The iterative decoder is an MSR (Maximum Sum Rate) receiver derived in [45] given by

$$(\mathbf{d}_{MSR,k}^{iter})^H = \zeta_m \left(\mathbf{\Psi}_k^{iter} \right) \tag{5.11}$$

$$\mathbf{\Psi}_k^{iter} = \left(\sum_{j=1, j\neq k}^{K} \mathbf{H}_k \mathbf{t}_j^{iter} (\mathbf{t}_j^{iter})^H (\mathbf{H}_k)^H + N_0 \mathbf{I} \right)^{-1} \mathbf{H}_k \mathbf{t}_k^{iter} (\mathbf{t}_k^{iter})^H (\mathbf{H}_k)^H \tag{5.12}$$

The description of the iterative algorithm and its convergence is given in chapter 3.

5.3.3 SVH Precoder

This last family of precoders is derived based on the SR (Sum-Rate) maximization. For that a Lagrangian optimization problem is solved. The obtained solution named "method 2.1" is given in [29] and describes the optimal solution derived for the quasi-convex optimization problem. Solving the problem using the bisection method gives the following system of equations (5.13).

$$\begin{cases} \mathbf{F}^{iter_{SVH}} & = diag\left(f_1, \cdots, f_K\right) & \text{(5.13a)} \\ \mathbf{G}^{iter_{SVH}} & = diag\left(g_1, \cdots, g_K\right) & \text{(5.13b)} \\ \mathbf{T}^{iter_{SVH}} & = \frac{\alpha \mathbf{H}^H \mathbf{G}^{iter_{SVH}}}{\left(\frac{N_0}{P_T} tr\left(\mathbf{F}^{iter_{SVH}}\right)\mathbf{I} + \mathbf{H}^H \mathbf{F}^{iter_{SVH}} \mathbf{H}\right)} & \text{(5.13c)} \end{cases}$$

The iterative algorithm is describes in **Algorithm 3**.

5.4 SR Calculation

To evaluate the performance of the algorithms, the sum-rate (SR) is evaluated for the different presented configurations. For a given system, the SR can be evaluated as the maximum of the mutual information between the received signal and the transmitted signal. Let us focus on user k and denote $\mathbf{X}_k$ the transmitted signal aimed to user k. Considering one stream per user, the transmitted signal $\mathbf{X}_k$ is given by $\mathbf{T}_k \mathbf{s}_k$ with $\mathbf{T}_k$ the precoder for user k and $\mathbf{s}_k$ a symbol vector of dimention Q_k. The received signal on the other hand (that we denote $\mathbf{Y}_k$) is given by

$$\mathbf{Y}_k = \mathbf{H}_k \mathbf{X}_k + \mathbf{H}_k \sum_{i=1; i \neq k}^{K} \mathbf{X}_i + \mathbf{V}_k \tag{5.14}$$

After decoding the received signal becomes

$$\hat{\mathbf{Y}}_k = \mathbf{D}_k \mathbf{H}_k \mathbf{X}_k + \mathbf{D}_k \mathbf{H}_k \sum_{i=1; i \neq k}^{K} \mathbf{X}_i + \mathbf{D}_k \mathbf{V}_k \tag{5.15}$$

Under these assumptions and considering perfect CSIT (Channel State Information at the Transmitter), the mutual information can be written

$$I\left(\mathbf{X}\left|\left(\widehat{\mathbf{Y}},\mathbf{H}\right)\right.\right) = I\left(\mathbf{X};\widehat{\mathbf{Y}}\left|\mathbf{H}\right.\right) \tag{5.16}$$

$$= h\left(\widehat{\mathbf{Y}}\right) - h\left(\widehat{\mathbf{Y}}\left|\mathbf{X}\right.\right) \tag{5.17}$$

$$= \log_2\left(\det\left(\mathbf{I} + \mathbf{D}_k\mathbf{H}_k\mathbf{R}_{\mathbf{X}_k}\mathbf{H}_k^H\mathbf{D}_k^H\mathbf{Q}_k^{-1}\right)\right) \tag{5.18}$$

with $\mathbf{Q}_k^{-1}$ is the covariance matrix of the interference and noise part.

$$\mathbf{Q}_k = \mathbf{D}_k\left(\mathbf{H}_k\sum_{i=1;i\neq k}^{K}\mathbf{R}_{\mathbf{X}_i}\mathbf{H}_k^H + \mathbf{R}_{\mathbf{V}_k}\right)\mathbf{D}_k^H \tag{5.19}$$

$\mathbf{R}_{\mathbf{X}_i}$ and $\mathbf{R}_{\mathbf{V}_k}$ are the covariance matrix of respectively the symbols and the noise.

The final expression of the sum rate for an SDMA system in the broadcast case with one stream per user is then given by

$$SR_{SDMA} = \sum_{k=1}^{K}\log_2\left(\mathbf{I} + \mathbf{T}_k^H\mathbf{H}_k^H\mathbf{D}_k^H\mathbf{D}_k\mathbf{H}_k\mathbf{T}_k\mathbf{R}_{\mathbf{s}_k}\left(\mathbf{D}_k\left(\Upsilon_k + N_0\mathbf{I}\right)\mathbf{D}_k^H\right)^{-1}\right) \tag{5.20}$$

where

$$\mathbf{\Upsilon}_k = \mathbf{H}_k\sum_{j=1,j\neq k}^{K}\mathbf{T}_j\mathbf{R}_{\mathbf{s}_j}\mathbf{T}_j^H\mathbf{H}_k^H \tag{5.21}$$

represents the interference generated by the other users and collected by user k. For a TDMA access mode the SR is given by

$$SR_{TDMA} = \sum_{\mathrm{k}=1}^{\mathrm{K}}\mathrm{Dis_K(k)}\log_2\left|\mathbf{I}_{N_{R_k}} + \mathbf{H}_k\mathbf{V}\mathbf{\Lambda}\mathbf{V}^H\mathbf{H}_k^H\right| \tag{5.22}$$

where $\mathrm{Dis_K(k)}$ is the temporal distribution given to the different users, $\mathbf{V}$ and $\mathbf{\Lambda}$ are respectively the precoding and the water-filling matrices.

5.5 Simulations And Results

5.5.1 Simlation Parameters

In all our simulations, we consider that we have only one stream per user $Q_k = 1$ and the number of receiving antennas is the same for all users N_{R_k}=N_R. Different channel configurations have been considered according to the schemes described in section II.

The simulation generates 10000 independent channel realizations for each user. To generate the total throughput of the system, we perform an average over all channel realizations on the quantity SR given in equation (5.22) or (5.20). The fading part of the channel coefficients $(h^k_{i,j})_{1\leq i\leq N_R, 1\leq j\leq N_T}$ are generated such as $\mathbb{E}\|h^k_{i,j}\|^2$=1.

The two convergence control parameters for the iterative algorithms ε_{SVH}, ε are fixed and equal to 0.001.

In all the following, the maximal number of iterations N^{iter}_{max} is fixed to 50.

For the SJNR precoder, we distribute the energy equally over all considered users according to P_k=P_T/K.

For the TDMA case, a uniform temporal distribution of the users is considered i.e. $Dis_K(k) = 1/K; \forall k \in [1 \dots K]$.

Finally for the θ distributed channel we consider $\theta \in \pi/360*[0, 45, 90, -45]$ for the 4 considered users and a variance of 10° around the nominal direction θ.

5.5.2 Simulation Results

In this subsection, the simulations results are presented. In all following figures we are going to adopt these notations: The names of the curves start with the corresponding Access technique C_{SDMA} or C_{TDMA} followed by the used precoder and decoder. The third element is indicating the family of the precoding technique namely $Iter$ for Iterative or CF for Closed Form and finally the value of the Rician factor K_{Rice}.

Figures 5.1-5.4 represent the evolution of the different considered algorithms for a variation of the Rician factor for $N_T = 4$ transmission antennas, $N_R = 4$ receiving antennas and $K = 4$ users. The first curves of figure 5.1 represents the CF algorithms namely the SJNR/MMSE, PU-MMSE/MMSE and the SVH/MMSE. This figure confirms the results obtained in [46] as the SJNR/MMSE performes better than the PU-MMSE algorithm. This result is here extended to all channel configurations and remains stable while

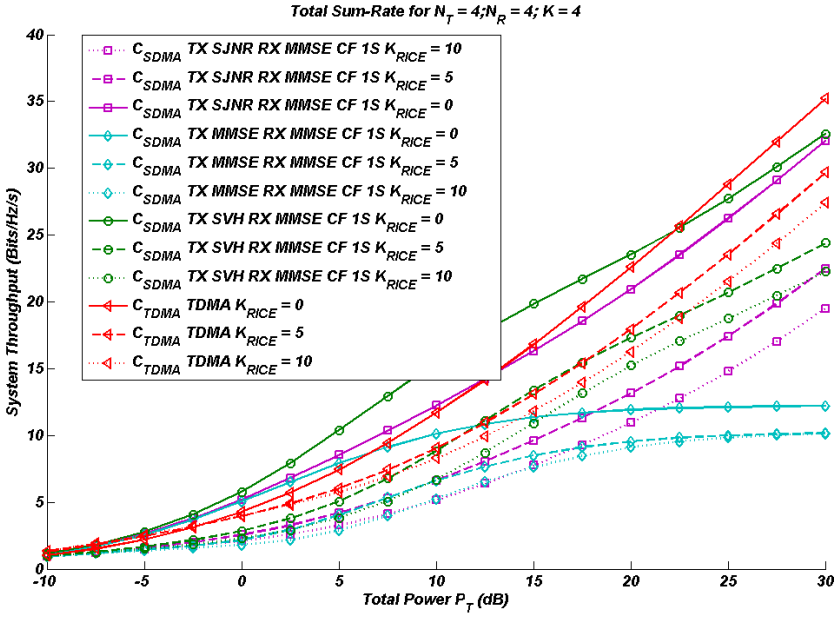


Figure 5.1: Throughput as a function of total transmit power P_T for $N_T = 4, N_R = 4$ and $K = 4$.

changing the Rician factor. It must be noted, at this point, that despite the fact that the SVH algorithm is an Iterative algorithm calculating the optimal receiver for the MU-MISO system, it has been proven in [45] that 2 iterations are sufficient for convergence. Moreover, taking a closer look at the algorithm, shows that the first iteration is playing the role of a simple initialization for the algorithm. In this case, the SVH/MSR iterative algorithm described in [45] and called ***Algorithm 4*** becomes equivalent to a CF algorithm when $iter_{SVH_{max}} = 2$ and $N_{iter} = 1$. In addition to that, it confirms the superiority of the SVH/MSR (equivalent to an SVH/MMSE) algorithm described in [45] as it is the optimal CF precoder associated with the optimal receiver maximizing mathematically the system sum-rates. We also remark in this curve the main problem encountered by the MMSE algorithm that always saturates at high SNRs when the system is fully charged. Comparing now these curves to the TDMA case, we can see that the water filling algorithm performed in this algorithm becomes stronger especially with an increase of the Rician factor. These observations combined with the results of figure 5.2 suggest that there might be better ways of extracting the system diversity. In fact the TDMA curves remains alway far beyond

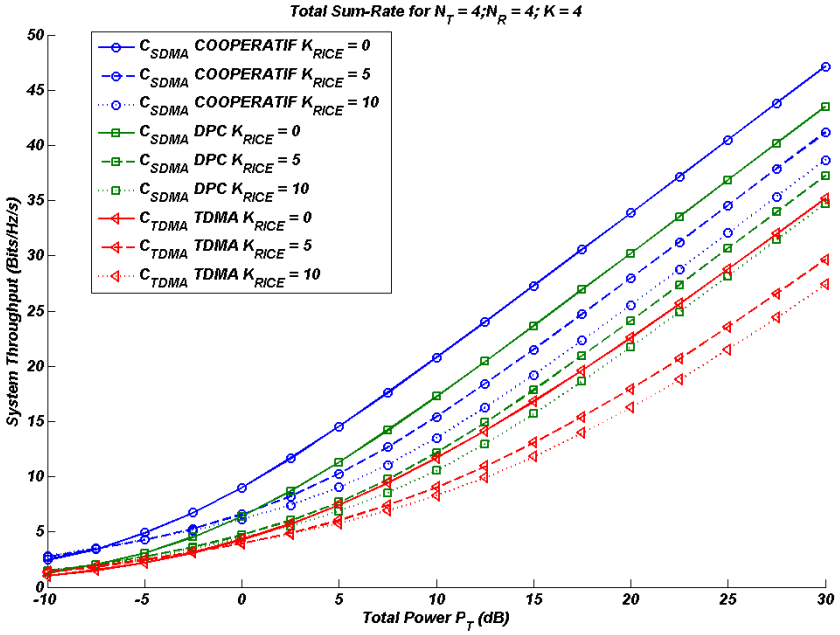


Figure 5.2: Throughput as a function of total transmit power P_T for $N_T = 4, N_R = 4$ and $K = 4$.

the DPC and cooperative curves.

Analyzing at the next curves 5.3 and 5.4 we see that by introducing the iterative procedure, the obtained results get closer to the DPC performances. The sum-rates offered by the TDMA system remain always lower than those offered by the iterative algorithms. Moreover, the impact of increase of the Rician factor (implying a decrease of the degrees of freedom generated by the multipahts) is thus much more stable.

The next figures, figures 5.5 and 5.6 gives the performances for a system with lower number of receiving antennas. In this case, we remark the almost the same phenomenons as the MMSE and SJNR CF algorithms saturate at high SNRs, with a domination of the SJNR. On the other hand, the SVH algorithm is capable of optimally eliminating the interference and no saturation is noted at high SNRs. Nevertheless, an overall decrease in performances must be noted.

In the iterative mode, the algorithms get weaker and the difference compared to the DPC gets higher as the Rician factor gets higher. This demonstrates the importance of the number of receiving antennas and thus the impact of the receiving structure in eliminating the interference part.

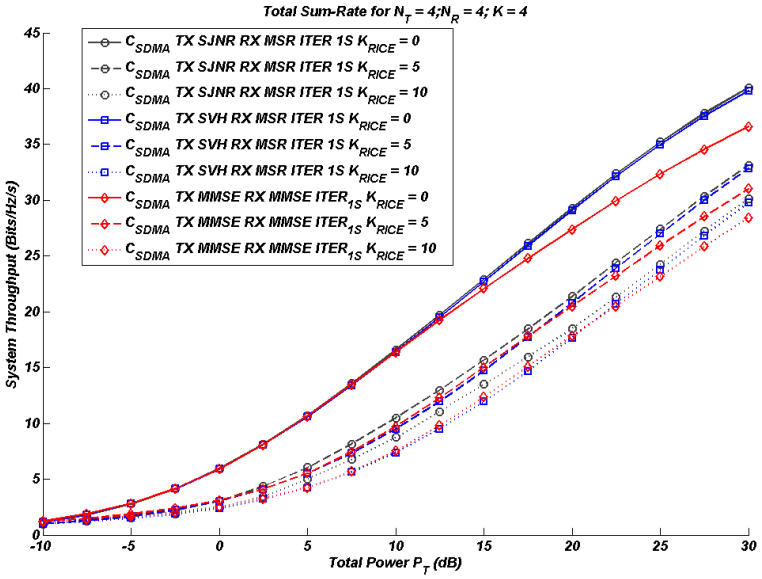


Figure 5.3: Throughput as a function of total transmit power P_T for $N_T = 4, N_R = 4$ and $K = 4$.

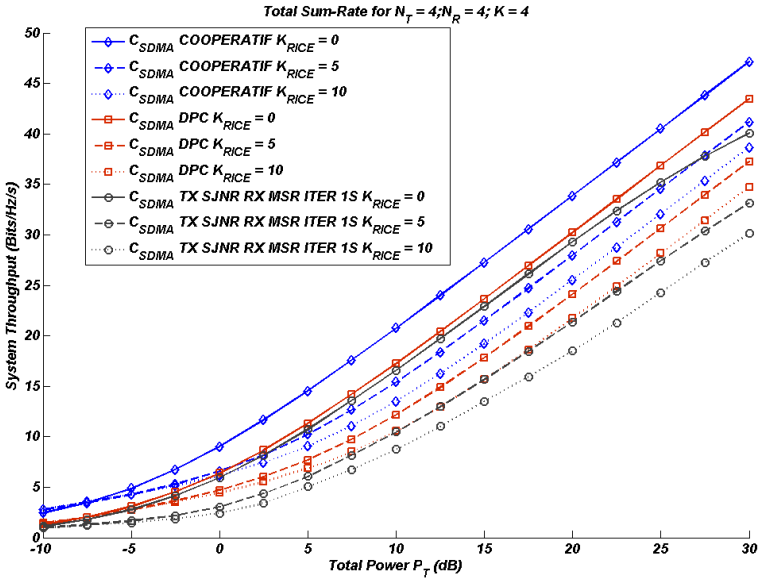


Figure 5.4: Throughput as a function of total transmit power P_T for $N_T = 4, N_R = 4$ and $K = 4$.

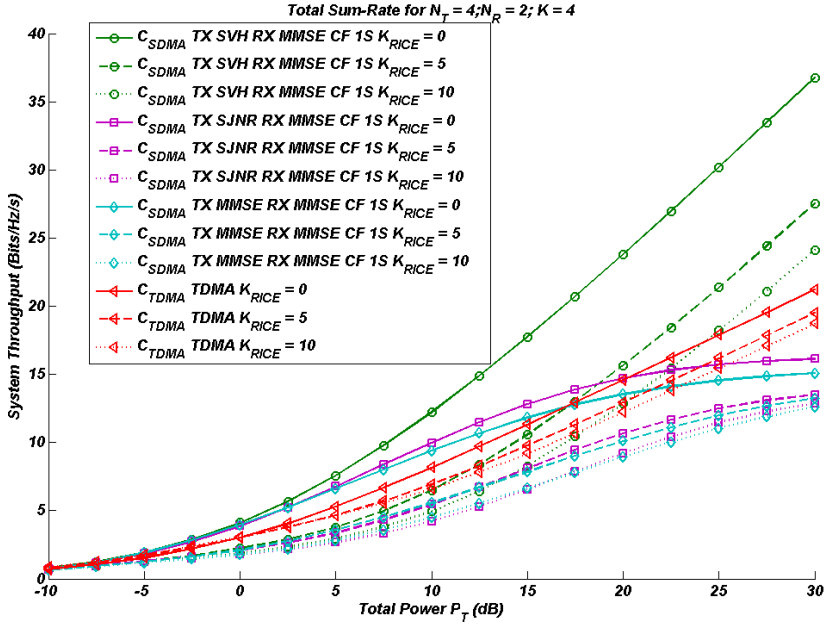


Figure 5.5: Throughput as a function of total transmit power P_T for $N_T = 4, N_R = 2$ and $K = 4$.

Figure 5.7 and 5.8 include some curves representing the performance of the cooperative system. The cooperative system is a configuration where all uers perfectly cooperate and can thus decode all the signals of the other users. The system becomes in this case equivalent to a single user MIMO system with the same number (N_T) of transmitting antennas and with $\sum_{k=1}^{K} N_{R_k}$ receiving antennas. The corresponding channel is then $\mathbf{H} = \left[\mathbf{H}_1^T, \ldots, \mathbf{H}_K^T\right]^T$. The performances of this system is obtained by an SVD decomposition over $\mathbf{H}$ followed by a Water Filling algorithm. We also plot out the perfoamnce achieved by the DPC algorithm implemented according to [25].

Figure 5.7 gives the results obtained with a fully charged 2 by 2 system ($N_T = 2$, $N_R = 2$ and $K = 2$). As expected from the previous observations, the performances of the iterative and the CF algorithms gets closer and are near the DPC limit. We also remark that the sequencing of the different algorithms remains the same. But, the most important observation to be noted here, is that even with the worst CF algorithm we can outperform the sum-rate offered by the TDMA system in the case of a perfect knowledge of the channel state information at the transmitter.

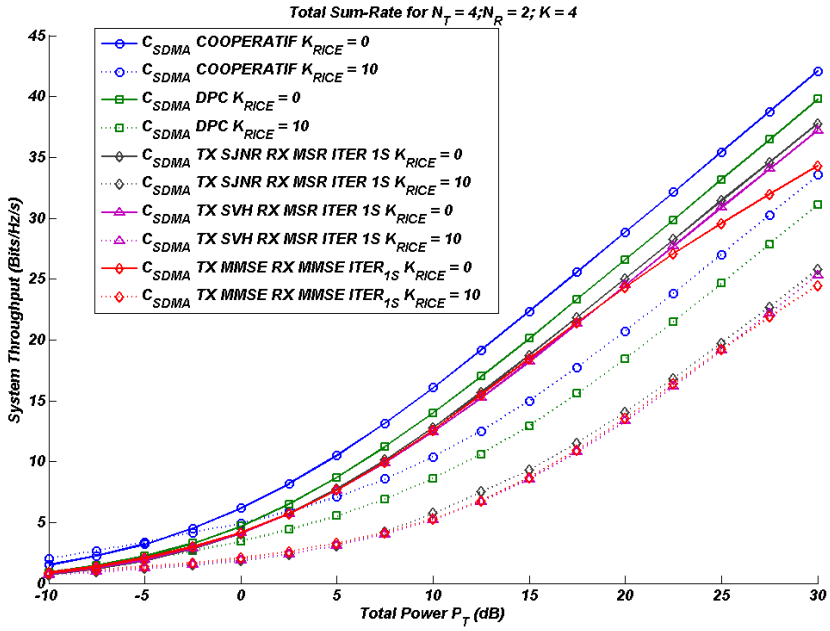


Figure 5.6: Throughput as a function of total transmit power P_T for $N_T = 4, N_R = 2$ and $K = 4$.

The last curves plotted on figure 5.8 are given for a θ distributed channel. Meaning that we considered specially uncorrelated channels for the different users. This case can easily be obtained with a good user scheduling choosing the users among a big set. These curves have been plotted for the case of a fully charged system with 4 transmitting antennas, 4 receiving antennas for each of the 4 users.

The obtained curves show that in this case, the impact of the Rician factor diminishes a lot and no longueur degrades the performances on the Cooperative system, the DPC algorithm and even on the SJNR/MSR iterative algorithm. This can easily be achieved using a good scheduling procedure selecting the least interfering users. This decreases also the constraints on the precoder making it easier to find the optimal solution maximizing the sum-rate.

5.6 Conclusion

In this chapter, we present a study of the impact of the channel geometry on some of the main precoding algorithms present in the literature. We consid-

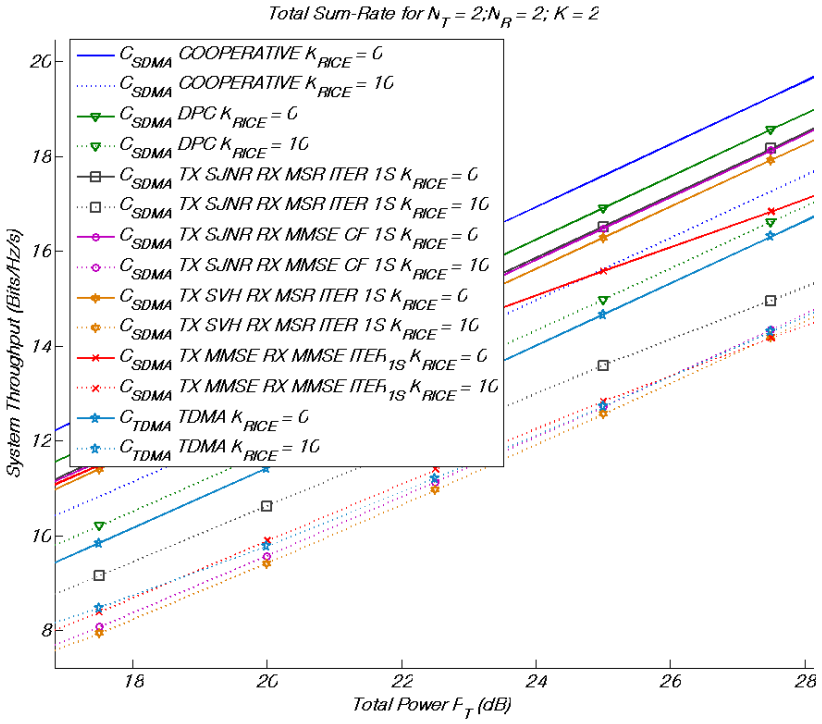


Figure 5.7: Throughput as a function of total transmit power P_T for $N_T = 2, N_R = 2$ and $K = 2$.

ered in fact, three main algorithms of the CF and iterative families. These algorithms are based on three different criteria to maximize the performance of the system. The MMSE minimizing the MSE, The SJNR minimizing the jamming signal and the SVH maximizing the throughput of the system. For our simulations we considered a Rician channel with variant K factor. The simulation results confirmed the fact that the main gain achieved by these MU-MIMO algorithms comes from the multipath structure of the channel. It shows also that the change in the channel configuration does not change the relative performances of the algorithms compared to each other.

On the other hand the second series of simulations using a θ distributed channel showed that a good scheduling algorithm and high number of users can limit the impact of a decrease of degrees of freedom of our channel.

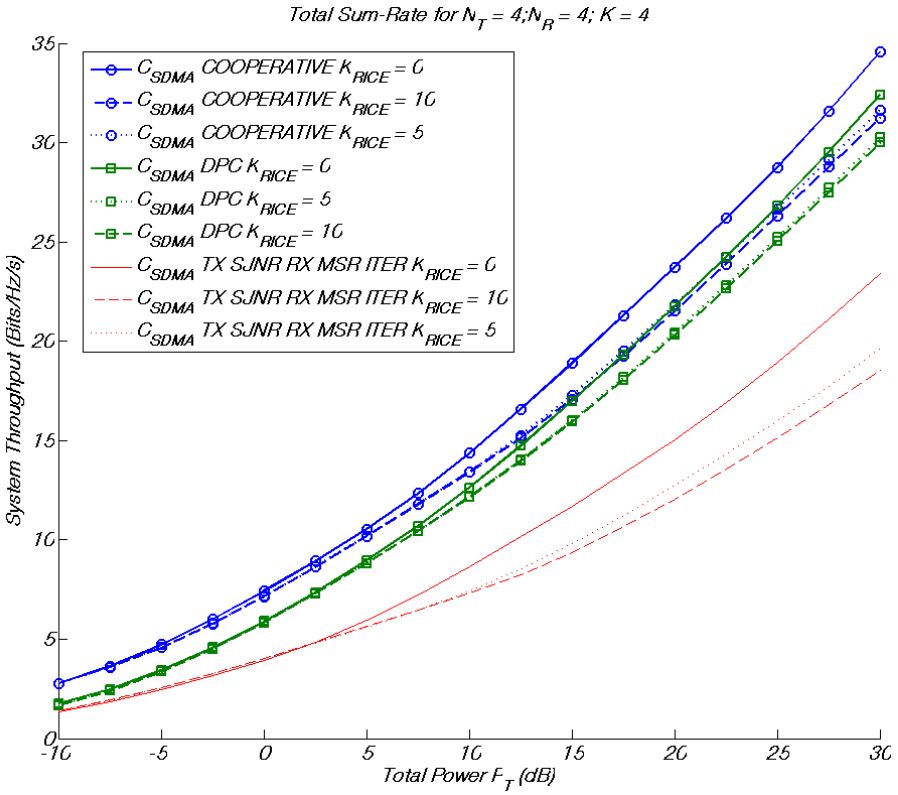


Figure 5.8: Throughput as a function of total transmit power P_T for $N_T = 4, N_R = 4$ and $K = 4$ with θ channel.

Chapter 6

Conclusions and Future Work

Conducting research in the multiuser MIMO downlink precoding techniques has generated many results, including but not limited to algorithms based on closed-form and iterative solutions, large system analysis. Furthermore many intuitive conclusions have been reached. We will try to summarize these conclusions below:

- Unlike the MISO case, in a MIMO configuration, multiple receiving antennas offer extra diversity. This diversity can be used to cancel out some of the interfering streams.
- Precoder optimization is related to receiver optimization and joint optimization process would be the best to do.
- Normalization factors for MMSE based iterative algorithms become very important in MIMO systems.
- Iterative solutions are very sensitive to initialization and suffer a lot from local maximum.
- Not all iterative algorithms offer a linear variation of the quality of the beams for example the proposed $SJNR/MF$ algorithm changes directions in function of power distributions at each iteration this offers a way of better exploring the precoder space.

- Combining algorithms sequentially leads to better performances and lower complexity if combination is correctly done.
- Sone among the proposed solutions are based on what we called the virtual channel. This cascade of the real channel and the receiver computed at step i in an iterative or recursive solution offers a practical way to easily extend the MISO based algorithms to the MIMO case with low complexity and significant gains.
- Maximum tracking proposed method allows us to optimize the precoder starting from a good initialization and thus offers better performances.
- Multi stream supporting solutions have been proposed for both linear and DPC precoding techniques as well as for iterative and closed form solutions.
- Optimizing precoders is system configuration dependent as well as channel affected. Thus dynamic adapting algorithms are one of the key solutions.
- Optimal beamforming solutions have demonstrated same sum rate slope at high SNRs. This slope is the same as the one obtained by DPC. This validates some theoretical analysis.
- The constant offset between beamforming and DPC is not only system configuration related as it depends on $\min(N_T, KN_R)$ supposing that N_R is constant for all users. But it is also decreasing with an increase of the number of users K.
- Performing multiple stream optimization in MU-MIMO systems are advantageous only in low loaded systems meaning the number of served users K is less then the number of transmitting antennas N_T. Otherwise, single stream double iterative proposed solution achieves the same performances.
- Stream selection algorithms are the bottleneck and the key solution in MU-MIMO systems in the case of multiple streams enabled. The proposed users selection algorithm is one of the possible solutions that obviously remains suboptimal but offers performances close to those obtained with DPC. Further improvements and complexity reduction can be obtained by adding some control and rate prediction during recursive greedy selection.

- The second proposed solution is the deterministic annealing procedure where the starting point is an optimal solution and the global solution at a given target SNR is tracked trough small power perturbations.

- The last chapter offered a new simple channel model that makes it possible to introduce the channel geometry and user positioning in space.

- Simulation results showed that MIMO structure is destroyed when line of sight is present in the channel. In this case differences in performances shrinkdown.

- Although differences between algorithms are reduced, the

- In any channel configuration, it is better to do SDMA precoding then time sharing TDMA between users if the algorithm is an iterative one.

- For closed form precoders TDMA becomes much better at high SNRs ($> 10dB$). And the performances of some precoders completely collapses for fully charged systems.

- If users are selected in geometrically separated direction or if the channel variance around the main direction has little overlapping, substantial increase in performances can be obtained.

- Importance of user scheduling

Despite the fact the proposed algorithms are practical algorithms that can be implemented in todays MIMO systems, it's true that there are still many questions that need to be answered and many uninvestigated issues that need to be addressed, in order to formulate a complete solution for the linear precoding problem. A few rather trivial extensions of the work presented in this document, include but are not limited to the following:

- Consider the impact of channel state information on the performances and stability of the algorithms.

- In TDD the reciprocity of the communication channel can be exploited to extrapolate the CSI but in the FDD case, feedback should be designed properly.

- Impact of CSI quantization and the usage of codebooks.

- So far, the only interference that we considered here is the inter user and inter stream interference but in real cellular systems (other than femtocells) there is a non negligible amount of interference will be coming from neighboring cells and from their users. This would imply more constraints on the optimization of the cost function and introduces higher challenges for CSI collection.
- Moreover strategies for treating and decisions taking based on the known CSI should be investigated. This can in fact be done in a centralized fashion or in a distributed manner.
- Another interesting point to investigate related to all the work done nowadays for relays would be how to extend these results to relay based systems.
- For the annealing solution, some problems remain, especially at phase transitions, and should be investigated deeper.

Bibliography

[1] M. Costa. Writing on dirty paper (corresp.). *Information Theory, IEEE Transactions on*, 29(3):439–441, May 1983.

[2] J.C. Liberti and T.S. Rappaport. A geometrically based model for line-of-sight multipath radio channels. volume 2, pages 844 –848 vol.2, apr. 1996.

[3] X. Giraud, E. Boutillon, and J.C. Belfiore. Algebraic tools to build modulation schemes for fading channels. *Information Theory, IEEE Transactions on*, 43(3):938 –952, may 1997.

[4] S.M. Alamouti. A simple transmit diversity technique for wireless communications. *Selected Areas in Communications, IEEE Journal on*, 16(8):1451 –1458, oct 1998.

[5] K. Rose. Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. *Proceedings of the IEEE*, 86(11):2210 –2239, nov 1998.

[6] V. Tarokh, H. Jafarkhani, and A.R. Calderbank. Space-time block codes from orthogonal designs. *Information Theory, IEEE Transactions on*, 45(5):1456 –1467, jul 1999.

[7] Mats Bengtsson and Björn Ottersten. Low-complexity estimators for distributed sources. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 48(8):2185–2194, 2000.

[8] O. Damen, A. Chkeif, and J.-C. Belfiore. Lattice code decoder for space-time codes. *Communications Letters, IEEE*, 4(5):161 –163, may 2000.

[9] N. Chiurtu, B. Rimoldi, and E. Telatar. On the capacity of multi-antenna gaussian channels. In *Information Theory, 2001. Proceedings. 2001 IEEE International Symposium on*, page 53, 2001.

[10] T. Haustein, C. von Helmolt, E. Jorswieck, V. Jungnickel, and V. Pohl. Performance of mimo systems with channel inversion. In *Vehicular Technology Conference, 2002. VTC Spring 2002. IEEE 55th*, volume 1, pages 35 – 39 vol.1, 2002.

[11] G. Caire and S. Shamai. On the achievable throughput of a multi-antenna gaussian broadcast channel. *Information Theory, IEEE Transactions on*, 49(7):1691 – 1706, july 2003.

[12] G. Caire and S. Shamai. On the achievable throughput of a multi-antenna gaussian broadcast channel. *Information Theory, IEEE Transactions on*, 49(7):1691–1706, July 2003.

[13] M.T. Ivrlac, R.L.U. Choi, R.D. Murch, and J.A. Nossek. Effective use of long-term transmit channel state information in multi-user MIMO communication systems. In *Vehicular Technology Conference, 2003. VTC 2003-Fall. 2003 IEEE 58th*, volume 1, pages 373–377 Vol.1, Oct. 2003.

[14] W. Keusgen. On limits of wireless communications when using multiple dual-polarized antennas. In *Telecommunications, 2003. ICT 2003. 10th International Conference on*, volume 1, pages 204 – 210 vol.1, feb.-1 march 2003.

[15] M. Sharif and B. Hassibi. On the capacity of mimo broadcast channel with partial side information. In *Signals, Systems and Computers, 2003. Conference Record of the Thirty-Seventh Asilomar Conference on*, volume 1, pages 958 – 962 Vol.1, 2003.

[16] David Tse and Pramod Viswanath. On the capacity of the multiple antenna broadcast channel. In *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 2003.

[17] H. Viswanathan, S. Venkatesan, and H. Huang. Downlink capacity evaluation of cellular networks with known-interference cancellation. *Selected Areas in Communications, IEEE Journal on*, 21(5):802 – 811, june 2003.

[18] Lizhong Zheng and D.N.C. Tse. Diversity and multiplexing: a fundamental tradeoff in multiple-antenna channels. *Information Theory, IEEE Transactions on*, 49(5):1073 – 1096, may 2003.

[19] H. El Gamal, G. Caire, and M.O. Damen. Lattice coding and decoding achieve the optimal diversity-multiplexing tradeoff of mimo channels. *Information Theory, IEEE Transactions on*, 50(6):968 –985, june 2004.

[20] N. Jindal and A. Goldsmith. Dirty paper coding vs. tdma for mimo broadcast channels. In *Communications, 2004 IEEE International Conference on*, volume 2, pages 682 – 686 Vol.2, june 2004.

[21] Q.H. Spencer, A.L. Swindlehurst, and M. Haardt. Zero-forcing methods for downlink spatial multiplexing in multiuser MIMO channels. *Signal Processing, IEEE Transactions on*, 52(2):461–471, Feb. 2004.

[22] H. Weingarten, Y. Steinberg, and S. Shamai. The capacity region of the gaussian mimo broadcast channel. In *Information Theory, 2004. ISIT 2004. Proceedings. International Symposium on*, page 174, june-2 july 2004.

[23] P. Dayal and M.K. Varanasi. An algebraic family of complex lattices for fading channels with application to space-time codes. *Information Theory, IEEE Transactions on*, 51(12):4184 – 4202, dec. 2005.

[24] G. Dimic and N.D. Sidiropoulos. On downlink beamforming with greedy user selection: performance analysis and a simple new algorithm. *Signal Processing, IEEE Transactions on*, 53(10):3857 – 3868, oct. 2005.

[25] Nihar Jindal, Wonjong Rhee, Syed Jafar, and Goldsmith. Sum power iterative water-filling for multi-antenna gaussian broadcast channels. *IEEE Trans. Inform. Theory*, 51:1570–1580, 2005.

[26] C.B. Peel, B.M. Hochwald, and A.L. Swindlehurst. A vector-perturbation technique for near-capacity multiantenna multiuser communication-part i: channel inversion and regularization. *Communications, IEEE Transactions on*, 53(1):195 – 202, jan. 2005.

[27] Yongle Wu, Jinfan Zhang, Mingguang Xu, Shidong Zhou, and Xibin Xu. Multiuser MIMO downlink precoder design based on the maximal sjnr criterion. In *Global Telecommunications Conference, 2005. GLOBECOM '05. IEEE*, volume 5, pages 5 pp.–2698, Dec. 2005.

[28] Jinfan Zhang, Yongle Wu, Shidong Zhou, and Jing Wang. Joint linear transmitter and receiver design for the downlink of multiuser MIMO systems. *Communications Letters, IEEE*, 9(11):991–993, Nov. 2005.

[29] M. Stojnic, H. Vikalo, and B. Hassibi. Rate maximization in multi-antenna broadcast channels with linear preprocessing. *Wireless Communications, IEEE Transactions on*, 5(9):2338–2342, September 2006.

[30] Taesang Yoo and A. Goldsmith. On the optimality of multiantenna broadcast scheduling using zero-forcing beamforming. *Selected Areas in Communications, IEEE Journal on*, 24(3):528 – 541, mar. 2006.

[31] C.-J. Chen and L.-C. Wang. Performance analysis of scheduling in multiuser mimo systems with zero-forcing receivers. *Selected Areas in Communications, IEEE Journal on*, 25(7):1435 –1445, september 2007.

[32] Min Lee and Seong Keun Oh. A per-user successive mmse precoding technique in multiuser MIMO systems. In *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, pages 2374–2378, April 2007.

[33] M. Sadek, A. Tarighat, and A.H. Sayed. A leakage-based precoding scheme for downlink multi-user mimo channels. *Wireless Communications, IEEE Transactions on*, 6(5):1711 –1721, may. 2007.

[34] M. Sharif and B. Hassibi. A comparison of time-sharing, dpc, and beamforming for mimo broadcast channels with many users. *Communications, IEEE Transactions on*, 55(1):11 –15, jan. 2007.

[35] Hanan Weingarten, Tie Liu, Shlomo Shamai, Yossef Steinberg, and Pramod Viswanath. The capacity region of the degraded mimo compound broadcast channel. In *Information Theory, 2007. ISIT 2007. IEEE International Symposium on*, pages 766 –770, june 2007.

[36] Schmidt D. Hunger R. and Joham M. An asymptotic analysis of the MIMO BC under linear filtering. *CoRR abs/0807.2724, Informal publication. [Online].*, Available: http://arxiv.org/abs/0807.2724., Jul. 2008.

[37] M.A. Maddah-Ali, M.A. Sadrabadi, and A.K. Khandani. Broadcast in mimo systems based on a generalized qr decomposition: Signaling and performance analysis. *Information Theory, IEEE Transactions on*, 54(3):1124 –1138, march 2008.

[38] Shuying Shi, M. Schubert, and H. Boche. Rate optimization for multiuser MIMO systems with linear processing. *Signal Processing, IEEE Transactions on*, 56(8):4020–4030, Aug. 2008.

[39] A.J. Tenenbaum and R.S. Adve. Improved sum-rate optimization in the multiuser MIMO downlink. In *Information Sciences and Systems, 2008. CISS 2008. 42nd Annual Conference on*, pages 984–989, March 2008.

[40] Hongmei Wang, Xibin Xu, Ming Zhao, Weiling Wu, and Yan Yao. Robust transmission for multiuser MIMO downlink systems with imperfect csit. In *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, pages 340–344, 31 2008-April 3 2008.

[41] Hongzhi Wang, P. Leray, and J. Palicot. An efficient mimo v-blast decoder based on a dynamically reconfigurable fpga including its reconfiguration management. In *Communications, 2008. ICC '08. IEEE International Conference on*, pages 746 –750, may 2008.

[42] S.S. Christensen, R. Agarwal, E. de Carvalho, and J.M. Cioffi. Weighted sum-rate maximization using weighted mmse for MIMO-BC beamforming design. In *Communications, 2009. ICC '09. IEEE International Conference on*, pages 1–6, June 2009.

[43] S. Kaviani and W.A. Krzymien. On the optimality of multiuser zero-forcing precoding in mimo broadcast channels. In *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, pages 1 –5, april 2009.

[44] M. Amara, Y. Yuan-Wu, and D. Slock. Double iterative precoder & receiver design for MU-MIMO broadcast channel. In *PIMRC 2010. 21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, September 2010.

[45] M. Amara, Y. Yuan-Wu, and D. Slock. Optimal MU-MIMO precoder with miso decomposition approach. In *Spawc 2010. IEEE International Workshop on Signal Processing Advances for Wireless Communications*, June 2010.

[46] M. Amara, Y. Yuan-Wu, and D. Slock. Optimized linear receivers and power allocation for two multi-user MIMO downlink schemes with linear precoding. In *ISCCSP 2010. IEEE International Symposium on Communications, Control and Signal Processing*, March 2010.

[47] M. Amara, Y. Yuan-Wu, and D. Slock. Performance of closed form and iterative MU-MIMO precoders for different broadcast channel configurations. In *The Second International Conference on Communications and Networking (ComNet2010)*, November 2010.

[48] M. Amara, Y. Yuan-Wu, and D. Slock. Receiver and transmitter iterative optimization using maximum sum-rate criterion for multi-user MIMO systems. In *ISCCSP 2010. IEEE International Symposium on Communications, Control and Signal Processing*, March 2010.

[49] L Sun and M Mckay. Eigen-based transceivers for the MIMO broadcast channel with semi-orthogonal user selection. *Signal Processing, IEEE Transactions on*, PP(99):1 –1, 2010.

[50] Eugene L. Allgower and Kurt Georg. *Introduction to Numerical Continuation Methods*. Society for Industrial Mathematics, U.S.A, 1987.

[51] I.S. Gradshteyn and I.M. Ryzhik. *Table of Integrals, Series, and Products, Seventh Edition*. Academic Press, U.S.A, 2007.

[52] M. G. Kendall. *A Course in the Geometry of n Dimensions*. Dover Publications, U.S.A, 2004.

[53] A. M. Kshirsagar. *Multivariate Analysis*. Marcel Dekker, U.S.A, 1972.