



HAL
open science

Egaliseurs multicanaux aveugles rapides et robustes aux erreurs de surestimation de l'ordre du canal

Ibrahim Kacha

► **To cite this version:**

Ibrahim Kacha. Egaliseurs multicanaux aveugles rapides et robustes aux erreurs de surestimation de l'ordre du canal. Traitement du signal et de l'image [eess.SP]. Ecole Nationale Polytechnique d'Alger, 2007. Français. NNT: . tel-00349331

HAL Id: tel-00349331

<https://theses.hal.science/tel-00349331>

Submitted on 29 Dec 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
École Nationale Polytechnique
Département d'Électronique



Thèse

présentée pour obtenir le grade de Docteur d'État en Électronique
Option : Traitement du signal

Ibrahim Kacha

Ingénieur, ENP 90.

Magister, ENP 93.

***Égaliseurs multicanaux aveugles rapides et robustes
aux erreurs de surestimation de l'ordre du canal***

Soutenue le : 01 Avril 2007

Devant le jury composé de :

<i>Amrane Houacine</i>	<i>Professeur (USTHB)</i>	<i>Président</i>
<i>Ahmed Zerguerras</i>	<i>Professeur (ENP)</i>	<i>Examineurs</i>
<i>Elbey Bourennane</i>	<i>Professeur (Université de Bourgogne, France)</i>	
<i>Latifa Hamami</i>	<i>Maître de Conférences (ENP)</i>	
<i>Braham Himed</i>	<i>Ph.D. (Air Force Research Laboratory, USA)</i>	<i>Invités</i>
<i>Mohamed Ouadjaout</i>	<i>Chargé de Cours (ENP)</i>	
<i>Adel Belouchrani</i>	<i>Professeur (ENP)</i>	<i>Directeurs de thèse</i>
<i>Karim Abed-Meraim</i>	<i>Maître de Conférences, HDR (Télécom Paris)</i>	

ملخص

يتناول موضوع هذه الأطروحة، أنظمة التعرف و التسوية العمياء، و التي تتميز باستغنائها عن سلسلة التمرن. مع ذلك فان انتهاج هذا النوع من الأساليب لا يخلو من بعض العوائق، مقارنة بالطرق التقليدية، حيث أنها تعاني من ضعف في الأداء و تكلفة حسابية عالية كما أنها جد حساسة للارتياح في تقدير رتبة القناة وهذا ما يقلل من تطبيقها عمليا . من اجل ذلك تأتي هذه الدراسة ، أولا لتلخيص و توضيح و تقييم ما تم انجازه من أعمال في هذا المجال ، ثانيا لاقتراح طريقتين جديدتين متكيفتين و سريعتين للتسوية العمياء من النوع الخطأ الرباعي المتوسط الأدنى و المتركزة على إحصائيات من الرتبة الثانية و الخاصة بأنظمة متعددة القنوات ذات إجابة نبضية منتهية . الطريقة الأولى و الخاصة بأنظمة متعددة المداخل و متعددة المخارج ، تتميز بحساسية جد ضعيفة إزاء الأخطاء الناتجة عن التقدير بالزيادة لرتبة القناة. أما الطريقة الثانية فهي موجهة لأنظمة وحيدة المدخل و متعددة المخارج و تتميز باستقلالية تامة عن رتبة القناة.

كلمات مفتاحية: التعرف و التسوية العمياء، الترشيح المتكيف السريع ، تقدير و ملاحقة الفضاءات.

Résumé

Ce travail de thèse se cadre autour de la thématique d'identification / égalisation aveugle (autodidacte) de canaux de transmission, dont l'atout majeur est, incontestablement, la suppression de la séquence d'apprentissage. Néanmoins, une telle démarche présente certains inconvénients, par rapport à un traitement classique (avec séquence d'apprentissage) du problème, à savoir, un niveau de performance plus faible, un coût de calcul plus élevé et une sensibilité plus importante aux incertitudes sur certains paramètres, tel l'ordre du canal. Ce qui joue en défaveur d'une large application des schémas d'identification / égalisation aveugle dans la pratique. Tenant compte de cette problématique et afin d'améliorer les performances des systèmes d'identification / égalisation aveugle et pallier à leurs carences, cette étude vient, dans un premier temps, résumer, clarifier et évaluer un certains nombre de travaux déjà existants, et dans un deuxième temps, proposer deux nouvelles approches adaptatives et rapides d'égalisation aveugle du type erreur quadratique moyenne minimale (*Minimum Mean Square Error, MMSE*), du second ordre, pour des systèmes multicanaux à réponse impulsionnelle finie. La première approche, destinée à des systèmes à entrées multiples sorties multiples (*Multiple Input Multiple Output, MIMO*), est robuste aux erreurs de surestimation de l'ordre du canal. Tandis que la deuxième approche, destinée à des systèmes à une entrée plusieurs sorties (*Single Input Multiple Output, SIMO*) est totalement indépendante de l'ordre du canal.

Mots clés : Identification / égalisation aveugle, Filtrage adaptatif rapide, Estimation et poursuite de sous-espace, Analyse asymptotique des performances.

Abstract

This thesis deals with the blind channel identification / equalization problem. Conventional non-blind equalization algorithms require training sequence or a priori knowledge of the channel. These solutions reduce the effective channel throughput, since a training sequence is usually sent periodically. It follows that the blind equalization of transmission channels represents a suitable alternative to traditional equalization, because they do not fully rely on training sequence or a priori channel knowledge. However, many of blind identification / equalization schemes suffer from various shortcomings, such as, slow convergence rate, high computational complexity and great sensitivity to channel order overestimation errors. Thus, our objective is, first, to summarize, clarify and evaluate various existing work. Then, we propose two new adaptive and fast blind minimum mean square error (MMSE) equalization algorithms, of noisy multi-channel finite impulse response (FIR) systems, that rely only on second order statistics. The proposed algorithms offer some advantages, such as, a low computational complexity, better robustness against channel order over-estimation errors for the first approach and a total independence from the channel order for the second approach.

Keywords: Blind identification / equalization, Fast adaptive filtering, Subspace estimating and tracking, Asymptotic performance analysis.

Remerciements

Je ne saurais présenter cette étude sans remercier tous ceux qui ont contribué à son aboutissement. Je tiens en tout premier lieu à remercier chaleureusement Adel Belouchrani, Professeur ENP, et Karim Abed-Meraim, Maître de conférences, HDR Télécom Paris. Adel, Karim, merci pour m'avoir mis sur de si bons rails, pour votre soutien sans faille, votre gentillesse et votre grande tolérance et patience vis à vis de mes faiblesses. Merci surtout pour la confiance que vous m'avez accordée et pour l'amitié que vous avez témoignée à mon égard. Et enfin, merci à toi Karim pour avoir été l'artisan de mes nombreux séjours à Télécom Paris. Je te remercie pour m'avoir si bien accueilli et n'avoir ménagé aucun effort pour que mes stages à Télécom Paris soient des réussites totales.

Je tiens ensuite à remercier Amrane Houacine, Professeur USTHB, pour l'honneur qu'il me fait en acceptant la présidence du jury de cette thèse. Mes remerciements s'adressent également à Ahmed Zergueras, Professeur ENP, Elbey Bourennane, Professeur Université de Bourgogne (France), Latifa Hamami, Maître de conférences ENP, Braham Himed, Ph.D. et chercheur, Air Force Research Laboratory (USA), et Mohamed Ouadjaout, Chargé de cours et Chef du département des Sciences Fondamentales ENP, et qui ont bien voulu me faire l'honneur d'examiner ce travail et d'être membres du jury.

Mes remerciements s'adressent également à Yves Grenier, Chef du département TSI (Télécom Paris), pour m'avoir accepté au sein de son département. Aussi, je tiens à remercier le Ministère de l'Enseignement Supérieur et de la Recherche Scientifique (Algérie) et le Centre Nationale des Oeuvres Universitaires (France), pour leur soutien financier.

Enfin, je ne saurais oublier dans mes remerciements Abdeldjalil Aissa-el-bey, mon collègue de bureau (durant dix huit mois) au département TSI (Télécom Paris) et Hicham Bousbia-Salah, mon copain de chambre (aussi durant dix huit mois) à la Maisel de Télécom Paris et collègue à l'ENP, pour leur soutien et leur amitié, ainsi que Mohamed-Oussaid Taghi mon ami de toujours et collègue au département des Sciences Fondamentales de l'ENP et qui a été à l'origine de mon premier contacte avec Karim Abed-Meraim.

Table des matières

Table des matières	1
Acronymes	5
Notations	7
Table des figures	9
Introduction	11
I État de l’art	15
1 Généralités sur l’identification / égalisation aveugle	17
1.1 Introduction	17
1.2 Formulation du problème	18
1.3 Structures d’égalisation	19
1.4 Diversité spatio-temporelle	20
1.4.1 Diversité temporelle	21
1.4.2 Diversité spatiale	22
1.5 Modèle de données	23
1.6 Exemple d’égalisation (indirecte) aveugle du second ordre :	
La méthode sous-espace	24
1.6.1 Décomposition sous-espace	24
1.6.2 Caractérisation de la réponse impulsionnelle du canal par le sous-espace bruit	25
1.6.3 Algorithme d’identification. Considérations pratiques	26
1.6.4 Égalisation	28
1.6.5 Simulations et discussion	29
1.7 Conclusion	32
2 Quelques méthodes d’identification / égalisation aveugles robustes aux erreurs de surestimation de l’ordre du canal	35
2.1 Introduction	35

2.2	Méthode de la prédiction linéaire	35
2.2.1	Définitions et notations	36
2.2.2	Égaliseurs ZF et MMSE	36
2.2.3	Égalisation aveugle	37
2.2.4	Implémentation adaptative	39
2.3	Méthode des égaliseurs mutuellement référencés	40
2.3.1	Solution asymptotique du critère MRE	41
2.3.2	Implémentation adaptative	42
2.4	Méthode des matrices d'autocorrélation décalées	43
2.5	Méthodes d'optimisation sous contraintes	44
2.5.1	Algorithme de Capon	45
2.5.2	Égalisation par annulateurs de lobes secondaires généralisée	46
2.5.3	Implémentation adaptative	47
2.6	Méthode sous-espace robuste	49
2.6.1	Sélection de l'ordre du canal	50
2.6.2	Algorithme	51
2.7	Méthode de la décomposition en produit extérieur	51
2.7.1	Produit extérieur	52
2.7.2	Algorithme	53
2.8	Méthode de lissage au sens des moindres carrés	54
2.8.1	Filtre de lissage	54
2.8.2	Algorithme pour l'estimation du canal et la détection de son ordre	56
2.9	Complexité de calcul	59
2.10	Simulations	59
2.11	Conclusion	62
II Développements algorithmiques et étude des performances		63
3 Égaliseur MMSE MIMO aveugle adaptatif rapide et robuste aux erreurs de surestimation de l'ordre du canal		65
3.1	Introduction	65
3.2	Développement de l'algorithme	66
3.2.1	Égaliseur MMSE	66
3.2.2	Égalisation aveugle	67
3.2.3	Implémentation	67
3.2.4	Selection du retard de l'égaliseur	69
3.2.5	Robustesse	71
3.3	Implémentation adaptative rapide	72
3.3.1	Le cas SIMO	73

3.3.2	Le cas MIMO	75
3.3.3	Procédure à deux étapes	77
3.4	Analyse asymptotique des performances	77
3.4.1	Perte asymptotique en performances	78
3.4.2	Validation des expressions des covariances asymptotiques	80
3.5	Résultats des simulations et discussion	82
3.5.1	Évaluation des performances	83
3.5.2	Robustesse aux erreurs de surestimation de l'ordre du canal	88
3.5.3	Robustesse aux faibles valeurs de $\mathbf{H}(0)$	88
3.5.4	Influence du nombre de capteurs	91
3.5.5	Discussion	92
3.6	Conclusion	93
4	Égaliseur aveugle indépendant de l'ordre du canal	95
4.1	Introduction	95
4.2	Développement de l'algorithme	95
4.3	Implémentation Adaptative	97
4.3.1	Procédure adaptative à deux étapes via l'algorithme du gradient stochastique normalisé	99
4.3.2	Procédure adaptative à deux étapes via l'algorithme des moindres carrés récursif	99
4.4	Résultats des simulations	101
4.4.1	Estimation du sous-espace mineur	101
4.4.2	Égalisation aveugle	101
4.5	Conclusion	102
	Conclusion générale	105
	Bibliographie	107
	Annexes	112
A	Démonstrations des lemmes et théorèmes	115
A.1	Démonstration du <i>Théorème 2.1</i>	115
A.2	Démonstration du <i>Théorème 2.2</i>	116
A.3	Démonstration du <i>Théorème 2.3</i>	116
A.4	Démonstration du <i>Lemme 2.1</i>	117
A.5	Démonstration du <i>Théorème 2.4</i>	117
A.6	Démonstration du <i>Théorème 3.1</i>	118
A.7	Démonstration du <i>Lemme 4.1</i>	119
B	Publications	121

Acronymes

Pour des raisons de lisibilité, la signification d'une abréviation ou d'un acronyme n'est souvent rappelée qu'à sa première apparition dans le texte. Par ailleurs, puisque nous utilisons toujours l'abréviation la plus usuelle, il est fréquent que ce soit le terme anglais qui est employé.

ACMA	: <i>Analytical Constant Modulus Algorithm</i>
A-CMS	: <i>Adaptive Constant Modulus Separation</i>
AIC	: <i>Akaike's Information theoretic Criteria</i>
DAB	: <i>Digital Audio Broadcast</i>
BSS	: <i>Blind Source Separation</i>
CDMA	: <i>Code Division Multiple Access</i>
CMA	: <i>Constant Modulus Algorithms</i>
DFE	: <i>Decision Feedback Equalizer</i>
DVB	: <i>Digital Video Broadcast</i>
EVD	: <i>Eigen Value Decomposition</i>
FDPM	: <i>Fast Data Projection Method</i>
FIR	: <i>Finite Impulse Response</i>
GMSK	: <i>Gaussian Minimum Shift Keying</i>
GRDA	: <i>Gazzah, Regalia, Delmas and Abed-Meraim</i>
GSM	: <i>Global System for Mobile communications</i>
GSC	: <i>Generalized Sidelobe Cancelers</i>
HFRANS	: <i>Householder transformation based implementation of Fast Rayleigh's quotient-based Adaptive Noise Subspace algorithm</i>
iid	: <i>independent identically distributed</i>
IIR	: <i>Infinite Impulse Response</i>
ISI	: <i>InterSymbol Interference</i>
LPA	: <i>Linear Prediction Algorithm</i>
LSS	: <i>Least Square Smoothing</i>
LC	: <i>Linear Constraint</i>
LMS	: <i>Least Mean Squares</i>

MDL	: <i>Minimum Description Length</i>
MIMO	: <i>Multiple Input Multiple Output</i>
MMSE	: <i>Minimum Mean Square Error</i>
MRE	: <i>Mutually Referenced Equalizers</i>
MSE	: <i>Mean Square Error</i>
MVDR	: <i>Minimum Variance Distorsionless Response</i>
OFDM	: <i>Orthogonal Frequency Division Multiplexing</i>
OOJA	: <i>Orthogonal OJA</i>
OPAST	: <i>Orthogonal Projection Approximation Subspace Tracking</i>
OPDA	: <i>Outer-Product Decomposition Algorithm</i>
PAM	: <i>Pulse Amplitude Modulation</i>
PAST	: <i>Projection Approximation Subspace Tracking</i>
PASTd	: <i>Projection Approximation Subspace Tracking with deflation</i>
PSK	: <i>Phase Shift Keying</i>
RLS	: <i>Recursive Least Squares</i>
RMSE	: <i>Relative Mean Square Error</i>
RSFA	: <i>Robust Subspace Fitting Algorithm</i>
QAM	: <i>Quadrature Amplitude Modulation</i>
QC	: <i>Quadrature Constraint</i>
SIMO	: <i>Single Input Multiple Output</i>
SISO	: <i>Single-Input Single-Output</i>
SNR	: <i>Signal to Noise Ratio</i>
SSA	: <i>Sub-Space Algorithm</i>
SVD	: <i>Singular Value Decomposition</i>
YAST	: <i>Yet Another Subspace Tracker</i>
ZF	: <i>Zero Forcing</i>

Notations

Sauf indication, nous adoptons les conventions sémantiques (standards) suivantes :

- Les lettres en style normale (comme $a, A, \gamma, \Gamma, \dots$) sont utilisées pour représenter des scalaires.
- Les lettres minuscules en style gras (comme $\mathbf{a}, \boldsymbol{\gamma}, \dots$) sont utilisées pour représenter des vecteurs.
- Les lettres majuscules en style gras (comme $\mathbf{A}, \boldsymbol{\Gamma}, \dots$) sont utilisées pour représenter des matrices.

Aussi, nous utilisons certaines notations informelles de MATLAB, telles $\mathbf{A}(:, i : j)$, $\mathbf{A}(i : j, :)$, `eig`, `svd`, ... pour désigner, respectivement, un bloc de colonnes de la matrice \mathbf{A} , un bloc de lignes de la matrice \mathbf{A} , une décomposition en valeurs/vecteurs propres, une décomposition en valeurs/vecteurs singuliers, ... Nous avons regroupé ci-dessous les principales notations employées dans les différents chapitres du document. Dans la mesure du possible, nous avons tenté de conserver les mêmes notations d'un chapitre à l'autre.

\mathbb{N}	: L'ensemble des entiers naturels
\mathbb{Z}	: Anneau des entiers relatifs
\mathbb{R}	: Corps des réels
\mathbb{C}	: Corps des complexes
$(\cdot)^*$: Conjugué
$(\cdot)^T$: Transposé
$(\cdot)^H$: Conjugué transposé
$(\cdot)^\#$: Pseudo-inverse au sens de Moore-Penrose
$ \cdot $: Valeur absolue
$\ \cdot\ $: Norme euclidienne (de Frobinious)
$\Re(\cdot)$: Partie réelle
$\Im(\cdot)$: Partie imaginaire
$E(\cdot)$: Espérance mathématique
$\delta(\tau)$: Impulsion de Kronecker (qui est égale à 1, pour $\tau = 0$ et à 0 sinon)
\mathbf{I}_n	: Matrice unité de dimension $n \times n$
$\mathbf{0}$: Matrice nulle de dimension appropriée
$\mathbf{0}_{n,m}$: Matrice nulle de dimension $n \times m$
$\mathcal{J}_{n,m,l}$: Matrice de troncation définie par $\mathcal{J}_{n,m,l} = \begin{bmatrix} \mathbf{0}_{m,n} & \mathbf{I}_m & \mathbf{0}_{m,l} \end{bmatrix}^T$

- \mathbf{J}_n : Matrice de troncation définie par $\mathbf{J}_n \stackrel{\text{def}}{=} \begin{cases} \begin{bmatrix} \mathbf{0}_{1,n-1} & 0 \\ \mathbf{I}_{n-1} & \mathbf{0}_{n-1,1} \end{bmatrix}, & \text{si } n \geq 2 \\ 0, & \text{si } n = 1 \end{cases}$
- avec $\mathbf{J}_n^0 = \mathbf{I}_n$ et $\mathbf{J}_n^{-1} = \mathbf{J}_n^T$, $n \in \mathbb{N}^*$
- * : Produit de convolution de deux signaux complexes.
- $x(t) * y(t) \stackrel{\text{def}}{=} \int_{-\infty}^{+\infty} x(\tau)y^*(t - \tau)d\tau$, dans le cas où les signaux $x(t)$ et $y(t)$ sont à temps continu.
- $x(k) * y(k) \stackrel{\text{def}}{=} \sum_{i \in \mathbb{Z}} x(i)y^*(k - i)$, dans le cas où les signaux $x(t)$ et $y(t)$ sont à temps discret.
- \otimes : Produit de Kronecker, $\mathbf{A} \otimes \mathbf{B} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{A}(1,1)\mathbf{B} & \cdots & \mathbf{A}(1,j)\mathbf{B} & \cdots \\ \vdots & \ddots & \vdots & \\ \mathbf{A}(i,1)\mathbf{B} & \cdots & \mathbf{A}(i,j)\mathbf{B} & \\ \vdots & & & \ddots \end{bmatrix}$
- $\text{diag}(\cdot)$: Matrice (bloc-) diagonale, $\text{diag}(\mathbf{A}_1, \dots, \mathbf{A}_n) \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 & & \vdots \\ \vdots & & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{A}_n \end{bmatrix}$
- $O(\cdot)$: Notation de Landau, $a_n = O(b_n) \Leftrightarrow \exists k > 0, \exists N > 0, \forall n > N : |a_n| < k|b_n|$
- $\text{vec}(\cdot)$: Vecteurisation-colonne de matrices, $\text{vec} \left(\begin{bmatrix} \mathbf{a}_1 & \cdots & \mathbf{a}_n \end{bmatrix} \right) \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{a}_1^T & \cdots & \mathbf{a}_n^T \end{bmatrix}^T$
- $\text{vec}^{-1}(\cdot)$: Opérateur inverse de $\text{vec}(\cdot)$
- $\text{range}(\mathbf{A})$: Image de la matrice $\mathbf{A} \in \mathbb{C}^{n \times m}$, $\text{range}(\mathbf{A}) = \{\mathbf{y} \in \mathbb{C}^n : \mathbf{y} = \mathbf{A}\mathbf{x}, \mathbf{x} \in \mathbb{C}^m\}$
- $\text{null}_l(\mathbf{A})$: Noyau gauche de la matrice $\mathbf{A} \in \mathbb{C}^{n \times m}$, $\text{null}_l(\mathbf{A}) = \{\mathbf{z} \in \mathbb{C}^n : \mathbf{z}^H \mathbf{A} = \mathbf{0}\}$
(Sous-espace vectoriel de \mathbb{C}^n , orthogonal au sous-espace $\text{range}(\mathbf{A})$)
- $\text{null}_r(\mathbf{A})$: Noyau droit de la matrice $\mathbf{A} \in \mathbb{C}^{n \times m}$, $\text{null}_r(\mathbf{A}) = \{\mathbf{z} \in \mathbb{C}^m : \mathbf{A}\mathbf{z} = \mathbf{0}\}$
Dans le cas où \mathbf{A} est hermétienne, $\text{null}_l(\mathbf{A}) = \text{null}_r(\mathbf{A}) = \text{null}(\mathbf{A})$

Table des figures

1.1	<i>Égaliseur à retour de décision (DFE).</i>	20
1.2	<i>Diversité temporelle.</i>	21
1.3	<i>Diversité spatiale.</i>	22
1.4	<i>Performances de l'algorithme sous-espace.</i>	30
1.5	<i>Dégradation des performances de l'algorithme sous-espace, quand l'ordre du canal est surestimé (l'ordre exacte $L=4$).</i>	31
2.1	<i>Comparaison des performances des égaliseurs.</i>	60
2.2	<i>Comparaison des coûts de calcul.</i>	61
2.3	<i>Robustesse aux erreurs de surestimation de l'ordre, l'ordre exacte $L=3$, $SNR=20$ dB.</i>	61
3.1	<i>Perte asymptotique en performances : contrainte quadratique.</i>	81
3.2	<i>Performances de l'égaliseur, cas SIMO.</i>	82
3.3	<i>Performances de l'égaliseur, cas MIMO.</i>	83
3.4	<i>Convergence de l'égaliseur adaptatif, cas SIMO.</i>	83
3.5	<i>Convergence de l'égaliseur adaptatif, cas MIMO.</i>	84
3.6	<i>Convergence de l'algorithme d'égalisation adaptatif dans le cas d'un canal variable dans le temps.</i>	85
3.7	<i>Comparaison des performances d'égaliseurs SIMO ($q = 1$, $p = 3$, $L = 4$, $N = 6$), dans le cas d'un traitement en bloc.</i>	86
3.8	<i>Comparaison des coûts de calcul d'algorithmes d'égalisation SIMO ($q = 1$, $p = 3$, $L = 4$, $N = 6$), dans le cas d'un traitement en bloc.</i>	86
3.9	<i>Comparaison des performances d'égaliseurs SIMO ($q = 1$, $p = 3$, $L = 4$, $N = 6$), dans le cas d'un traitement adaptatif.</i>	87
3.10	<i>Comparaison des coûts de calcul d'algorithmes d'égalisation SIMO ($q = 1$, $p = 3$, $L = 4$, $N = 6$), dans le cas d'un traitement adaptatif.</i>	87
3.11	<i>Comparaison de Robustesse (aux erreurs de surestimation de l'ordre du canal), l'ordre exacte est $L = 4$.</i>	89
3.12	<i>Comparaison de Robustesse (aux erreurs de surestimation de l'ordre du canal), l'ordre exacte est $L = 4$.</i>	89

3.13	<i>Comparaison de Robustesse (aux erreurs de surestimation de l'ordre du canal), cas d'un canal à évanouissement en fréquence, l'ordre exacte est $L = 4$.</i>	90
3.14	<i>Robustesse aux faibles valeurs de $\mathbf{H}(0)$, SNR=15dB.</i>	90
3.15	<i>Robustesse aux faibles valeurs de $\mathbf{H}(0)$, SNR=30dB.</i>	91
3.16	<i>MSE en fonction du nombre de capteurs, SNR=5dB.</i>	92
3.17	<i>MSE en fonction du nombre de capteurs, SNR=15dB.</i>	92
4.1	<i>Performances des algorithmes de suivi de sous-espaces mineurs.</i>	102
4.2	<i>Performance de l'égaliseur, $K = 500$.</i>	103
4.3	<i>Convergence de l'égaliseur adaptatif, SNR=15dB.</i>	103
4.4	<i>Robustesse aux erreurs de surestimation de l'ordre du canal, l'ordre exacte $L = 4$, $K = 500$ et SNR=15dB.</i>	104

Introduction

L'égalisation aveugle (autodidacte) est un processus durant lequel, une séquence de données inconnues à l'entrée d'un canal est récupérée uniquement à partir des signaux recueillis à la sortie de ce canal. De ce fait, l'égalisation aveugle est devenue un axe de recherche important en traitement numérique du signal. En effet, les trois dernières décennies ont vu la publication d'un nombre impressionnant de travaux qui concernent directement ce problème (voir [28, 40] ainsi que les références qui s'y trouvent). L'importance du rôle que joue un égaliseur canal dans un système de communications numériques est bien connue et différentes approches (non-aveugles) d'optimisation et d'estimation d'égaliseurs ont été proposées (e.g. voir [66]). Comme la majorité des systèmes de communications sont souvent soumis aux contraintes de limitation de la bande de transmission, il est alors souhaitable d'avoir à la réception un égaliseur canal optimal, sans pour autant occuper une partie de la bande passante du canal. En éliminant la séquence d'apprentissage et en maximisant la capacité du canal à transmettre l'information utile, l'égalisation aveugle du canal présente une solution efficace pour compenser les distorsions du canal. De même, l'égalisation aveugle trouve tout son intérêt dans certains systèmes de communications, où les symboles de la séquence d'apprentissage ne sont pas toujours disponibles (e.g. systèmes point à multi-points). Comparée aux approches traditionnelles (avec séquence d'apprentissage), l'égalisation aveugle donne lieu à de véritables défis et enjeux qui prennent de plus en plus d'ampleur.

Vu le nombre important de travaux publiés, qui sont en relation avec le problème de l'égalisation aveugle, on pensera que toutes les questions ayant trait à ce problème sont réglées. Ce qui n'est pas le cas en réalité. Une question toute simple peut se poser : *Pourquoi n'y a-t-il pas une large utilisation, en pratique, des schémas d'égalisation aveugle, pour ainsi en bénéficier de leur avantage majeur (comparés aux schémas d'égalisation classiques) à savoir la suppression de la séquence d'apprentissage.* On pourra donner plusieurs réponses à cette question, chacune de ces réponses peut, à elle seule, constituer un axe de recherche qui justifierait la nécessité de déployer des efforts. On peut alors citer comme une des raisons, le fait qu'un grand nombre d'ingénieurs n'ont pas une bonne connaissance de tous les aspects du problème. Des années durant, les travaux en égalisation aveugle sont restés à l'intérieur du cercle limité des académiciens. Souvent, dans beaucoup de ces travaux, on met plus l'accent sur la nouveauté des schémas proposés que sur leurs retombées en pratique. Ce qui joue en défaveur d'une large utilisation pratique de ces schémas. Par conséquent, le premier objectif visé par

cette étude c'est de contribuer à faire connaître ce domaine (identification / égalisation aveugle) à une large frange de chercheurs et d'ingénieurs, notamment, ceux qui activent en traitement du signal et en systèmes de communications.

Une autre raison fait que jusqu'à présent, peu d'algorithmes d'égalisation aveugle sont répandus en pratique, est la différence en performance, en complexité de calcul et en robustesse entre égaliseurs aveugles et égaliseurs classiques. En effet, dans plusieurs applications pratiques, en particulier dans un environnement mobile, une convergence rapide de l'égaliseur est nécessaire pour répondre aux variations dans le temps des paramètres du canal. Alors que dans la plus part des algorithmes d'égalisation aveugle existants, un flux important de données est requis afin d'atteindre un niveau de convergence acceptable. De même, l'implémentation de la plus part des méthodes d'égalisation aveugle est d'une complexité de calcul supérieure à celle des méthodes classiques. En plus, les performances des égaliseurs aveugles, en particulier ceux basés sur des statistiques d'ordre deux des signaux reçus, sont très sensibles aux erreurs de surestimation de l'ordre du canal. Même si beaucoup de progrès ont été réalisés pour améliorer la vitesse de convergence, réduire la complexité de calcul et atténuer la sensibilité des égaliseurs aveugles, ça n'empêche que les ingénieurs ont toujours tendance à favoriser les schémas bien connus avec séquence d'apprentissage dans leurs conceptions de nouveaux systèmes de communications.

Quoi qu'il en soit, on dira que les problèmes liés à l'égalisation aveugle sont loins d'être entièrement résolus et que beaucoup d'efforts de recherche restent à faire, afin de pouvoir développer des égaliseurs aveugles plus rapides, plus simples et plus robustes, et de prétendre à une réelle compétitivité avec les égaliseurs traditionnelles. Se cadrant autour de cette problématique, cette étude vient, dans un premier temps, résumer, clarifier (si nécessaire) et évaluer un certain nombre de travaux déjà existants, et dans un deuxième temps, proposer de nouvelles approches et identifier de nouveaux axes de recherche afin d'améliorer les performances des systèmes d'identification / égalisation aveugle et de pallier à leurs carences.

Dans ce travail de thèse, nous développons deux techniques originales d'égalisation du type erreur quadratique moyenne minimale (*Minimum Mean Square Error*, MMSE) aveugle du second ordre, pour des systèmes multicanaux à réponse impulsionnelle finie. La première approche, destinée à des systèmes à entrées multiples sorties multiples (*Multiple Input Multiple Output*, MIMO), est robuste aux erreurs de surestimation de l'ordre du canal. Tandis que la deuxième approche, destinée à des systèmes à une entrée plusieurs sorties (*Single Input Multiple Output*, SIMO), est totalement indépendante de l'ordre du canal. Dans les deux approches, nous incluons une procédure dite à *deux étapes* qui, d'une part, permet de compenser la baisse en performances comparée au cas non-aveugle (cette baisse est due au processus d'estimation aveugle), d'autre part, elle permet de choisir des retards d'égalisation non nuls. De même, nous développons, pour les deux approches, des implémentations adaptatives rapides, ayant une complexité de calcul linéaire. Concernant la première approche, nous effectuons une analyse asymptotique des performances, dont les résultats théoriques seront validés par des simulations numériques. En marge de ces travaux, nous présentons une étude comparative d'algorithmes rapides d'estimation et de suivi de sous-espaces.

La première partie de ce travail est consacrée à l'état de l'art. Après une introduction générale sur les systèmes d'identification / égalisation aveugle, on donne une description relativement détaillée des méthodes les plus en vue d'identification / égalisation aveugle et qui ont la particularité d'être robustes aux erreurs de surestimation de l'ordre du canal et de présenter un coût de calcul faible relativement à d'autres méthodes, telles la méthode de la prédiction linéaire, la méthode de la décomposition en produit extérieur, la méthode de lissage au sens des moindres carrés, etc. Dans la deuxième partie, nous décrivons les différentes approches que nous proposons . nous présentons de façon détaillée (développement algorithmique et discussion des résultats) une première approche d'égalisation MMSE aveugle de systèmes MIMO, robuste aux erreurs de surestimation de l'ordre du canal, suivit d'une seconde approche d'égalisation MMSE aveugle de systèmes SIMO et qui a la particularité d'être totalement indépendante de l'ordre du canal.

Première partie

État de l'art

Chapitre 1

Généralités sur l'identification / égalisation aveugle

1.1 Introduction

Dans le cadre des transmissions numériques à hauts débits, la propagation induit une dispersion des impulsions dans le temps, en particulier par des trajets multiples de grande durée par rapport au trajet le plus court. Dans la plupart des conditions de propagation (hormis satellite), la dispersion se traduit par une transformation linéaire du signal émis. Si on considère le signal reçu après échantillonnage, cette dispersion conduit à deux types d'évanouissements en temps et en fréquence. Nous nous intéressons plus particulièrement à ces derniers qui se traduisent par des interférences entre symboles (*InterSymbol Interference*, ISI) successifs que l'on modélise par un filtre linéaire invariant par décalage. Cette modélisation suppose les conditions de propagation invariantes pendant la durée de transmission. Les détecteurs de symboles à seuil classiques voient leurs performances dégradées au point de les rendre inutilisables. Il faut alors introduire un traitement, l'*égalisation*, pour réduire l'effet des ISI ou pour ajuster le détecteur au canal. La principale alternative au problème de l'égalisation consiste à utiliser une modulation par répartition de fréquence (e.g. de type *Orthogonal Frequency Division Multiplexing*, OFDM) qui transforme les ISI en évanouissements temporels moyennant un intervalle de garde suffisant. Ces techniques ne seront pas abordées dans le cadre de ce travail.

Quand cela est possible, le réglage de l'égaliseur est effectué à l'aide d'une séquence d'apprentissage (parfois appelée symboles pilotes), des symboles connus d'avance du récepteur. Malheureusement ces symboles pilotes ne sont pas toujours disponibles, en particulier dans les systèmes point à multi-points (diffusions radio et télévision (*Digital Audio Broadcast*, DAB, *Digital Video Broadcast*, DVB), télévision numérique câblée [41]) ou en quantité suffisante (canaux de propagation variant dans le temps dans le cas de transmissions avec des mobiles à grandes vitesses). De plus quand elle existe, la séquence d'apprentis-

sage occupe une part importante des données transmises (e.g. 1/5 de chaque trame pour le *Global System for Mobile communications*, GSM). On peut alors vouloir supprimer la séquence d'apprentissage ou tenter de la réduire pour gagner un débit utile. Dans tous ces cas, on veut pouvoir *égaliser sans séquence d'apprentissage : en aveugle*. Lorsque l'on se contente de réduire la taille de la séquence d'apprentissage, en combinant une technique aveugle et l'apprentissage classique, on parle d'égalisation semi-aveugle. Même si l'égalisation (avec séquence d'apprentissage) de canaux de transmission a suscité une intense activité de recherche depuis la première implémentation adaptative d'un égaliseur canal en 1965 par R. Lucky [60, 61], le concept d'égalisation sans séquence d'apprentissage n'a eut vraiment d'échos qu'en 1975, quand Y. Sato [68] a présenté un égaliseur linéaire simple pour des signaux modulés en amplitude pulsé (*Pulse Amplitude Modulation*, PAM) dans un système discret mono-entrée mono-sortie (SISO : *Single-Input Single-Output*). Néanmoins, la première percé remarquable dans le domaine de l'égalisation aveugle a été réalisé en 1980 par A. Benveniste *et al* [17], qui établirent les principes d'une déconvolution aveugle pour des signaux analogiques à l'entrée d'un canal SISO, en plus ils généralisèrent la méthode originale de Y. Sato [68]. Le terme *égalisation aveugle* a été officiellement introduit par A. Benveniste et M. Goursat dans leur article de 1982 [18]. Les premières méthodes proposées pour l'égalisation aveugle (A. Benveniste *et al* [17]), repose sur des statistiques d'ordre supérieur. Certains algorithmes ont été développés indépendamment, dans le cadre de la déconvolution de données ne provenant pas de communications numériques (sismique, imagerie,...), voir [13]. A partir de 1991, Gardner [32] et Tong *et al* [74] ont suggéré de suréchantillonner le signal reçu pour donner une nature cyclique aux statistiques du second ordre du signal reçu qui est équivalente à un caractère à phase minimal du signal. Depuis 1992, de nombreuses méthodes ont été proposées qui repose sur la diversité temporelle ainsi créée ou bien sur la diversité spatiale (utilisation de plusieurs capteurs en réception) qui lui est équivalente. De façon non-exhaustive, la famille d'algorithmes du second ordre destinés à l'identification / égalisation du canal inclue, entre autres, les approches basées sur la décomposition sous-espace, celles des moindres carrés, du maximum de vraisemblance et de la prédiction linéaire.

1.2 Formulation du problème

Afin de transmettre une information, le signal est d'abord converti en une suite de symboles $\{s(t)\}_{t \in \mathbb{Z}}$ (la variable t désigne un temps discret), choisis dans un alphabet fini de phase et/ou d'amplitude (*Quadrature Amplitude Modulation*, QAM, *Phase Shift Keying*, PSK, *Pulse Amplitude Modulation*, PAM etc.) et qui sont supposés indépendants. Ces symboles sont issus d'un codeur de source, puis d'un codeur de canal, éventuellement d'un entrelaceur et d'un convertisseur bits-symboles (code de Gray par exemple). Nous supposons toutes ces opérations parfaitement réalisées et nous nous intéressons à la transmission des symboles $\{s(t)\}_{t \in \mathbb{Z}}$.

Le signal émis autour de la fréquence porteuse f_0 est de la forme $\Re\{s_e(t_c)e^{-j2\pi f_0 t_c}\}$ ¹, où $s_e(t_c) = \sum_l s(l)h_e(t_c - lT_s)$ est l'enveloppe complexe du signal et où $h_e(t_c)$ est le filtre d'émission qui forme les

¹Ici, la variable réelle t_c désigne un temps continu et le symbole j désigne un nombre complexe tel que $j^2 = -1$. Pour la suite du rapport et sauf indication, le symbole j désigne un entier.

impulsions émises au rythme symbole $1/T_s$.

Après suppression de la porteuse (supposée sans résidu) et du filtre de réception, le signal complexe reçu en bande de base s'écrit sous la forme

$$x(t_c) = h_r(t_c) * c(t_c) * s_e(t_c) + h_r(t_c) * b'(t_c),$$

où $h_r(t_c)$ est le filtre de réception qui est en général choisi adapté au filtre d'émission $h_e(t_c) = h_e^*(-t_c)$ de manière à limiter le signal à la bande passante du signal utile. Le filtre d'émission $h_e(t_c)$ est le plus souvent une racine de cosinus surélevé afin de respecter la condition de Nyquist (pas d'ISI) en l'absence de canal de propagation $c(t_c)$, voir [66] pour plus de détails. Enfin, $\tilde{b}(t_c)$ est un bruit additif supposé blanc centré.

Après échantillonnage au rythme symbole, on obtient

$$x(t) \stackrel{\text{def}}{=} x(t_c)|_{t_c=tT_s} = \sum_{k=0}^L h(k)s(t-k) + b(t),$$

où

$$h(k) \stackrel{\text{def}}{=} h_r(t_c) * c(t_c) * h_e(t_c)|_{t_c=kT_s}$$

est le k -ième coefficient de la réponse impulsionnelle du filtre, appelé *canal discret équivalent*, qui rassemble l'effet des filtres d'émission et de réception et du canal de propagation. En général, on suppose le canal à réponse impulsionnelle finie (*Finite Impulse Response, FIR*) de degré L . Cette approximation est validée par de nombreuses expériences à condition de considérer une valeur de L suffisamment grande. Le bruit discret $b(t)$ est donné par

$$b(t) \stackrel{\text{def}}{=} h_r(t_c) * b'(t_c)|_{t_c=tT_s}.$$

Le problème de l'égalisation aveugle consiste à estimer la suite des symboles émis $\{s(t)\}_{t \in \mathbb{Z}}$ à partir de la seule connaissance de la séquence $\{x(t)\}_{t \in \mathbb{Z}}$ et éventuellement du type de l'alphabet utilisé pour engendrer les symboles. Aucune connaissance *a priori* de la séquence de symboles émis ou du canal discret équivalent n'est disponible.

1.3 Structures d'égalisation

Dans la littérature, on trouve une multitude d'approches d'égalisation aveugle, qu'on peut classer, comme en égalisation avec séquence d'apprentissage, en méthodes directes et méthode indirectes. Dans une approche d'égalisation aveugle directe, le filtre égaliseur est conçu directement à partir des statistiques des symboles émis et des signaux reçus observés. Dans le cas d'une démarche indirecte, on identifie d'abord la réponse impulsionnelle du canal, puis on récupère les symboles émis au moyen d'une technique d'estimation des symboles. Comme on peut le constater, en communications numériques, il existe un lien très étroit entre les notions identification canal et égalisation canal. C'est pourquoi, dans certaines approches (indirectes) on ne décrit que la partie proprement aveugle de l'égalisation (la partie

identification canal). Pour les techniques d'estimation des symboles utilisées une fois le canal identifié, on peut se référer à [66]. Il en découle de cette classification trois familles de structures (comme pour le cas avec séquence d'apprentissage) pour résoudre le problème de l'égalisation :

- Égalisation aveugle par un filtre linéaire. Nous recherchons alors un filtre $\mathbf{v} = [v(0) \cdots v(n-1)]^T$ tel que $\hat{s}(t) = \sum_{k=0}^{n-1} v(k) * x(t-k)$ soit une bonne estimée (à une constante près) de $s(t-t')$ où t' est un retard de traitement arbitraire. Cette structure est préférée dans le cas de canaux longs (plus de dix coefficients) et d'un bruit pas trop important.
- Égalisation aveugle par un filtre non linéaire. La forme la plus classique est celle du filtre à retour de décision (*Decision Feedback Equalizer*, DFE) [12] qui est constitué de filtres linéaires et d'un organe de décision (voir Fig.1.1). Les filtres du DFE peuvent être estimés soit directement, soit à partir d'une estimée du canal. Cette approche peut être préférée pour des canaux courts difficiles² et des alphabets de taille importante (QAM-16 et plus, PSK-8).

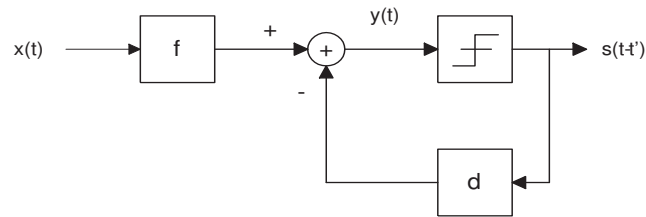


FIG. 1.1 – Égaliseur à retour de décision (DFE).

- Identification aveugle du canal discret équivalent, suivie d'une estimation des symboles par une technique du type maximum de vraisemblance (algorithme de Viterbi, voir [30]). Cette méthode est potentiellement la plus performante, mais sa complexité limite son utilisation pour des canaux courts (quelques coefficients) et des alphabets avec peu d'états (PSK-2, *Gaussian Minimum Shift Keying*, GMSK).

La valeur prise par les filtres intervenant dans ces différentes structures est déterminée de manière aveugle.

1.4 Diversité spatio-temporelle

Ce n'est qu'à partir de 1993 que l'apport potentiel de la diversité spatio-temporelle a été réellement compris, plus précisément suite aux travaux de Tong *et al.* [74] et de Gardner [32].

²Des canaux qui présentent des évanouissements en fréquence.

1.4.1 Diversité temporelle

La diversité temporelle (suréchantillonnage du signal reçu) a été utilisée dans les systèmes de communications numériques depuis les années 1970 afin d'effectuer des tâches telles que la synchronisation. L'utilisation de la diversité temporelle pour l'égalisation avait été proposée par Macchi et Guidou [62], puis étendue par Weinstein et Gitlin [83], sans que l'étude analytique de ses potentiels et limitations soit effectuée. Elle est pourtant mise en œuvre dans la plupart des récepteurs aveugles existants (voir [77]). La *diversité temporelle* correspond à l'échantillonnage du signal reçu à une cadence n_e fois plus grande que celle de la séquence d'émission, c.à.d à une fréquence $1/T' = n_e/T_s$. La séquence obtenue au rythme $1/T'$ est cyclostationnaire de période n_e . Cette propriété de cyclostationnarité peut être directement utilisée pour réaliser l'égalisation aveugle au second ordre, comme on le verra plus loin. Le signal cyclostationnaire peut également être décrit sous une forme multivariée.

En effet, considérons le cas général où n_e est rationnel, $n_e = p/q$ avec p et q deux entiers naturels

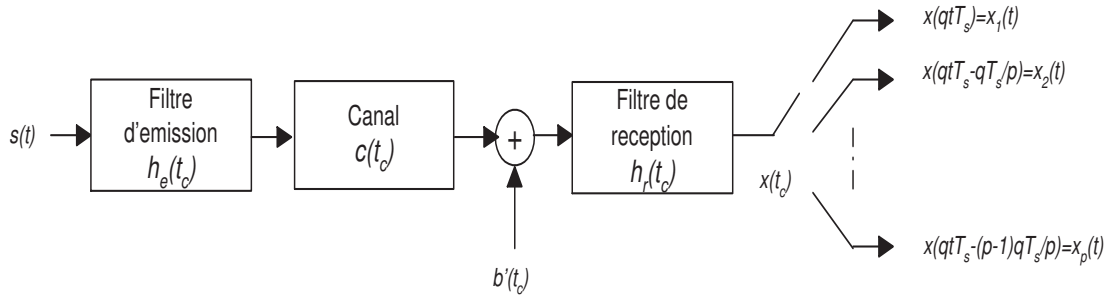


FIG. 1.2 – Diversité temporelle.

premiers entre eux et tels que $p > q \geq 1$. Après suréchantillonnage du signal reçu à un pas T' (voir Fig.1.2), on obtient

$$\begin{aligned} x_i(t) &\stackrel{\text{def}}{=} x(t_c)|_{t_c=(qt-(i-1)q/p)T_s}, \quad 1 \leq i \leq p \\ &= \sum_{k=0}^L \sum_{j=1}^q h_{ij}(k) s_j(t-k) + b_i(t), \end{aligned}$$

avec

$$\begin{aligned} h_{ij}(k) &\stackrel{\text{def}}{=} h_r(t_c) * c(t_c) * h_e(t_c)|_{t_c=(qk+j-1-(i-1)q/p)T_s}, \\ b_i(t) &\stackrel{\text{def}}{=} h_r(t_c) * b'(t_c)|_{t_c=(qt-(i-1)q/p)T_s}, \\ s_j(t) &= s(qt-j+1), \quad 1 \leq j \leq q. \end{aligned}$$

Posons

$$\mathbf{x}(t) = [x_1(t) \cdots x_p(t)]^T,$$

on obtient alors

$$\mathbf{x}(t) = \sum_{k=0}^L \mathbf{H}(k) \mathbf{s}(t-k) + \mathbf{b}(t), \quad (1.1)$$

où

$$\mathbf{H}(k) = \begin{bmatrix} h_{11}(k) & \cdots & h_{1q}(k) \\ \vdots & \ddots & \vdots \\ h_{p1}(k) & \cdots & h_{pq}(k) \end{bmatrix}, \quad (1.2)$$

$$\mathbf{s}(t) = [s_1(t) \cdots s_q(t)]^T, \quad (1.3)$$

$$\mathbf{b}(t) = [b_1(t) \cdots b_p(t)]^T.$$

Il est immédiat de constater que le signal $\mathbf{x}(t)$ de dimension p est stationnaire au sens large, c'est-à-dire que $E(\mathbf{x}(t + \alpha)\mathbf{x}^H(t))$ ne dépend que de α . L'expression (1.1) représente ainsi un modèle de filtrage entrées multiples sorties multiples (*Multiple Input Multiple Output*, MIMO), à q entrées et p sorties, au rythme $1/qT_s$ avec un bruit additif.

1.4.2 Diversité spatiale

A l'aide du modèle multi-dimensionnel (1.1), on voit facilement que la diversité temporelle est équivalente à propager à travers le canal des symboles émis par q sources différentes et à observer le signal reçu à l'aide d'un réseau de p capteurs. Chaque capteur i reçoit une contribution d'une source j filtrée par $h_{ij}(z) \stackrel{\text{def}}{=} \sum_{k=0}^L h_{ij}(k)z^{-k}$. On obtient ainsi l'équivalence de la diversité temporelle avec la diversité spatiale où q sources et p capteurs sont disposés dans des localisations différentes (voir Fig.1.3). Bien entendu, on peut combiner la diversité spatiale et la diversité temporelle pour augmenter le facteur de diversité p .

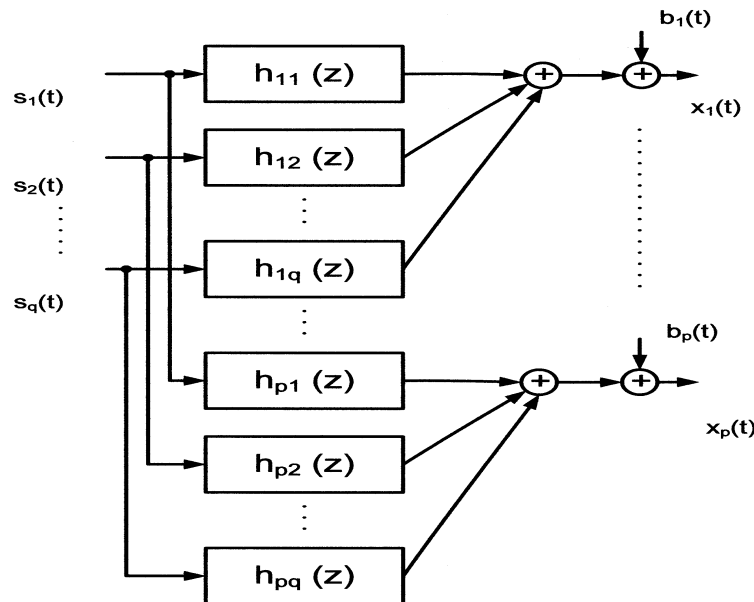


FIG. 1.3 – Diversité spatiale.

1.5 Modèle de données

Dans les deux cas, diversité spatiale et/ou temporelle, le canal discret équivalent est représenté par le système MIMO à q entrées et p sorties (avec $p > q$), donné par l'équation (1.1), qu'on peut représenter d'une manière compacte (et symbolique) en utilisant la fonction de transfert du canal (transformé en z de la réponse impulsionnelle du canal)

$$\begin{aligned}\mathbf{x}(t) &= [\mathbf{H}(z)]\mathbf{s}(t) + \mathbf{b}(t), \\ \mathbf{H}(z) &= \sum_{k=0}^L \mathbf{H}(k)z^{-k}.\end{aligned}$$

La fonction de transfert $\mathbf{H}(z)$ est un filtre FIR inconnu, de dimension $p \times q$, causal et stable. A noter que le modèle FIR est souvent suffisant dans des applications pratiques du fait qu'un système à réponse impulsionnelle infinie (*Infinite Impulse Response*, IIR) peut toujours être bien approximé par un modèle FIR si le degré L de ce dernier est suffisamment grand. Il reste que le modèle FIR permet, comme on le verra, des développements algorithmiques très simples pour réaliser l'égalisation aveugle. On suppose que :

- **(H1)** $\mathbf{H}(z)$ est irréductible et à colonnes réduites, i.e. $\forall z \in \mathbb{C} \cup \{\infty\} : \text{rank}(\mathbf{H}(z)) = q$ et $\mathbf{H}(L)$ est à rang colonne plein ($\text{rank}(\mathbf{H}(L)) = q$).
- **(H2)** Le signal d'entrée $\mathbf{s}(t)$ (non observable) est un vecteur aléatoire de dimension q , supposé être un processus indépendant identiquement distribué (iid) de moyenne nulle et de variance unité, i.e. $E(\mathbf{s}(t + \tau)\mathbf{s}^H(t)) = \delta(\tau)\mathbf{I}_q$.
- **(H3)** $\mathbf{b}(t)$ est un bruit additif, supposé blanc spatialement et temporellement, de variance σ_b^2 (i.e., $E(\mathbf{b}(t)\mathbf{b}^H(t)) = \sigma_b^2\mathbf{I}_p$) et indépendant de la séquence émise $\{\mathbf{s}(t)\}$.

Ces hypothèses sont les principales conditions nécessaires à l'identifiabilité des coefficients de la matrice polynômiale $\mathbf{H}(z)$ à partir seulement des observations du signal reçu $\mathbf{x}(t)$.

Le modèle de données que nous utiliserons (qui est adopté dans la plus part des méthodes d'identification / égalisation aveugle) est celui donné par la construction suivante :

On empile N observations successives du signal reçu $\mathbf{x}(t)$, N (taille de la fenêtre d'observation) étant pour l'instant un entier quelconque, il en résulte le modèle de données suivant

$$\begin{aligned}\mathbf{x}_N(t) &= [\mathbf{x}^T(t) \ \mathbf{x}^T(t-1) \ \cdots \ \mathbf{x}^T(t-N+1)]^T \\ &= \mathbf{H}_N \mathbf{s}_m(t) + \mathbf{b}_N(t),\end{aligned}\tag{1.4}$$

où $\mathbf{x}_N(t)$ est un vecteur de dimension $n = Np$, $\mathbf{s}_m(t)$ et $\mathbf{b}_N(t)$ deux vecteurs de dimension d et n respectivement, donnés par

$$\mathbf{s}_m(t) = [\mathbf{s}^T(t) \ \cdots \ \mathbf{s}^T(t-m+1)]^T, \tag{1.5}$$

$$\mathbf{b}_N(t) = [\mathbf{b}^T(t) \ \cdots \ \mathbf{b}^T(t-N+1)]^T, \tag{1.6}$$

$d = qm$, $m = N + L$, \mathbf{H}_N est la matrice de convolution du canal de dimension $n \times d$, donnée par

$$\mathbf{H}_N = \begin{bmatrix} \mathbf{H}(0) & \cdots & \mathbf{H}(L) & \mathbf{0} \\ & \ddots & & \ddots \\ \mathbf{0} & & \mathbf{H}(0) & \cdots & \mathbf{H}(L) \end{bmatrix}. \quad (1.7)$$

La matrice \mathbf{H}_N , dite matrice de filtrage, possède une structure de Sylvestre dont les propriétés sont exploitées par la plus part des techniques d'identification / égalisation aveugle. Une des propriétés importante de la matrice \mathbf{H}_N est donnée par le lemme suivant (voir [1] pour la démonstration)

Lemme 1.1 *Si la taille de la fenêtre d'observation est supérieure au degré du canal ($N > qL$), et si les conditions de l'hypothèse (H1) sont vérifiées, alors les colonnes de la matrice \mathbf{H}_N sont linéairement indépendantes (\mathbf{H}_N est une matrice à rang colonne plein : $\text{rank}(\mathbf{H}_N) = d$).*

De même, il existe un lien directe entre les structures de Sylvestre et les opérations de multiplication de polynômes. Soit $V(z) = \sum_{k=0}^{N-1} \mathbf{V}^*(k)z^{-k}$ un polynôme matriciel de dimension $p \times q$ auquel on associe la matrice $\mathbf{V} = [\mathbf{V}^H(0) \cdots \mathbf{V}^H(N-1)]^H$.

Dans ce cas, au polynôme

$$P(z) = \mathbf{V}^T(z)\mathbf{H}(z) = \sum_{k=0}^{m-1} \mathbf{P}(k)z^{-k},$$

est associé la matrice

$$\mathbf{P} = [\mathbf{P}(0) \cdots \mathbf{P}(m-1)] = \mathbf{V}^H \mathbf{H}_N.$$

1.6 Exemple d'égalisation (indirecte) aveugle du second ordre : La méthode sous-espace

Dans ce paragraphe, nous allons présenté une des premières techniques du second ordre d'identification / égalisation aveugle [64]. Le but étant de montrer la validité d'une telle démarche, ses avantages et ses faiblesses. Pour simplifier, prenant le cas ($q = 1$) d'un canal à une entrée et plusieurs sorties (*Single Input Multiple Output*, SIMO). Dans ce cas, les coefficients matriciels du canal $\mathbf{H}(k)$, $0 \leq k \leq L$ de dimension $p \times q$ et le vecteur source $\mathbf{s}(t)$ de dimension q , défini par les équations (1.2) et (1.3) respectivement, seront remplacés par des coefficients vectoriels $\mathbf{h}(k)$, $0 \leq k \leq L$ de dimension p et le scalaire $s(t)$, respectivement.

1.6.1 Décomposition sous-espace

Soit \mathbf{C}_N la matrice d'autocorrélation associée au vecteur de données $\mathbf{x}_N(t)$, défini par l'équation (1.4)

$$\mathbf{C}_N = E(\mathbf{x}_N(t)\mathbf{x}_N^H(t)), \quad (1.8)$$

tenant compte des hypothèses (H2) et (H3), l'expression de \mathbf{C}_N devient

$$\mathbf{C}_N = \mathbf{H}_N \mathbf{H}_N^H + \sigma_b^2 \mathbf{I}_n. \quad (1.9)$$

D'après l'équation (1.9), il est facile d'établir que les vecteurs propres de $\mathbf{H}_N \mathbf{H}_N^H$ (souvent appelée partie signal, ou bien, partie utile de \mathbf{C}_N) sont identiques à ceux de \mathbf{C}_N . En effet, si $\mathbf{H}_N \mathbf{H}_N^H \mathbf{u} = \alpha \mathbf{u}$ (α est une valeur propre de $\mathbf{H}_N \mathbf{H}_N^H$ et \mathbf{u} un vecteur propre correspondant), alors $\mathbf{C}_N \mathbf{u} = (\alpha + \sigma_b^2) \mathbf{u}$. La réciproque étant évidente.

Supposons, maintenant, qu'on y choisi $N > L$, ce qui implique que $Np = n > d = N + L$ (\mathbf{H}_N possède plus de lignes que de colonnes). D'après le **Lemme 1.1**, \mathbf{H}_N est de rang colonne plein ($\text{rank}(\mathbf{H}_N) = d$). Soient $\lambda_1, \lambda_2, \dots, \lambda_n$ les valeurs propres de \mathbf{C}_N , comme la matrice $\mathbf{H}_N \mathbf{H}_N^H$ est à rang déficient, i.e. $\text{rank}(\mathbf{H}_N \mathbf{H}_N^H) = d$, il s'en suit que

$$\begin{aligned} \lambda_k &> \sigma_b^2, & \text{pour } k = 1, \dots, d, \\ \lambda_k &= \sigma_b^2, & \text{pour } k = d + 1, \dots, n. \end{aligned}$$

Soient $\mathbf{u}_1^s, \mathbf{u}_2^s, \dots, \mathbf{u}_d^s$ les vecteurs propres à norme-unité associés aux valeurs propres $\lambda_1, \lambda_2, \dots, \lambda_d$ et $\mathbf{u}_1^b, \mathbf{u}_2^b, \dots, \mathbf{u}_{d'}^b$, $d' = n - d$ ceux associés à $\lambda_{d+1}, \lambda_{d+2}, \dots, \lambda_n$. Si on définit

$$\begin{aligned} \mathbf{U}_s &= [\mathbf{u}_1^s \cdots \mathbf{u}_d^s], \\ \mathbf{U}_b &= [\mathbf{u}_1^b \cdots \mathbf{u}_{d'}^b], \end{aligned}$$

alors la matrice d'autocorrélation \mathbf{C}_N s'exprime aussi par

$$\begin{aligned} \mathbf{C}_N &= [\mathbf{U}_s \ \mathbf{U}_b] \text{diag}(\lambda_1, \dots, \lambda_n) [\mathbf{U}_s \ \mathbf{U}_b]^H \\ &= \mathbf{U}_s \text{diag}(\lambda_1, \dots, \lambda_d) \mathbf{U}_s^H + \sigma_b^2 \mathbf{U}_b \mathbf{U}_b^H. \end{aligned} \quad (1.10)$$

Les colonnes de \mathbf{U}_s génèrent ce qu'on appelle le *sous-espace signal* (de dimension d), tandis que les colonnes de \mathbf{U}_b génèrent son complément orthogonal, qu'on appelle *sous-espace bruit*. Il est aisé de vérifier que le sous-espace signal est isomorphe à l'image de $\mathbf{H}_N \mathbf{H}_N^H$ ($\text{range}(\mathbf{H}_N \mathbf{H}_N^H) = \text{range}(\mathbf{H}_N)$: sous-espace vectoriel généré par les colonnes de \mathbf{H}_N). De même le sous-espace bruit est isomorphe au noyau de $\mathbf{H}_N \mathbf{H}_N^H$ ($\text{null}(\mathbf{H}_N \mathbf{H}_N^H) = \{\mathbf{z} \in \mathbb{C}^n : \mathbf{H}_N \mathbf{H}_N^H \mathbf{z} = \mathbf{0}\} = \{\mathbf{z} \in \mathbb{C}^n : \mathbf{z}^H \mathbf{H}_N = \mathbf{0}\} = \text{null}_l(\mathbf{H}_N)$), par conséquent, le sous-espace bruit est isomorphe au noyau gauche de \mathbf{H}_N .

1.6.2 Caractérisation de la réponse impulsionnelle du canal par le sous-espace bruit

Établissons maintenant le lien entre le sous-espace bruit et la réponse impulsionnelle du canal $\{\mathbf{h}(k)\}_{0 \leq k \leq L}$. On note d'abord que pour $k = 1, \dots, d'$, on a : $\mathbf{C}_N \mathbf{u}_k^b = \mathbf{H}_N \mathbf{H}_N^H \mathbf{u}_k^b + \sigma_b^2 \mathbf{u}_k^b = \sigma_b^2 \mathbf{u}_k^b$, ce qui équivaut à écrire

$$(\mathbf{u}_k^b)^H \mathbf{H}_N = \mathbf{0}, \quad k = 1, \dots, d'. \quad (1.11)$$

Définissons le vecteur \mathbf{h} par

$$\mathbf{h} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{h}(0) \\ \vdots \\ \mathbf{h}(L) \end{bmatrix} \quad (1.12)$$

et partitionons le vecteur \mathbf{u}_k^b en N sous-vecteur de dimension p

$$\mathbf{u}_k^b = \begin{bmatrix} \mathbf{u}_k^b(0) \\ \mathbf{u}_k^b(1) \\ \vdots \\ \mathbf{u}_k^b(N-1) \end{bmatrix}.$$

Le produit $(\mathbf{u}_k^b)^H \mathbf{H}_N$ obéit à l'identité structurelle suivante (voir [64] pour démonstration)

$$(\mathbf{u}_k^b)^H \mathbf{H}_N = \mathbf{h}^T \mathbf{U}_k^*, \quad k = 1, \dots, d', \quad (1.13)$$

où \mathbf{U}_k est une matrice de dimension $(L+1)p \times d$ ayant une structure de Sylvestre, donnée par

$$\mathbf{U}_k = \begin{bmatrix} \mathbf{u}_k^b(0) & \cdots & \mathbf{u}_k^b(N-1) & \mathbf{0} \\ & \ddots & & \ddots \\ \mathbf{0} & & \mathbf{u}_k^b(0) & \cdots & \mathbf{u}_k^b(N-1) \end{bmatrix}. \quad (1.14)$$

D'après (1.11) et (1.13), $\mathbf{h}^T \mathbf{U}_k^* = \mathbf{0}$, $k = 1, \dots, d'$, ou d'une manière équivalente, $\mathbf{h}^H \mathbf{U}_k = \mathbf{0}$, $k = 1, \dots, d'$, ce qui donne

$$\mathbf{U}^H \mathbf{h} = \mathbf{0} \quad (1.15)$$

avec $\mathbf{U} = [\mathbf{U}_1 \mathbf{U}_2 \cdots \mathbf{U}_{d'}]$.

Montrons maintenant que l'équation (1.15) caractérise de façon bi-univoque (à une constante multiplicative près) la réponse impulsionnelle du canal représentée par le vecteur \mathbf{h} . En effet, soit $\mathbf{h}' = [\mathbf{h}'(0)^T \mathbf{h}'(1)^T \cdots \mathbf{h}'(L)^T]^T$ un vecteur appartenant au noyau droit de \mathbf{U}^H , par conséquent $\mathbf{h}'^T \mathbf{U}_k^* = \mathbf{0}$, $k = 1, \dots, d'$ d'où $\mathbf{u}_k^{bH} \mathbf{H}'_N = \mathbf{0}$, $k = 1, \dots, d'$, avec

$$\mathbf{H}'_N = \begin{bmatrix} \mathbf{h}'(0) & \cdots & \mathbf{h}'(L) & \mathbf{0} \\ & \ddots & & \ddots \\ \mathbf{0} & & \mathbf{h}'(0) & \cdots & \mathbf{h}'(L) \end{bmatrix},$$

on obtient ainsi

$$\mathbf{U}_b^H \mathbf{H}'_N = \mathbf{0}.$$

Comme \mathbf{U}_b génère le complément orthogonal de $\text{range}(\mathbf{H}_N)$, alors $\text{range}(\mathbf{H}'_N) \subset \text{range}(\mathbf{H}_N)$. D'après le *Théorème 2* présenté dans [64], l'image de \mathbf{H}'_N est incluse dans l'image \mathbf{H}_N si et seulement si $\mathbf{h}' = r\mathbf{h}$, r est une constante scalaire non-nulle donnée. En d'autre terme, le noyau droit de \mathbf{U}^H est de dimension un. Noter que la constante non-nulle r est une indétermination inhérente aux systèmes d'identification aveugle SIMO, utilisant des statistiques du second ordre [1].

1.6.3 Algorithme d'identification. Considérations pratiques

En pratique, la matrice d'autocorrélation de l'équation (1.8) est remplacée par sa version échantillonnée, obtenue à partir de la séquence $\{\mathbf{x}_N(t)\}_{0 \leq t \leq K-1}$ (K étant le nombre d'échantillons), en utilisant un estimateur standard

$$\mathbf{C}_N = \frac{1}{K} \sum_{t=0}^{K-1} \mathbf{x}_N(t) \mathbf{x}_N^H(t). \quad (1.16)$$

En théorie (en utilisant des statistiques exactes), le noyau droit de la matrice \mathbf{U}^H dans l'équation (1.15) est de dimension un. En pratique, les erreurs d'estimation intervenant lors du calcul de \mathbf{C}_N et de ses vecteurs propres, font que \mathbf{U}^H est une matrice à rang colonne plein. On approxime alors le noyau droit de \mathbf{U}^H (à une constante multiplicative près) en minimisant la forme quadratique $\mathbf{z}^H \mathbf{Q} \mathbf{z}$, avec $\mathbf{Q} \stackrel{\text{def}}{=} \mathbf{U} \mathbf{U}^H$, sous la contrainte norme-unité

$$\mathbf{h} = \arg \min_{\|\mathbf{z}\|=1} (\mathbf{z}^H \mathbf{Q} \mathbf{z}).$$

L'algorithme sous-espace d'identification du canal (SSA : *Sub-Space Algorithm*) est alors résumé comme suit :

1. Connaissant le degré du canal L et le nombre de capteurs p . Choisir $N > L$ et empiler N observations successives

$$\mathbf{x}_N(t) = \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}(t-1) \\ \vdots \\ \mathbf{x}(t-N+1) \end{bmatrix}.$$

2. Estimer

$$\mathbf{C}_N = \frac{1}{K} \sum_{k=N-1}^{N+K-2} \mathbf{x}_N(k) \mathbf{x}_N^H(k).$$

3. Calculer $d' = n - d = N(p-1) - L$ et déterminer les vecteurs propres de \mathbf{C}_N , \mathbf{u}_k^b , $k = 1, \dots, d'$, associés à ses d' plus petites valeurs propres.
4. Pour $k = 1, \dots, d'$, construire les matrices \mathbf{U}_k définies en (1.14) à partir des vecteurs \mathbf{u}_k^b et calculer

$$\mathbf{Q} = \sum_{k=1}^{d'} \mathbf{U}_k \mathbf{U}_k^H. \quad (1.17)$$

5. Trouver \mathbf{h}' , vecteur propre de \mathbf{Q} , associé à sa plus petite valeur propre.
6. Partitioner \mathbf{h}' en $(L+1)$ sous-vecteurs de dimension p et identifier la réponse impulsionnelle du canal

$$\mathbf{h}(k) = \frac{1}{r} \mathbf{h}'(k), \quad k = 0, \dots, L, \quad (1.18)$$

où r est une constante scalaire non-nulle donnée.

TAB. 1.1 – *Algorithme SSA.*

1.6.4 Égalisation

Une fois la réponse impulsionnelle du canal identifiée, on peut récupérer le signal source en utilisant, ici, un filtre linéaire \mathbf{v}_τ du type erreur quadratique moyenne minimale (*Minimum Mean Square Error*, MMSE) ou bien du type forçage à zéro (*Zero Forcing*, ZF), de sorte que $\hat{s}(t - \tau) = \mathbf{v}_\tau^H \mathbf{x}_N(t)$ soit une bonne approximation d'une réplique de retard τ du signal source.

Égaliseur MMSE

Considérons un égaliseur MMSE à retard $\tau \in \{0, \dots, m - 1\}$. Tenant compte du modèle de données (1.4), on peut facilement montrer que le filtre égaliseur correspondant à la solution recherchée est donnée par

$$\begin{aligned} \mathbf{v}_\tau &= \arg \min_{\mathbf{v}} E(|s(t - \tau) - \mathbf{v}^H \mathbf{x}_N(t)|^2) \\ &= \mathbf{C}_N^{-1} \mathbf{g}_\tau, \end{aligned} \quad (1.19)$$

où \mathbf{g}_τ est un vecteur de dimension n donné par

$$\begin{aligned} \mathbf{g}_\tau &\stackrel{\text{def}}{=} E(\mathbf{x}_N(t) s^*(t - \tau)) \\ &= \mathbf{H}_N(:, \tau + 1). \end{aligned} \quad (1.20)$$

Noter que $\mathbf{H}_N(:, \tau + 1)$ désigne la $(\tau + 1)$ -ième colonne de la matrice \mathbf{H}_N . En utilisant l'expression (1.10) de \mathbf{C}_N et sachant que les colonnes de \mathbf{U}_b et \mathbf{H}_N sont orthogonales, le vecteur \mathbf{v}_τ est alors donné par

$$\mathbf{v}_\tau = \mathbf{U}_s \text{diag}(\lambda_1^{-1}, \dots, \lambda_d^{-1}) \mathbf{U}_s^H \mathbf{H}_N(:, \tau + 1). \quad (1.21)$$

Égaliseur ZF

Comme la matrice de filtrage \mathbf{H}_N est à rang colonne plein, on peut alors estimer le vecteur source \mathbf{s}_m par une simple inversion à gauche de \mathbf{H}_N dans (1.4)

$$\begin{aligned} \hat{\mathbf{s}}_m(t) &= \mathbf{H}_N^\# \mathbf{x}_N(t) \\ &= \mathbf{s}_m(t) + \mathbf{H}_N^\# \mathbf{b}_N(t), \end{aligned}$$

où

$$\begin{aligned} \mathbf{H}_N^\# &= (\mathbf{H}_N^H \mathbf{H}_N)^{-1} \mathbf{H}_N^H \\ &= \mathbf{H}_N^H \mathbf{U}_s \text{diag}((\lambda_1 - \sigma_b^2)^{-1}, \dots, (\lambda_d - \sigma_b^2)^{-1}) \mathbf{U}_s^H. \end{aligned} \quad (1.22)$$

Chaque ligne $i \in \{1, \dots, m\}$ de $\mathbf{H}_N^\#$ constitue un filtre égaliseur dit à forçage à zéro (car il force les ISI à s'annuler) de retard $\tau = i - 1$ et qui a pour expression

$$\mathbf{v}_\tau = \mathbf{U}_s \text{diag}((\lambda_1 - \sigma_b^2)^{-1}, \dots, (\lambda_d - \sigma_b^2)^{-1}) \mathbf{U}_s^H \mathbf{H}_N(:, \tau + 1). \quad (1.23)$$

Noter que le terme $\mathbf{U}_s \text{diag}((\lambda_1 - \sigma_b^2)^{-1}, \dots, (\lambda_d - \sigma_b^2)^{-1}) \mathbf{U}_s^H$ n'est autre que la matrice pseudo-inverse de la matrice d'autocorrélation débruité $\mathbf{H}_N \mathbf{H}_N^H$. En effet, tenant compte de (1.9) et de (1.10) on aura alors

$$\begin{aligned} \tilde{\mathbf{C}}_N &\stackrel{\text{def}}{=} \mathbf{C}_N - \sigma_b^2 \mathbf{I}_n \\ &= \mathbf{H}_N \mathbf{H}_N^H \\ &= \mathbf{U}_s \text{diag}((\lambda_1 - \sigma_b^2), \dots, (\lambda_d - \sigma_b^2)) \mathbf{U}_s^H, \end{aligned} \quad (1.24)$$

or, d'après (1.20) et (1.23), le filtre égaliseur \mathbf{v}_τ aura pour expression

$$\mathbf{v}_\tau = \tilde{\mathbf{C}}_N^\# \mathbf{g}_\tau. \quad (1.25)$$

Remarques

- Il est à noter que dans le cas d'un canal non-bruité ($\sigma_b^2 = 0$), les expressions des égaliseurs MMSE et ZF sont identiques.
- Afin d'éviter une inversion matricielle lors d'une implémentation, il est préférable d'utiliser les expressions (1.21) et (1.23). Les vecteurs propres et les valeurs propres sont déjà obtenus lors de l'étape identification. La variance du bruit peut être estimée comme suit

$$\sigma_b^2 = \frac{1}{n-d} \sum_{k=d+1}^n \lambda_k. \quad (1.26)$$

1.6.5 Simulations et discussion

On donne ici quelques exemples de simulations pour illustrer les performances de l'algorithme présenté précédemment. De même, on fera quelques commentaires sur des éventuelles faiblesses de cet algorithme en particulier, et ceux des systèmes d'identification / égalisation aveugle en générale.

Évaluation des performances

On considère un canal SIMO ($q = 1, p = 3$ et $L = 4$), dont Les coefficients sont générés aléatoirement selon une distribution gaussienne complexe. Le signal d'entrée est une séquence iid, appartenant à un alphabet fini de phase et d'amplitude à quatre états QAM4 ($s(t) = Ae^{j\theta}$, $A \in \{-1, 1\}$ et $\theta \in \{\frac{\pi}{4}, \frac{3\pi}{4}\}$)³. Comme indice de performance, on estime l'erreur quadratique moyenne (*Mean Square Error*, MSE) par

$$\text{MSE} = \frac{1}{K} \sum_{t=\tau}^{K+\tau+1} |s(t-\tau) - \hat{\mathbf{v}}_\tau^H \mathbf{x}_N(t)|^2,$$

moyenné sur 100 réalisations indépendantes. $\hat{\mathbf{v}}_\tau$ étant une estimation échantillonnée (obtenue à partir de K échantillons) du filtre égaliseur \mathbf{v}_τ (obtenu à partir de statistiques exactes). Le MSE est comparé au MSE optimale

$$\text{MSE}_{opt} = E(|s(t-\tau) - \mathbf{v}_\tau^H \mathbf{x}_N(t)|^2) = \begin{cases} 1 - \mathbf{g}_\tau^H \mathbf{C}_N^{-1} \mathbf{g}_\tau, & \text{égaliseur MMSE,} \\ \sigma_b^2 \mathbf{g}_\tau^H (\tilde{\mathbf{C}}_N^\#)^2 \mathbf{g}_\tau, & \text{égaliseur ZF.} \end{cases}$$

³Ici, le symbole j désigne un nombre complexe tel que $j^2 = -1$.

De même, pour permettre l'évaluation de l'algorithme, l'indétermination (1.18)⁴ est levée en estimant la constante r par

$$r = \arg \min_{\alpha} \|\mathbf{h}' - \alpha \mathbf{h}\|^2 = \frac{\mathbf{h}^H \mathbf{h}'}{\|\mathbf{h}\|^2}.$$

Le rapport signal à bruit (*Signal to Noise Ratio*, SNR) est donné par $\text{SNR} = -20 \log(\sigma_b)$.

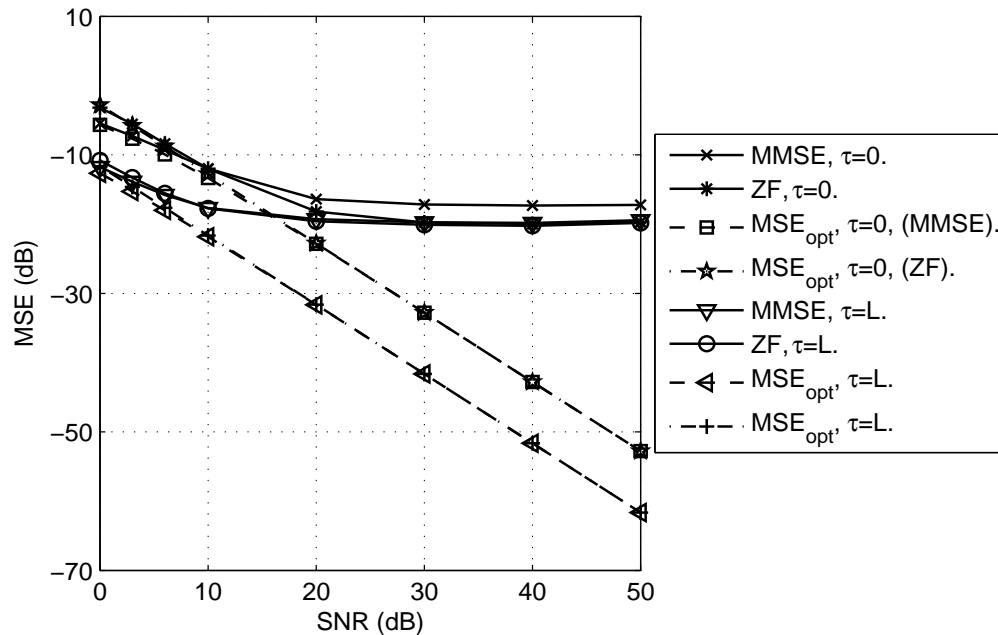


FIG. 1.4 – Performances de l'algorithme sous-espace.

Dans la Fig. 1.4, le MSE est représenté en fonction du SNR, pour un nombre d'échantillons $K = 500$. On remarque qu'il y a une baisse en performance (par rapport au MSE_{opt}), chose qui est dû à la procédure d'estimation du canal.

Robustesse aux erreurs d'estimation de l'ordre du canal

La Fig. 1.5 illustre l'inconvénient majeur de l'approche sous-espace, à savoir sa sensibilité aux erreurs d'estimation de l'ordre du canal L (et par conséquent, de la dimension du sous-espace signal $d = N + L$ (resp. sous-espace bruit $d' = n - d = N(p - 1) - L$), qui n'est pas toujours bien défini⁵. Ce qui perturbe et dégrade énormément les performances des méthodes type sous-espace [58]. En pratique, au lieu d'estimer l'ordre du canal L , il est préférable d'estimer directement la dimension du sous-espace signal via des critères de la théorie de l'information, en l'occurrence le critère AIC [7, 8]

⁴Rappelons, que cette indétermination est un problème inhérent aux systèmes d'identification aveugle SIMO utilisant des statistiques du second ordre [1]. Voir aussi discussion ci-dessous.

⁵En général, on estime la borne supérieure de l'ordre du canal. C'est pourquoi, on étudie en général, la robustesse aux erreurs de surestimation de l'ordre du canal L .

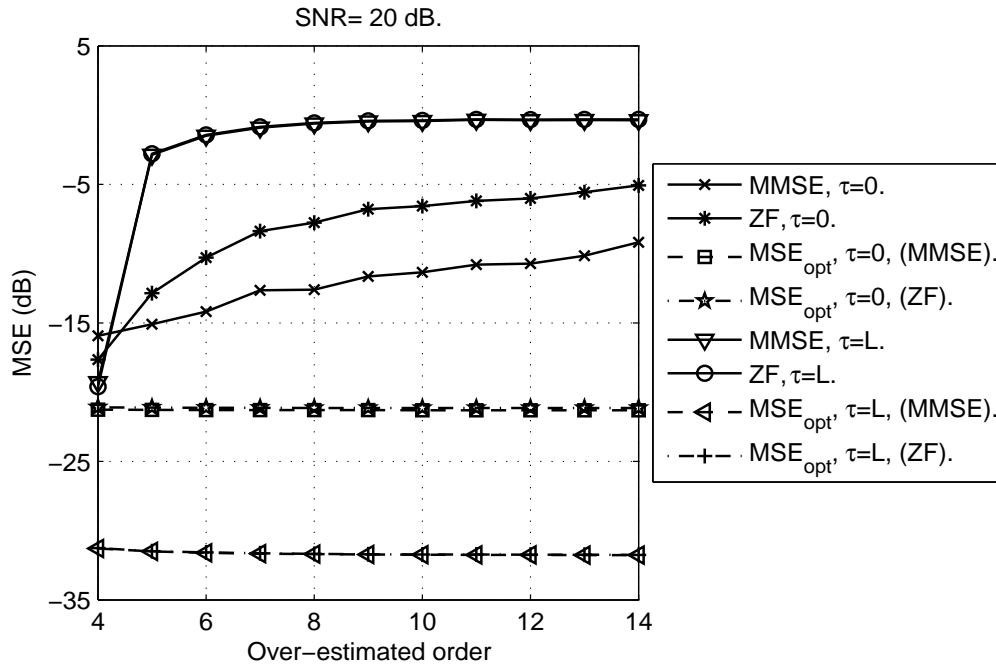


FIG. 1.5 – Dégradation des performances de l’algorithme sous-espace, quand l’ordre du canal est sur-estimé (l’ordre exacte $L=4$).

(Akaike’s Information theoretic Criteria) ou le critère MDL [67, 69] (Minimum Description Length)

$$\hat{d} = \arg \min_{0 \leq k \leq n-1} Cr(k),$$

$$Cr(k) = \begin{cases} -2 \log \left(\frac{\prod_{i=k+1}^n \lambda_i^{1/(n-k)}}{\frac{1}{n-k} \sum_{i=k+1}^n \lambda_i} \right)^{(n-k)K} + 2k(2n-k), & \text{critère AIC,} \\ -\log \left(\frac{\prod_{i=k+1}^n \lambda_i^{1/(n-k)}}{\frac{1}{n-k} \sum_{i=k+1}^n \lambda_i} \right)^{(n-k)K} + \frac{1}{2}k(2n-k) \log(K), & \text{critère MDL,} \end{cases}$$

où $\lambda_1, \lambda_2, \dots, \lambda_n$ sont des estimés des valeurs propres de la matrice d’autocorrélation donnée par (1.16), K étant le nombre d’échantillons.

Complexité de calcul

L’algorithme présenté précédemment est destiné, bien entendu, à un schéma de traitement en bloc des données qui nécessite une décomposition en valeurs singulières (*Singular Value Decomposition*, SVD) ou une décomposition en valeur propre (*Eigen Value Decomposition*, EVD). Le coût de calcul (nombre d’opérations) d’une SVD (ou EVD), relativement élevé (de l’ordre de $O(n^3)$ opérations, n étant la dimension du vecteur de données), ce qui constitue un sérieux obstacle pour un traitement en temps réel, comme dans le cas de canaux variables dans le temps, où une poursuite du sous-espace signal et/ou du sous-espace bruit s’avère nécessaire. On y trouve en littérature un nombre assez important de contributions qui y traite de ce problème [5, 6, 9–11, 21, 23, 29, 53, 73, 87, 88]. En général on y développe des approches adaptatives de poursuite de sous-espace, ayant un coût de calcul de l’ordre de $O(nd^2)$

ou $O(nd)$ (d étant la dimension du sous-espace). Néanmoins, cela ne suffit pas à réduire le coût de calcul total de la procédure d'identification / égalisation aveugle. En effet, l'étape d'optimisation de l'identification / égalisation canal s'avère aussi coûteuse, en terme de complexité de calcul, que celle de l'estimation du sous-espace. C'est ainsi qu'on y trouve dans la littérature pas mal de méthodes (autre que l'approche sous-espace) [38, 39, 54–57, 86], qui y vont dans ce sens, à savoir proposer des algorithmes adaptatifs d'identification / égalisation aveugle du second ordre, dont le coût de calcul globale est moyen ou faible (de l'ordre de $O(n^2)$ ou $O(nd)$). Une grande partie de ces algorithmes utilise des schémas adaptatifs classiques du type moindres carrés récursifs (*Recursive Least-Squares*, RLS) ou moindres carrés moyens (*Least Mean Squares*, LMS). De telles approches souffrent d'un inconvénient de taille, à savoir une convergence très lente.

Indéterminations

À l'inverse d'un schéma classique d'identification / égalisation, dans un schéma aveugle, il est impossible d'identifier entièrement un système linéaire $H(z)$, dans le cas où son entrée $s(t)$ est inconnue et vis vers ça. Cette observation vient du fait qu'on peut trouver une infinité de modèles $(H'(z), s'(t))$ dont la sortie coïncide avec celle de $(H(z), s(t))$. En effet, soient $H'(z) = H(z)\mathbf{R}$ et $s'(t) = \mathbf{R}^{-1}s(t)$, où \mathbf{R} est une matrice non-singulière (on peut considérer \mathbf{R} comme étant le produit d'une matrice diagonale non-singulière et une matrice de permutation). Il est clair, dans ce cas, que $[H(z)]s(t) = [H'(z)]s'(t)$ et par conséquent $H(z)$ (resp. $s(t)$) est identifiable à une matrice constante non-singulière près. Ces indéterminations présentent un handicap de taille quand à la mise en œuvre pratique d'algorithmes d'identification / égalisation totalement aveugle. En communications numériques, ces indéterminations peuvent être plus ou moins compensés par l'utilisations de modulations différentielles [66], ou par l'introduction d'une certaine forme d'information additionnelle. Ce qui équivaut à introduire une séquence d'apprentissage très courte. Ce genre de méthodes qui exploitent des critères d'identification / égalisation aveugle et des informations issues de symboles (*a priori* connus) forment ce qui est appelé méthodes d'identification / égalisation semi-aveugle [25].

1.7 Conclusion

Dans ce chapitre, nous avons d'abord donné un bref historique sur l'évolution des recherches dans le domaine de l'identification / égalisation aveugle, suivi d'une description générale du problème d'identification / égalisation aveugle des canaux de transmission numérique. Nous avons présenté le modèle du canal de transmission. Il s'agit d'un canal MIMO à réponse impulsionnelle finie causal et stable. De même, nous avons présenté une description du modèle de données que nous avons adopté. La construction de ce modèle fait apparaître une matrice de filtrage qui dépend des coefficients du canal et qui possède une structure de Sylvestre dont les propriétés sont exploitées pour développer des schémas d'identification / égalisation aveugle. Et afin que le lecteur se familiarise avec les techniques d'identification / égalisation aveugle, nous avons présenté un exemple illustratif, désormais classique, d'identification / égalisation aveugle à savoir la méthode sous-espace [64]. De même, cette méthode illustre bien les deux problèmes

fondamentaux que nous traitons dans ce travail. Il s'agit de la robustesse des méthodes d'égalisation aveugle vis à vis de la surestimation de l'ordre du canal de transmission et de leur complexité numérique. Ces deux aspects feront l'objet de la suite de ce travail.

Chapitre 2

Quelques méthodes d'identification / égalisation aveugles robustes aux erreurs de surestimation de l'ordre du canal

2.1 Introduction

Comme il a été montré au chapitre 1, outre leur coût de calcul élevé, l'inconvénient majeur des méthodes type sous-espace (comme c'est d'ailleurs le cas pour la plus part des méthodes d'identification / égalisation aveugle), est leur sensibilité aux erreurs de surestimation de l'ordre du canal. Par ailleurs, de considérables efforts de recherche ont été déployés afin d'y remédier à cette carence. On y trouve des approches qui tentent d'améliorer les techniques d'estimation de l'ordre du canal [35, 45, 59], alors que d'autres se fixent pour objectif le développement de techniques d'identification / égalisation aveugle robustes aux erreurs de surestimation de l'ordre du canal [2, 33, 34, 36–39, 43, 44, 54–57, 70–72, 75, 76, 80, 84–86], parmi lesquelles nous allons présenter quelques unes, dont la mise en œuvre est relativement simple, tout en ayant un coût de calcul moyen ou faible.

2.2 Méthode de la prédiction linéaire

La méthode de la prédiction linéaire (*Linear Prediction Algorithm*, LPA) a été présentée pour la première fois, dans un contexte de diversité spatio-temporelle (cas SIMO), par D. Slock [72]. Elle a été ensuite raffinée et étendue au cas MIMO (voir [2, 44] par exemple). Depuis, un grand nombre de variantes a été proposées [37, 39, 55, 56, 70]. Il est montré dans [2] que la méthode LPA est relativement robuste aux erreurs de surestimation de l'ordre du canal. Nous présentons ici l'algorithme proposé par X. Li et H. Fan [55], dans le cas d'un canal SIMO. Le choix de cet algorithme est motivé par le fait qu'il est simple à mettre en œuvre. Il permet une égalisation directe à retard contrôlé du canal. De même, il présente une

version adaptative à faible coût de calcul. Il s'agit, dans un premier temps, d'estimer un égaliseur ZF à retard nul \mathbf{v}_0^{zf} via une prédiction linéaire, pour ainsi en déduire des égaliseurs MMSE et ZF à retards non-nuls à partir de \mathbf{v}_0^{zf} .

2.2.1 Définitions et notations

Soient $a \in \mathbb{N}^*$ et \mathbf{J}_a la matrice définie par

$$\mathbf{J}_a \stackrel{\text{def}}{=} \begin{cases} \begin{bmatrix} \mathbf{0}_{1,a-1} & 0 \\ \mathbf{I}_{a-1} & \mathbf{0}_{a-1,1} \end{bmatrix}, & \text{si } a \geq 2, \\ 0, & \text{si } a = 1, \end{cases} \quad (2.1)$$

où \mathbf{I}_β est la matrice identité de dimension $\beta \times \beta$.

Si, par convention on pose

$$\mathbf{J}_a^0 = \mathbf{I}_a \quad \text{et} \quad \mathbf{J}_a^{-1} = \mathbf{J}_a^T, \quad a \in \mathbb{N}^*, \quad (2.2)$$

alors, les matrices d'autocorrélation du vecteur $\mathbf{x}_N(t)$ sont données par

$$\begin{aligned} \mathbf{C}_{N,k} &\stackrel{\text{def}}{=} E(\mathbf{x}_N(t+k)\mathbf{x}_N^H(t)), \quad k \in \mathbb{Z} \\ &= \mathbf{H}_N \mathbf{J}_m^k \mathbf{H}_N^H + \sigma_b^2 \mathbf{J}_N^k \otimes \mathbf{I}_p. \end{aligned} \quad (2.3)$$

La partie signal de $\mathbf{C}_{N,k}$ est alors définie par

$$\begin{aligned} \tilde{\mathbf{C}}_{N,k} &\stackrel{\text{def}}{=} E(\tilde{\mathbf{x}}_N(t+k)\tilde{\mathbf{x}}_N^H(t)), \quad k \in \mathbb{Z} \\ &= \mathbf{H}_N \mathbf{J}_m^k \mathbf{H}_N^H \\ &= \mathbf{C}_{N,k} - \sigma_b^2 \mathbf{J}_N^k \otimes \mathbf{I}_p, \end{aligned} \quad (2.4)$$

avec

$$\tilde{\mathbf{x}}_N(t) = \mathbf{x}_N(t) - \mathbf{b}_N(t) = \mathbf{H}_N \mathbf{s}_m(t). \quad (2.5)$$

Remarquer que souvent $\mathbf{C}_{N,0}$ et $\tilde{\mathbf{C}}_{N,0}$ sont notées simplement par \mathbf{C}_N et $\tilde{\mathbf{C}}_N$, respectivement.

2.2.2 Égaliseurs ZF et MMSE

Rappelons que l'expression d'un égaliseur ZF d'un canal SIMO à retard τ (voir paragraphe 1.6) est donnée par

$$\mathbf{v}_\tau^{zf} = \tilde{\mathbf{C}}_N^\# \mathbf{H}_N(:, \tau + 1). \quad (2.6)$$

Pour $\tau = 0$, l'égaliseur ZF à retard zéro est alors donné par

$$\mathbf{v}_0^{zf} = \tilde{\mathbf{C}}_N^\# [\mathbf{h}(0)^T \quad \mathbf{0}_{1 \times (n-p)}]^T, \quad (2.7)$$

comme $\mathbf{H}_N^H \mathbf{v}_0^{zf} = [1 \quad \mathbf{0}_{1 \times (d-q)}]^T$, il s'en suit que

$$\begin{aligned} \tilde{\mathbf{C}}_{N,\tau} \mathbf{v}_0^{zf} &= \mathbf{H}_N \mathbf{J}_m^\tau \mathbf{H}_N^H \mathbf{v}_0^{zf} \\ &= \mathbf{H}_N(:, \tau + 1), \end{aligned}$$

d'où, \mathbf{v}_τ^{zf} s'exprime aussi par

$$\mathbf{v}_\tau^{zf} = \tilde{\mathbf{C}}_N^\# \tilde{\mathbf{C}}_{N,\tau} \mathbf{v}_0^{zf}, \quad (2.8)$$

de même, le filtre égaliseur MMSE du canal SIMO de l'équation (1.19) s'exprime aussi par

$$\begin{aligned} \mathbf{v}_\tau^{mmse} &= \mathbf{C}_N^{-1} \mathbf{H}_N(:, \tau + 1) \\ &= \mathbf{C}_N^{-1} \tilde{\mathbf{C}}_{N,\tau} \mathbf{v}_0^{zf}. \end{aligned}$$

2.2.3 Égalisation aveugle

Pour estimer \mathbf{v}_0^{zf} (et par conséquent \mathbf{v}_τ^{zf} et \mathbf{v}_τ^{mmse}), on aura besoin d'estimer $\tilde{\mathbf{C}}_N$ et $\mathbf{h}(0)$. $\tilde{\mathbf{C}}_N$ sera obtenu par débruitage de \mathbf{C}_N . Pour estimer $\mathbf{h}(0)$, nous adopterons une approche de prédiction linéaire. Considérons le problème de la prédiction linéaire suivant

$$\begin{cases} \min_{\mathbf{P}} J(\mathbf{P}), \\ J(\mathbf{P}) = E(\|\tilde{\boldsymbol{\varepsilon}}(t)\|^2), \end{cases} \quad (2.9)$$

$\tilde{\boldsymbol{\varepsilon}}(t)$ étant l'erreur de la prédiction, donnée par

$$\begin{aligned} \tilde{\boldsymbol{\varepsilon}}(t) &= \tilde{\mathbf{x}}(t) - \sum_{k=1}^{N-1} \mathbf{P}_k \tilde{\mathbf{x}}(t-k) \\ &= [\mathbf{I}_p - \mathbf{P}] \tilde{\mathbf{x}}_N(t) \end{aligned} \quad (2.10)$$

où $\tilde{\mathbf{x}}(t)$ est le signal reçu débruité, i.e. $\tilde{\mathbf{x}}(t) = \mathbf{x}(t) - \mathbf{b}(t) = \sum_{k=0}^L \mathbf{h}(k)s(t-k)$, et $\tilde{\mathbf{x}}_N(t)$ est le vecteur de données, donné par l'équation (2.5). Les matrices \mathbf{P}_k , $1 \leq k \leq N-1$, de dimension $p \times p$, sont les coefficients du filtre de la prédiction $\mathbf{P} = [\mathbf{P}_1 \cdots \mathbf{P}_{N-1}]$.

Théorème 2.1 *Supposons que les conditions de l'hypothèse (H1) sont vérifiées et que $N > L$. Si \mathbf{P}_{opt} est une solution optimale du problème (2.9), alors*

$$[\mathbf{I}_p - \mathbf{P}_{opt}] \mathbf{H}_N = [\mathbf{h}(0) \mathbf{0}_{p \times (d-1)}], \quad (2.11)$$

$$\tilde{\boldsymbol{\varepsilon}}_{min}(t) = \mathbf{h}(0)s(t). \quad (2.12)$$

Démonstration : voir Annexe A.

Pour calculer le filtre de la prédiction linéaire \mathbf{P}_{opt} , on exploite le fait que $s(t)$ est orthogonal à $\tilde{\mathbf{x}}(t-k)$, $k \geq 1$, i.e. : $E(s(t)\tilde{\mathbf{x}}^H(t-k)) = \mathbf{0}$, $k \geq 1$. En effet, d'après (2.12)

$$E(\tilde{\boldsymbol{\varepsilon}}_{min}(t)\tilde{\mathbf{x}}_{N-1}^H(t-1)) = \mathbf{h}(0)E(s(t)\tilde{\mathbf{x}}_{N-1}^H(t)) = \mathbf{0}. \quad (2.13)$$

D'autre part, d'après (2.10) et (2.13)

$$\begin{aligned} E(\tilde{\boldsymbol{\varepsilon}}_{min}(t)\tilde{\mathbf{x}}_{N-1}^H(t-1)) &= E\left([\mathbf{I}_p - \mathbf{P}_{opt}] \begin{bmatrix} \tilde{\mathbf{x}}(t) \\ \tilde{\mathbf{x}}_{N-1}(t-1) \end{bmatrix} \tilde{\mathbf{x}}_{N-1}^H(t-1)\right) \\ &= \tilde{\mathbf{C}}' - \mathbf{P}_{opt} \tilde{\mathbf{C}}_{N-1} = \mathbf{0}, \end{aligned}$$

avec

$$\tilde{\mathbf{C}}' \stackrel{\text{def}}{=} E(\tilde{\mathbf{x}}(t)\tilde{\mathbf{x}}_{N-1}^H(t-1)) = \tilde{\mathbf{C}}_N(1:p, p+1:n).$$

D'où l'expression

$$\mathbf{P}_{opt} = \tilde{\mathbf{C}}'\tilde{\mathbf{C}}_{N-1}^\#.$$

Pour estimer $\mathbf{h}(0)$, multiplions d'abord l'équation (2.11) par \mathbf{H}_N^H , ce qui donne (vu la structure Sylvestre de \mathbf{H}_N)

$$[\mathbf{I}_p \quad -\mathbf{P}_{N-1}^{opt}] \tilde{\mathbf{C}}_N = [\mathbf{h}(0)\mathbf{h}^H(0) \quad \mathbf{0}_{p \times (n-p)}]. \quad (2.14)$$

Si on tronque les $n-p$ dernières colonnes du système (2.14), on obtient

$$\mathbf{D} \stackrel{\text{def}}{=} [\mathbf{I}_p \quad -\mathbf{P}_{opt}] \tilde{\mathbf{C}}'' = \mathbf{h}(0)\mathbf{h}^H(0), \quad (2.15)$$

avec

$$\tilde{\mathbf{C}}'' \stackrel{\text{def}}{=} E(\tilde{\mathbf{x}}_N(t)\tilde{\mathbf{x}}^H(t)) = \tilde{\mathbf{C}}_N(:, 1:p). \quad (2.16)$$

Alors, $\mathbf{h}(0)$ correspond (à une constante multiplicative scalaire non nulle près) au vecteur propre dominant de la matrice \mathbf{D} . Évidemment, \mathbf{v}_0^{zf} peut être calculé en utilisant (2.7), ou encore plus simplement, en tenant compte de (2.14), \mathbf{v}_0^{zf} s'exprime aussi par

$$\mathbf{v}_0^{zf} = \frac{1}{\|\mathbf{h}(0)\|^2} [\mathbf{I}_p \quad -\mathbf{P}_{opt}^H]^H \mathbf{h}(0).$$

En supposant que l'ordre du canal et la variance du bruit σ_b^2 connus ou estimés via un critère du type AIC ou MDL, et la relation (1.26) respectivement, l'algorithme d'égalisation en schéma bloc est résumé dans la TABLE.2.1.

\mathbf{C}_N	$= \frac{1}{K} \sum_{k=N-1}^{K+N-2} \mathbf{x}_N(k)\mathbf{x}_N^H(k),$ (K : nombre d'échantillons)
$\mathbf{C}_{N,\tau}$	$= \frac{1}{K} \sum_{k=\tau+N-1}^{K+\tau+N-2} \mathbf{x}_N(k)\mathbf{x}_N^H(k-\tau)$
$\tilde{\mathbf{C}}_N$	$= \mathbf{C}_N - \sigma_b^2 \mathbf{I}_n$
$\tilde{\mathbf{C}}_{N,\tau}$	$= \mathbf{C}_{N,\tau} - \sigma_b^2 \mathbf{J}_N^\tau \otimes \mathbf{I}_p$
$\tilde{\mathbf{C}}'$	$= \tilde{\mathbf{C}}_N(1:p, p+1:n)$
$\tilde{\mathbf{C}}_{N-1}$	$= \tilde{\mathbf{C}}_N(p+1:n, p+1:n)$
\mathbf{P}_{opt}	$= \tilde{\mathbf{C}}'\tilde{\mathbf{C}}_{N-1}^\#$
\mathbf{D}	$= [\mathbf{I}_p \quad -\mathbf{P}_{opt}] \tilde{\mathbf{C}}_N(:, 1:p)$
\mathbf{h}	$=$ vecteur propre dominant de \mathbf{D}
\mathbf{v}_0^{zf}	$= \frac{1}{\ \mathbf{h}\ ^2} [\mathbf{I}_p \quad -\mathbf{P}_{opt}^H]^H \mathbf{h},$ (à une constante multiplicative scalaire non-nulle près)
\mathbf{v}_τ^{zf}	$= \tilde{\mathbf{C}}_N^\# \tilde{\mathbf{C}}_{N,\tau} \mathbf{v}_0^{zf}$
\mathbf{v}_τ^{mmse}	$= \mathbf{C}_N^{-1} \tilde{\mathbf{C}}_{N,\tau} \mathbf{v}_0^{zf}$

TAB. 2.1 – Algorithme d'égalisation aveugle de type prédiction linéaire.

2.2.4 Implémentation adaptative

Comme il apparaît au paragraphe précédent (voir TABLE 2.1), l'implémentation en bloc de l'algorithme d'égalisation nécessite un calcul explicite de la pseudo-inverse de la matrice d'autocorrélation débruitée $\tilde{\mathbf{C}}_N$ et/ou de l'inverse de la matrice d'autocorrélation \mathbf{C}_N , ce qui est excessivement coûteux. Dans ce qui suit, nous allons présenter une implémentation adaptative de l'algorithme, basée sur la technique du gradient descendant et qui permettra une estimation récursive du filtre égaliseur, ayant une complexité de calcul relativement faible (de l'ordre de $O(n^2)$ opérations par itération).

Du moment que le gradient de $J(\mathbf{P})$ est donné par

$$\begin{aligned}\nabla J(\mathbf{P}) &= -[\mathbf{I}_p \quad -\mathbf{P}]\tilde{\mathbf{C}}''', \\ \tilde{\mathbf{C}}''' &\stackrel{\text{def}}{=} E(\tilde{\mathbf{x}}_N(t)\tilde{\mathbf{x}}_{N-1}^H(t-1)) = \tilde{\mathbf{C}}_N(:, p+1:n),\end{aligned}$$

alors, $\mathbf{P}_{opt}(t)$ est mis à jour comme suit

$$\begin{aligned}\mathbf{P}_{opt}(t) &= \mathbf{P}_{opt}(t-1) - \mu \nabla J(\mathbf{P}_{opt}(t-1)) \\ &= \mathbf{P}_{opt}(t-1) + \mu [\mathbf{I}_p \quad -\mathbf{P}_{opt}(t-1)]\tilde{\mathbf{C}}''',\end{aligned}$$

où $\mu > 0$ est le pas d'adaptation, Si la matrice $\tilde{\mathbf{C}}'''$ est remplacée par sa valeur instantanée avec des signaux reçus bruités, i.e., $\tilde{\mathbf{C}}''' \approx \mathbf{x}_N(t)\mathbf{x}_{N-1}^H(t-1)$, on aura finalement alors

$$\begin{aligned}\mathbf{P}_{opt}(t) &= \mathbf{P}_{opt}(t-1) + \mu \boldsymbol{\varepsilon}(t)\mathbf{x}_{N-1}^H(t-1), \\ \boldsymbol{\varepsilon}(t) &= [\mathbf{I}_p \quad -\mathbf{P}_{opt}(t-1)]\mathbf{x}_N(t).\end{aligned}$$

La matrice $\tilde{\mathbf{C}}''(t)$ donnée par (2.16) est estimée récursivement (avec des signaux reçus bruités) par

$$\tilde{\mathbf{C}}''(t) = \beta \tilde{\mathbf{C}}''(t-1) + \mathbf{x}_N(t)\mathbf{x}_N^H(t),$$

β étant le facteur d'oubli. La mise à jour de $\mathbf{D}(t)$ est alors donnée par

$$\mathbf{D}(t) = \beta \mathbf{D}(t-1) + \boldsymbol{\varepsilon}(t)\mathbf{x}_N^H(t).$$

Montrons maintenant que l'égaliseur ZF à retard τ non-nul, donné par (2.8), peut être estimé par une prédiction linéaire. Considérons le problème de la prédiction linéaire suivant

$$\begin{cases} \min_{\mathbf{Q}} J'(\mathbf{Q}), \\ J'(\mathbf{Q}) = E(\|\tilde{\boldsymbol{\varepsilon}}'(t)\|^2), \end{cases} \quad (2.17)$$

$\tilde{\boldsymbol{\varepsilon}}'(t)$ étant l'erreur de la prédiction, donnée par

$$\tilde{\boldsymbol{\varepsilon}}'(t) = \tilde{\mathbf{x}}_N(t-\tau) - \mathbf{Q}^H \tilde{\mathbf{x}}_N(t),$$

\mathbf{Q} est une matrice de projection de dimension $n \times n$. Dans ce cas on a le résultat suivant

Théorème 2.2 Si \mathbf{Q}_{opt} est la solution optimale du problème (2.17), alors l'égaliseur ZF à retard τ de (2.8), est donné en fonction de l'égaliseur ZF à retard zéro par

$$\mathbf{v}_\tau^{zf} = \mathbf{Q}_{opt} \mathbf{v}_0^{zf}$$

Démonstration : voir Annexe A

Comme le gradient de $J'(\mathbf{Q})$ est donné par $\nabla J'(\mathbf{Q}) = -\tilde{\mathbf{C}}_{N,\tau} + \tilde{\mathbf{C}}_N \mathbf{Q}$, alors le filtre \mathbf{Q}_{opt} est mis à jour comme suit

$$\begin{aligned}\mathbf{Q}_{opt}(t) &= \mathbf{Q}_{opt}(t-1) - \mu' \nabla J'(\mathbf{Q}_{opt}(t-1)) \\ &= \mathbf{Q}_{opt}(t-1) + \mu' (\tilde{\mathbf{C}}_{N,\tau} - \tilde{\mathbf{C}}_N \mathbf{Q}_{opt}(t-1)),\end{aligned}$$

où $\mu' > 0$ est le pas d'adaptation. Si les matrices $\tilde{\mathbf{C}}_N$ et $\tilde{\mathbf{C}}_{N,\tau}$ sont remplacées par leurs valeurs instantanées avec des signaux reçus bruités, i.e., $\tilde{\mathbf{C}}_N \approx \mathbf{x}_N(t) \mathbf{x}_N^H(t)$ et $\tilde{\mathbf{C}}_{N,\tau} \approx \mathbf{x}_N(t) \mathbf{x}_N^H(t-\tau)$, on obtient alors

$$\begin{aligned}\mathbf{Q}_{opt}(t) &= \mathbf{Q}_{opt}(t-1) + \mu' \mathbf{x}_N(t) \boldsymbol{\varepsilon}'^H(t), \\ \boldsymbol{\varepsilon}'(t) &= \mathbf{x}_N(t-\tau) - \mathbf{Q}_{opt}^H(t-1) \mathbf{x}_N(t).\end{aligned}$$

La version adaptative de l'algorithme d'égalisation est alors donnée par la TABLE 2.2. Noter que le coût globale de cette implémentation est de l'ordre de $O(n^2)$ par itération.

$\boldsymbol{\varepsilon}(t)$	$= [\mathbf{I}_p - \mathbf{P}_{opt}(t-1)] \mathbf{x}_N(t)$
$\mathbf{P}_{opt}(t)$	$= \mathbf{P}_{opt}(t-1) + \mu' \boldsymbol{\varepsilon}(t) \mathbf{x}_N^{H-1}(t-1)$
$\mathbf{D}(t)$	$= \beta \mathbf{D}(t-1) + \boldsymbol{\varepsilon}(t) \mathbf{x}_N^H(t)$
$\mathbf{h}(t)$	$=$ vecteur propre dominant de $\mathbf{D}(t)$
$\mathbf{v}_0^{zf}(t)$	$= \frac{1}{\ \mathbf{h}(t)\ ^2} [\mathbf{I}_p - \mathbf{P}_{opt}^H(t)]^H \mathbf{h}(t)$, (à une constante multiplicative scalaire non-nulle près)
$\boldsymbol{\varepsilon}'(t)$	$= \mathbf{x}_N(t-\tau) - [\mathbf{Q}_{opt}(t-1)]^H \mathbf{x}_N(t)$
$\mathbf{Q}_{opt}(t)$	$= \mathbf{Q}_{opt}(t-1) + \mu' \mathbf{x}_N(t) \boldsymbol{\varepsilon}'^H(t)$
$\mathbf{v}_\tau^{zf}(t)$	$= \mathbf{Q}_{opt}(t) \mathbf{v}_0^{zf}(t)$

TAB. 2.2 – Algorithme adaptatif d'égalisation aveugle de type prédiction linéaire.

2.3 Méthode des égaliseurs mutuellement référencés

Cette méthode introduite dans [38] pour le cas SIMO, estime directement un ensemble d'égaliseurs ZF dits mutuellement référencés. L'existence de tels égaliseurs relève de celle de l'inverse à gauche (pseudo-inverse) de la matrice de filtrage \mathbf{H}_N , lorsque celle-ci est à rang-colonne plein. En effet, en l'absence du bruit ($\mathbf{b}(t) = \mathbf{0}$) on a

$$\mathbf{H}_N^\# \mathbf{x}_N(t) = \mathbf{s}_m(t).$$

Il est clair que le $i^{\text{ème}}$ vecteur-ligne de la matrice $\mathbf{H}_N^\#$, $1 \leq i \leq m$, est un égaliseur ZF associé au retard $\tau = i + 1$. La méthode des égaliseurs mutuellement référencés (*Mutually Referenced Equalizers*, MRE) exploite cette diversité de retard afin de permettre une estimation de l'ensemble des pseudo-inverses du

canal. L'idée de la méthode MRE est alors la suivante. Considérons les égaliseurs ZF \mathbf{v}_τ et $\mathbf{v}_{\tau+1}$ de retard τ et $\tau + 1$ respectivement, en l'absence du bruit on a

$$\begin{aligned}\mathbf{v}_\tau^H \mathbf{x}_N(t) &= s(t - \tau), \\ \mathbf{v}_{\tau+1}^H \mathbf{x}_N(t) &= s(t - \tau - 1),\end{aligned}$$

il en découle

$$\mathbf{v}_\tau^H \mathbf{x}_N(t) = \mathbf{v}_{\tau+1}^H \mathbf{x}_N(t + 1), \quad 0 \leq \tau \leq m - 2, \quad (2.18)$$

où les sorties des égaliseurs sont dites référencées l'une à l'autre. Il est montré dans [38], que les $m - 1$ relations (2.18) suffisent pour que les filtres $\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_{m-1}$ soient des égaliseurs ZF. Nous avons alors le théorème suivant

Théorème 2.3 Soient $\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_{m-1}$ m vecteurs vérifiant (2.18) pour tout t . Dans ce cas on a

- i) $\mathbf{V}^H \mathbf{H}_N = r \mathbf{I}_m$, où $\mathbf{V} \stackrel{\text{def}}{=} [\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_{m-1}]$ et r est une constante scalaire. Ce qui équivaut à
- ii) $\mathbf{v}_\tau^H \mathbf{x}_N(t) = r s(t - \tau)$, $0 \leq \tau \leq m - 1$.

Démonstration : voir Annexe A

En pratique (en présence du bruit additif : $\mathbf{b}(t) \neq \mathbf{0}$), on estime la matrice \mathbf{V} de façon à ce qu'elle soit solution du problème d'optimisation sous contrainte suivant

$$\begin{cases} \min_{\mathcal{C}(\mathbf{V})=0} J(\mathbf{V}), & J(\mathbf{V}) = E(\|\boldsymbol{\varepsilon}(t)\|^2), \\ \boldsymbol{\varepsilon}(t) = [\mathbf{I}_{m-1} \ \mathbf{0}_{m-1,1}] \mathbf{V}^H \mathbf{x}_N(t) - [\mathbf{0}_{m-1,1} \ \mathbf{I}_{m-1}] \mathbf{V}^H \mathbf{x}_N(t + 1). \end{cases} \quad (2.19)$$

La contrainte $\mathcal{C}(\mathbf{V}) = 0$, peut être quadratique (e.g. $\mathcal{C}(\mathbf{V}) = \text{trace}(\mathbf{V}^H \mathbf{V}) - 1$) ou linéaire (e.g. $\mathcal{C}(\mathbf{V}) = \text{trace}(\mathbf{U}^H \mathbf{V}) - 1$, où $\mathbf{U} \in \mathbb{C}^{n \times m}$ est une matrice arbitraire), son rôle n'est pas seulement d'éviter la solution triviale $\mathbf{V} = \mathbf{0}$, mais aussi d'éviter que \mathbf{V} soit une matrice de 'blocage', c'est à dire satisfaisant $\mathbf{V}^H \mathbf{H}_N = \mathbf{0}$, qui correspond à $r = 0$. Dans [38] des contraintes linéaires et quadratiques sont considérées pour garder la procédure de minimisation aussi simple que possible. Les différents choix de contraintes correspondent à différentes implémentations et différentes performances de l'algorithme MRE.

2.3.1 Solution asymptotique du critère MRE

Dans l'équation (2.19), le critère MRE est présenté en tant que problème d'optimisation de matrice. Néanmoins, un formalisme permettant de le transformer en un problème d'optimisation de vecteur simplifiera énormément les développements algébriques et facilitera le côté algorithmique. Si on considère le vecteur $\mathbf{v} = \text{vec}(\mathbf{V})$, le lemme suivant [38] donne alors une expression simple, en fonction de \mathbf{v} , du critère MRE de l'équation (2.19).

Lemme 2.1 Soient $\mathbf{C}_{N,0} = E(\mathbf{x}_N(t) \mathbf{x}_N^H(t))$ et $\mathbf{C}_{N,1} = E(\mathbf{x}_N(t + 1) \mathbf{x}_N^H(t))$ les matrices d'auto-corrélation d'ordre zéro et un respectivement du vecteur observé $\mathbf{x}_N(t)$. Notons par \mathbf{C} la matrice de

dimension $nm \times nm$, donnée par

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_{N,0} & -\mathbf{C}_{N,1}^H & \mathbf{0} & \cdots & \mathbf{0} \\ -\mathbf{C}_{N,1} & 2\mathbf{C}_{N,0} & \ddots & \ddots & \vdots \\ \mathbf{0} & \ddots & \ddots & & \mathbf{0} \\ \vdots & \ddots & & 2\mathbf{C}_{N,0} & -\mathbf{C}_{N,1}^H \\ \mathbf{0} & \cdots & \mathbf{0} & -\mathbf{C}_{N,1} & \mathbf{C}_{N,0} \end{bmatrix}.$$

Dans ce cas la fonction coût $J(\mathbf{V})$ de l'équation (2.19) est aussi donnée par

$$J(\mathbf{V}) = \mathcal{J}(\mathbf{v}) = \mathbf{v}^H \mathbf{C} \mathbf{v},$$

où $\mathbf{v} = \text{vec}(\mathbf{V})$ ($\text{vec}(\cdot)$ étant l'opérateur de vecteurisation-colonne d'une matrice).

Démonstration : voir Annexe A

Il s'en suit alors que la solution du problème (2.19), dans le cas où la contrainte est quadratique, est donnée par

$$\mathbf{v}_{opt} = \arg \min_{\|\mathbf{v}\|=1} \mathbf{v}^H \mathbf{C} \mathbf{v}.$$

Il est bien évident que \mathbf{v}_{opt} n'est autre que le vecteur propre à norme unité, associé à la plus petite valeur propre de la matrice \mathbf{C} .

Dans le cas où la contrainte est linéaire, résoudre le problème (2.19) revient à minimiser $\mathcal{J}(\mathbf{v})$ sous une contrainte $\mathbf{u}^H \mathbf{v} = 1$, où $\mathbf{u} = \text{vec}(\mathbf{U})$. La solution, bien connue à ce problème, est alors donnée par

$$\mathbf{v}_{opt} = \frac{\mathbf{C}^{-1} \mathbf{u}}{\mathbf{u}^H \mathbf{C}^{-1} \mathbf{u}}.$$

L'algorithme d'égalisation MRE en schéma bloc est résumé par la TABLE 2.3.

$\mathbf{C}_{N,0}$	$= \frac{1}{K} \sum_{k=N-1}^{K+N-2} \mathbf{x}_N(k) \mathbf{x}_N^H(k),$ (K : nombre d'échantillons)
$\mathbf{C}_{N,1}$	$= \frac{1}{K} \sum_{k=N}^{K+N} \mathbf{x}_N(k) \mathbf{x}_N^H(k-1)$
\mathbf{C}	$= \text{diag}(1, 2, \dots, 2, 1) \otimes \mathbf{C}_{N,0} - \mathbf{J}_{m-1} \otimes \mathbf{C}_{N,1} - \mathbf{J}_{m-1}^{-1} \otimes \mathbf{C}_{N,1}^H$
\mathbf{v}_{opt}	$= \begin{cases} \text{vecteur propre mineur de } \mathbf{C}, & (\text{cas d'une contrainte quadratique}) \\ \frac{\mathbf{C}^{-1} \mathbf{u}}{\mathbf{u}^H \mathbf{C}^{-1} \mathbf{u}}, & (\text{cas d'une contrainte linéaire}) \end{cases}$
\mathbf{V}_{opt}	$= \text{vec}^{-1}(\mathbf{v}_{opt})$

TAB. 2.3 – Algorithme d'égalisation aveugle de type MRE.

2.3.2 Implémentation adaptative

Pour une implémentation adaptative de l'algorithme MRE, plusieurs schémas peuvent être adoptés. On se contentera ici de donner une solution présentée dans [38], basée sur la technique du gradient avec

contrainte quadratique ou linéaire, cette approche est parmi les moins coûteuses en terme de complexité de calcul. Sachant que le gradient de $J(\mathbf{V})$ est donné par

$$\begin{aligned} \nabla J(\mathbf{V}) = \frac{\partial J(\mathbf{V})}{\partial \mathbf{V}^H} &= (\mathbf{C}_{N,0} \mathbf{V} [\mathbf{I}_{m-1} \mathbf{0}_{m-1,1}]^T - \mathbf{C}_{N,-1} \mathbf{V} [\mathbf{0}_{m-1,1} \mathbf{I}_{m-1}]^T) [\mathbf{I}_{m-1} \mathbf{0}_{m-1,1}] \\ &\quad - (\mathbf{C}_{N,1} \mathbf{V} [\mathbf{I}_{m-1} \mathbf{0}_{m-1,1}]^T - \mathbf{C}_{N,0} \mathbf{V} [\mathbf{0}_{m-1,1} \mathbf{I}_{m-1}]^T) [\mathbf{0}_{m-1,1} \mathbf{I}_{m-1}], \end{aligned}$$

si $\mathbf{V}(t)$ est une estimation de la matrice d'égalisation à l'instant t , alors en approximant la corrélation $E(\mathbf{x}_N(t+\alpha)\mathbf{x}_N(t+\beta))$ par sa valeur instantanée, i.e., $E(\mathbf{x}_N(t+\alpha)\mathbf{x}_N^H(t+\beta)) \simeq \mathbf{x}_N(t+\alpha)\mathbf{x}_N^H(t+\beta)$, on obtient alors l'expression suivante du gradient

$$\nabla J(\mathbf{V}(t)) = \mathbf{x}_N(t)\boldsymbol{\varepsilon}^H(t) [\mathbf{I}_{m-1} \mathbf{0}_{m-1,1}] - \mathbf{x}_N(t+1)\boldsymbol{\varepsilon}^H(t) [\mathbf{0}_{m-1,1} \mathbf{I}_{m-1}],$$

avec

$$\boldsymbol{\varepsilon}(t) = [\mathbf{I}_{m-1} \mathbf{0}_{m-1,1}] \mathbf{V}^H(t)\mathbf{x}_N(t) - [\mathbf{0}_{m-1,1} \mathbf{I}_{m-1}] \mathbf{V}^H(t)\mathbf{x}_N(t+1).$$

Si μ est le pas d'adaptation, la matrice d'égalisation $\mathbf{V}(t)$ est alors mise à jour comme suit

$$\begin{aligned} \tilde{\mathbf{V}}(t) &= \mathbf{V}(t-1) - \mu \nabla J(\mathbf{V}(t-1)), \\ \mathbf{V}(t) &= \frac{\tilde{\mathbf{V}}(t)}{\|\tilde{\mathbf{V}}(t)\|}, \text{ (cas d'une contrainte quadratique),} \\ V_{i,j}(t) &= \begin{cases} \tilde{V}_{i,j}(t), & (i,j) \neq (1,1) \\ 1, & i=j=1 \end{cases}, \text{ (cas d'une contrainte linéaire).} \end{aligned}$$

La version adaptative de l'algorithme d'égalisation MRE est alors donné par la TABLE 2.4. Noter que le coût global de cette implémentation est de l'ordre de $O(mn)$ opérations par itération.

$\begin{aligned} \boldsymbol{\varepsilon}'(t) &= [\mathbf{I}_{m-1} \mathbf{0}_{m-1,1}] \mathbf{V}^H(t-1)\mathbf{x}_N(t-1) - [\mathbf{0}_{m-1,1} \mathbf{I}_{m-1}] \mathbf{V}^H(t-1)\mathbf{x}_N(t) \\ \tilde{\mathbf{V}}(t) &= \mathbf{V}(t-1) - \mu(\mathbf{x}_N(t-1)\boldsymbol{\varepsilon}'(t)^H [\mathbf{I}_{m-1} \mathbf{0}_{m-1,1}] \\ &\quad - \mathbf{x}_N(t)\boldsymbol{\varepsilon}'(t)^H [\mathbf{0}_{m-1,1} \mathbf{I}_{m-1}]) \\ \mathbf{V}(t) &= \frac{\tilde{\mathbf{V}}(t)}{\ \tilde{\mathbf{V}}(t)\ }, \text{ (cas d'une contrainte quadratique)} \\ V_{i,j}(t) &= \begin{cases} \tilde{V}_{i,j}(t), & (i,j) \neq (1,1) \\ 1, & i=j=1 \end{cases}, \text{ (cas d'une contrainte linéaire)} \end{aligned}$

TAB. 2.4 – Algorithme adaptatif d'égalisation aveugle de type MRE.

2.4 Méthode des matrices d'autocorrélation décalées

Dans cette technique, proposée par H. Gazzah et al [33, 34]¹ pour le cas SIMO, on démontre qu'on peut extraire un égaliseur ZF à retard $\tau \in \{0, \dots, m-2\}$, du noyau d'une certaine matrice construite à

¹Cette technique est souvent désignée par l'acronyme GRDA, qui fait référence à ses auteurs Gazzah, Regalia, Delmas, et Abed-Meraim.

partir des matrices d'autocorrélation d'ordre τ et $\tau + 1$ respectivement, du vecteur de données $\mathbf{x}_N(t)$. En effet, si $N > L$ la partie signal de la matrice d'autocorrélation $\tilde{\mathbf{C}}_{N,k} = \mathbf{C}_{N,k} - \sigma_b^2 \mathbf{I}_p \otimes \mathbf{J}_N^k = \mathbf{H}_N \mathbf{J}_m^k \mathbf{H}_N^H$, $k \in \mathbb{Z}$ (voir définitions dans les équations (2.3) et (2.4)) est une matrice à rang déficient, i.e., $\text{rank}(\tilde{\mathbf{C}}_{N,k}) = m - k$. Soit $\text{null}_l(\tilde{\mathbf{C}}_{N,k}) = \{\mathbf{z} \in \mathbb{C}^n : \mathbf{z}^H \tilde{\mathbf{C}}_{N,k} = \mathbf{0}\}$, $k \in \{0, \dots, m - 1\}$, un sous espace vectoriel de dimension $n - m + k$ (noyau gauche de $\tilde{\mathbf{C}}_{N,k}$). Dans ce cas, on a le résultat suivant

Théorème 2.4 Soient $\mathbf{M}_1 \in \mathbb{C}^{n \times (n-m+\tau)}$ et $\mathbf{M}_2 \in \mathbb{C}^{n \times (n-m+\tau+1)}$ deux bases orthonormales de $\text{null}_l(\tilde{\mathbf{C}}_{N,\tau})$ et $\text{null}_l(\tilde{\mathbf{C}}_{N,\tau+1})$ respectivement, et $\mathbf{Q}_1 \stackrel{\text{def}}{=} \mathbf{M}_1^H \tilde{\mathbf{C}}_{N,0} \mathbf{M}_2$ une matrice de dimension $(n - m + \tau) \times (n - m + \tau + 1)$. On suppose $N > L$. Dans ce cas, il existe un vecteur complexe \mathbf{v}'_τ de dimension $n - m + \tau + 1$, tel que

$$\mathbf{Q}_1 \mathbf{v}'_\tau = \mathbf{0}, \quad (2.20)$$

si et seulement si, $\mathbf{v}_\tau = \mathbf{M}_2 \mathbf{v}'_\tau$ est un égaliseur ZF de retard τ du système (1.4).

Démonstration : voir Annexe A

Reste maintenant à choisir le meilleur filtre \mathbf{v}'_τ , correspondant à l'égaliseur, et qui appartient au noyau droit de la matrice \mathbf{Q}_1 du Théorème 2.4. Ce noyau étant un sous-espace vectoriel défini par $\text{null}_r(\mathbf{Q}_1) = \{\mathbf{z} \in \mathbb{C}^{n-m+\tau+1} : \mathbf{Q}_1 \mathbf{z} = \mathbf{0}\}$, de dimension $l = \dim(\text{null}_r(\mathbf{Q}_1)) = \max\{n - 2m + \tau + 1, 1\}$. Si $\mathbf{W} \in \mathbb{C}^{(n-m+\tau+1) \times l}$ est une base du sous-espace $\text{null}_r(\mathbf{Q}_1)$, alors $\mathbf{v}'_\tau = \mathbf{W} \mathbf{v}''_\tau$, où \mathbf{v}''_τ est un vecteur complexe de dimension l . En utilisant le critère d'égalisation par pique (EPC : *Equalization Peak Criterion*) [4], on peut estimer \mathbf{v}''_τ parmi les vecteurs à norme unité, de sorte que l'énergie à la sortie du filtre égaliseur appliqué au vecteur de données $\mathbf{x}_N(t)$ soit maximale. i.e.,

$$\begin{aligned} \mathbf{v}''_\tau &= \arg \max_{\|\mathbf{z}\|=1} E (|(\mathbf{M}_2 \mathbf{W} \mathbf{z})^H \mathbf{x}_N(t)|^2) \\ &= \arg \max_{\|\mathbf{z}\|=1} \mathbf{z}^H \mathbf{Q}_2 \mathbf{z}, \\ \mathbf{Q}_2 &\stackrel{\text{def}}{=} \mathbf{W}^H \mathbf{M}_2^H \mathbf{C}_{N,0} \mathbf{M}_2 \mathbf{W}. \end{aligned}$$

Ainsi, la TABLE 2.5 résume l'algorithme d'égalisation par la méthode des matrices d'autocorrélation décalées.

2.5 Méthodes d'optimisation sous contraintes

A l'origine, ces méthodes ont été développées pour des problèmes liés au traitement d'antennes (formateurs de voies de Capon et à réponse sans distorsion à variance minimale (*Minimum Variance Distorsionless Response*, MVDR) [48]. Par la suite, elles ont été adoptées à certains problèmes en communications numériques multiutilisateurs (égalisation multicanaux, récepteurs du type *Code Division Multiple Access*, CDMA) [47, 79, 80, 86]. L'idée de cette technique (en égalisation) est de minimiser la variance à la sortie de l'égaliseur sous certaines contraintes appropriées. Le rôle de ces contraintes n'est pas seulement d'éviter des solutions triviales, mais aussi de garantir que la composante désirée du signal reçu ne sera pas annulée. Pour la suite de ce paragraphe, on supposera qu'on est dans un cas SIMO. On présentera deux approches proposées par M. K. Tsatsanis et Z. Xu [80, 86].

$\mathbf{C}_{N,0}$	$= \frac{1}{K} \sum_{k=N-1}^{K+N-2} \mathbf{x}_N(k) \mathbf{x}_N^H(k),$ (K : nombre d'échantillons)
$\tilde{\mathbf{C}}_{N,0}$	$= \mathbf{C}_{N,0} - \sigma_b^2 \mathbf{I}_n,$ (σ_b^2 estimée à partir de (1.26))
$\tilde{\mathbf{C}}_{N,\tau}$	$= \frac{1}{K} \sum_{k=N+\tau-1}^{K+N+\tau-2} \mathbf{x}_N(k) \mathbf{x}_N^H(k - \tau) - \hat{\sigma}_b^2 \mathbf{J}_N^\tau \otimes \mathbf{I}_p$
$\tilde{\mathbf{C}}_{N,\tau+1}$	$= \frac{1}{K} \sum_{k=N+\tau}^{K+N+\tau-1} \mathbf{x}_N(k) \mathbf{x}_N^H(k - \tau - 1) - \hat{\sigma}_b^2 \mathbf{J}_N^{\tau+1} \otimes \mathbf{I}_p$
$(\mathbf{U}_1, \mathbf{E}_1, \mathbf{V}_1)$	$= \text{svd}(\tilde{\mathbf{C}}_{N,\tau})$
\mathbf{M}_1	$= \mathbf{U}_1(:, m - \tau + 1 : n)$
$(\mathbf{U}_2, \mathbf{E}_2, \mathbf{V}_2)$	$= \text{svd}(\tilde{\mathbf{C}}_{N,\tau+1})$
\mathbf{M}_2	$= \mathbf{U}_2(:, m - \tau : n)$
\mathbf{Q}_1	$= \mathbf{M}_1^H \tilde{\mathbf{C}}_{N,0} \mathbf{M}_2$
$(\mathbf{U}_3, \mathbf{E}_3, \mathbf{V}_3)$	$= \text{svd}(\mathbf{Q}_1)$
l	$= \max\{n - 2m + \tau + 1, 1\}$
\mathbf{W}	$= \mathbf{V}_3(:, n - l - m + \tau + 2 : n - m + \tau + 1)$
\mathbf{Q}_2	$= \mathbf{W}^H \mathbf{M}_2^H \mathbf{C}_{N,0} \mathbf{M}_2 \mathbf{W}$
\mathbf{v}_τ''	$=$ vecteur propre dominant de \mathbf{Q}_2
\mathbf{v}_τ	$= \mathbf{M}_2 \mathbf{W} \mathbf{v}_\tau''$, à une constante multiplicative scalaire non-nulle près

TAB. 2.5 – Algorithme d'égalisation aveugle par la méthode des matrices d'autocorrélation décalées.

2.5.1 Algorithme de Capon

Soit $\mathbf{v}_\tau \in \mathbb{C}^n$, un égaliseur à retard τ du système (1.4), i.e., $\mathbf{v}_\tau^H \mathbf{x}_N(t) = \hat{s}(t - \tau)$. Dans ce cas, le principe du MVDR est adapté à l'égalisation, en le transformant en problème d'optimisation sous contraint, i.e.

$$\begin{cases} \min_{\mathbf{v}_\tau^H \mathbf{g}_\tau = 1} J(\mathbf{v}_\tau), \\ J(\mathbf{v}_\tau) = E(|\mathbf{v}_\tau^H \mathbf{x}_N(t)|^2) = \mathbf{v}_\tau^H \mathbf{C}_N \mathbf{v}_\tau, \end{cases} \quad (2.21)$$

où \mathbf{C}_N est la matrice d'autocorrélation associée au vecteur de données $\mathbf{x}_N(t)$ et qui est donnée par les équations (1.8) et (1.9). \mathbf{g}_τ étant le vecteur donné par (1.20) et qui désigne la $(\tau + 1)$ -ième colonne de la matrice de filtrage \mathbf{H}_N donnée par (1.7). La solution du problème (2.21) est bien connue, que l'on peut obtenir en utilisant les multiplieurs de Lagrange [63]

$$\mathbf{v}_\tau^{opt} = \arg \min_{\mathbf{v}_\tau^H \mathbf{g}_\tau = 1} \mathbf{v}_\tau^H \mathbf{C}_N \mathbf{v}_\tau = \frac{\mathbf{C}_N^{-1} \mathbf{g}_\tau}{\mathbf{g}_\tau^H \mathbf{C}_N^{-1} \mathbf{g}_\tau}.$$

Noter que d'après [70], de meilleures performances sont obtenues en choisissant $\tau \in \{L, \dots, N - 1\}$. Le vecteur

$$\mathbf{g}_\tau = \begin{bmatrix} \mathbf{0}_{(\tau-L)p \times 1} \\ \mathbf{h}(L) \\ \vdots \\ \mathbf{h}(0) \\ \mathbf{0}_{(N-\tau-1)p \times 1} \end{bmatrix}$$

peut être alors linéairement paramétré en fonction des coefficients du canal. En effet, soient \mathcal{A}_a et \mathcal{P}_τ deux matrices de dimensions $a \times a$ et $n \times (L+1)p$ respectivement, définies par

$$\mathcal{A}_a \stackrel{\text{def}}{=} \begin{bmatrix} 0 & \cdots & 0 & 1 \\ \vdots & & 1 & 0 \\ 0 & 1 & & \vdots \\ 1 & 0 & \cdots & 0 \end{bmatrix},$$

$$\mathcal{P}_\tau \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{0}_{(\tau-L)p \times (L+1)p} \\ \mathcal{A}_{(L+1)p} \\ \mathbf{0}_{(N-\tau-1)p \times (L+1)p} \end{bmatrix},$$

et \mathbf{h} étant le vecteur donné par (1.12), contenant les coefficients du canal. Dans ce cas, le vecteur \mathbf{g}_τ s'exprime par

$$\mathbf{g}_\tau = \mathcal{P}_\tau \mathbf{h},$$

d'où, le vecteur \mathbf{v}_τ^{opt} aura pour expression

$$\mathbf{v}_\tau^{opt} = \frac{\mathbf{C}_N^{-1} \mathcal{P}_\tau \mathbf{h}}{\mathbf{h}^H \mathcal{P}_\tau^T \mathbf{C}_N^{-1} \mathcal{P}_\tau \mathbf{h}}.$$

La variance minimale à la sortie de l'égaliseur est alors donnée par

$$\mathcal{J}(\mathbf{h}) = J(\mathbf{v}_\tau^{opt}) = (\mathbf{h}^H \mathcal{P}_\tau^T \mathbf{C}_N^{-1} \mathcal{P}_\tau \mathbf{h})^{-1}.$$

Le principe de l'égaliseur de Capon est d'estimer le vecteur \mathbf{h} de façon à maximiser l'énergie du signal désiré à la sortie de l'égaliseur après suppression des interférences. i.e.,

$$\mathbf{h}_{opt} = \arg \max_{\|\mathbf{h}\|=1} \mathcal{J}(\mathbf{h}) = \arg \min_{\|\mathbf{h}\|=1} \mathbf{h}^H \mathcal{P}_\tau^T \mathbf{C}_N^{-1} \mathcal{P}_\tau \mathbf{h}.$$

Il est démontré dans [86], qu'à fort SNR \mathbf{h}_{opt} converge vers $\mathbf{h}/\|\mathbf{h}\|$. Ainsi, la TABLE 2.6 résume l'algorithme d'égalisation de Capon.

$\mathbf{C}_N = \frac{1}{K} \sum_{k=N-1}^{K+N-2} \mathbf{x}_N(k) \mathbf{x}_N^H(k), \quad (K : \text{nombre d'échantillons})$ $\mathbf{A} = \mathcal{P}_\tau^T \mathbf{C}_N^{-1} \mathcal{P}_\tau$ $\mathbf{h}_{opt} = \text{vecteur propre mineur de } \mathbf{A}$ $\mathbf{v}_\tau^{opt} = \frac{\mathbf{C}_N^{-1} \mathcal{P}_\tau \mathbf{h}_{opt}}{\mathbf{h}_{opt}^H \mathbf{A} \mathbf{h}_{opt}}$
--

TAB. 2.6 – Algorithme d'égalisation de Capon.

2.5.2 Égalisation par annulateurs de lobes secondaires généralisée

Une autre technique semblable à celle de Capon (voir paragraphe 2.5.1), inspirée elle aussi d'une technique de traitement d'antennes, connue sous le nom de formateur de voies par annulateurs de lobes

secondaires généralisé (*Generalized Sidelobe Cancellers*, GSC). Considérons le problème d'optimisation suivant

$$\begin{cases} \min_{\mathcal{P}_\tau^T \mathbf{v}_\tau = \mathbf{u}} J(\mathbf{v}_\tau), \\ J(\mathbf{v}_\tau) = E(|\mathbf{v}_\tau^H \mathbf{x}_N(t)|^2) = \mathbf{v}_\tau^H \mathbf{C}_N \mathbf{v}_\tau, \end{cases} \quad (2.22)$$

où le vecteur \mathbf{u} est un vecteur à déterminer. Noter que la contrainte dans (2.22) garantit aussi la non-suppression de la composante désirée du signal reçu. En effet

$$\mathbf{v}_\tau^H \mathbf{g}_\tau = \mathbf{v}_\tau^H \mathcal{P}_\tau \mathbf{h} = \mathbf{u}^H \mathbf{h} = \text{constante.}$$

D'une manière analogue que pour l'algorithme de Capon (voir paragraphe 2.5.1), la solution du problème (2.22) est donnée par

$$\mathbf{v}_\tau^{\text{opt}} = \mathbf{C}_N^{-1} \mathcal{P}_\tau (\mathcal{P}_\tau^H \mathbf{C}_N^{-1} \mathcal{P}_\tau)^{-1} \mathbf{u}.$$

Dans ce cas la variance minimale de la sortie de l'égaliseur est donnée par

$$\mathcal{J}(\mathbf{h}) = J(\mathbf{v}_\tau^{\text{opt}}) = \mathbf{u}^H (\mathcal{P}_\tau^T \mathbf{C}_N^{-1} \mathcal{P}_\tau)^{-1} \mathbf{u}.$$

Comme $\mathcal{J}(\mathbf{h})$ dépend du paramètre \mathbf{u} , là aussi la même idée du min/max est applicable. i.e.,

$$\mathbf{u}_{\text{opt}} = \arg \max_{\|\mathbf{u}\|=1} \mathbf{u}^H (\mathcal{P}_\tau^T \mathbf{C}_N^{-1} \mathcal{P}_\tau)^{-1} \mathbf{u} = \arg \min_{\|\mathbf{u}\|=1} \mathbf{u}^H \mathcal{P}_\tau^T \mathbf{C}_N^{-1} \mathcal{P}_\tau \mathbf{u}.$$

La TABLE 2.7 résume l'algorithme d'égalisation du type GSC.

\mathbf{C}_N	$= \frac{1}{K} \sum_{k=N-1}^{K+N-2} \mathbf{x}_N(k) \mathbf{x}_N^H(k)$, (K : nombre d'échantillons)
\mathbf{A}	$= \mathcal{P}_\tau^T \mathbf{C}_N^{-1} \mathcal{P}_\tau$
\mathbf{u}_{opt}	$=$ vecteur propre mineur de \mathbf{A}
$\mathbf{v}_\tau^{\text{opt}}$	$= \mathbf{C}_N^{-1} \mathcal{P}_\tau \mathbf{A}^{-1} \mathbf{u}_{\text{opt}}$

TAB. 2.7 – Algorithme d'égalisation du type GSC.

2.5.3 Implémentation adaptative

On présentera ici deux implémentations adaptatives de l'algorithme d'égalisation GSC (décrit au paragraphe 2.5.2). L'une basée sur la méthode du gradient stochastique, qui fut adoptée la première fois pour ce genre de problème par O. L. Frost [31]. L'avantage de cette approche est, sans conteste, son faible coût de calcul (de l'ordre de $O(n(L+1)p)$ opérations par itération), tandis que son principal inconvénient est sa faible vitesse de convergence. L'autre, basée sur la méthode des moindres carrés récurrente (*Recursive Least Squares*, RLS) qui est relativement coûteuse, est nettement rapide.

Adaptation via la méthode du gradient

Le lagrangien relatif au problème (2.22) est explicitement donné par

$$\mathcal{L}(\mathbf{v}_\tau, \boldsymbol{\lambda}, \mathbf{u}) = \mathbf{v}_\tau^H \mathbf{C}_N \mathbf{v}_\tau + \boldsymbol{\lambda}^H (\mathcal{P}_\tau^T \mathbf{v}_\tau - \mathbf{u}) + (\mathbf{v}_\tau^H \mathcal{P}_\tau - \mathbf{u}^H) \boldsymbol{\lambda}, \quad (2.23)$$

où le vecteur $\boldsymbol{\lambda}$ correspond au multiplicateur de Lagrange. En dérivant par rapport aux variables \mathbf{v}_τ , $\boldsymbol{\lambda}$ et \mathbf{u} respectivement, on obtient les expressions suivantes des gradients de \mathcal{L}

$$\nabla_{\mathbf{v}_\tau} \mathcal{L}(\mathbf{v}_\tau, \boldsymbol{\lambda}, \mathbf{u}) = \frac{\partial \mathcal{L}}{\partial \mathbf{v}_\tau^H} = \mathbf{C}_N \mathbf{v}_\tau + \mathcal{P}_\tau \boldsymbol{\lambda}, \quad (2.24)$$

$$\nabla_{\boldsymbol{\lambda}} \mathcal{L}(\mathbf{v}_\tau, \boldsymbol{\lambda}, \mathbf{u}) = \frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}^H} = \mathcal{P}_\tau^T \mathbf{v}_\tau - \mathbf{u}, \quad (2.25)$$

$$\nabla_{\mathbf{u}} \mathcal{L}(\mathbf{v}_\tau, \boldsymbol{\lambda}, \mathbf{u}) = \frac{\partial \mathcal{L}}{\partial \mathbf{u}^H} = -\boldsymbol{\lambda}. \quad (2.26)$$

Le but est de minimiser \mathcal{L} par rapport à \mathbf{v}_τ et de le maximiser par rapport à \mathbf{u} , ce qui donne les mises à jour suivantes pour \mathbf{v}_τ et \mathbf{u} respectivement

$$\mathbf{v}_\tau(t) = \mathbf{v}_\tau(t-1) - \mu_{\mathbf{v}_\tau} \nabla_{\mathbf{v}_\tau} \mathcal{L}(\mathbf{v}_\tau(t-1), \boldsymbol{\lambda}(t-1), \mathbf{u}(t-1)), \quad (2.27)$$

$$\tilde{\mathbf{u}}(t) = \mathbf{u}(t-1) + \mu_{\mathbf{u}} \boldsymbol{\Pi}(t-1) \nabla_{\mathbf{u}} \mathcal{L}(\mathbf{v}_\tau(t-1), \boldsymbol{\lambda}(t-1), \mathbf{u}(t-1)), \quad (2.28)$$

$$\mathbf{u}(t) = \frac{\tilde{\mathbf{u}}(t)}{\|\tilde{\mathbf{u}}(t)\|},$$

$$\boldsymbol{\Pi}(t) = \mathbf{I}_{(L+1)p} - \frac{\mathbf{u}(t) \mathbf{u}^H(t)}{\mathbf{u}^H(t) \mathbf{u}(t)}.$$

L'utilisation du projecteur $\boldsymbol{\Pi}(t)$ dans (2.28) intervient pour simplifier l'expression de $\mathbf{u}(t)$, son introduction n'affecte en rien les performances de l'égaliseur. La mise à jour finale des équations est obtenue en remplaçant les gradients (2.24) et (2.26) dans les équations (2.27) et (2.28) respectivement. On détermine le vecteur $\boldsymbol{\lambda}(t-1)$ en forçant la contrainte $\mathcal{P}_\tau^T \mathbf{v}_\tau(t) = \mathbf{u}(t-1)$ pour chaque itération (voir aussi [31]). Si la matrice d'autocorrélation \mathbf{C}_N est approximée par sa valeur instantanée, i.e., $\mathbf{C}_N \approx \mathbf{x}_N(t) \mathbf{x}_N^H(t)$, on arrivera en définitif aux récursions suivantes

$$\begin{aligned} \mathbf{v}_\tau(t) &= (\mathbf{I}_n - \mathcal{P}_\tau \mathcal{P}_\tau^T) (\mathbf{v}_\tau(t-1) - \mu_{\mathbf{v}_\tau} \mathbf{x}_N(t) \mathbf{x}_N^H(t) \mathbf{v}_\tau(t-1)) + \mathcal{P}_\tau \mathbf{u}(t-1), \\ \tilde{\mathbf{u}}(t) &= \mathbf{u}(t-1) + \frac{\mu_{\mathbf{v}_\tau}}{\mu_{\mathbf{u}}} \boldsymbol{\Pi}(t-1) \mathcal{P}_\tau^T (\mu_{\mathbf{v}_\tau} \mathbf{x}_N(t) \mathbf{x}_N^H(t) \mathbf{v}_\tau(t-1) - \mathbf{v}_\tau(t-1)). \end{aligned}$$

La TABLE 2.8 résume la version adaptative, basée sur la méthode du gradient, de l'algorithme d'égalisation du GSC. Noter que le coût global de cette implémentation est de l'ordre de $O(n(L+1)p)$ opérations par itération.

Adaptation via la méthode RLS

Certes, l'implémentation présentée à la TABLE 2.8 est à faible complexité de calcul (de l'ordre de $O(n(L+1)p)$ opérations par itération), néanmoins sa vitesse de convergence est faible aussi. On peut y remédier à cette situation, moyennant un coût de calcul plus important, mais nettement inférieur à celui de l'implémentation présentée au paragraphe 2.5.2, et ce en réduisant le coût de calcul de la matrice

$$\begin{aligned}
 \mathbf{v}_\tau(t) &= (\mathbf{I}_n - \mathcal{P}_\tau \mathcal{P}_\tau^T) (\mathbf{v}_\tau(t-1) - \mu_{\mathbf{v}_\tau} \mathbf{x}_N(t) \mathbf{x}_N^H(t) \mathbf{v}_\tau(t-1)) + \mathcal{P}_\tau \mathbf{u}(t-1) \\
 \tilde{\mathbf{u}}(t) &= \mathbf{u}(t-1) + \frac{\mu_{\mathbf{v}_\tau}}{\mu_{\mathbf{u}}} \mathbf{\Pi}(t-1) \mathcal{P}_\tau^T (\mu_{\mathbf{v}_\tau} \mathbf{x}_N(t) \mathbf{x}_N^H(t) \mathbf{v}_\tau(t-1) - \mathbf{v}_\tau(t-1)) \\
 \mathbf{u}(t) &= \frac{\tilde{\mathbf{u}}(t)}{\|\tilde{\mathbf{u}}(t)\|} \\
 \mathbf{\Pi}(t) &= \mathbf{I}_{(L+1)p} - \frac{\mathbf{u}(t) \mathbf{u}^H(t)}{\mathbf{u}^H(t) \mathbf{u}(t)}
 \end{aligned}$$

TAB. 2.8 – Algorithme d'égalisation adaptative du type GSC (méthode du gradient).

\mathbf{C}_N^{-1} dans l'implémentation de la TABLE 2.7 et qui est de l'ordre de $O(n^3)$, à $O(n^2)$. Ce qui donne un coût global de $O(n^2)$, du moment que la matrice $\mathbf{A} = \mathcal{P}_\tau^T \mathbf{C}_N^{-1} \mathcal{P}_\tau$ (voir TABLE 2.7) est de dimension $(L+1)p \times (L+1)p$, donc sa décomposition en valeurs propres/vecteurs propres et son inversion ont une complexité de calcul de l'ordre de $O((L+1)^3 p^3)$ seulement. Comme la matrice d'autocorrélation peut être estimée récursivement par $\mathbf{C}_N(t) = \beta \mathbf{C}_N(t-1) + \mathbf{x}_N(t) \mathbf{x}_N^H(t)$, où β est un facteur d'oubli, par application du lemme d'inversion matricielle [46], on aura alors

$$\begin{aligned}
 \mathbf{C}_N^{-1}(t) &= \frac{1}{\beta} \mathbf{C}_N^{-1}(t-1) - \mathbf{y}(t) \mathbf{y}^H(t), \\
 \mathbf{y}(t) &= \frac{\mathbf{C}_N^{-1}(t-1) \mathbf{x}_N(t)}{\beta \sqrt{1 + \frac{1}{\beta} \mathbf{x}_N^H(t) \mathbf{C}_N^{-1}(t-1) \mathbf{x}_N(t)}}.
 \end{aligned}$$

La TABLE 2.9 résume la version adaptative, basée sur la méthode RLS, de l'algorithme d'égalisation du GSC. Noter que le coût global de cette implémentation est de l'ordre de $O(n^2)$ opérations par itération.

$$\begin{aligned}
 \mathbf{y}(t) &= \frac{\mathbf{C}_N^{-1}(t-1) \mathbf{x}_N(t)}{\beta \sqrt{1 + \frac{1}{\beta} \mathbf{x}_N^H(t) \mathbf{C}_N^{-1}(t-1) \mathbf{x}_N(t)}} \\
 \mathbf{C}_N^{-1}(t) &= \frac{1}{\beta} \mathbf{C}_N^{-1}(t-1) - \mathbf{y}(t) \mathbf{y}^H(t) \\
 \mathbf{A}(t) &= \mathcal{P}_\tau^T \mathbf{C}_N^{-1}(t) \mathcal{P}_\tau \\
 \mathbf{u}(t) &= \text{vecteur propre mineur de } \mathbf{A}(t) \\
 \mathbf{v}_\tau(t) &= \mathbf{C}_N^{-1}(t) \mathcal{P}_\tau \mathbf{A}^{-1}(t) \mathbf{u}(t)
 \end{aligned}$$

TAB. 2.9 – Algorithme d'égalisation adaptative du type GSC (méthode RLS).

2.6 Méthode sous-espace robuste

Cette approche proposée par A. Gorokhov *et al.* [43], est une version robuste de la méthode sous-espace décrite au paragraphe 1.6 (voir aussi [64]). Sa robustesse aux erreurs de surestimation de l'ordre du canal est due au fait qu'elle permet conjointement de détecter l'ordre du système et de l'identifier. Rappelons tout d'abord un résultat de base de la méthode sous-espace, qui se résume dans le lemme suivant (voir [64] pour la démonstration) :

Lemme 2.2 Soient $h(z)$ et $h'(z)$ deux fonctions de transfert de systèmes SIMO-FIR, de dimension $p \times 1$ et d'ordre L et L' respectivement. i.e., $h(z) = \sum_{k=0}^L \mathbf{h}(k) z^{-k}$ et $h'(z) = \sum_{k=0}^{L'} \mathbf{h}'(k) z^{-k}$, \mathbf{H}_N et \mathbf{H}'_N

sont les matrices de filtrage correspondant aux fonctions $h(z)$ et $h'(z)$ respectivement (voir définitions du paragraphe 1.5). Dans ce cas, pour $N > L' \geq L$, les sous-espaces vectoriel générés par les colonnes des matrices \mathbf{H}'_N et \mathbf{H}_N coïncident, si et seulement si, $h'(z) = r(z)h(z)$, où $r(z)$ est un polynôme scalaire de degrés $L' - L$.

Notons par $h(z)$ la fonction de transfert du vrai système, L la valeur exacte de son ordre et L' sa valeur estimée. Dans ce cas, si \mathbf{U}_b est une base orthonormale du sous-espace bruit de la matrice d'autocorrélation \mathbf{C}_N associée au vecteur de données $\mathbf{x}_N(t)$ (voir équations (1.4), (1.8) et (1.9)), et si l'ordre du canal est connu avec exactitude, i.e., $L' = L$ alors la fonction $h(z)$ est identifiable via l'équation $\mathbf{U}_b^H \mathbf{H}_N = \mathbf{0}$, qui équivaut à $\mathbf{h}^H \mathbf{Q}_{L'} \mathbf{h} = 0$, où \mathbf{h} étant le vecteur donné par (1.12), contenant les coefficients du canal, $\mathbf{Q}_{L'}$ étant la matrice donnée par (1.17) (ici l'indice L' fait référence à l'utilisation de l'ordre estimé du canal), et qui est construite à partir des colonnes de la matrice \mathbf{U}_b . En pratique, on estime \mathbf{h} en minimisant la forme quadratique $\mathbf{h}^H \mathbf{Q}_{L'} \mathbf{h}$ sous des contraintes, qui peuvent être linéaires ou quadratiques (voir l'algorithme du paragraphe 1.6 et [64] pour plus de détails). Dans ce qui suit on aura besoin de plusieurs estimations indépendantes du canal. Chose qu'on pourra obtenir en utilisant différentes contraintes linéaires pour résoudre le problème de minimisation suivant

$$\begin{aligned} \hat{\mathbf{h}}^{(k)} &= \arg \min_{\mathbf{b}_k^H \mathbf{z} = \alpha_k} \mathbf{z}^H \mathbf{Q}_{L'} \mathbf{z} \\ &= \frac{\alpha_k \mathbf{Q}_{L'}^{-1} \mathbf{b}_k}{\mathbf{b}_k^H \mathbf{Q}_{L'}^{-1} \mathbf{b}_k}, \quad 1 \leq k \leq n_e, \end{aligned} \quad (2.29)$$

où \mathbf{b}_k est un vecteur-colonne et α_k un scalaire non-nul, n_e étant le nombre d'estimations indépendantes de \mathbf{h} .

2.6.1 Sélection de l'ordre du canal

Si le niveau du bruit est relativement faible, il s'en suit, et d'après le **Lemme** 2.2, que les différentes estimées de $h(z)$, par la méthode sous-espace, satisferont $\hat{h}^{(k)}(z) \approx r^{(k)}(z)h(z)$, $1 \leq k \leq n_e$, les polynômes $r^{(k)}(z)$ prennent des valeurs arbitraires. Dans le cas où $L' = L$, les polynômes $r^{(k)}(z)$ sont de degré zéro et les estimés $\hat{h}^{(k)}(z)$ tendent toutes vers $h(z)$ à une constante multiplicative arbitraire près. Ceci s'explique aussi par le fait que le noyau de la matrice $\mathbf{Q}_{L'}$ est un sous-espace de dimension un, généré par \mathbf{h} . Comme tous les $\hat{\mathbf{h}}^{(k)}$ appartiennent au noyau $\mathbf{Q}_{L'}$, quand $\sigma_b^2 \rightarrow 0$ ou $K \rightarrow +\infty$ (K étant le nombre d'échantillons), ce qui mène à conclure que dans ce cas tous les $\hat{\mathbf{h}}^{(k)}$ sont des estimations consistantes de \mathbf{h} . Si $L' < L$ ou $L' > L$ (qui est fréquemment le cas), les n_e estimations $\hat{h}^{(k)}(z)$ de $h(z)$ sont alors différentes, du moment que les polynômes $r^{(k)}(z)$ sont différents et dépendent des réalisations du bruit. Ce qu'on peut aussi expliquer par le fait que, si $L' < L$ alors la matrice $\mathbf{Q}_{L'}$ est de rang plein, donc pour différentes contraintes $\mathbf{b}_k^H \mathbf{h} \neq 0$ on aura des estimations $\hat{\mathbf{h}}^{(k)}$ différentes, et dans le cas $L' > L$, la matrice $\mathbf{Q}_{L'}$ est à rang déficient et son noyau est de dimension $L' - L + 1$, avec la présence du bruit, pour des contraintes linéaires différentes on obtient des estimations différentes de \mathbf{h} . Donc, si on considère la fonction $\mathcal{D}_{L'}(\hat{\mathbf{h}}^{(1)}, \dots, \hat{\mathbf{h}}^{(n_e)})$ comme étant une mesure de la proximité des estimations $\hat{\mathbf{h}}^{(k)}$ pour différentes valeurs de L' , e.g., pour $n_e = 2$, on peut prendre comme mesure de la proximité

la corrélation normalisée (cosinus de l'angle entre deux vecteurs) de $\hat{\mathbf{h}}^{(1)}$ et $\hat{\mathbf{h}}^{(2)}$, alors on sélectionne l'ordre du canal en choisissant la valeur de L' correspondant à la valeur maximale de $\mathcal{D}_{L'}(\hat{\mathbf{h}}^{(1)}, \dots, \hat{\mathbf{h}}^{(n_e)})$.

2.6.2 Algorithme

Pour simplifier, considérons seulement le cas $n_e = 2$. Choisissons d'abord des contraintes telles que $\mathbf{b}_k^H \mathbf{h} \neq 0$. Une solution à ce problème consiste à partitionner $\mathbf{U}_b = [\mathbf{U}_b^T(0) \cdots \mathbf{U}_b^T(N-1)]^T$, vu la structure particulière de la matrice de filtrage \mathbf{H}_N , l'équation $\mathbf{U}_b^H \mathbf{H}_N = \mathbf{0}$ implique $\mathbf{U}_b^H(0) \mathbf{h}(0) = \mathbf{0}$ et $\mathbf{U}_b^H(N-1) \mathbf{h}(L) = \mathbf{0}$. Comme les noyaux de $\mathbf{U}_b(0) \mathbf{U}_b^H(0)$ et $\mathbf{U}_b(N-1) \mathbf{U}_b^H(N-1)$ sont des sous-espaces de dimension un, il s'en suit que $\mathbf{g}_1^H \mathbf{h}(0) \neq 0$ et $\mathbf{g}_2^H \mathbf{h}(L) \neq 0$, où \mathbf{g}_1 et \mathbf{g}_2 sont les vecteurs propres des noyaux de $\mathbf{U}_b(0) \mathbf{U}_b^H(0)$ et $\mathbf{U}_b(N-1) \mathbf{U}_b^H(N-1)$ respectivement. Les contraintes \mathbf{b}_1 et \mathbf{b}_2 sont alors données par $\mathbf{b}_1 = [\mathbf{g}_1^T \quad \mathbf{0}_{1 \times pL}]^T$ et $\mathbf{b}_2 = [\mathbf{0}_{1 \times pL} \quad \mathbf{g}_2^T]^T$. Si on adopte comme mesure de la proximité des estimations $\hat{\mathbf{h}}^{(1)}$ et $\hat{\mathbf{h}}^{(2)}$ leur corrélation normalisée, alors la TABLE 2.10 résume l'algorithme sous-espace robuste (*Robust Subspace Fitting Algorithm*, RSFA) comme suit :

1. Spécifier les valeurs minimale et maximale possible de L , notées L_{min} et L_{max} respectivement.
2. Pour $L = L_{max}$, calculer \mathbf{C}_N , son sous-espace bruit \mathbf{U}_b et former la matrice $\mathcal{Q}_{L_{max}}$ (comme indiqué dans l'algorithme du paragraphe 1.6).
3. Calculer \mathbf{g}_1 et \mathbf{g}_2 (comme indiqué précédemment).
4. Pour $L' = L_{min} : L_{max}$
 - Calculer $\mathbf{b}_1 = [\mathbf{g}_1^T \quad \mathbf{0}_{1 \times pL'}]^T$, $\mathbf{b}_2 = [\mathbf{0}_{1 \times pL'} \quad \mathbf{g}_2^T]^T$, $\hat{\mathbf{h}}^{(1)} = \mathcal{Q}_{L'}^{-1} \mathbf{b}_1$ et $\hat{\mathbf{h}}^{(2)} = \mathcal{Q}_{L'}^{-1} \mathbf{b}_2$, où $\mathcal{Q}_{L'}$ est le $(L'+1)p \times (L'+1)p$ bloc supérieur de $\mathcal{Q}_{L_{max}}$, i.e., $\mathcal{Q}_{L'} = \mathcal{Q}_{L_{max}}(1 : (L'+1)p, 1 : (L'+1)p)$.
 - Calculer
$$\mathcal{D}_{L'}(\hat{\mathbf{h}}^{(1)}, \hat{\mathbf{h}}^{(2)}) = \frac{|\hat{\mathbf{h}}^{(1)H} \hat{\mathbf{h}}^{(2)}|}{\|\hat{\mathbf{h}}^{(1)}\| \|\hat{\mathbf{h}}^{(2)}\|}.$$
5. Sélectionner l'ordre optimal $L_{opt} = \arg \max_{L'} \mathcal{D}_{L'}(\hat{\mathbf{h}}^{(1)}, \hat{\mathbf{h}}^{(2)})$ et prendre $\hat{\mathbf{h}}^{(1)}$ qui lui correspond comme estimation de \mathbf{h} .

TAB. 2.10 – Algorithme RSFA.

2.7 Méthode de la décomposition en produit extérieur

Comme on a pu constater au paragraphe 2.2, l'algorithme LPA repose d'abord sur une estimation de l'élément matriciel $\mathbf{h}(0)$ de la réponse impulsionnelle du canal, via une décomposition de l'équation (2.15). Dans le cas où $\mathbf{h}(0)$ est faible (ce qui est généralement le cas), les effets du bruit et de la longueur

finie des données, rendent cette approche inefficace. C'est dans le but de pallier à cette carence que Z. Ding [26,27] proposa une approche, basée sur les statistiques d'ordre deux, et qui permet une formulation des coefficients du canal sous forme d'un produit extérieur, d'où le nom d'algorithme de décomposition en produit extérieur (*Outer-Product Decomposition Algorithm*, OPDA).

2.7.1 Produit extérieur

Définissons, dans le cas d'un canal MIMO, la matrice bloc-Hankel suivante, de dimension $(L+1)p \times (L+N)q$

$$\mathcal{H} = \begin{bmatrix} \mathbf{H}(0) & \mathbf{H}(1) & \cdots & \mathbf{H}(L) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{H}(1) & \mathbf{H}(2) & \cdots & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \cdots & \vdots \\ \mathbf{H}(L) & \mathbf{0} & \cdots & \mathbf{0} & \cdots & \cdots & \mathbf{0} \end{bmatrix},$$

il s'en suit que

$$\mathcal{H}\mathcal{H}^H = \begin{bmatrix} \sum_{k=0}^L \mathbf{H}(k)\mathbf{H}(k)^H & \sum_{k=0}^{L-1} \mathbf{H}(k)\mathbf{H}(k+1)^H & \cdots & \mathbf{H}(0)\mathbf{H}(L)^H \\ \sum_{k=1}^L \mathbf{H}(k)\mathbf{H}(k-1)^H & \sum_{k=1}^L \mathbf{H}(k)\mathbf{H}(k+1)^H & \cdots & \mathbf{H}(1)\mathbf{H}(L)^H \\ \sum_{k=2}^L \mathbf{H}(k)\mathbf{H}(k-2)^H & \sum_{k=2}^L \mathbf{H}(k)\mathbf{H}(k-2)^H & \cdots & \mathbf{H}(2)\mathbf{H}(L)^H \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{H}(L)\mathbf{H}(0)^H & \mathbf{H}(L)\mathbf{H}(1)^H & \cdots & \mathbf{H}(L)\mathbf{H}(L)^H \end{bmatrix}. \quad (2.30)$$

En utilisant la matrice de décalage définie par l'équation (2.1), on obtient

$$\mathbf{J}_{(L+1)p}^{-p} \mathcal{H}\mathcal{H}^H [\mathbf{J}_{(L+1)p}^{-p}]^T = \begin{bmatrix} \sum_{k=1}^L \mathbf{H}(k)\mathbf{H}(k)^H & \sum_{k=1}^{L-1} \mathbf{H}(k)\mathbf{H}(k+1)^H & \cdots & \mathbf{H}(1)\mathbf{H}(L)^H & \mathbf{0} \\ \sum_{k=2}^L \mathbf{H}(k)\mathbf{H}(k-1)^H & \sum_{k=2}^L \mathbf{H}(k)\mathbf{H}(k)^H & \cdots & \mathbf{H}(2)\mathbf{H}(L)^H & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{H}(L)\mathbf{H}(1)^H & \mathbf{H}(L)\mathbf{H}(2)^H & \cdots & \mathbf{H}(L)\mathbf{H}(L)^H & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \end{bmatrix}. \quad (2.31)$$

En soustrayant l'équation (2.31) de l'équation (2.30), on obtient

$$\begin{aligned} \Delta &= \mathcal{H}\mathcal{H}^H - \mathbf{J}_{(L+1)p}^{-p} \mathcal{H}\mathcal{H}^H [\mathbf{J}_{(L+1)p}^{-p}]^T \\ &= \begin{bmatrix} \mathbf{H}(0)\mathbf{H}(0)^H & \mathbf{H}(0)\mathbf{H}(1)^H & \cdots & \mathbf{H}(0)\mathbf{H}(L)^H \\ \mathbf{H}(1)\mathbf{H}(0)^H & \mathbf{H}(1)\mathbf{H}(1)^H & \cdots & \mathbf{H}(1)\mathbf{H}(L)^H \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{H}(L)\mathbf{H}(0)^H & \mathbf{H}(L)\mathbf{H}(1)^H & \cdots & \mathbf{H}(L)\mathbf{H}(L)^H \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{H}(0) \\ \mathbf{H}(1) \\ \vdots \\ \mathbf{H}(L) \end{bmatrix} \begin{bmatrix} \mathbf{H}(0)^H & \mathbf{H}(1)^H & \cdots & \mathbf{H}(L)^H \end{bmatrix} \\ &= \mathbf{H}\mathbf{H}^H, \end{aligned}$$

où \mathbf{H} est la matrice contenant les coefficients du canal, définie par

$$\mathbf{H} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{H}(0) \\ \mathbf{H}(1) \\ \vdots \\ \mathbf{H}(L) \end{bmatrix}.$$

La matrice Δ constitue alors, ce qui est appelé *produit extérieur* des paramètres de la matrice canal \mathbf{H} . Une matrice $(L+1)p \times q$ dont les colonnes constituent les vecteurs propres de Δ correspondant à ses q plus grandes valeurs propres, donne une estimation de \mathbf{H} à une matrice $q \times q$ non-singulière près.

2.7.2 Algorithme

Le point crucial lors du développement de l'algorithme est de pouvoir estimer le produit matriciel $\mathcal{H}\mathcal{H}^H$ à partir des statistiques du second ordre des signaux reçus. Si on adopte les notations et les définitions données dans les paragraphes 1.5 et 2.2.1, et on tient compte du fait que $\mathbf{C}_{i,j} = \mathbf{0}$, pour $|j| > L$ et $\mathbf{C}_{i,-j} = \mathbf{C}_{i,j}^H$, il sera alors facile de vérifier que la matrice d'autocorrélation $\mathbf{C}_{N,0}$ du vecteur de données $\mathbf{x}_N(t)$ s'écrit comme suit

$$\mathbf{C}_{N,0} = \begin{bmatrix} \mathbf{C}_{1,0} & \mathbf{C}_{1,1} & \cdots & \mathbf{C}_{1,L} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{C}_{1,1}^H & \mathbf{C}_{1,0} & \mathbf{C}_{1,1} & \cdots & \mathbf{C}_{1,L} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & \ddots & \mathbf{0} \\ \vdots & & & & & & \mathbf{C}_{1,L} & \\ \mathbf{C}_{1,L}^H & & & & & & \vdots & \\ \mathbf{0} & \ddots & & & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & & \mathbf{C}_{1,1}^H & \mathbf{C}_{1,0} & \mathbf{C}_{1,1} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{C}_{1,L}^H & \cdots & \mathbf{C}_{1,1}^H & \mathbf{C}_{1,0} \end{bmatrix} = \mathbf{H}_N \mathbf{H}_N^H + \sigma_b^2 \mathbf{I}_n \quad (2.32)$$

et que

$$\mathcal{C} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{C}_{1,0} - \sigma_b^2 \mathbf{I}_p & \mathbf{C}_{1,1} & \cdots & \mathbf{C}_{1,L} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{C}_{1,1} & \mathbf{C}_{1,2} & \cdots & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \cdots & \vdots \\ \mathbf{C}_{1,L} & \mathbf{0} & \cdots & \mathbf{0} & \cdots & \cdots & \mathbf{0} \end{bmatrix} = \mathcal{H} \mathcal{H}_N^H. \quad (2.33)$$

Noter que \mathbf{H}_N est une matrice à rang-colonne plein, alors l'égalité suivante est vérifiée

$$\mathbf{H}_N^H (\mathbf{H}_N \mathbf{H}_N^H)^{\#} \mathbf{H}_N = \mathbf{I}_d,$$

ce qui mène à déduire une estimation de $\mathcal{H}\mathcal{H}^H$ à partir de

$$\begin{aligned} \mathcal{C} (\mathbf{C}_{N,0} - \sigma_b^2 \mathbf{I}_n)^{\#} \mathcal{C}^H &= \mathcal{H} \mathbf{H}_N^H (\mathbf{H}_N \mathbf{H}_N^H)^{\#} \mathbf{H}_N \mathcal{H}^H \\ &= \mathcal{H} \mathcal{H}^H. \end{aligned}$$

Ainsi la TABLE 2.11 résume l'algorithme OPDA.

1. Pour $\tau = 0 \cdot L$, calculer $\mathbf{C}_{1,\tau} = \frac{1}{K} \sum_{k=\tau}^{K+\tau-1} \mathbf{x}(k)\mathbf{x}^H(k-\tau)$.
2. Former les matrices $\mathbf{C}_{N,0}$ et \mathbf{C} d'après les relations (2.32) et (2.33) respectivement.
3. Calculer $\mathbf{\Delta} = \mathbf{C}(\mathbf{C}_{N,0} - \sigma_b^2 \mathbf{I}_n) \# \mathbf{C}^H - \mathbf{J}_{(L+1)p}^{-p} \mathbf{C}(\mathbf{C}_{N,0} - \sigma_b^2 \mathbf{I}_n) \# \mathbf{C}^H [\mathbf{J}_{(L+1)p}^{-p}]^T$
4. Les colonnes de la matrice \mathbf{H} correspondent au q vecteurs propres dominants de la matrice $\mathbf{\Delta}$.

TAB. 2.11 – *Algorithme OPDA.*

2.8 Méthode de lissage au sens des moindres carrés

Comme pour la technique OPDA (voir paragraphe 2.7), la technique de lissage au sens des moindres carrés (*Least Square Smoothing*, LSS), proposée par L. Tong et Q. Zhao [75], estime la réponse impulsionnelle du canal SIMO après l'avoir exprimé sous forme d'un produit extérieur. On montre dans [75] que l'approche LSS est applicable au cas déterministe aussi bien qu'au cas stochastique. Dans le cas déterministe l'hypothèse qui stipule que les signaux d'entrée sont temporellement blancs peut être relaxer. Un autre avantage de cette approche, c'est qu'elle permet d'estimer conjointement le canal et son ordre, ce qui lui procure une certaine robustesse vis à vis des erreurs de surestimation de l'ordre. De même une implémentation adaptative efficace à été proposé dans le cas déterministe [76]. Nous présentons ici la version stochastique dans le cas d'un canal SIMO.

2.8.1 Filtre de lissage

L'idée de base du lissage consiste à concevoir un estimateur linéaire du vecteur des signaux reçus actuels à partir de données passés et futures. Commençons d'abord par former les vecteurs de données reçus actuels, futures et passés

$$\mathbf{x}^a(t) \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}(t-1) \\ \vdots \\ \mathbf{x}(t-L') \end{bmatrix}; \quad \mathbf{x}^f(t) \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{x}(t+L') \\ \mathbf{x}(t+L'-1) \\ \vdots \\ \mathbf{x}(t+1) \end{bmatrix}; \quad \mathbf{x}^p(t) \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{x}(t-L'-1) \\ \mathbf{x}(t-L'-2) \\ \vdots \\ \mathbf{x}(t-2L') \end{bmatrix},$$

où L' est la taille de la fenêtre d'observation, choisie supérieure à L . D'après les relations d'entrée/sortie du modèle de données (voir équations (1.4), (1.5) et (1.6)), on obtient

$$\begin{aligned} \mathbf{x}^a(t) &= \mathbf{x}_{L'+1}(t) \\ &= \mathbf{H}_{L'+1} \mathbf{s}_{L'+1}(t) + \mathbf{b}_{L'+1}(t), \end{aligned} \tag{2.34}$$

$$\begin{aligned} \mathbf{x}^f(t) &= \mathbf{x}_{L'}(t+L') \\ &= \mathbf{H}_{L'} \mathbf{s}_{L'}(t+L') + \mathbf{b}_{L'}(t+L'), \end{aligned} \tag{2.35}$$

$$\begin{aligned} \mathbf{x}^p(t) &= \mathbf{x}_{L'}(t-L'-1) \\ &= \mathbf{H}_{L'} \mathbf{s}_{L'}(t-L'-1) + \mathbf{b}_{L'}(t-L'-1). \end{aligned} \tag{2.36}$$

Le principe du lissage au sens des moindres carrés est d'estimer le vecteur $\mathbf{x}^a(t)$ à partir des vecteurs $\mathbf{x}^f(t)$ et $\mathbf{x}^p(t)$. Soit $\mathbf{B}_{opt} = \begin{bmatrix} \mathbf{B}_{opt}^f & \mathbf{B}_{opt}^p \end{bmatrix}$ le filtre d'estimation linéaire donné par

$$\begin{cases} \mathbf{B}_{opt} = \arg \min_{\mathbf{B}} E(\|\boldsymbol{\varepsilon}(t)\|^2), \\ \boldsymbol{\varepsilon}(t) = \mathbf{x}^a(t) - \mathbf{B} \begin{bmatrix} \mathbf{x}^f(t) \\ \mathbf{x}^p(t) \end{bmatrix}. \end{cases} \quad (2.37)$$

La solution du problème (2.37) est alors donnée par

$$\mathbf{B}_{opt}^f = \mathbf{M}_f \mathbf{C}_{L'}^{-1}; \quad \mathbf{B}_{opt}^p = \mathbf{M}_p \mathbf{C}_{L'}^{-1},$$

où

$$\begin{aligned} \mathbf{C}_{L'} &\stackrel{\text{def}}{=} E(\mathbf{x}_{L'}(t) \mathbf{x}_{L'}^H(t)) \\ &= \mathbf{H}_{L'} \mathbf{H}_{L'}^H + \sigma_b^2 \mathbf{I}_{L'p}, \\ \mathbf{M}_f &\stackrel{\text{def}}{=} E(\mathbf{x}_{L'+1}(t-L') \mathbf{x}_{L'}^H(t)) \\ &= \mathbf{H}_{L'+1} \mathcal{J}_f \mathbf{H}_{L'}^H, \\ \mathbf{M}_p &\stackrel{\text{def}}{=} E(\mathbf{x}_{L'+1}(t) \mathbf{x}_{L'}^H(t-L'-1)) \\ &= \mathbf{H}_{L'+1} \mathcal{J}_p \mathbf{H}_{L'}^H, \\ \mathcal{J}_f &= \begin{bmatrix} \mathbf{0}_{L \times L'} & \mathbf{I}_L \\ \mathbf{0}_{(L'+1) \times L'} & \mathbf{0}_{(L'+1) \times L} \end{bmatrix}, \\ \mathcal{J}_p &= \begin{bmatrix} \mathbf{0}_{(L'+1) \times L} & \mathbf{0}_{(L'+1) \times L'} \\ \mathbf{I}_L & \mathbf{0}_{L \times L'} \end{bmatrix}. \end{aligned}$$

L'erreur d'estimation est alors donnée par

$$\boldsymbol{\varepsilon}_{min}(t) = \mathbf{x}^a(t) - \mathbf{B}_{opt}^f \mathbf{x}^f(t) - \mathbf{B}_{opt}^p \mathbf{x}^p(t),$$

tenant compte de (2.35) et de (2.36), on obtient

$$\begin{aligned} \mathbf{x}^a(t) &= \mathbf{B}_{opt}^f \mathbf{H}_{L'} \mathbf{s}_{L+L'}(t+L') + \mathbf{B}_{opt}^p \mathbf{H}_{L'} \mathbf{s}_{L+L'}(t-L'-1) \\ &\quad + \boldsymbol{\varepsilon}_{min}(t) + \mathbf{B}_{opt}^f \mathbf{b}_{L'}(t+L') + \mathbf{B}_{opt}^p \mathbf{b}_{L'}(t-L'-1). \end{aligned} \quad (2.38)$$

D'autre part, si on partitionne la matrice de filtrage bloc toeplitz $\mathbf{H}_{L'+1}$ (voir équation (1.7)) telle que

$$\mathbf{H}_{L'+1} = \begin{bmatrix} \mathcal{F}' & \mathcal{F} & \mathcal{F}'' \end{bmatrix},$$

les blocs $\mathcal{F}' \in \mathbb{C}^{(L'+1)p \times L}$, $\mathcal{F}'' \in \mathbb{C}^{(L'+1)p \times L}$ et $\mathcal{F} \in \mathbb{C}^{(L'+1)p \times (L'-L+1)}$ sont alors donnés comme suit

$$\mathcal{F}' = \begin{bmatrix} \mathbf{h}(0) & \cdots & \mathbf{h}(L-1) \\ \mathbf{0} & \ddots & \vdots \\ & \ddots & \mathbf{h}(0) \\ \vdots & & \mathbf{0} \\ & & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix},$$

$$\mathcal{F}'' = \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & & \\ \mathbf{0} & & \vdots \\ \mathbf{h}(L) & \ddots & \\ \vdots & \ddots & \mathbf{0} \\ \mathbf{h}(1) & \cdots & \mathbf{h}(L) \end{bmatrix},$$

$$\mathcal{F} = \begin{bmatrix} \mathbf{h}(L) & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{h}(0) & & \ddots & \mathbf{0} \\ \mathbf{0} & \ddots & & \mathbf{h}(L) \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{h}(0) \end{bmatrix},$$

il s'en suit que

$$\begin{aligned} \mathbf{x}^a(t) &= \mathcal{F}' \begin{bmatrix} \mathbf{s}(t) \\ \mathbf{s}(t-1) \\ \vdots \\ \mathbf{s}(t-L+1) \end{bmatrix} + \mathcal{F}'' \begin{bmatrix} \mathbf{s}(t-L'-1) \\ \mathbf{s}(t-L'-2) \\ \vdots \\ \mathbf{s}(t-L'-L) \end{bmatrix} + \mathcal{F} \begin{bmatrix} \mathbf{s}(t-L) \\ \mathbf{s}(t-L-1) \\ \vdots \\ \mathbf{s}(t-L') \end{bmatrix} \\ &\quad + \mathbf{b}_{L'+1}(t) \\ &= \begin{bmatrix} \mathbf{0}_{(L'+1)p \times L'} & \mathcal{F}' \end{bmatrix} \mathbf{s}_{L+L'}(t+L') + \begin{bmatrix} \mathcal{F}'' & \mathbf{0}_{(L'+1)p \times L'} \end{bmatrix} \mathbf{s}_{L+L'}(t-L'-1) \\ &\quad + \mathcal{F} \mathbf{s}_{L'-L+1}(t-L) + \mathbf{b}_{L'+1}(t). \end{aligned} \quad (2.39)$$

En comparant (2.38) et (2.39), on en déduit

$$\varepsilon_{min}(t) = \mathcal{F} \mathbf{s}_{L'-L+1}(t-L) + \mathbf{b}_{L'+1}(t) - \mathbf{B}_{opt}^f \mathbf{b}_{L'}(t+L') - \mathbf{B}_{opt}^p \mathbf{b}_{L'}(t-L'-1).$$

2.8.2 Algorithme pour l'estimation du canal et la détection de son ordre

Soit \mathfrak{F} la matrice de dimension $(L'+1)p \times (L'+1)p$, définie par

$$\mathfrak{F} \stackrel{\text{def}}{=} E(\varepsilon_{min}(t) \varepsilon_{min}^H(t)) - \sigma_b^2 \mathbf{I}_{(L'+1)p} - \sigma_b^2 \mathbf{B}_{opt} \mathbf{B}_{opt}^H = \mathcal{F} \mathcal{F}^H,$$

vu la structure de \mathcal{F} , il est clair que \mathfrak{F} est une matrice à rang déficient, i.e., $\text{rank}(\mathfrak{F}) = (L' - L + 1)$. Par conséquent, le noyau de \mathfrak{F} , i.e., $\text{null}(\mathfrak{F}) = \{\mathbf{z} \in \mathbb{C}^{(L'+1)p} : \mathfrak{F}\mathbf{z} = \mathbf{0}\}$ est un sous-espace vectoriel de dimension $d = (L' + 1)p - (L' - L + 1)$. soit $\mathbf{U} \in \mathbb{C}^{(L'+1)p \times d}$ une matrice dont les colonnes forment une base orthonormale de $\text{null}(\mathfrak{F})$, par conséquent $\mathfrak{F}\mathbf{U} = \mathbf{0}$, ce qui implique

$$\mathbf{U}^H \mathcal{F} = \mathbf{0}. \quad (2.40)$$

Si on partitionne la matrice \mathbf{U} en $L' + 1$ blocs de p lignes et de d colonnes, i.e.,

$$\mathbf{U} = \begin{bmatrix} \mathbf{U}^T(0) & \mathbf{U}^T(1) & \cdots & \mathbf{U}^T(L') \end{bmatrix}^T,$$

alors d'après (2.40) on a l'équivalence suivante

$$\begin{bmatrix} \mathbf{U}^H(0) & \mathbf{U}^H(1) & \cdots & \mathbf{U}^H(L') \end{bmatrix} \begin{bmatrix} \mathbf{h}(L) & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{h}(0) & & \ddots & \mathbf{0} \\ \mathbf{0} & \ddots & & \mathbf{h}(L) \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{h}(0) \end{bmatrix} = \mathbf{0}$$

$$\Downarrow$$

$$\mathfrak{U}_L^H \check{\mathbf{h}} = \mathbf{0}, \quad (2.41)$$

avec

$$\mathfrak{U}_k = \begin{bmatrix} \mathbf{U}(0) & \mathbf{U}(1) & \cdots & \mathbf{U}(L' - k) \\ \mathbf{U}(1) & \mathbf{U}(2) & \cdots & \mathbf{U}(L' - k + 1) \\ \vdots & \vdots & \cdots & \vdots \\ \mathbf{U}(k) & \mathbf{U}(k + 1) & \cdots & \mathbf{U}(L') \end{bmatrix}; \quad \check{\mathbf{h}} = \begin{bmatrix} \mathbf{h}(L) \\ \mathbf{h}(L - 1) \\ \vdots \\ \mathbf{h}(0) \end{bmatrix}. \quad (2.42)$$

En d'autres termes, les coefficients du canal vérifient l'équation linéaire homogène (2.41). Reste à démontrer que la solution de (2.41) est unique à une constante multiplicative non-nulle près. Or le théorème suivant (voir [75] pour la démonstration) nous permettra d'établir un algorithme qui joint à l'estimation du canal la détection de son ordre.

Théorème 2.5 Soit \mathfrak{U}_k la matrice définie dans (2.42) pour $0 \leq k < L'$. L'équation linéaire homogène

$$\mathfrak{U}_k^H \mathbf{z} = \mathbf{0}, \quad \mathbf{z} \in \mathbb{C}^{(k+1)p}, \quad (2.43)$$

admet une solution non-triviale unique dans le cas où $k = L$, cette solution correspond au coefficients du canal $\check{\mathbf{h}}$ (à une constante multiplicative scalaire non-nulle près), i.e.,

$$\mathbf{z} = r\check{\mathbf{h}},$$

où r est une constante scalaire non-nulle donnée. Dans le cas où $k \neq L$, l'équation (2.43) n'admet que des solutions triviales.

La TABLE 2.12 résume l'algorithme LSS.

$$\begin{aligned}
 \mathbf{C}_{L'} &= \frac{1}{K} \sum_{k=L'-1}^{K+L'-1} \mathbf{x}_{L'}(k) \mathbf{x}_{L'}^H(k), \quad (K : \text{Nombre d'échantillons}) \\
 \mathbf{M}_f &= \frac{1}{K} \sum_{k=2L'}^{K+2L'} \mathbf{x}_{L'+1}(k-L') \mathbf{x}_{L'}^H(k) \\
 \mathbf{M}_p &= \frac{1}{K} \sum_{k=2L'}^{K+2L'} \mathbf{x}_{L'+1}(k) \mathbf{x}_{L'}^H(k-L'-1) \\
 \mathbf{B}_{opt} &= \begin{bmatrix} \mathbf{M}_f \mathbf{C}_{L'}^{-1} & \mathbf{M}_p \mathbf{C}_{L'}^{-1} \end{bmatrix} \\
 \boldsymbol{\varepsilon}_{min}(t) &= \mathbf{x}_{L'+1}(t-L') - \mathbf{B}_{opt} \begin{bmatrix} \mathbf{x}_{L'}(t) \\ \mathbf{x}_{L'}(t-2L'-1) \end{bmatrix} \\
 \mathfrak{F} &= \frac{1}{K} \sum_{k=3L'}^{K+3L'} \boldsymbol{\varepsilon}_{min}(k) \boldsymbol{\varepsilon}_{min}^H(k) - \sigma_b^2 \mathbf{I}_{(L'+1)p} - \sigma_b^2 \mathbf{B}_{opt} \mathbf{B}_{opt}^H \\
 (\mathbf{E}, \boldsymbol{\Sigma}, \mathbf{V}) &= \text{svd}(\mathfrak{F}) \\
 n' &= (L'+1)p \\
 \text{For } k &= 0, \dots, L'-1 : \\
 d' &= L' - k + 2 \\
 \mathbf{U} &= \begin{bmatrix} \mathbf{U}(0) \\ \vdots \\ \mathbf{U}(L') \end{bmatrix} \\
 &= \mathbf{E}(:, d' : n') \\
 \boldsymbol{\mathfrak{U}}_k &= \begin{bmatrix} \mathbf{U}(0) & \mathbf{U}(1) & \dots & \mathbf{U}(L'-k) \\ \mathbf{U}(1) & \mathbf{U}(2) & \dots & \mathbf{U}(L'-k+1) \\ \vdots & \vdots & \dots & \vdots \\ \mathbf{U}(k) & \mathbf{U}(k+1) & \dots & \mathbf{U}(L') \end{bmatrix} \\
 \mathbf{z}_k &= \arg \min_{\|\mathbf{z}\|=1} (\|\boldsymbol{\mathfrak{U}}_k^H \mathbf{z}\|^2) \\
 \text{end} \\
 \hat{L} &= \arg \min_k (\|\boldsymbol{\mathfrak{U}}_k^H \mathbf{z}_k\|^2) \\
 \check{\mathbf{h}} &= \mathbf{z}_{\hat{L}} \quad (\text{à une constante multiplicative scalaire non-nulle près})
 \end{aligned}$$

TAB. 2.12 – Algorithmes LSS.

2.9 Complexité de calcul

Dans ce paragraphe, nous présentons une comparaison des complexités de calcul des différentes approches exposées ci-dessus (voir TABLE 2.13), dans le cas SIMO. Le nombre d'opérations donné pour chaque implémentation, correspond au nombre de multiplications complexes des étapes prépondérantes de l'algorithme. Pour les implémentations adaptatives, le coût de calcul donné correspond au nombre de multiplications complexes par itération. Pour la version adaptative de l'algorithme GSC, le coût de calcul donné par la TABLE 2.13 correspond à sa version via la méthode du gradient.

<i>Algorithme</i>	<i>Nombre total de multiplications complexes (implémentation en bloc)</i>	<i>Nombre de multiplications complexes par itération (implémentation adaptative)</i>
SSA	$(K + d + 1)n^2 + dd'n'^2 + O(n^3) + O(n'^3)$	
LPA	$3n^3 + (2K + d + p + 1)n^2 - (2d + p - 3)np + dp^2 + p^3 + O(n^3) + O(p^2)$	$3n^2 + 4pn + O(p^3)$
MRE	$(2K + d^2)n^2 + nd + O(d^3n^3)$	$6nd$
GRDA	$(3K + d')n^2 + (d'^2 + d')n + 4O(n^3) + O(d'^3)$	
Capon	$Kn^2 + n'n + (n' + 1)n' + O(n^3) + O(n'^3)$	
GSC	$Kn^2 + n'n + O(n^3) + 2O(n'^3)$	$(n' + 2)n + n'^2$
RSFA	$(K + d + 1)n^2 + (2(L + 1) + dd')n'^2 + 3(L + 1)n' + p^2d' + O(n^3) + (L + 1)O(n'^3) + 2O(p^3)$	
OPDA	$(K + 2n' + d + 1)n^2 + 2O(n^3) + O(n'^2)$	
LSS	$2K(2n + p)(n + p) + (N + 4)n^3 + (5p + d + 1)n^2 + p(2p + 1)n + 2O((n + p)^3) + O(n^3)$	

TAB. 2.13 – Comparaison des coûts de calcul.

On rappelle que :

- K est le nombre d'échantillons utilisés pour estimer les statistiques.
- $n = Np$ est la taille du vecteur de données $\mathbf{x}_N(t)$, où N est la taille de la fenêtre d'observation et p est le nombre de capteurs.
- $d = N + L$ est la dimension du sous-espace signal, L étant l'ordre du canal de transmission.
- $d' = n - d$ est la dimension du sous-espace bruit.
- $n' = (L + 1)p$ est la dimension du vecteur qui contient les coefficients de la réponse impulsionnelle du canal de transmission.

2.10 Simulations

Dans ce paragraphe, nous présentons quelques exemples de simulation afin d'évaluer et de comparer les performances des différentes approches déjà présentées. Comme pour les expérimentations présentées

au paragraphe 1.6.5, on considère un canal SIMO ($q = 1, p = 3$ et $L = 3$), dont Les coefficients sont générés aléatoirement selon une distribution gaussienne complexe. Le signal d'entrée est une séquence iid, appartenant à un alphabet fini de phase et d'amplitude à quatre états QAM4. Comme indice de performance, on estime l'erreur quadratique moyenne (*Mean Square Error*, MSE) par

$$\text{MSE} = \frac{1}{K} \sum_{t=\tau}^{K+\tau+1} |s(t - \tau) - \hat{\mathbf{v}}_{\tau}^H \mathbf{x}_N(t)|^2,$$

moyenné sur 100 réalisations indépendantes. $\hat{\mathbf{v}}_{\tau}$ étant une estimation échantillonnée (obtenue à partir de K échantillons) du filtre égaliseur \mathbf{v}_{τ} (obtenu à partir de statistiques exactes). Noter que dans le cas d'une égalisation indirecte (comme c'est le cas des algorithmes RSFA, OPDA et LSS), et après estimation des coefficients du canal, on calcul le filtre égaliseur \mathbf{v}_{τ} en utilisant, respectivement, l'expression (1.21) dans le cas d'une égalisation MMSE, et l'expression (1.23) dans le cas d'une égalisation ZF. Le rapport signal à bruit (SNR : Signal to Noise Ration) est défini par $\text{SNR} = -20 \log(\sigma_b)$.

La Fig. 2.1 compare les performances des différentes approches proposées. Les courbes représentent

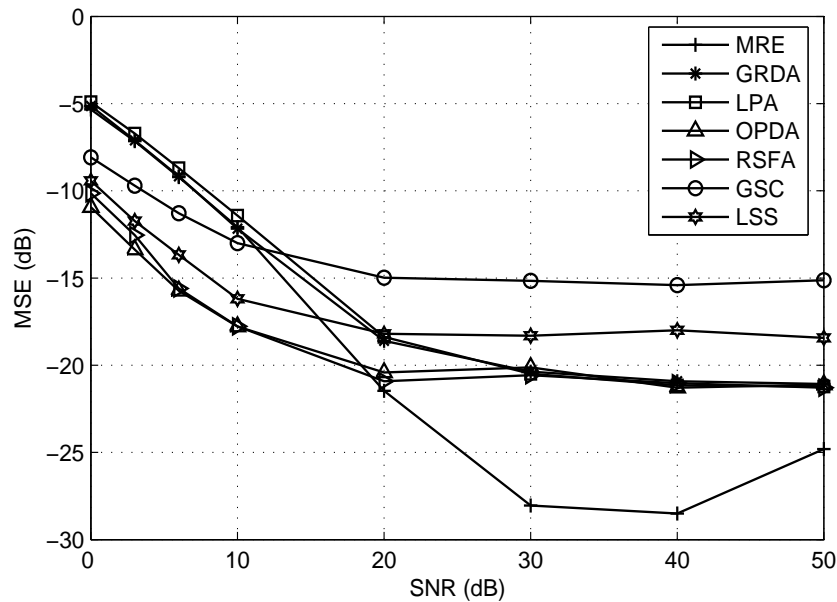


FIG. 2.1 – Comparaison des performances des égaliseurs.

le MSE estimé en fonction du SNR, pour un nombre d'échantillons $K = 500$ et une fenêtre d'observation de largeur $N = 6$. On prend un retard d'égalisation $\tau = 4$ (du moment que de meilleurs performances sont obtenues en choisissant τ dans $\{L, L + 1, \dots, N - 1\}$, voir [70] pour plus de détails). Pour les approches LPA, RSFA, OPDA et LSS, nous avons représenté dans la Fig. 2.1 les performances de l'égaliseur MMSE (des résultats semblables sont obtenues à l'aide de l'égaliseur ZF). Comme on peut l'observer, l'approche MRE donne de meilleurs performances pour un fort SNR. Tandis que pour un faible SNR, les algorithmes RSFA et OPDA donnent de meilleurs résultats que les autres.

Dans la Fig 2.2, on donne une comparaison des complexités de calcul pour les différentes approches

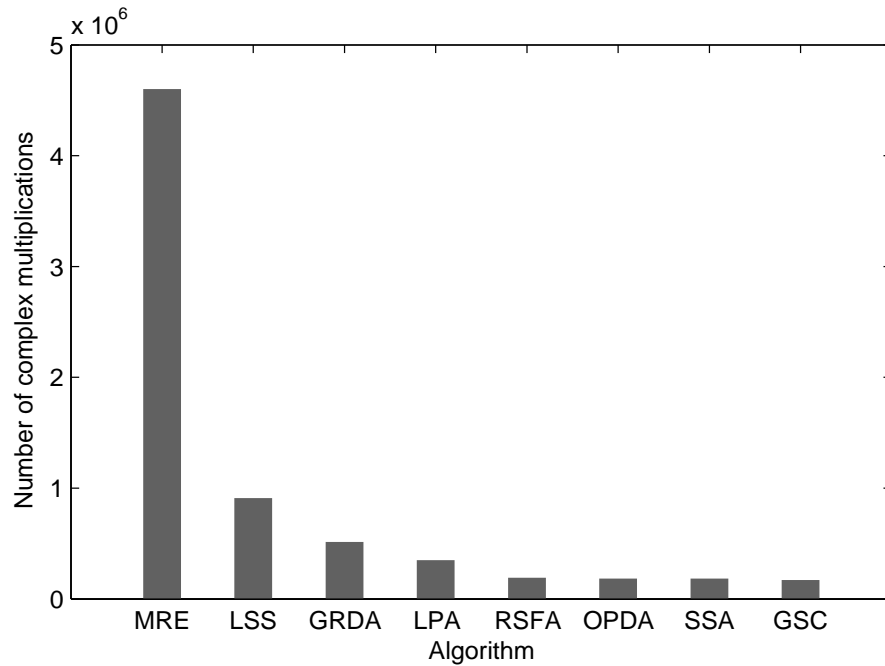


FIG. 2.2 – Comparaison des coûts de calcul.

présentées, il est clair que l'approche MRE est la plus coûteuse, alors que les approches RSFA, OPDA, SSA et GSC restent les moins coûteuses.

La Fig. 2.1 compare la robustesse aux erreurs de surestimation de l'ordre, des différentes approches

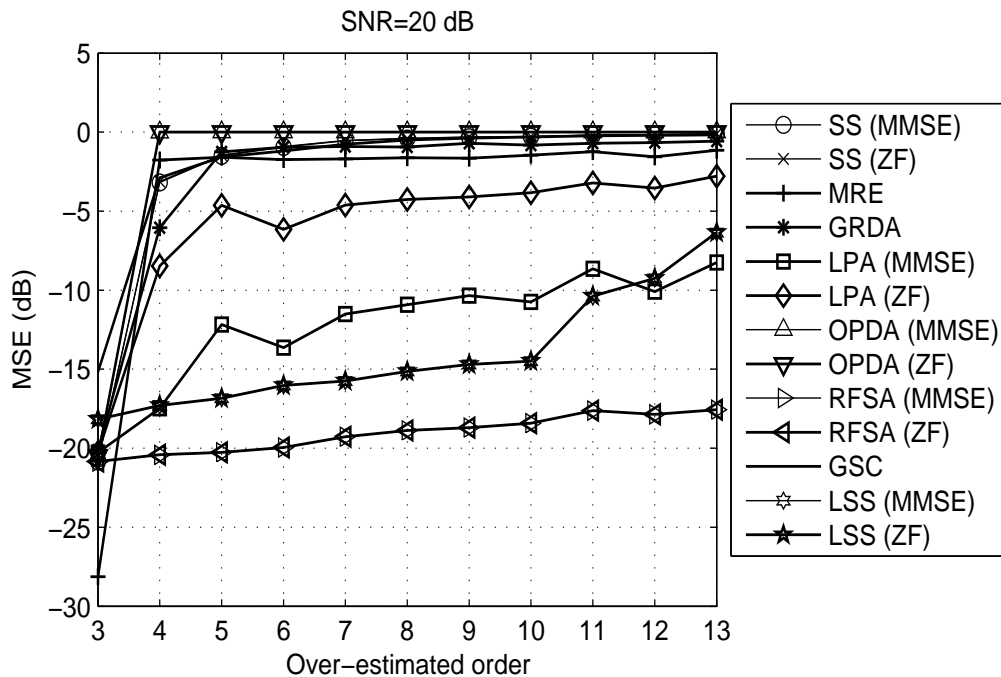


FIG. 2.3 – Robustesse aux erreurs de surestimation de l'ordre, l'ordre exacte $L=3$, $SNR=20$ dB.

étudiées. Les courbes représentent le MSE estimé en fonction de l'ordre surestimé du canal (l'ordre exacte du canal étant $L=3$), pour un nombre d'échantillons $K = 500$. On voit bien que dans ces conditions, les meilleures performances sont données par l'algorithme RSFA, qui est pratiquement insensible aux erreurs de surestimation de l'ordre du canal, vu qu'il permet aussi de détecter l'ordre exacte du canal. Même si l'algorithme LSS permet de détecter l'ordre exacte du canal, néanmoins ses résultats sont moins fiable que ceux de l'algorithme RSFA. Pour les autres approches (et qui n'incluent pas de procédure de détection de l'ordre exacte du canal), le filtre égaliseur MMSE basé sur l'approche LPA donne de meilleurs résultats.

2.11 Conclusion

Dans ce chapitre, nous avons rassemblé les principales techniques d'identification / égalisation aveugle basées sur des statistiques du second ordre et qui se veulent robustes aux erreurs de surestimation de l'ordre du canal de transmission. Certaines de ces approches (e.g. les algorithmes LPA, MRE, GRDA, Capon et GSC) estiment directement le filtre d'égalisation, alors que les autres approches (e.g. les algorithmes RSFA, OPDA et LSS) identifient d'abord les coefficients du canal, pour construire ensuite le filtre d'égalisation (MMSE ou ZF). Une implémentation adaptative rapide des algorithmes LPA, MRE, Capon et GSC a été donnée. Une étude comparative, des coûts de calcul des implémentations en bloc et adaptative des différentes approches étudiées a été présentée. Enfin, des exemples de simulation ont été donnés pour évaluer les performances des différentes techniques étudiées et de tester leur robustesse aux erreurs de surestimation de l'ordre du canal.

Deuxième partie

Développements algorithmiques et étude des performances

Chapitre 3

Égaliseur MMSE MIMO aveugle adaptatif rapide et robuste aux erreurs de surestimation de l'ordre du canal

3.1 Introduction

Dans ce chapitre nous développons un algorithme d'égalisation adaptatif aveugle du second ordre, qui présente deux avantages majeurs à savoir, une robustesse aux erreurs de surestimation de l'ordre et un faible coût de calcul. Plus précisément, il sera question d'une nouvelle technique MMSE d'égalisation directe de systèmes MIMO à réponse impulsionnelle finie, ayant une complexité de calcul de l'ordre de $O(qnd)$ (q étant le nombre de sources, n la dimension du vecteur de données et d la dimension du sous-espace signal) et une faible sensibilité aux erreurs de modélisation du canal. On montre que les colonnes du filtre égaliseur matriciel à retard zéro appartiennent, simultanément, au sous-espace signal et au noyau d'une matrice issue de la troncature de la matrice d'autocorrélation. Cette propriété nous conduit à concevoir une méthode simple pour l'estimation du filtre égaliseur, après minimisation d'une certaine forme quadratique soumise à des contraintes bien appropriées. On présente une implémentation adaptative rapide de ce nouvel algorithme, incluant une procédure à deux étapes qui nous permet de compenser la baisse des performances comparées au cas non-aveugle, et de choisir des retards d'égalisation non-nuls. De même, on effectue une analyse asymptotique des performances de notre méthode, qui aboutit à la détermination de l'expression théorique caractérisant la baisse en performances due au processus aveugle d'égalisation.

3.2 Développement de l'algorithme

3.2.1 Égaliseur MMSE

Considérons le modèle de données du canal MIMO décrit au paragraphe 1.5. Tenant compte des hypothèses faites sur ce modèle, l'expression de la matrice d'égalisation MMSE à retard τ ($\tau \in \{0, 1, \dots, m-1\}$) est alors donnée par

$$\begin{aligned} \mathbf{V}_\tau &= \arg \min_{\mathbf{V}} E(\|\mathbf{s}(t-\tau) - \mathbf{V}^H \mathbf{x}_N(t)\|^2) \\ &= \mathbf{C}_N^{-1} \mathbf{G}_\tau, \end{aligned} \quad (3.1)$$

où \mathbf{C}_N est la matrice d'autocorrélation donnée par les équations (1.8) et (1.9), \mathbf{G}_τ est une matrice $n \times q$ donnée par

$$\begin{aligned} \mathbf{G}_\tau &\stackrel{\text{def}}{=} E(\mathbf{x}_N(t) \mathbf{s}^H(t-\tau)) \\ &= \mathbf{H}_N \mathcal{J}_{q\tau, q, q(m-\tau-1)}, \end{aligned} \quad (3.2)$$

$\mathcal{J}_{j,k,l}$ est une matrice de troncation définie comme suit

$$\mathcal{J}_{j,k,l} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{0}_{j \times k} \\ \mathbf{I}_k \\ \mathbf{0}_{l \times k} \end{bmatrix}. \quad (3.3)$$

Noter que $\mathbf{H}_N \mathcal{J}_{q\tau, q, q(m-\tau-1)}$ désigne une sous-matrice de \mathbf{H}_N donnée par les vecteurs-colonne dont les indices appartiennent à l'intervalle $[\tau q + 1, \dots, (\tau + 1)q]$. Des équations (3.1), (1.9) et (3.2) et en utilisant le lemme d'inversion matricielle [46], on arrive aussi à exprimer la matrice \mathbf{V}_τ par

$$\mathbf{V}_\tau = \mathbf{H}_N \bar{\mathbf{V}}_\tau,$$

où $\bar{\mathbf{V}}_\tau$ est une matrice $d \times q$ donnée par

$$\bar{\mathbf{V}}_\tau = \frac{1}{\sigma_b^2} \left(\mathbf{I}_d - \frac{1}{\sigma_b^4} (\sigma_b^2 \mathbf{I}_d + \mathbf{H}_N^H \mathbf{H}_N)^{-1} \mathbf{H}_N^H \mathbf{H}_N \right) \mathcal{J}_{q\tau, q, q(m-\tau-1)}.$$

Il est clair que les colonnes du filtre matriciel MMSE \mathbf{V}_τ appartiennent au *sous-espace signal* (i.e. $\text{range}(\mathbf{H}_N)$), alors on peut écrire

$$\mathbf{V}_\tau = \mathbf{W} \tilde{\mathbf{V}}_\tau, \quad (3.4)$$

où \mathbf{W} est une matrice $n \times d$ dont les colonnes forment une base orthonormale du sous-espace signal (il existe une matrice \mathbf{P} de dimension $d \times d$ telle que $\mathbf{W} = \mathbf{H}_N \mathbf{P}$) et $\tilde{\mathbf{V}}_\tau$ est une matrice $d \times q$.

3.2.2 Égalisation aveugle

Notre objectif ici est d'arriver à une estimation aveugle de l'égaliseur MMSE à retard zéro \mathbf{V}_0 . Des équations (3.1), (3.2), (3.3) and (3.4), on peut écrire $\mathbf{V}_0 = \mathbf{W}\tilde{\mathbf{V}}_0$, avec

$$\mathbf{C}_N \mathbf{W} \tilde{\mathbf{V}}_0 = \begin{bmatrix} \mathbf{H}(0) \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}. \quad (3.5)$$

Si on tronque les p premières lignes du système (3.5), on obtient alors

$$\mathbf{T} \tilde{\mathbf{V}}_0 = \mathbf{0}, \quad (3.6)$$

où \mathbf{T} est une matrice $(n-p) \times d$ donnée par

$$\mathbf{T} \stackrel{\text{def}}{=} \bar{\mathbf{C}} \mathbf{W} \quad (3.7)$$

$$\bar{\mathbf{C}} = \mathbf{C}_N(p+1:n,:) = \mathcal{J}_{p,n-p,0}^T \mathbf{C}_N. \quad (3.8)$$

La matrice $\bar{\mathbf{C}}$ est une sous-matrice de \mathbf{C} donnée par ses $n-p$ dernières lignes. L'équation (3.6) montre que les colonnes de $\tilde{\mathbf{V}}_0$ appartiennent au noyau droit de \mathbf{T} ($\text{null}_r(\mathbf{T}) = \{\mathbf{z} \in \mathbb{C}^d : \mathbf{T}\mathbf{z} = \mathbf{0}\}$). Inversement, on peut établir que l'équation (3.6) caractérise de façon bi-univoque l'égaliseur MMSE à retard zéro. Dans ce cas on a le résultat suivant

Théorème 3.1 *Tenant compte des hypothèses ci-dessus et pour $N > qL + 1$ la solution de l'équation homogène*

$$\mathbf{T} \tilde{\mathbf{V}} = \mathbf{0}, \quad (3.9)$$

soumise à la contrainte

$$\text{rank}(\tilde{\mathbf{V}}) = q, \quad (3.10)$$

est unique et correspond à l'égaliseur MMSE désiré (à une matrice $q \times q$ non-singulière près), i.e.,

$$\tilde{\mathbf{V}} = \tilde{\mathbf{V}}_0 \mathbf{R}, \quad (3.11)$$

\mathbf{R} est une matrice $q \times q$ inversible donnée.

Démonstration : voir Annexe A

3.2.3 Implémentation

Le cas SIMO

Dans le cas SIMO ($q = 1$), La matrice $\tilde{\mathbf{V}}$ est remplacée par le vecteur $\tilde{\mathbf{v}}$ de dimension d et l'équation (3.9) est résolue simplement au sens des moindres carrés, sous contrainte norme unité

$$\tilde{\mathbf{v}} = \arg \min_{\|\mathbf{z}\|=1} \mathbf{z}^H \mathbf{Q} \mathbf{z}, \quad (3.12)$$

où \mathbf{Q} est une matrice $(d \times d)$ définie par

$$\mathbf{Q} \stackrel{\text{def}}{=} \mathbf{T}^H \mathbf{T}. \quad (3.13)$$

D'après (3.4) et (3.11), on obtient alors le vecteur égaliseur MMSE $\mathbf{v}_0 = r\mathbf{v}$, où r est une constante scalaire non-nulle donnée et \mathbf{v} est un vecteur de dimension n donné par

$$\mathbf{v} = \mathbf{W}\tilde{\mathbf{v}}. \quad (3.14)$$

La TABLE 3.1 résume l'implémentation en schéma bloc de l'algorithme d'égalisation MMSE aveugle pour le cas SIMO. Le coût de calcul de cette implémentation est de l'ordre de $(K + d)n^2 + (d - p + 1)nd - pd^2 + O(dn^2) + O(d^3)$.

\mathbf{C}_N	$= \frac{1}{K} \sum_{t=L-1}^{K+L-2} \mathbf{x}_N(t)\mathbf{x}_N^H(t)$, (K : Nombre d'échantillons)
$(\mathbf{W}, \mathbf{\Lambda})$	$= \text{eigs}(\mathbf{C}_N, d)$, (extrait les d principaux vecteurs propres de \mathbf{C}_N)
\mathbf{T}	$= \mathbf{C}_N(p + 1 : n, :)\mathbf{W}$
\mathbf{Q}	$= \mathbf{T}^H \mathbf{T}$
$\tilde{\mathbf{v}}$	$=$ vecteur propre mineur de \mathbf{Q}
\mathbf{v}	$= \mathbf{W}\tilde{\mathbf{v}}$

TAB. 3.1 – Algorithme d'égalisation MMSE aveugle pour un canal SIMO.

Le cas MIMO

Dans ce cas, il serait plus simple de remplacer le système (3.9) par l'expression équivalente suivante (voir [19] pour plus de détails)

$$(\mathbf{I}_q \otimes \mathbf{T})\text{vec}(\tilde{\mathbf{V}}) = \mathbf{0}. \quad (3.15)$$

Dans cette situation, la contrainte quadratique sur $\text{vec}(\tilde{\mathbf{V}})$ ne peut plus garantir la condition (3.10) dans le Théorème 3.1. On y remédiera en choisissant une contrainte linéaire telle que le premier bloc $q \times q$ de la matrice $\tilde{\mathbf{V}}$ soit triangulaire inférieur

$$\tilde{\mathbf{V}}(1 : q, 1 : q) = \begin{bmatrix} 1 & \cdots & 0 \\ \times & \ddots & \vdots \\ \times & \times & 1 \end{bmatrix}, \quad (3.16)$$

ce qui garantira que la matrice $\tilde{\mathbf{V}}$ sera de rang colonne plein q .

En tenant compte de la contrainte triangulaire inférieure (3.16), l'équation (3.15) devient alors

$$\mathbf{a} + \mathbf{A}\tilde{\mathbf{v}} = \mathbf{0}, \quad (3.17)$$

où

$$\begin{aligned}
 \bar{\mathbf{v}} &= \mathcal{J}^T \text{vec}(\tilde{\mathbf{V}}), \\
 \mathbf{a} &= \text{vec}(\mathbf{T} \mathcal{J}_{0,q,d-q}), \\
 \mathbf{A} &= (\mathbf{I}_q \otimes \mathbf{T}) \mathcal{J}, \\
 \mathcal{J} &= \text{diag}(\mathcal{J}_1, \mathcal{J}_2, \dots, \mathcal{J}_q), \\
 \mathcal{J}_k &= \mathcal{J}_{k,d-k,0}, \quad k = 1, \dots, q.
 \end{aligned} \tag{3.18}$$

La solution de l'équation (3.17) est donnée par

$$\bar{\mathbf{v}} = -\mathbf{A}^\# \mathbf{a}. \tag{3.19}$$

La matrice $\tilde{\mathbf{V}}$, solution de l'équation (3.9), est alors donnée par $\tilde{\mathbf{V}} = \text{vec}^{-1}(\bar{\mathbf{v}})$, où $\bar{\mathbf{v}}$ est obtenu à partir de $\bar{\mathbf{v}}$ en lui rajoutant des uns et des zéros à des endroits appropriés par le biais de la relation

$$\bar{\mathbf{v}} = \mathcal{J} \bar{\mathbf{v}} + \text{vec}(\mathcal{J}_{0,q,d-q}). \tag{3.20}$$

Des équations (3.4) et (3.11), on obtient la matrice d'égalisation MMSE $\mathbf{V}_0 = \mathbf{V} \mathbf{R}^{-1}$, où \mathbf{R} est une matrice $q \times q$, constante et inversible et \mathbf{V} est une matrice ($n \times q$) donnée par

$$\mathbf{V} = \mathbf{W} \tilde{\mathbf{V}}. \tag{3.21}$$

La TABLE 3.2 résume l'implémentation en schéma bloc de l'algorithme d'égalisation MMSE aveugle pour le cas MIMO. Noter que la matrice constante \mathbf{R} est une indétermination inhérente aux systèmes

\mathbf{C}_N	$= \frac{1}{K} \sum_{t=N-1}^{K+N-2} \mathbf{x}_N(t) \mathbf{x}_N^H(t)$, (K : Nombre d'échantillons)
$(\mathbf{W}, \mathbf{\Lambda})$	$= \text{eigs}(\mathbf{C}_N, d)$, (extrait les d principaux vecteurs propres de \mathbf{C}_N)
\mathbf{T}	$= \mathbf{C}_N(p+1:n, :)\mathbf{W}$
\mathbf{a}	$= \text{vec}(\mathbf{T}(:, 1:q))$
\mathbf{A}	$= (\mathbf{I}_q \otimes \mathbf{T}) \mathcal{J}$
$\bar{\mathbf{v}}$	$= -\mathbf{A}^\# \mathbf{a}$
$\tilde{\mathbf{V}}$	$= \text{vec}^{-1}(\mathcal{J} \bar{\mathbf{v}}) + \mathcal{J}_{0,q,d-q}$
\mathbf{V}	$= \mathbf{W} \tilde{\mathbf{V}}$

TAB. 3.2 – Algorithme d'égalisation MMSE aveugle pour un canal MIMO.

d'identification aveugle MIMO, utilisant des statistiques du second ordre [1]. En général, ces indéterminations sont levées par l'application d'algorithmes de séparation de source. Le coût globale de cette implémentation est de l'ordre de $(K+d)n^2 - (p-q)nd + q^2d^2 + O(dn^2) + O(q^3d^3)$.

3.2.4 Selection du retard de l'égaliseur

Il est bien connu que le choix du retard de l'égaliseur affecte et d'une manière significative les performances de l'égalisation que se soit pour des systèmes SIMO ou bien MIMO. En effet les égaliseurs

à retard non-nul donnent de meilleures performances comparés à ceux avec retard zéro [70]. A cet effet, le vecteur spatio-temporel dans (1.4) peut être réécrit comme suit

$$\mathbf{x}_N(t) = \sum_{k=0}^{m-1} \mathbf{G}_k \mathbf{s}(t-k) + \mathbf{b}_N(t), \quad (3.22)$$

où la matrice \mathbf{G}_k est celle définie dans (3.2). On peut observer que $\|\mathbf{G}_0\| \leq \|\mathbf{G}_1\| \leq \dots \leq \|\mathbf{G}_L\| = \|\mathbf{G}_{L+1}\| = \dots = \|\mathbf{G}_{N-1}\|$ et $\|\mathbf{G}_{N-1}\| \geq \|\mathbf{G}_N\| \geq \dots \geq \|\mathbf{G}_{m-1}\|$. En d'autres termes, les symboles d'entrée ayant des retards τ tels que $L \leq \tau \leq N-1$ sont multipliés par des coefficients (matriciels) dont la norme est maximale. Par conséquent, les meilleurs retards d'égalisation sont ceux appartenant à l'intervalle $[L, \dots, N-1]$. De même, on peut observer que le gain en performance des égaliseurs dont les retards appartiennent à l'intervalle $[L, \dots, N-1]$ par rapport à ceux avec des retards extrêmes, i.e., $\tau = 0$ ou $\tau = m-1$, est considérable. Alors qu'en général, cette différence de gain devient négligeable si on considère deux égaliseurs avec des retards à l'intérieur de l'intervalle $[L, \dots, N-1]$ (voir [70]). En pratique, la recherche d'un retard d'égalisation optimal s'avère une opération induisant un coût de calcul supplémentaire élevé et inutile. En général, on obtient un *bon* retard d'égalisation en le choisissant dans l'intervalle $[L, \dots, N-1]$, comme c'est le cas pour nous, e.g. $\tau = L$.

De plus, il est montré dans les paragraphes 1.6 et 3.4 que le processus aveugle d'estimation du filtre MMSE induit une perte en performance comparée au cas non-aveugle. Pour compenser cette perte et du coup obtenir des égaliseurs avec des retards non-nuls d'où une amélioration des performances, on propose ici une approche à *deux étapes* pour estimer l'égaliseur MMSE aveugle. Dans la première étape, on estime \mathbf{V}_0 d'après les algorithmes précédents, tandis que dans la deuxième étape, on raffine cette estimation en exploitant la connaissance a priori de l'alphabet auquel appartiennent les symboles $\mathbf{s}(t)$. Ce qui revient à effectuer une décision sur les symboles qui seront par la suite utilisés pour ré-estimer \mathbf{V}_τ en utilisant les équations (3.1) and (3.2)¹.

Plus précisément, en opérant avec le filtre égaliseur \mathbf{V} de (3.21) (ou de (3.14) pour le cas SIMO) sur le vecteur de données reçu $\mathbf{x}_N(t)$ de (1.4), on obtient alors, d'après (3.11) et (3.4), une estimation du signal émis $\tilde{\mathbf{s}}(t) = \mathbf{V}^H \mathbf{x}_N(t) = \mathbf{R}^H \mathbf{V}_0^H \mathbf{x}_N(t)$, comme $\mathbf{V}_0^H \mathbf{x}_N(t) = \mathbf{s}(t) + \boldsymbol{\epsilon}(t)$, où $\boldsymbol{\epsilon}(t)$ représente une erreur d'estimation résiduelle (de la variance minimale) de $\mathbf{s}(t)$, il s'en suit

$$\tilde{\mathbf{s}}(t) = \mathbf{R}^H \mathbf{s}(t) + \tilde{\boldsymbol{\epsilon}}(t), \quad (3.23)$$

où $\tilde{\boldsymbol{\epsilon}}(t) = \mathbf{R}^H \boldsymbol{\epsilon}(t)$. Il est clair, d'après (3.23), que le signal estimé $\tilde{\mathbf{s}}(t)$ est un mélange instantané du signal émis $\mathbf{s}(t)$ affecté d'un bruit additif coloré $\tilde{\boldsymbol{\epsilon}}(t)$. Ainsi, une identification de \mathbf{R} (i.e. lever l'indétermination) est alors nécessaire afin d'extraire le signal original. Ceci est réalisé par application (dans le cas d'un traitement en bloc ou bien en adaptatif) d'algorithmes aveugles de séparation de sources (*Blind Sources Separation*, BSS) [14, 16, 65] à la sortie de l'égaliseur (3.23), suivit d'une décision sur les symboles. En ce qui nous concerne, on utilisera l'algorithme ACMA (*Analytical Constant Modulus Algorithm*) de [81] dans le cas d'un traitement en bloc et l'algorithme A-CMS (*Adaptive Constant Modulus Separation*) de [15] (cf. TABLE 3.7) dans le cas adaptatif. Ceci est dû au fait que les algorithmes

¹On suppose ici qu'on utilise une modulation différentielle pour ainsi se débarrasser de l'indétermination de phase qui est inhérente au problème d'égalisation aveugle.

du genre module constant (*Constant Modulus Algorithms*, CMA) ont, relativement, un faible coût de calcul et ils sont d'une grande efficacité pour quand il s'agit de séparer des signaux (à alphabet fini) de communications. L'algorithme aveugle d'égalisation MMSE à deux étapes est résumé dans TABLE 3.1, TABLE 3.2 et TABLE 3.3. Il en résulte alors un coût globale de $(K + d + q)n^2 + Kqn - (p - q)nd + q^2d^2 + O(n^3) + O(dn^2) + O(q^3d^3)$

1. Estimer $\tilde{\mathbf{s}}(t)$, en utilisant \mathbf{V} donné par la TABLE 3.1 ou la TABLE 3.2, i.e. $\tilde{\mathbf{s}}(t) = \mathbf{V}^H \mathbf{x}_N(t)$.
2. Estimer $\hat{\mathbf{s}}(t)$ à partir du modèle 3.23, en utilisant un algorithme BSS (e.g. ACMA de [81]).
3. Calculer
 - $\mathbf{G}_\tau = \frac{1}{K} \sum_{t=\tau}^{K+\tau-1} \mathbf{x}_N(t) \hat{\mathbf{s}}^H(t - \tau)$
 - $\mathbf{V}_\tau = \mathbf{C}_N^{-1} \mathbf{G}_\tau$

TAB. 3.3 – *Procédure d'égalisation à deux étapes.*

3.2.5 Robustesse

Ici on étudiera la robustesse de l'égaliseur MMSE aveugle proposé au erreurs de surestimation de l'ordre du canal. Pour simplifier, considérons le cas SIMO où l'ordre du canal est utilisé pour déterminer le nombre de colonnes $d = L + N$ de la matrice \mathbf{W} (qui, en pratique, correspond à la dimension du sous espace dominant de la matrice \mathbf{C}_N). Soit $L' > L$ l'ordre surestimé du canal, d'où $d' = L' + N$ est le nombre de colonnes de \mathbf{W} , i.e., on considère le sous-espace généré par les d' vecteurs propres dominant de \mathbf{C}_N . On a remarqué alors, que tant que le nombre de capteurs p auquel s'ajoute l'erreur de surestimation de l'ordre $L' - L$ est inférieur à la dimension du sous-espace bruit, i.e., $p + L' - L < n - d$, la résolution de (3.9) au sens des moindres carrés donne une estimation consistante de l'égaliseur MMSE. Cette observation est justifiée par ce qui suit :

Noter qu'en utilisant l'équation (1.9), la matrice $\bar{\mathbf{C}}$ définie par (3.8) s'exprime par $\bar{\mathbf{C}} = [\mathbf{H}' \ \mathbf{C}_{N-1}]$, où \mathbf{H}' est une matrice $(n - p) \times p$. Comme \mathbf{C}_{N-1} est une matrice à rang plein, il s'en suit que le noyau droit de la matrice $\bar{\mathbf{C}}$: $\text{null}_r(\bar{\mathbf{C}}) = \{\mathbf{z} \in \mathbb{C}^n : \bar{\mathbf{C}}\mathbf{z} = \mathbf{0}\}$, est un sous-espace de dimension p . De même, on peut affirmer que seulement une direction de $\text{null}_r(\bar{\mathbf{C}})$ appartient au sous-espace signal, du moment que $\text{null}_r(\bar{\mathbf{C}}) \cap \text{range}(\mathbf{H}_N) = \text{null}_r(\bar{\mathbf{C}}\mathbf{H}_N) = \text{null}_r(\bar{\mathbf{C}}\mathbf{W})$ (cette dernière égalité vient du fait que \mathbf{H}_N et \mathbf{W} génèrent le même sous-espace signal). D'après la preuve du *Théorème*. 3.1, $\dim(\text{null}_r(\bar{\mathbf{C}}\mathbf{W})) = 1$. Soit $\mathbf{b}_1, \dots, \mathbf{b}_p$ une base de $\text{null}_r(\bar{\mathbf{C}})$ telle que \mathbf{b}_1 appartient au sous-espace signal (i.e. $\text{range}(\mathbf{H}_N)$). Or, la solution de (3.9) est unique (à une constante multiplicative scalaire non-nulle près) si :

$$\text{range}(\mathbf{W}) \cap \text{range}([\mathbf{b}_1 \ \dots \ \mathbf{b}_p]) = \text{range}(\mathbf{b}_1),$$

ce qui équivaut à

$$\text{range}(\mathbf{W}) \cap \text{range}([\mathbf{b}_2 \ \dots \ \mathbf{b}_p]) = \{\mathbf{0}\}.$$

Les conditions ci-dessus sont vérifiées si l'intersection du sous-espace généré par les projections de $\mathbf{b}_2, \dots, \mathbf{b}_p$ dans le sous-espace signal et le sous-espace généré par les $L' - L$ vecteurs bruit de \mathbf{W} introduit par l'erreur de surestimation de l'ordre est vide (à l'exception du vecteur nul). Comme ces derniers sont introduits aléatoirement par la décomposition en valeurs / vecteurs propres de \mathbf{C}_N et comme $p + L' - L < n - d$, alors on peut s'attendre à ce que cette intersection de sous-espace soit presque tout le temps vide.

Noter aussi que l'utilisation d'une contrainte linéaire donne une meilleure robustesse que celle obtenue avec une contrainte quadratique. La raison est que la solution de (3.9) est, en général, une combinaison linéaire de la solution désirée \mathbf{v}_0 (qui appartient au sous-espace signal) et des vecteurs du sous-espace bruit (introduit par les erreurs de surestimation de l'ordre). Cependant, il a été observé que pour une taille d'échantillons finie et pour des SNRs modérés ou élevés, la contribution de la solution désirée \mathbf{v}_0 dans l'équation (3.9) est plus élevée que celle des vecteurs du sous-espace bruit. Ceci est dû au fait que la faible énergie à la sortie des vecteurs du sous-espace bruit, vient de leur orthogonalité avec la matrice de filtrage du système \mathbf{H}_N (c'est une propriété structurelle, indépendante de la taille des échantillons), tandis que l'appartenance de la solution désirée \mathbf{v}_0 au noyau de $\bar{\mathbf{C}}$ est due à la propriété de décorrélation des signaux d'entrée, qui est asymptotiquement valide lorsque la taille des échantillons est large. En effet, on peut observer (voir Fig. 3.11. du paragraphe 3.5) qu'en augmentant K (la taille des échantillons), la robustesse de l'égaliseur avec contrainte quadratique s'améliore de façon très significative. Par conséquent, dans un contexte où la taille des échantillons est moyenne ou faible, résoudre (3.9) au sens des moindres carrés sous contrainte norme unité, mène à une solution qui appartient presque tout le temps au sous-espace bruit (i.e., la partie de \mathbf{v}_0 dans la solution finale devient très faible). D'autre part, en résolvant (3.9) soumise à une contrainte linéaire (équations (3.19) et (3.20)), on obtient une solution où le facteur linéaire de \mathbf{v}_0 est plus significatif (ce qui est due au fait que le vecteur \mathbf{a} dans (3.19) appartient à l'image de la matrice \mathbf{A}).

Ce raisonnement, qui toute fois n'est pas une démonstration rigoureuse de la robustesse, a été confirmé par des résultats de simulation (voir l'exemple de simulation présenté ci-dessous, où on peut voir que la dégradation des performance due aux erreurs de surestimation de l'ordre du canal reste relativement limitée).

3.3 Implémentation adaptative rapide

Dans des applications de poursuite, on est intéressé par une estimation récursive à faible coût du vecteur égaliseur. Nous introduirons ici une implémentation adaptative rapide des algorithmes d'égalisation aveugle proposés. La réduction du coût de calcul est réalisée en exploitant l'idée de l'approximation de la projection [88] et la propriété du décalage-invariant des matrices d'autocorrélation des données temporelles.

La matrice \mathbf{C}_N est remplacée par son estimation récursive

$$\mathbf{C}_N(t) = \sum_{k=0}^t \beta^{t-k} \mathbf{x}_N(k) \mathbf{x}_N^H(k) = \beta \mathbf{C}_N(t-1) + \mathbf{x}_N(t) \mathbf{x}_N^H(t), \quad (3.24)$$

où $0 < \beta < 1$ est le facteur d'oubli. La matrice \mathbf{W} , correspondant au d vecteurs propres dominants de la matrice \mathbf{C}_N , peut être estimée en utilisant un algorithme rapide d'estimation et de suivi de sous-espace. Dans ce travail, on utilisera l'algorithme YAST (*Yet Another Subspace Tracker*) [11]. Le choix de l'algorithme YAST est motivé par ses remarquables performances en poursuite comparé à d'autres algorithmes existants de suivi de sous-espaces ayant la même complexité de calcul (PAST [88], OPAST [5], ..). La TABLE 3.4 résume l'algorithme YAST. Noter qu'avec YAST, seulement $O(nd)$ opérations par itération sont requises (au lieu de $O(n^3)$ pour une EVD complète) pour l'estimation de $\mathbf{W}(t)$. Le vecteur $\mathbf{x}'_N(t) = \mathbf{C}_N(t-1)\mathbf{x}_N(t)$ dans la TABLE 3.4 est calculé en $O(np)$ opérations, en utilisant la propriété du décalage-invariant de la matrice d'autocorrélation (voir détails dans [52]).

Par application, à l'équation (3.7), de l'approximation de la projection

$$\mathbf{C}_N(t)\mathbf{W}(t) \approx \mathbf{C}_N(t)\mathbf{W}(t-1),$$

qui est valide quand la matrice $\mathbf{W}(t)$ varie lentement dans le temps [5], on obtient

$$\mathbf{T}(t) = \beta\mathbf{T}(t-1) + \mathcal{J}_{p,n-p,0}^T \mathbf{x}_N(t) \mathbf{y}^H(t), \quad (3.25)$$

où le vecteur $\mathcal{J}_{p,n-p,0}^T \mathbf{x}_N(t)$ est un sous-vecteur de $\mathbf{x}_N(t)$ donné par ses $(n-p)$ derniers éléments et le vecteur $\mathbf{y}(t) = \mathbf{W}^H(t-1)\mathbf{x}_N(t)$ est calculé par YAST (cf. TABLE 3.4).

3.3.1 Le cas SIMO

Dans ce cas, on cherche une estimation récursive du vecteur $\tilde{\mathbf{v}}$ de l'équation (3.12), sachant que $\tilde{\mathbf{v}}$ correspond au vecteur propre de la matrice \mathbf{Q} associé à sa plus petite valeur propre, ou d'une manière équivalente, au vecteur propre associé à la plus grande valeur propre de son inverse². En utilisant (3.25), l'équation (3.13) est alors remplacé par la relation récursive suivante

$$\mathbf{Q}(t) = \beta^2\mathbf{Q}(t-1) - \mathbf{D}_Q(t)\mathbf{\Gamma}_Q^{-1}(t)\mathbf{D}_Q^H(t), \quad (3.26)$$

où $\mathbf{D}_Q(t)$ est une matrice $d \times 2$ donnée par

$$\mathbf{D}_Q(t) = \begin{bmatrix} \beta\mathbf{T}^H(t-1)\mathcal{J}_{p,n-p,0}^T \mathbf{x}_N(t) & \mathbf{y}(t) \end{bmatrix}, \quad (3.27)$$

et $\mathbf{\Gamma}_Q(t)$ est une matrice 2×2 non-singulière donnée par

$$\mathbf{\Gamma}_Q(t) = \begin{bmatrix} \|\mathcal{J}_{p,n-p,0}^T \mathbf{x}_N(t)\|^2 & -1 \\ -1 & 0 \end{bmatrix}. \quad (3.28)$$

Soit $\mathbf{F}(t)$ la matrice $d \times d$, définie par

$$\mathbf{F}(t) \stackrel{\text{def}}{=} \mathbf{Q}^{-1}(t),$$

² \mathbf{Q} est une matrice singulière dans le cas où les statistiques sont exactes. Quoi qu'il en soit, quand la matrice \mathbf{C} est estimée à partir d'un nombre finis d'échantillons, du fait des erreurs d'estimation et de l'approximation de la projection, l'estimé de la matrice \mathbf{Q} est presque tout le temps non-singulière.

$\mathbf{y}(t)$	$= \mathbf{W}^H(t-1)\mathbf{x}_N(t)$
$\mathbf{x}'_N(t)$	$= \mathbf{C}_N(t-1)\mathbf{x}_N(t)$
$\mathbf{y}'(t)$	$= \mathbf{W}^H(t-1)\mathbf{x}'_N(t)$
$\sigma(t)$	$= (\mathbf{x}_N^H(t)\mathbf{x}_N(t) - \mathbf{y}^H(t)\mathbf{y}(t))^{\frac{1}{2}}$
$\mathbf{h}(t)$	$= \mathbf{Z}(t-1)\mathbf{y}(t)$
$\gamma(t)$	$= (\beta + \mathbf{y}^H(t)\mathbf{h}(t))^{-1}$
$\tilde{\mathbf{Z}}(t)$	$= \frac{1}{\beta}(\mathbf{Z}(t-1) - \mathbf{h}(t)\gamma(t)\mathbf{h}^H(t))$
$\alpha(t)$	$= \mathbf{x}_N^H(t)\mathbf{x}_N(t)$
$\mathbf{y}''(t)$	$= \beta\mathbf{y}'(t) + \mathbf{y}(t)\alpha(t)$
$c_{yy}(t)$	$= \beta\mathbf{x}_N^H(t)\mathbf{x}'_N(t) + \alpha^*(t)\alpha(t)$
$\mathbf{h}'(t)$	$= \tilde{\mathbf{Z}}(t-1)\mathbf{y}''(t)$
$\gamma'(t)$	$= (c_{yy}(t) - [\mathbf{y}''(t)]^H\mathbf{h}'(t))^{-1}$
$\mathbf{h}''(t)$	$= \mathbf{h}'(t) - \mathbf{y}(t)$
$\tilde{\mathbf{Z}}'(t)$	$= \tilde{\mathbf{Z}}(t) + \mathbf{h}''(t)\gamma'(t)[\mathbf{h}''(t)]^H$
$\mathbf{g}(t)$	$= \mathbf{h}''(t)\gamma'(t)\sigma^*(t)$
$\gamma''(t)$	$= \sigma(t)\gamma'(t)\sigma^*(t)$
$\mathbf{Z}'(t)$	$= [\tilde{\mathbf{Z}}'(t), -\mathbf{g}(t); -\mathbf{g}^H(t), \gamma''(t)]$
$(\phi(t), \lambda(t))$	$= \text{eigs}(\mathbf{Z}'(t), 1)$
$\varphi(t)$	$= \phi_{(1:d)}(t)$
$z(t)$	$= \phi_{(d+1)}(t)$
$\rho(t)$	$= z(t) $
$\theta(t)$	$= e^{j \arg(z(t))}$
$\mathbf{f}(t)$	$= \varphi(t)\theta^*(t)$
$\mathbf{f}'(t)$	$= \mathbf{f}(t)(1 + \rho(t))^{-1}$
$\mathbf{y}'''(t)$	$= \mathbf{y}(t)\sigma^{-1}(t) - \mathbf{f}'(t)$
$\mathbf{e}(t)$	$= \mathbf{x}(t)\sigma^{-1}(t) - \mathbf{W}(t-1)\mathbf{y}'''(t)$
$\mathbf{W}(t)$	$= \mathbf{W}(t-1) - \mathbf{e}(t)\mathbf{f}^H(t)$
$\mathbf{g}'(t)$	$= \mathbf{g}(t) + \mathbf{f}'(t)(\gamma''(t) - \theta(t)\lambda(t)\theta^*(t))$
$\mathbf{Z}(t)$	$= \tilde{\mathbf{Z}}'(t) + \mathbf{g}'(t)[\mathbf{f}'(t)]^H + \mathbf{f}'(t)\mathbf{g}^H(t)$

TAB. 3.4 – *Algorithme YAST.*

en utilisant le lemme d'inversion matricielle [46], on obtient

$$\mathbf{F}(t) = \frac{1}{\beta^2} \mathbf{F}(t-1) + \mathbf{D}_F(t) \mathbf{\Gamma}_F(t) \mathbf{D}_F^H(t),$$

où $\mathbf{D}_F(t)$ est une matrice $d \times 2$ donnée par

$$\mathbf{D}_F(t) = \frac{1}{\beta^2} \mathbf{F}(t-1) \mathbf{D}_Q(t),$$

et $\mathbf{\Gamma}_F(t)$ est une matrice 2×2 donnée par

$$\mathbf{\Gamma}_F(t) = (\mathbf{\Gamma}_Q(t) - \mathbf{D}_F^H(t) \mathbf{D}_Q(t))^{-1}.$$

L'extraction du vecteur propre dominant de $\mathbf{F}(t)$ est obtenu en utilisant la méthode des puissance

$$\tilde{\mathbf{v}}(t) = \frac{\mathbf{F}(t) \tilde{\mathbf{v}}(t-1)}{\|\mathbf{F}(t) \tilde{\mathbf{v}}(t-1)\|}.$$

La TABLE 3.5 résume ainsi, l'algorithme d'égalisation MMSE adaptative aveugle dans le cas SIMO. Noter que le processus tout entier requiert seulement $3nd + 5d^2 + d + O(nd)$ opérations par itération.

<p>– Mise à jour de $\mathbf{W}(t)$ et $\mathbf{y}(t)$ en utilisant YAST (cf. TABLE 3.4)</p> <p>$\bar{\mathbf{x}}(t) = \mathbf{x}_N(t)_{(p+1:n)}$</p> <p>$\mathbf{\Gamma}_Q(t) = \begin{bmatrix} \ \bar{\mathbf{x}}(t)\ ^2 & -1 \\ -1 & 0 \end{bmatrix}$</p> <p>$\mathbf{D}_Q(t) = [\beta \mathbf{T}^H(t-1) \bar{\mathbf{x}}(t) \quad \mathbf{y}(t)]$</p> <p>$\mathbf{D}_F(t) = \frac{1}{\beta^2} \mathbf{F}(t-1) \mathbf{D}_Q(t)$</p> <p>$\mathbf{\Gamma}_F(t) = (\mathbf{\Gamma}_Q(t) - \mathbf{D}_F^H(t) \mathbf{D}_Q(t))^{-1}$</p> <p>$\mathbf{F}(t) = \frac{1}{\beta^2} \mathbf{F}(t-1) + \mathbf{D}_F(t) \mathbf{\Gamma}_F(t) \mathbf{D}_F^H(t)$</p> <p>$\tilde{\mathbf{v}}(t) = \frac{\mathbf{F}(t) \tilde{\mathbf{v}}(t-1)}{\ \mathbf{F}(t) \tilde{\mathbf{v}}(t-1)\ }$</p> <p>$\mathbf{v}(t) = \mathbf{W}(t) \tilde{\mathbf{v}}(t)$</p> <p>$\mathbf{T}(t) = \beta \mathbf{T}(t-1) + \bar{\mathbf{x}}(t) \mathbf{y}^H(t)$</p>
--

TAB. 3.5 – Algorithme d'égalisation MMSE adaptative aveugle pour un canal SIMO.

3.3.2 Le cas MIMO

Ici, nous introduirons une version adaptative rapide de l'algorithme d'égalisation MMSE aveugle dans le cas d'un canal MIMO donnée par la TABLE 3.2. Tout d'abord, notant les effets de l'approximation de la projection et de la taille finie des échantillons rendent la matrice \mathbf{A} presque tout le temps à rang-colonne plein, d'où

$$\mathbf{A}^\# = (\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H. \quad (3.29)$$

Par conséquent le vecteur $\bar{\mathbf{v}}$ dans (3.19) s'exprime par

$$\bar{\mathbf{v}}(t) = \left[\bar{\mathbf{v}}_1^T(t) \quad \bar{\mathbf{v}}_2^T(t) \quad \dots \quad \bar{\mathbf{v}}_q^T(t) \right]^T,$$

où les vecteurs $\bar{\mathbf{v}}_k(t)$ pour $k = 1, \dots, q$, sont donnés par

$$\begin{aligned}\bar{\mathbf{v}}_k(t) &= -\mathbf{F}_k(t) \mathbf{f}_k(t), \\ \mathbf{F}_k(t) &= (\mathcal{J}_k^T \mathbf{Q}(t) \mathcal{J}_k)^{-1}, \\ \mathbf{f}_k(t) &= \mathcal{J}_k^T \mathbf{Q}(t) \mathcal{J}_{k-1,1,d-k}.\end{aligned}$$

En utilisant (3.26) et le lemme d'inversion matricielle [46], les matrices $\mathbf{F}_k(t)$ sont mises à jour par les relations récursives suivantes

$$\begin{aligned}\mathbf{F}_k(t) &= \frac{1}{\beta^2} \mathbf{F}_k(t-1) + \mathbf{D}_{F_k}(t) \mathbf{\Gamma}_{F_k}(t) \mathbf{D}_{F_k}^H(t), \\ \mathbf{D}_{F_k}(t) &= \frac{1}{\beta^2} \mathbf{F}_k(t-1) \mathcal{J}_k^T \mathbf{D}_Q(t), \\ \mathbf{\Gamma}_{F_k}(t) &= (\mathbf{\Gamma}_Q(t) - \mathbf{D}_{F_k}^H(t) \mathcal{J}_k^T \mathbf{D}_Q(t))^{-1},\end{aligned}$$

où les matrices $\mathbf{D}_Q(t)$ et $\mathbf{\Gamma}_Q(t)$ sont données par les équations (3.27) et (3.28).

La TABLE 3.6 résume ainsi, l'algorithme d'égalisation MMSE adaptative aveugle dans le cas MIMO. Noter que le processus tout entier requiert une complexité de calcul de l'ordre de $O(qnd)$ opérations par itération.

<p>– Mise à jour de $\mathbf{W}(t)$ et $\mathbf{y}(t)$ en utilisant YAST (cf. TABLE 3.4)</p> $\bar{\mathbf{x}}(t) = \mathbf{x}_N(t)_{(p+1:n)}$ $\mathbf{\Gamma}_Q(t) = \begin{bmatrix} \ \bar{\mathbf{x}}(t)\ ^2 & -1 \\ -1 & 0 \end{bmatrix}$ $\mathbf{D}_Q(t) = [\beta \mathbf{T}^H(t-1) \bar{\mathbf{x}}(t) \quad \mathbf{y}(t)]$ $\mathbf{Q}(t) = \beta^2 \mathbf{Q}(t-1) - \mathbf{D}_Q(t) \mathbf{\Gamma}_Q^{-1}(t) \mathbf{D}_Q^H(t)$ <p>For $k = 1, \dots, q$:</p> $\mathbf{f}_k(t) = \mathbf{Q}(t)_{(k+1:d,k)}$ $\mathbf{D}_{F_k}(t) = \frac{1}{\beta^2} \mathbf{F}_k(t-1) \mathbf{D}_Q(t)_{(k+1:d,:)}$ $\mathbf{\Gamma}_{F_k}(t) = (\mathbf{\Gamma}_Q(t) - \mathbf{D}_{F_k}^H(t) \mathbf{D}_Q(t)_{(k+1:d,:)})^{-1}$ $\mathbf{F}_k(t) = \frac{1}{\beta^2} \mathbf{F}_k(t-1) + \mathbf{D}_{F_k}(t) \mathbf{\Gamma}_{F_k}(t) \mathbf{D}_{F_k}^H(t)$ $\bar{\mathbf{v}}_k(t) = -\mathbf{F}_k(t) \mathbf{f}_k(t)$ <p>end</p> $\bar{\mathbf{v}}(t) = \begin{bmatrix} \bar{\mathbf{v}}_1^T(t) & \bar{\mathbf{v}}_2^T(t) & \dots & \bar{\mathbf{v}}_q^T(t) \end{bmatrix}^T$ $\tilde{\mathbf{V}}(t) = \text{vec}^{-1}(\mathcal{J} \bar{\mathbf{v}}(t)) + \mathcal{J}_{0,q,d-q}$ $\mathbf{V}(t) = \mathbf{W}(t) \tilde{\mathbf{V}}(t)$ $\mathbf{T}(t) = \beta \mathbf{T}(t-1) + \bar{\mathbf{x}}(t) \mathbf{y}^H(t)$

TAB. 3.6 – Algorithme d'égalisation MMSE adaptative aveugle pour un canal MIMO.

3.3.3 Procédure à deux étapes

Soit $\mathbf{W} \in \mathbb{C}^{n \times d}$ une base orthonormale du sous-espace signal. Comme \mathbf{G}_τ appartient au sous-espace signal, on peut écrire (voir [22])

$$\mathbf{V}_\tau = \mathbf{W}(\mathbf{W}^H \mathbf{C}_N \mathbf{W})^{-1} \mathbf{W}^H \mathbf{G}_\tau. \quad (3.30)$$

Cette expression de \mathbf{V}_τ est utilisée pour une implémentation adaptative rapide de l'algorithme à deux étapes, du moment que $\mathbf{Z} = (\mathbf{W}^H \mathbf{C}_N \mathbf{W})^{-1}$ est déjà calculé dans YAST. L'expression récursive du vecteur \mathbf{G}_τ est alors donnée par

$$\mathbf{G}_\tau(t) = \beta \mathbf{G}_\tau(t-1) + \mathbf{x}_N(t) \hat{\mathbf{s}}^H(t-\tau),$$

où $\hat{\mathbf{s}}(t)$ est une estimée de $\mathbf{s}(t)$ obtenue par application d'algorithmes BSS aux signaux $\tilde{\mathbf{s}}(t)$ de l'équation (3.23). Dans nos simulations, nous avons utilisé l'algorithme A-CMS [15] que résume la TABLE 3.7.

Pour chaque instant t :

1. Forcer le module de chaque composante du vecteur $\hat{\mathbf{s}}(t-1)$ à une constante α (e.g. $\alpha = 1$), i.e., $\bar{\mathbf{s}}(t) = \text{diag}(|\hat{s}_1(t-1)|^{-1}, \dots, |\hat{s}_q(t-1)|^{-1}) \hat{\mathbf{s}}(t-1)$.
2. Calculer les statistiques :
 - $\mathbf{C}_{\tilde{\mathbf{s}}\tilde{\mathbf{s}}}(t) = (1 - \mu) \mathbf{C}_{\tilde{\mathbf{s}}\tilde{\mathbf{s}}}(t-1) + \mu \tilde{\mathbf{s}}(t) \tilde{\mathbf{s}}^H(t)$.
 - $\mathbf{C}_{\mathbf{s}\mathbf{s}}(t) = (1 - \mu) \mathbf{C}_{\mathbf{s}\mathbf{s}}(t-1) + \mu \bar{\mathbf{s}}(t) \bar{\mathbf{s}}^H(t)$.

μ étant une séquence décroissante et positive.
3. Calculer $\hat{\mathbf{s}}(t) = \mathbf{C}_{\mathbf{s}\mathbf{s}}(t) \mathbf{C}_{\tilde{\mathbf{s}}\tilde{\mathbf{s}}}(t)^\# \tilde{\mathbf{s}}(t)$.

TAB. 3.7 – Algorithme A-CMS.

Ainsi, l'équation (3.30) est remplacée par la récursion suivante

$$\begin{aligned} \mathbf{V}_\tau(t) &= \beta \mathbf{V}_\tau(t-1) + \mathbf{z}(t) \hat{\mathbf{s}}^H(t-\tau), \\ \mathbf{z}(t) &= \mathbf{W}(t) \mathbf{Z}(t) \mathbf{W}^H(t) \mathbf{x}_N(t). \end{aligned}$$

Noter, qu'en choisissant un retard d'égalisation non-nul, on améliorera les performances de l'égaliseur, comme on le verra plus bas. Ainsi, l'algorithme d'égalisation MMSE aveugle à deux-étapes est résumé par TABLE 3.5, TABLE 3.6 et TABLE 3.8. Le coût de calcul total de cet algorithme est de $(q+8)nd + O(qn + qd^2)$ opérations par itération.

3.4 Analyse asymptotique des performances

Comme il a été déjà mentionné plus haut (voir paragraphe 3.2.4), l'extraction de la matrice d'égalisation nécessite l'utilisation d'algorithmes BSS afin de pouvoir résoudre le problème des indéterminations qui est inhérent aux méthodes du second ordre d'identification aveugle MIMO. Ainsi les performances de

Pour chaque instant t :

1. Estimer $\tilde{\mathbf{s}}(t)$, en utilisant $\mathbf{V}(t)$ donné par la TABLE 3.5 ou la TABLE 3.6, i.e. $\tilde{\mathbf{s}}(t) = \mathbf{V}^H(t)\mathbf{x}_N(t)$.
2. Estimer $\hat{\mathbf{s}}(t)$ à partir du modèle 3.23, en utilisant l'algorithme A-CMS (cf. TABLE 3.7).
3. Mise à jour de $\mathbf{W}(t)$ et $\mathbf{Z}(t)$ en utilisant YAST (cf. TABLE 3.4)
4. Calculer
 - $\mathbf{z}(t) = \mathbf{W}(t)\mathbf{Z}(t)\mathbf{W}^H(t)\mathbf{x}_N(t)$
 - $\mathbf{V}_\tau(t) = \beta\mathbf{V}_\tau(t-1) + \mathbf{z}(t)\hat{\mathbf{s}}^H(t-\tau)$

TAB. 3.8 – Procédure d'égalisation adaptative à deux-étapes.

notre algorithme d'égalisation MIMO dépendent, en partie, du choix de l'algorithme BSS, ce qui va mener à des analyses asymptotiques de convergence très encombrantes. Pour simplifier, on se contentera d'étudier l'expression asymptotique du MSE estimé de l'égaliseur aveugle à retard nul dans le cas SIMO seulement, où le vecteur égaliseur est obtenu à une constante multiplicative scalaire non-nul inconnue près. Afin de pouvoir évaluer les performances de notre algorithme, une estimation de cette constante est donnée par

$$r = \arg \min_{\alpha} \|\mathbf{v}_0 - \alpha\mathbf{v}\|^2 = \frac{\mathbf{v}^H\mathbf{v}_0}{\|\mathbf{v}\|^2},$$

où \mathbf{v}_0 représente la valeur exacte de l'égaliseur MMSE à retard zéro et \mathbf{v} est l'égaliseur MMSE aveugle présenté précédemment.

3.4.1 Perte asymptotique en performances

Théoriquement, le MSE optimal est donné par

$$\text{MSE}_{opt} = E(|s(t) - \mathbf{v}_0^H \mathbf{x}_N(t)|^2) = 1 - \mathbf{g}_0^H \mathbf{C}_N^{-1} \mathbf{g}_0,$$

où \mathbf{g}_0 est le vecteur donné par l'équation (3.2) (pour $q = 1, \tau = 0$). Soit $\widehat{\text{MSE}}_{opt}$, le MSE correspondant à une estimation $\hat{\mathbf{v}}_0$ de \mathbf{v}_0

$$\widehat{\text{MSE}}_{opt} \stackrel{\text{def}}{=} E(|s(t) - \hat{\mathbf{v}}_0^H \mathbf{x}_N(t)|^2).$$

En terme de MSE, l'estimation aveugle donne lieu à une perte en performance égale à

$$\widehat{\text{MSE}}_{opt} - \text{MSE}_{opt} = \text{trace}(\mathbf{C}_N(\hat{\mathbf{v}}_0 - \mathbf{v}_0)(\hat{\mathbf{v}}_0 - \mathbf{v}_0)^H).$$

Asymptotiquement (i.e. pour un nombre d'échantillons K très grand), cette perte en performance est donnée par

$$\varepsilon \stackrel{\text{def}}{=} \lim_{K \rightarrow +\infty} K E(\widehat{\text{MSE}}_{opt} - \text{MSE}_{opt}) = \text{trace}(\mathbf{C}_N \boldsymbol{\Sigma}_{\mathbf{v}}),$$

où $\boldsymbol{\Sigma}_{\mathbf{v}}$ est la matrice de covariance asymptotique du vecteur $\hat{\mathbf{v}}_0$. Comme $\hat{\mathbf{v}}_0$ est une 'fonction' de la matrice d'autocorrélation échantillonnée du signal observé $\mathbf{x}_N(t)$, notée ici $\hat{\mathbf{C}}_N$ et donnée à partir de K

observations, par

$$\hat{\mathbf{C}}_N = \frac{1}{K} \sum_{t=0}^{K-1} \mathbf{x}_N(t) \mathbf{x}_N^H(t),$$

il clair que $\Sigma_{\mathbf{v}}$ dépend de la matrice de covariance asymptotique de $\hat{\mathbf{C}}_N$. Le lemme suivant donne l'expression explicite de la matrice de covariance asymptotique du vecteur aléatoire $\hat{\mathbf{c}}_N = \text{vec}(\hat{\mathbf{C}}_N)$.

Lemme 3.1 Soient $\mathbf{C}_{N,k}$ les matrices d'autocorrélation du vecteur $\mathbf{x}_N(t)$ définies par (2.3) et $\text{cum}(x_1, x_2, \dots, x_r)$ le cumulants d'ordre r des variables aléatoires (x_1, x_2, \dots, x_r) .

Si les hypothèses **H1-H3** sont vérifiées et qu'en plus, l'entrée $s(t)$ est un processus complexe circulaire, dont les moments d'ordre quatre sont finis (i.e., $E(s(t+\tau)s(t)) = 0$, $E(|s(t)|^4) < \infty$), et le bruit additive $\mathbf{b}(t)$ est gaussien, alors la séquence d'estimateurs $\hat{\mathbf{c}}_N = \text{vec}(\hat{\mathbf{C}}_N)$ est asymptotiquement normale avec une moyenne $\mathbf{c}_N = \text{vec}(\mathbf{C}_N)$ et une covariance $\Sigma_{\mathbf{c}}$. i.e.,

$$\sqrt{K}(\hat{\mathbf{c}}_N - \mathbf{c}_N) \xrightarrow{\mathcal{L}} \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{c}}).$$

La covariance $\Sigma_{\mathbf{c}}$ est donnée par

$$\begin{aligned} \Sigma_{\mathbf{c}} &= \kappa \tilde{\mathbf{c}}_N \tilde{\mathbf{c}}_N^H + \sum_{k=-(m-1)}^{m-1} \mathbf{C}_{N,k}^T \otimes \mathbf{C}_{N,k}^H, \\ \tilde{\mathbf{c}}_N &= \text{vec}(\mathbf{C}_N - \sigma_b^2 \mathbf{I}_n), \\ \kappa_s &= \text{cum}(s(t), s^*(t), s(t), s^*(t)), \end{aligned} \quad (3.31)$$

(On dit que κ_s est le kurtosis du signal d'entrée $s(t)$, dans le cas où $s(t)$ est soumis à l'hypothèse **H2**).

Démonstration : voir [52]

Pour établir la normalité asymptotique du vecteur $\hat{\mathbf{v}}_0$, on utilise ce qui est communément appelé 'théorème de continuité' qui stipule qu'une statistique asymptotiquement normale 'transmit' sa normalité asymptotique à n'importe quel autre paramètre estimé à partir de cette statistique, à condition que la relation qui lie ce paramètre à la statistique soit suffisamment régulière sur un voisinage de la valeur exacte (asymptotique) de la statistique. Plus précisément, on a le théorème suivant [20]

Théorème 3.2 : "Théorème de continuité"

Soient $\boldsymbol{\theta}_K$ une séquence de vecteurs aléatoires, asymptotiquement normale, de moyenne asymptotique $\boldsymbol{\theta}$ et de matrice de covariance asymptotique $\Sigma_{\boldsymbol{\theta}}$ et $\boldsymbol{\omega}(\boldsymbol{\theta}_K) = [\omega_1(\boldsymbol{\theta}_K) \ \dots \ \omega_{n_{\boldsymbol{\omega}}}(\boldsymbol{\theta}_K)]^T$ une fonction vectorielle réelle définie sur un voisinage de $\boldsymbol{\theta}$, telle que chacun de ses éléments a une différentielle non-nulle au point $\boldsymbol{\theta}$, i.e. $\mathbf{D}\omega_k(\boldsymbol{\theta}) \neq \mathbf{0}$, $k = 1, \dots, n_{\boldsymbol{\omega}}$. Dans ce cas, $\boldsymbol{\omega}(\boldsymbol{\theta}_K)$ est une séquence de vecteurs aléatoires de dimension $n_{\boldsymbol{\omega}}$, de moyenne asymptotique $\boldsymbol{\omega}(\boldsymbol{\theta})$ et de covariance asymptotique $\Sigma_{\boldsymbol{\omega}} = [\Sigma_{\omega_i, \omega_j}]_{1 \leq i, j \leq n_{\boldsymbol{\omega}}}$ donnée par

$$\Sigma_{\omega_i, \omega_j} = \mathbf{D}\omega_i^T(\boldsymbol{\theta}) \Sigma_{\boldsymbol{\theta}} \mathbf{D}\omega_j(\boldsymbol{\theta}).$$

Démonstration : voir [20]

L'application du théorème précédent aux estimés de \mathbf{v}_0 mène au théorème suivant

Théorème 3.3 *Tenant compte des hypothèses citées ci-dessus, le vecteur aléatoire $\hat{\mathbf{v}}_0$ suit, asymptotiquement, une loi gaussienne de moyenne \mathbf{v}_0 et de covariance $\Sigma_{\mathbf{v}}$, i.e.*

$$\sqrt{K}(\hat{\mathbf{v}}_0 - \mathbf{v}_0) \xrightarrow{\mathcal{L}} \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{v}}).$$

L'expression de $\Sigma_{\mathbf{v}}$ est donnée par

$$\Sigma_{\mathbf{v}} = \mathbf{M}\Sigma_{\mathbf{c}}\mathbf{M}^H,$$

où $\Sigma_{\mathbf{c}}$ est la matrice de covariance asymptotique de l'estimation échantillonnée du vecteur $\mathbf{c}_N = \text{vec}(\mathbf{C}_N)$ donnée par le **Lemme 3.1** et \mathbf{M} est une matrice donnée par

$$\begin{aligned} \mathbf{M} &= r \left(\mathbf{I}_n - \frac{\mathbf{v}\mathbf{v}^H}{\|\mathbf{v}\|^2} \right) [(\tilde{\mathbf{v}}^T \otimes \mathbf{I}_n)\mathbf{\Gamma} - \mathbf{W}\mathbf{M}_2\mathbf{M}_1], \\ \mathbf{\Gamma} &= \begin{bmatrix} \mathbf{W}^T(:, 1) \otimes (\lambda_1 \mathbf{I}_n - \mathbf{C}_N)^\# \\ \vdots \\ \mathbf{W}^T(:, d) \otimes (\lambda_d \mathbf{I}_n - \mathbf{C}_N)^\# \end{bmatrix}, \\ \mathbf{M}_1 &= [(\mathbf{C}_N \mathcal{J}_{p,n-p,0} \mathbf{T})^T \otimes \mathbf{I}_d] \mathbf{U}_{n,d} \mathbf{\Gamma}^* \mathbf{U}_{n,n} + [\mathbf{I}_d \otimes (\mathbf{T}^H \mathcal{J}_{p,n-p,0}^T \mathbf{C}_N)] \mathbf{\Gamma} \\ &\quad + (\mathcal{J}_{p,n-p,0} \mathbf{T})^T \otimes \mathbf{W}^H + \mathbf{W}^T \otimes (\mathbf{T}^H \mathcal{J}_{p,n-p,0}^T), \\ \mathbf{M}_2 &= \tilde{\mathbf{v}}^T \otimes \mathbf{Q}', \\ \mathbf{U}_{\alpha,\beta} &= \sum_{i=1}^{\alpha} \sum_{j=1}^{\beta} (\mathbf{e}_i^\alpha [\mathbf{e}_j^\beta]^T) \otimes (\mathbf{e}_j^\beta [\mathbf{e}_i^\alpha]^T), \\ \mathbf{Q}' &= \begin{cases} \mathbf{Q}^\#, & \text{dans le cas d'une contrainte quadratique} \\ \mathcal{J}_1 (\mathcal{J}_1^T \mathbf{Q} \mathcal{J}_1)^{-1} \mathcal{J}_1^T, & \text{dans le cas d'une contrainte linéaire} \end{cases}, \end{aligned}$$

$\mathbf{U}_{\alpha,\beta}$ est une matrice de permutation, \mathbf{e}_k^l est le k -ème vecteur-colonne de la matrice \mathbf{I}_l et $\lambda_1 > \lambda_2 \geq \dots \geq \lambda_d$ sont les d principales valeurs propres de la matrice \mathbf{C}_N associées aux vecteurs propres $\mathbf{W}(:, 1), \dots, \mathbf{W}(:, d)$, respectivement.

Démonstration : voir [52].

3.4.2 Validation des expressions des covariances asymptotiques

Dans ce paragraphe, on se propose de valider par des simulations numériques les expressions théoriques obtenues dans le paragraphe précédent. On considère un canal SIMO ($q = 1$, $p = 3$, et $L = 4$), généré aléatoirement en utilisant une distribution de Rayleigh. Le signal d'entrée est une séquence iid QAM4. La largeur de la fenêtre d'observation est $N = 6$. Les expressions théoriques sont comparées aux estimations empiriques obtenues sur un moyennage de 100 réalisations indépendantes. Le critère de performance utilisé ici, est l'erreur quadratique moyenne relative (*Relative Mean Square Error*, RMSE), définie comme étant la moyenne échantillonnée sur l'ensemble des réalisations indépendantes de l'estimation totale de la perte en MSE, i.e., $\widehat{\text{MSE}}_{opt} - \text{MSE}_{opt}$. Cette quantité est comparée avec son expression asymptotique exacte divisée par le nombre d'échantillons K , $\varepsilon_K = \frac{1}{K}\varepsilon = \frac{1}{K}\text{trace}(\mathbf{C}_N \Sigma_{\mathbf{v}})$. Le rapport signal à bruit est défini (en dB) par $\text{SNR} = -20 \log(\sigma_b)$.

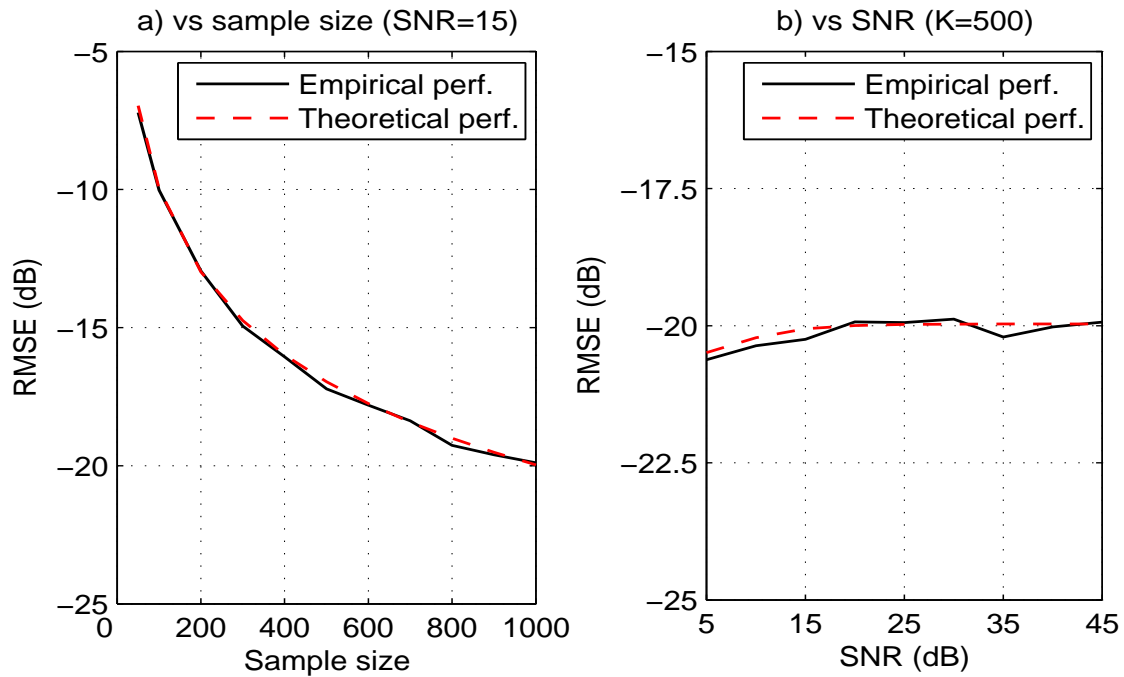


FIG. 3.1 – Perte asymptotique en performances : contrainte quadratique.

La Fig. 3.1a compare, dans le cas d'une contrainte quadratique, le RMSE empirique (trait continu) à son correspondant théorique ε_K (trait discontinu) en fonction du nombre d'échantillons K , pour un SNR égale à 15dB. On voit que l'expression théorique du RMSE est valide même pour une cinquantaine d'échantillons, ce qui veut dire que les conditions asymptotiques sont atteintes pour une courte taille d'échantillons. Dans la Fig. 3.1b les RMSE empirique (trait continu) et théorique (trait discontinu) sont tracés en fonction du SNR, pour un nombre d'échantillons $K = 500$. Cette figure démontre la parfaite concordance entre résultats théoriques et expérimentaux. Des résultats similaires sont obtenus dans le cas d'une contrainte linéaire.

3.5 Résultats des simulations et discussion

Dans cette section nous illustrons les performances des algorithmes d'égalisation aveugle proposés par quelques exemples de simulation. Nos testes sont basés sur des canaux SIMO et MIMO dont les coefficients sont générés aléatoirement, pour chaque exécution, suivant une distribution gaussienne complexe. Comme dans le paragraphe 1.6.5, les signaux d'entrée sont des séquences iid appartenant à un alphabet fini de phase et d'amplitude à quatre états QAM4 ($s(t) = Ae^{j\theta}$, $A \in \{-1, 1\}$ et $\theta \in \{\frac{\pi}{4}, \frac{3\pi}{4}\}$)³. Comme indice de performance on estime le MSE moyen par

$$\text{MSE} = \frac{1}{qK} \sum_{t=\tau}^{K+\tau-1} \|\mathbf{s}(t - \tau) - \hat{\mathbf{V}}_{\tau}^H \mathbf{x}_N(t)\|^2,$$

dans le cas d'un traitement par bloc, et le MSE instantané est estimé par

$$\text{MSE}(t) = \frac{1}{q} \|\mathbf{s}(t - \tau) - \hat{\mathbf{V}}_{\tau}^H \mathbf{x}_N(t)\|^2$$

dans le cas d'un traitement adaptative, le tout est moyenné sur une centaine de réalisations indépendantes. $\hat{\mathbf{V}}_{\tau}$ étant une estimation échantillonnée (obtenue à partir de K échantillons) du filtre égaliseur \mathbf{V}_{τ} (obtenu à partir de statistiques exactes). Le MSE est alors comparé au MSE optimal donné par

$$\text{MSE}_{opt} = \frac{1}{q} E(\|\mathbf{s}(t - \tau) - \mathbf{V}_{\tau}^H \mathbf{x}_N(t)\|^2) = \frac{1}{q} \text{trace}(\mathbf{I}_q - \mathbf{G}_{\tau}^H \mathbf{C}_N^{-1} \mathbf{G}_{\tau}).$$

Le rapport signal à bruit est défini par $\text{SNR} = -20 \log(\sigma_b)$.

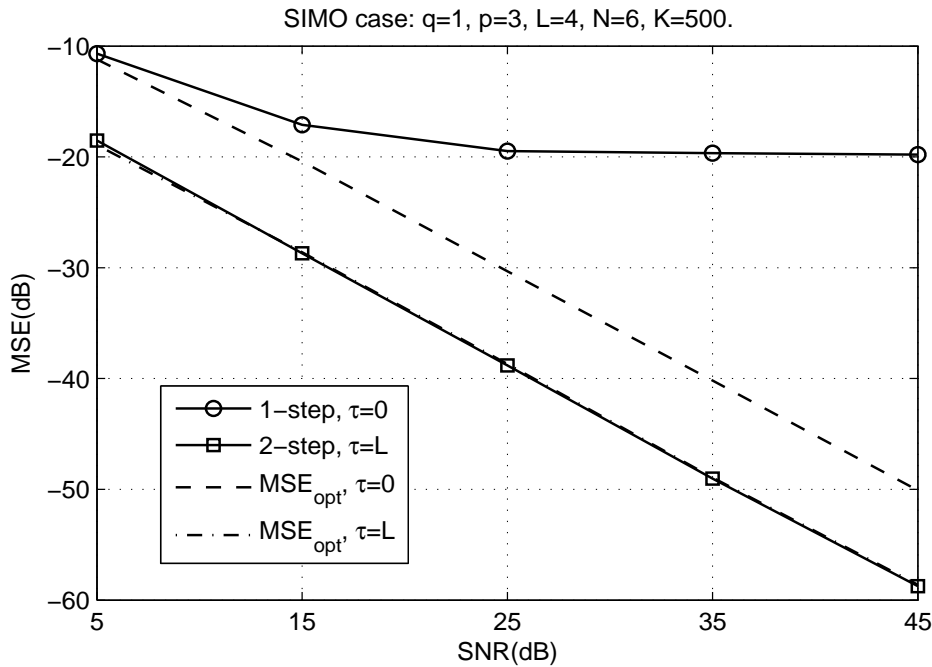


FIG. 3.2 – Performances de l'égaliseur, cas SIMO.

³Ici, le symbole j désigne un nombre complexe tel que $j^2 = -1$.

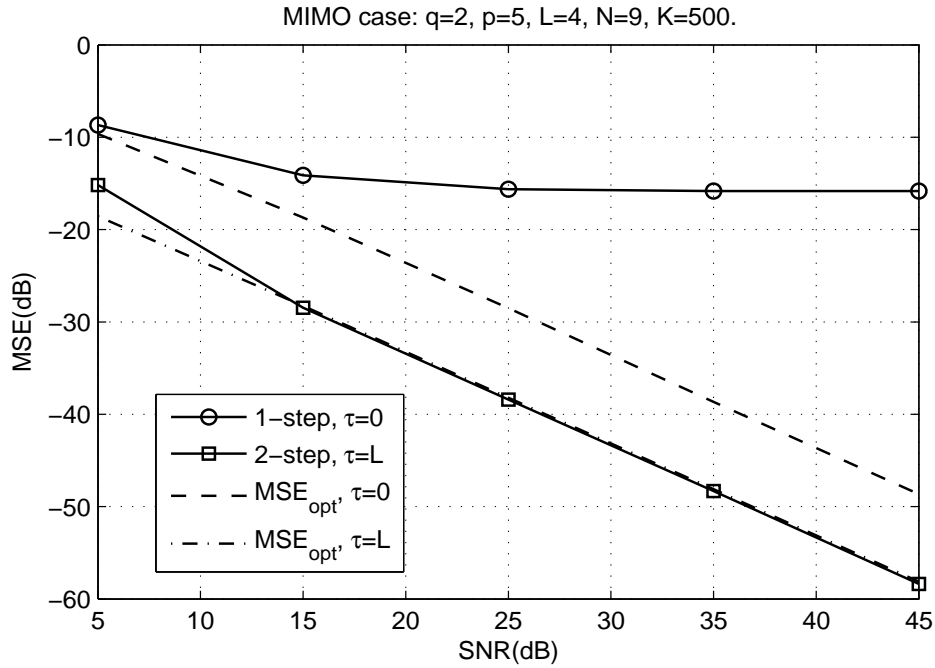


FIG. 3.3 – Performances de l'égaliseur, cas MIMO.

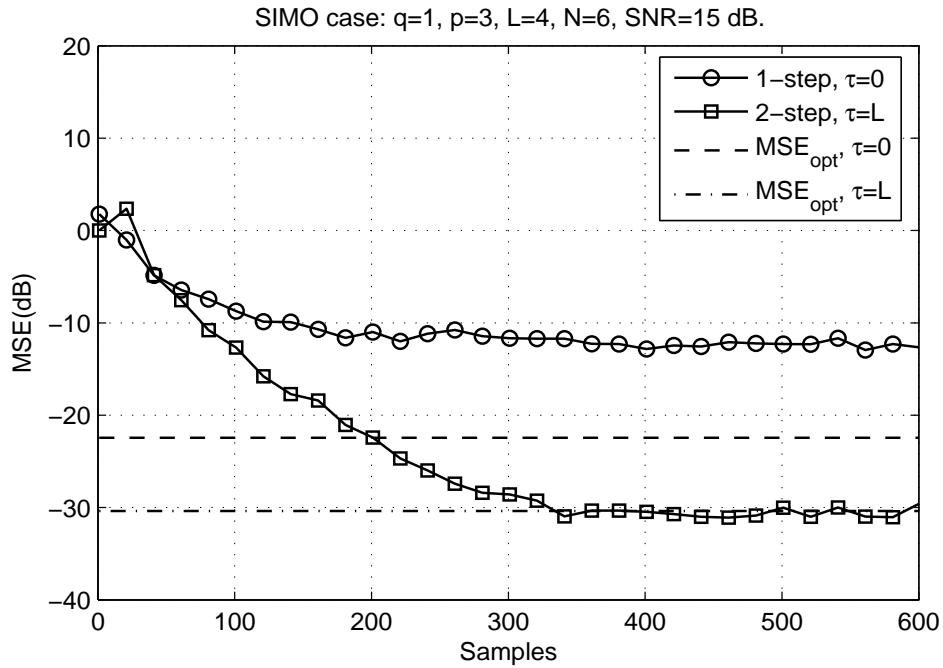


FIG. 3.4 – Convergence de l'égaliseur adaptatif, cas SIMO.

3.5.1 Évaluation des performances

Dans cette expérimentation, nous allons explorer les performances de notre algorithme. Dans la Fig. 3.2 (cas SIMO avec contrainte quadratique) et la Fig. 3.3 (cas MIMO) on trace le MSE (en dB) en

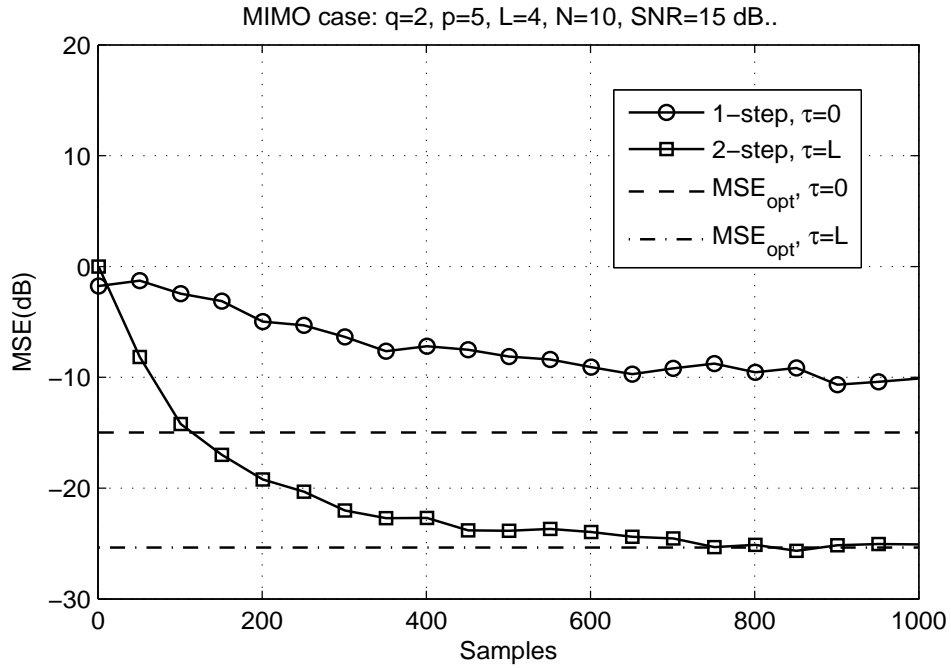


FIG. 3.5 – Convergence de l'égaliseur adaptatif, cas MIMO.

fonction du SNR (en dB) pour un nombre d'échantillons $K = 500$. On peut observer la baisse en performance du filtre MMSE à retard zéro comparé au filtre optimal, chose qui est due (comme c'est expliquée plus haut) à la procédure d'estimation aveugle. Aussi, ça illustre l'efficacité de l'approche à deux-étapes, qui non seulement compense la perte en performance mais aussi elle permet de choisir des retards d'égalisation non-nuls, ce qui aide à améliorer les performances dans l'ensemble. La Fig. 3.4 (cas SIMO avec contrainte quadratique) et la Fig. 3.5 (cas MIMO) donne une idée sur la convergence de la version adaptative de l'algorithme et du bon fonctionnement de la procédure à deux-étapes même dans le cas adaptatif. Le SNR étant fixé à 15 dB.

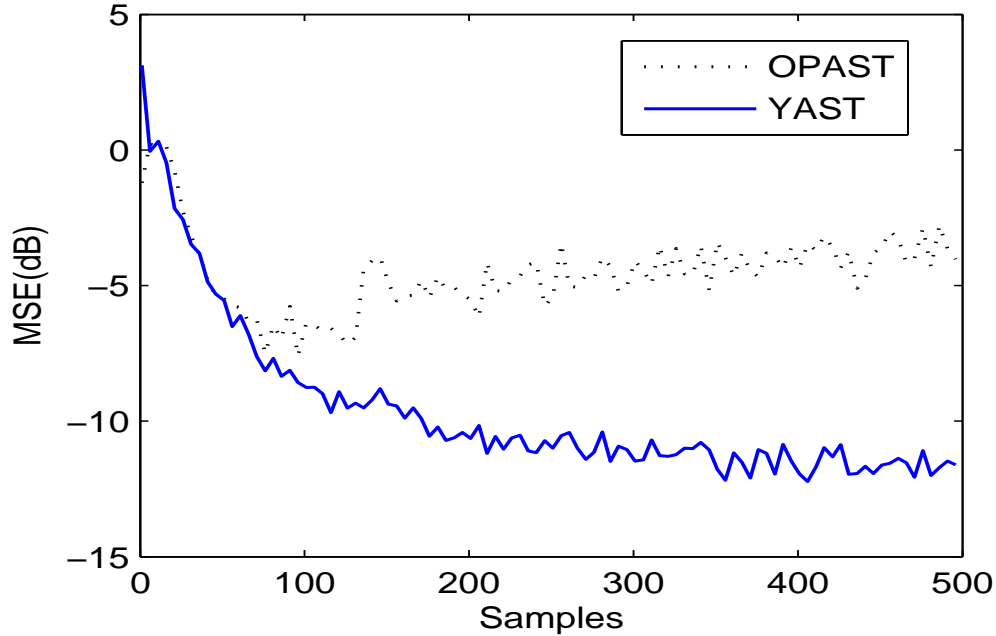


FIG. 3.6 – Convergence de l'algorithme d'égalisation adaptatif dans le cas d'un canal variable dans le temps.

La Fig. 3.6 compare, dans le cas où les paramètres du canal varient rapidement, les performances en poursuite de l'algorithme adaptatif en utilisant YAST [11] et OPASt [5] (*Orthogonal Projection Approximation Subspace Tracking algorithm*) respectivement, comme algorithmes suiveurs de sous-espaces. Nous avons choisi comme modèle de canal variable dans le temps celui de la référence [78] (cas SIMO : $q = 1$, $p = 3$ et $L = 4$), pour un SNR=15dB. Comme on peut l'observer, l'algorithme d'égalisation adaptative utilisant YAST arrive à suivre les variations du canal, alors qu'il échoue à le faire dans le cas où c'est OPASt qui est utilisé.

La Fig. 3.7 compare dans le cas SIMO ($q = 1$, $p = 3$ et $L = 4$) les performances de notre égaliseur MMSE à retard zéro, dans le cas d'un traitement en bloc, cas d'une contrainte linéaire (des résultats similaires sont obtenus dans le cas d'une contrainte quadratique), avec ceux des algorithmes GRDA [33], LPA [71], OPDA [27] RSFA [43], LSS [75] et MRE [38], respectivement. On rappelle que dans le cas d'une égalisation indirecte (comme c'est le cas des algorithmes RSFA, OPDA et LSS), et après estimation des coefficients du canal, on calcule le filtre égaliseur MMSE \mathbf{v}_0 en utilisant l'expression (1.21). Les courbes représentent le MSE estimé en fonction du SNR, pour un nombre d'échantillons $K = 500$. Comme on peut l'observer, toutes ces approches donnent des résultats très proches les uns des autres, exception faite pour l'algorithme MRE qui présente de meilleures performances à fort SNR. En outre, l'approche MRE est la plus coûteuse du point de vue calcul numérique, alors que notre approche présente la plus faible complexité de calcul (voir Fig. 3.8).

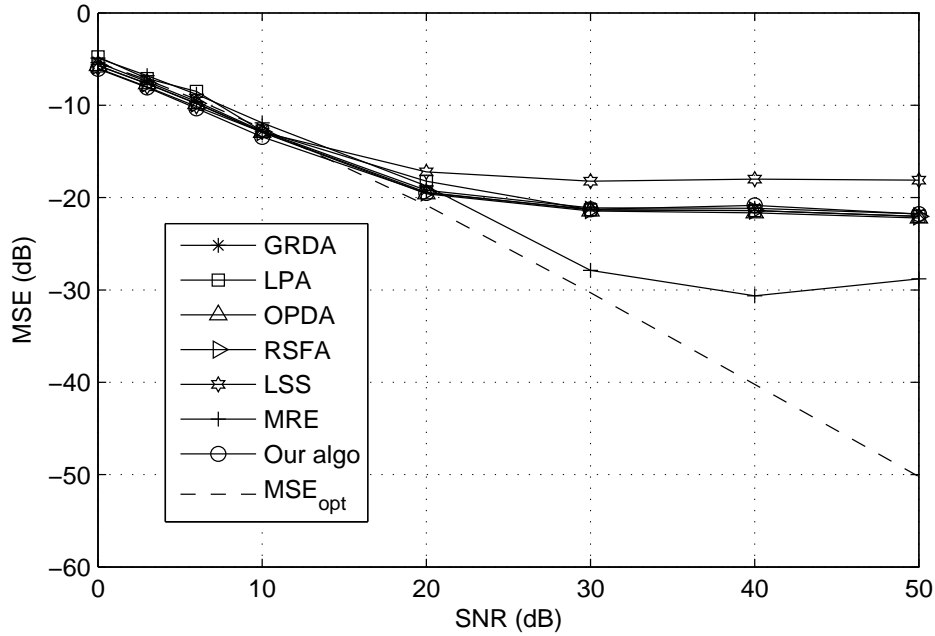


FIG. 3.7 – Comparison des performances d'égaliseurs SIMO ($q = 1$, $p = 3$, $L = 4$, $N = 6$), dans le cas d'un traitement en bloc.

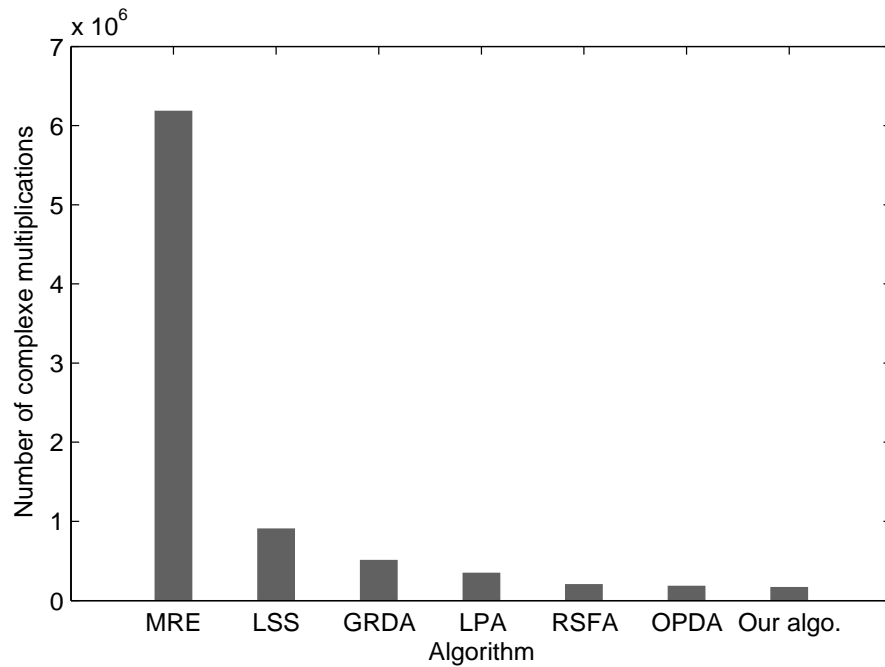


FIG. 3.8 – Comparison des coûts de calcul d'algorithmes d'égalisation SIMO ($q = 1$, $p = 3$, $L = 4$, $N = 6$), dans le cas d'un traitement en bloc.

La Fig. 3.9 compare dans le cas SIMO ($q = 1$, $p = 3$ et $L = 4$) les performances, dans le cas d'un traitement adaptatif, de notre égaliseur MMSE à retard zéro avec ceux des algorithmes LPA [71], et MRE

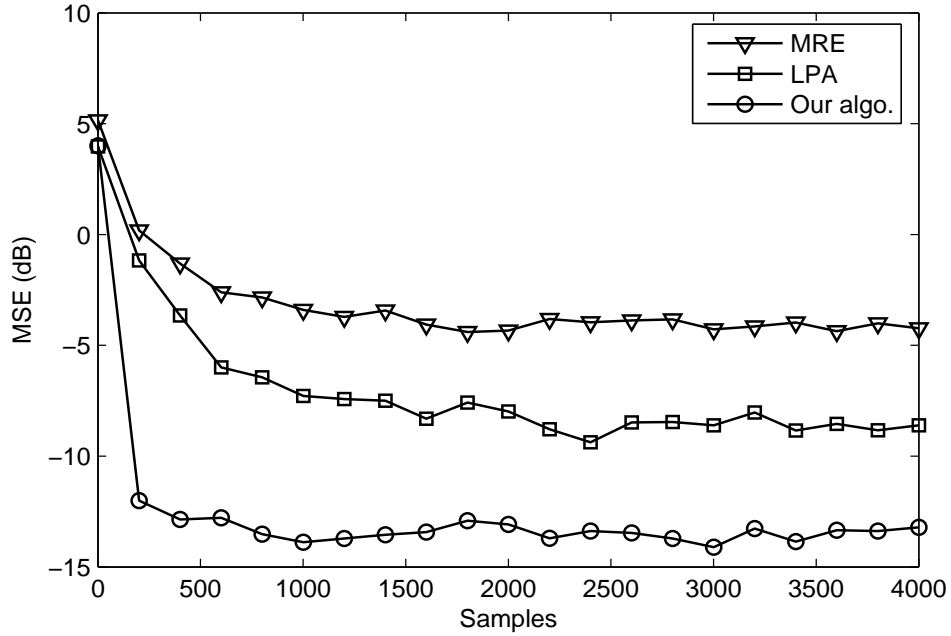


FIG. 3.9 – *Comparison des performances d'égaliseurs SIMO ($q = 1, p = 3, L = 4, N = 6$), dans le cas d'un traitement adaptatif.*

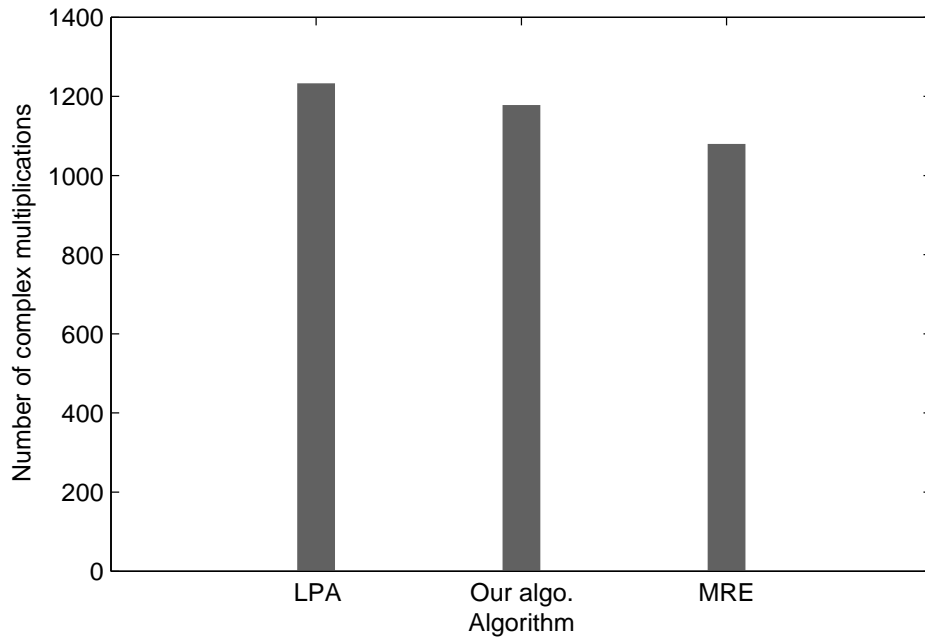


FIG. 3.10 – *Comparison des coûts de calcul d'algorithmes d'égalisation SIMO ($q = 1, p = 3, L = 4, N = 6$), dans le cas d'un traitement adaptatif.*

[38], respectivement. Les courbes représentent le MSE estimé en fonction du temps, pour un SNR=15 dB. Il est clair que même si les trois approches ont une complexité de calcul du même ordre (comme

le montre bien la Fig. 3.10), néanmoins, notre approche est de loin celle qui a la plus grande vitesse de convergence. Ceci est dû au fait que dans son implémentation, on utilise le principe de l'approximation par projection [88]. Cette approximation accélère, de manière très significative, la convergence de notre algorithme par rapport à des approches telles LPA ou MRE, dont l'implémentation est basée sur la technique du gradient stochastique.

3.5.2 Robustesse aux erreurs de surestimation de l'ordre du canal

Cette expérimentation, est dédié à l'étude de la robustesse aux erreurs de surestimation de l'ordre du canal. La Fig. 3.11 (resp. Fig. 3.12) représente le MSE en fonction de l'ordre surestimé du canal, pour un SNR=15dB et un nombre d'échantillons $K = 500$ (resp. $K = 2000$). Les courbes comparent, dans le cas SIMO ($q = 1, p = 3$ et l'ordre exacte du canal est $L = 4$), le MSE obtenu par notre algorithme avec contrainte quadratique (*Quadratic Constraint, Q.C.*) et contrainte linéaire (*Linear Constraint, L.C.*) respectivement, à ceux obtenus par les algorithmes LPA [71], OPDA [27] RSFA [43] et LSS [75], respectivement. Il est facile de constater que notre approche et l'approche RSFA ont un même niveau de performances et qui est nettement supérieur à celui des autres approches. En effet, notre approche et l'approche RSFA sont pratiquement insensibles aux erreurs de surestimation de l'ordre du canal. Néanmoins, il faut noter que notre approche offre de meilleurs avantages par rapport à la technique RSFA, à savoir, que notre méthode n'a nullement besoin de connaître l'ordre du canal avec exactitude, alors que la technique RSFA (et même la technique LSS) inclue une procédure de détection de l'ordre du canal, ce qui induit un coût de calcul supplémentaire. De même, dans le cas où le canal de transmission présente un évanouissement en fréquence, i.e. l'hypothèse **H1** donnée dans le paragraphe 1.5 n'est plus respectée⁴. Même si ses performances subissent une dégradation, notre approche donne de meilleurs résultats que ceux de la technique RSFA, comme le montre la Fig. 3.13, et qui compare les performances de notre égaliseur à ceux obtenues par la méthode RSFA, dans le cas d'un canal SIMO ($q = 1, p = 3$ l'ordre exacte du canal est $L = 4$), présentant un évanouissement en fréquence (i.e., les polynômes scalaires qui forment la fonction de transfert vectoriel du canal $\mathbf{h}(z) = \sum_{k=0}^L \mathbf{h}(k)z^{-k}$ ont au moins un zéro commun). D'autre part, il est clair que l'utilisation de la contrainte linéaire améliore de façon significative la robustesse aux erreurs de surestimation de l'ordre de notre filtre MMSE. Noter aussi, comme c'est expliqué en section 3.2.5, que les résultats de notre algorithme dans le cas d'une contrainte quadratique peuvent être améliorés d'avantage et ceci en augmentant le nombre d'échantillons K , c'est ce qu'on observe lorsqu'on compare les résultats des Fig. 3.11 et Fig. 3.12 dans le cas d'une contrainte quadratique.

3.5.3 Robustesse aux faibles valeurs de $\mathbf{H}(0)$

L'inconvénient majeur d'un égaliseur à retard nul est, en général, sa sensibilité aux faibles valeurs du premier coefficient du canal $\mathbf{H}(0)$. Les figures Fig. 3.14 et Fig. 3.15, illustrent la robustesse de l'algo-

⁴Noter que dans le cas d'un canal SIMO, l'hypothèse **H1** donnée dans le paragraphe 1.5 se réduit à ce que les polynômes scalaires qui forment la fonction de transfert vectoriel du canal $\mathbf{h}(z) = \sum_{k=0}^L \mathbf{h}(k)z^{-k}$ n'ont pas de zéros communs.

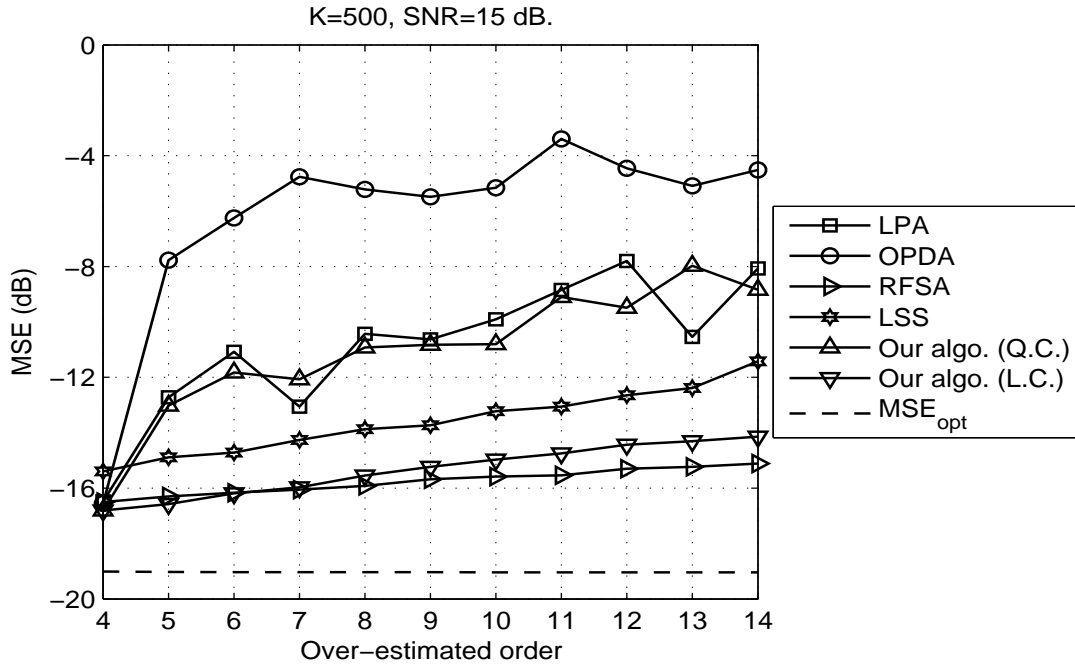


FIG. 3.11 – Comparaison de Robustesse (aux erreurs de surestimation de l'ordre du canal), l'ordre exacte est $L = 4$.

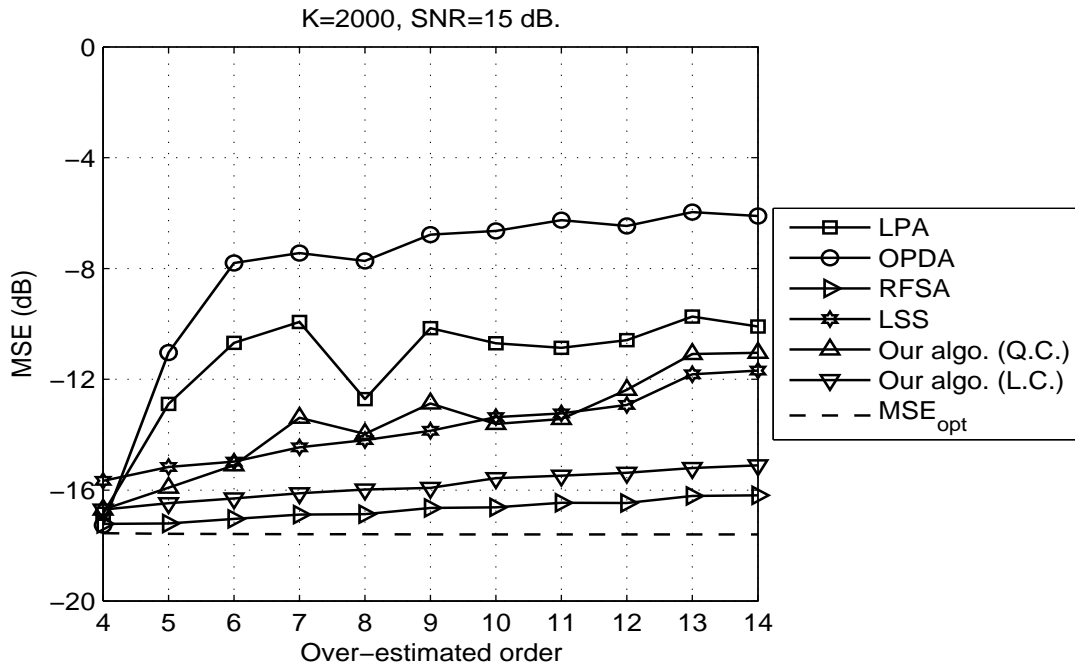


FIG. 3.12 – Comparaison de Robustesse (aux erreurs de surestimation de l'ordre du canal), l'ordre exacte est $L = 4$.

rithme proposé, quand $\mathbf{H}(0)$ prend de faibles valeurs. Plus précisément, on représente le MSE en fonction de la variance de $\mathbf{H}(0)$: $\sigma_{\mathbf{H}(0)}^2 \stackrel{\text{def}}{=} E(\|\mathbf{H}(0)\|^2)$, dans le cas d'un canal SIMO ($q = 1$, $p = 3$ et $L = 4$)

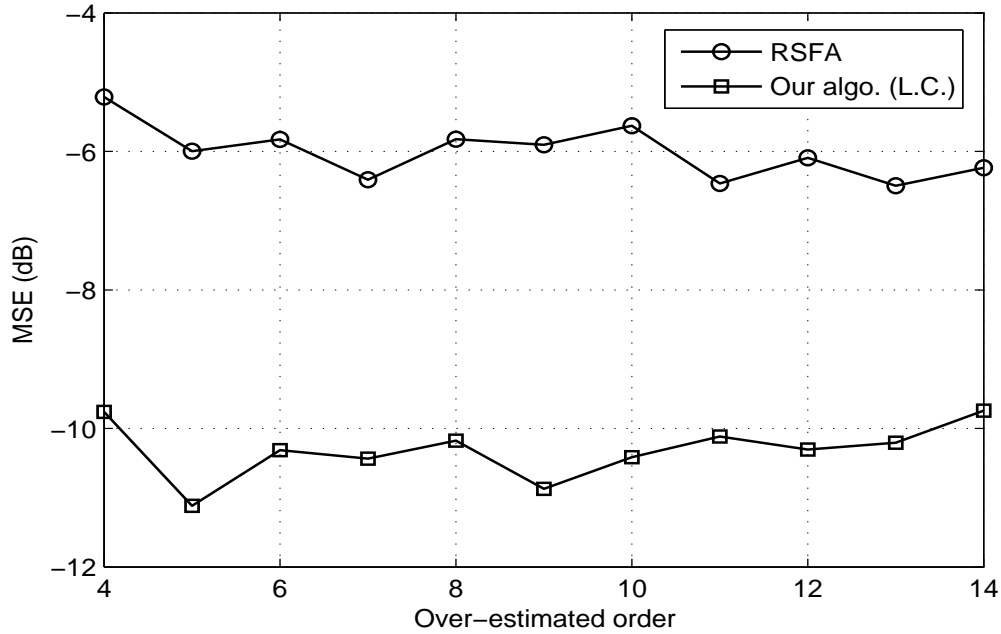


FIG. 3.13 – Comparaison de Robustesse (aux erreurs de surestimation de l'ordre du canal), cas d'un canal à évanouissement en fréquence, l'ordre exacte est $L = 4$.

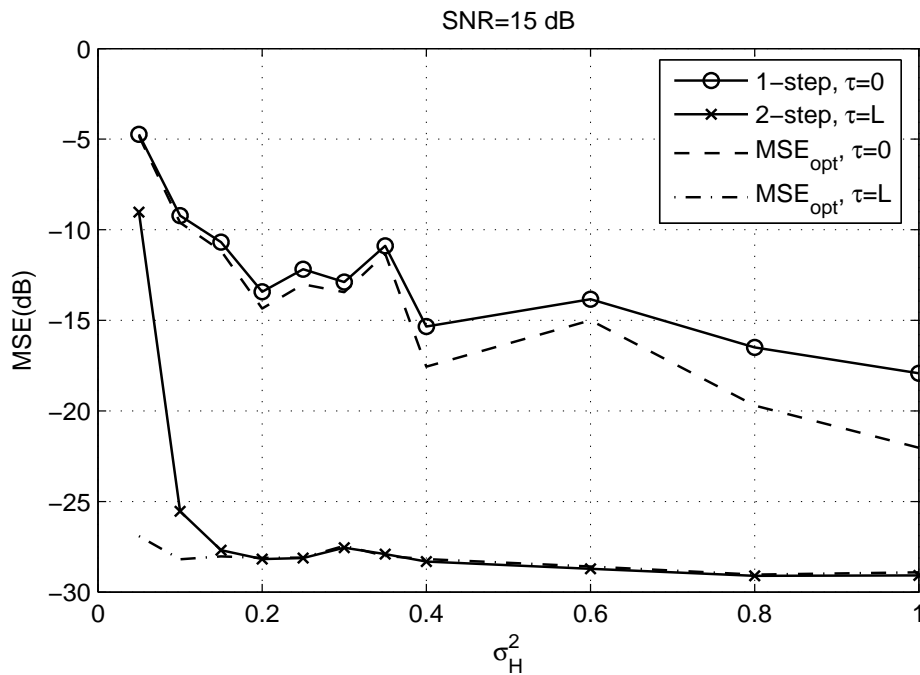


FIG. 3.14 – Robustesse aux faibles valeurs de $\mathbf{H}(0)$, SNR=15dB.

pour un nombre d'échantillons $K = 500$ et un SNR=15dB pour la Fig. 3.14 (resp. SNR=30dB pour la Fig. 3.15). Il est clair que pour un SNR faible ou moyen, l'algorithme donne des résultats satisfaisants si la variance de $\mathbf{H}(0)$ n'est pas inférieure à un certain seuil minimum (comme on le voit sur la courbe, il

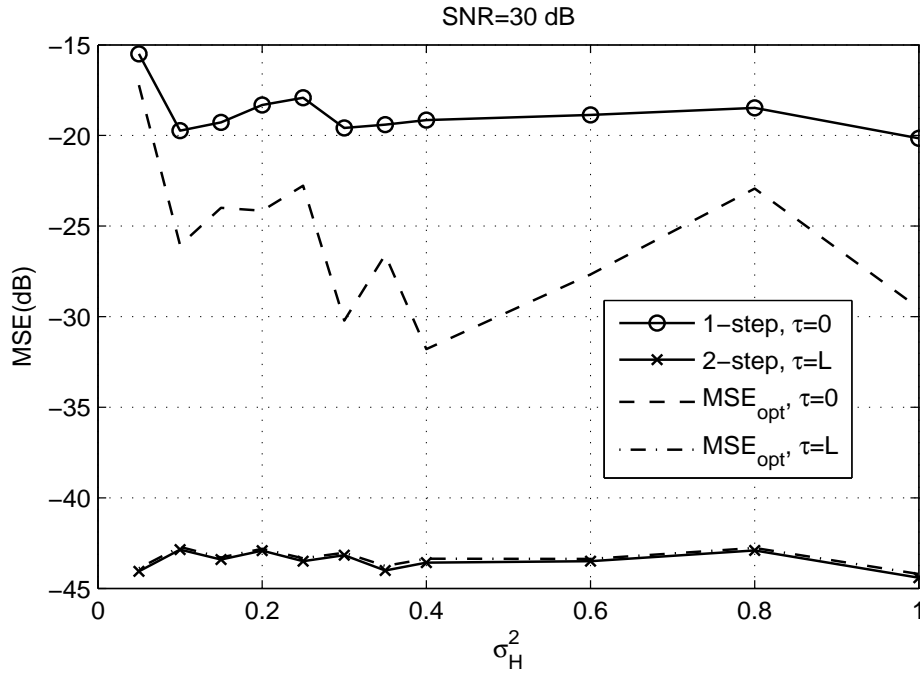


FIG. 3.15 – Robustesse aux faibles valeurs de $\mathbf{H}(0)$, SNR=30dB.

est nécessaire que $\sigma_{\mathbf{H}(0)}^2 \geq 0.15$). Cependant, ce seuil est encore d'avantage plus faible dans le cas d'un fort SNR, comme le montre la Fig. 3.15

3.5.4 Influence du nombre de capteurs

Les figures Fig. 3.16 et Fig. 3.17 représentent l'évolution du MSE en fonction du nombre de capteurs pour $q = 1$, $K = 500$ et SNR=5dB dans la Fig. 3.16 (resp. SNR=15dB dans la Fig. 3.17). On peut observer que pour un faible SNR, l'algorithme requière un minimum de degré de liberté en terme de nombre de capteurs (typiquement $p - q$ doit être supérieur à 2 ou 3), tandis que dans le cas d'un SNR moyen ou fort, p peut être aussi petit que $q + 1$. Des résultats similaires ont été observés dans le cas MIMO.

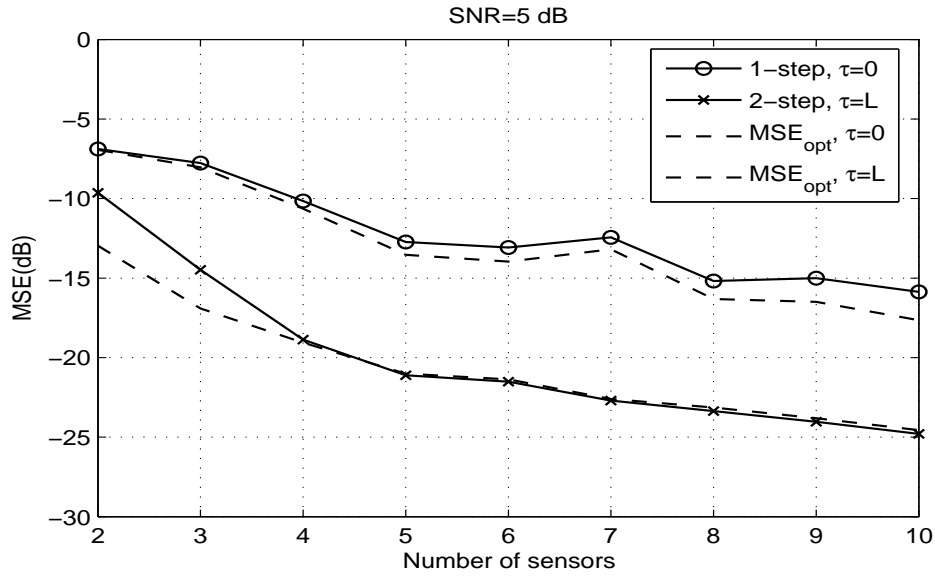


FIG. 3.16 – MSE en fonction du nombre de capteurs, SNR=5dB.

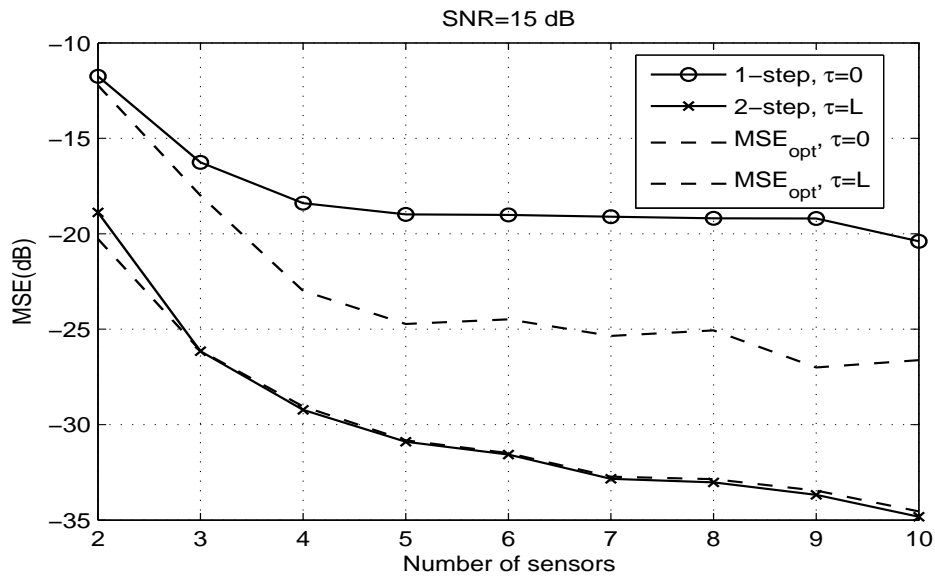


FIG. 3.17 – MSE en fonction du nombre de capteurs, SNR=15dB.

3.5.5 Discussion

Ces résultats que nous avons présenté ci-dessus, mettent en évidence les nombreux atouts dont dispose notre méthode. Un de ses principaux avantages est sa faible sensibilité aux erreurs de surestimation de l'ordre du canal. De même, quand l'ordre du canal est connu, l'algorithme proposé donne d'aussi bons résultats que les autres approches déjà existantes, tout en ayant le plus faible coût de calcul. Un autre point, sur lequel notre approche prend un meilleur avantage par rapport à d'autres approches, est son faible coût de calcul et la convergence rapide de sa version adaptative. Cependant, des méthodes telles

celles décrites plus haut, ont l'avantage d'estimer directement (en une seule étape) le filtre égaliseur à retard non-nul, chose qui s'avérera importante dans certains cas limites, où l'égaliseur à retard zéro faillit à donner des performances satisfaisantes (voir Fig. 3.14 et Fig. 3.15).

3.6 Conclusion

Dans ce chapitre, nous avons présenté une méthode originale d'égalisation aveugle du second ordre, de systèmes multicanaux (SIMO et MIMO) à réponse impulsionnelle finie. Des implémentations en bloc et adaptative rapide sont proposées. Une procédure dite à *deux-étapes* a été développée et qui exploite la connaissance a priori de l'alphabet auquel appartiennent les symboles émis, afin d'obtenir des retards d'égalisation non-nuls et du coup améliorer les performances. Une étude sur la robustesse de cette approche, aux erreurs de surestimation de l'ordre du canal, et qui a été confirmée par des simulations, a été effectuée. De même, dans le cas SIMO, nous avons effectué une analyse asymptotique des performances, dont les résultats théoriques ont été validés par des simulations numériques. En fin, des expérimentations par simulation sont présentées, afin d'illustrer les performances de cette nouvelle approche, tout en les comparant à ceux des approches déjà existantes.

Chapitre 4

Égaliseur aveugle indépendant de l'ordre du canal

4.1 Introduction

Dans ce chapitre nous développons une nouvelle méthode d'égalisation MMSE aveugle du second ordre, de systèmes SIMO à réponse impulsionnelle finie. Les principaux atouts de cette nouvelle technique sont, d'abord, sa totale indépendance de l'ordre du canal, ensuite le faible coût de calcul de sa version adaptative tout en gardant une convergence rapide. Plus précisément, on estime d'abord l'égaliseur à retard zéro en exploitant le fait qu'il appartient au noyau de la matrice d'autocorrélation tronqué. En suite et de la même façon et pour les mêmes raisons que pour la méthode présentée au chapitre 3, une procédure à deux étapes est incluse dans l'algorithme afin de permettre la compensation de la baisse en performances comparé au cas non-aveugle, et de choisir des retards d'égalisation non-nuls. Tirant profit de l'idée de l'approximation de la projection [88] et de la propriété du décalage-invariant des matrices d'autocorrélation des données temporelles, on donne une implémentation adaptative rapide de l'algorithme pour ainsi réduire la complexité de calcul de $O(n^3)$ à $O(np)$ (p étant le nombre de capteurs et n la dimension du vecteur de données).

4.2 Développement de l'algorithme

Considérons le modèle de données du canal décrit au paragraphe 1.5 dans le cas SIMO ($q = 1$). Tenant compte des relations (3.1) et (3.2), l'égaliseur MMSE à retard zéro \mathbf{v}_0 est alors donné implicitement

par

$$\mathbf{C}_N \mathbf{v}_0 = \begin{bmatrix} \mathbf{h}(0) \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}. \quad (4.1)$$

Si on tronque les p premières lignes du système (4.1), on obtient alors

$$\bar{\mathbf{C}} \mathbf{v}_0 = \mathbf{0} \quad (4.2)$$

où $\bar{\mathbf{C}}$ est la matrice donnée par l'équation (3.8). Il est évident que \mathbf{v}_0 appartient au noyau droit de la matrice $\bar{\mathbf{C}}$, alors en utilisant quelques résultats théoriques rapportés dans [4], on pourra établir le lien entre la structure du noyau droit de $\bar{\mathbf{C}}$ et l'égaliseur MMSE à retard zéro. Le lemme suivant permet de caractériser la structure du noyau droit de $\bar{\mathbf{C}}$

Lemme 4.1 *Tenant compte des hypothèses citées plus haut et pour $N > L + 1$, le noyau droit de $\bar{\mathbf{C}}$: $\text{null}_r(\bar{\mathbf{C}}) = \{\mathbf{z} \in \mathbb{C}^n : \bar{\mathbf{C}}\mathbf{z} = \mathbf{0}\}$, est un sous-espace vectoriel de dimension p , où seulement une seule direction de ce sous-espace appartient au sous-espace signal.*

Démonstration : voir Annexe A.

Pour extraire la direction de $\text{null}_r(\bar{\mathbf{C}})$ qui appartient au sous-espace signal (la direction de \mathbf{v}_0), on utilise le critère d'égalisation par pique (*Equalization Peak Criterion, EPC*) [4], on estime \mathbf{v}_0 parmi les vecteurs à norme unité, de sorte que l'énergie à la sortie du filtre égaliseur appliqué au vecteur de données $\mathbf{x}_N(t)$ soit maximale. i.e.,

$$\max_{\mathbf{v} \in \text{null}_r(\bar{\mathbf{C}})} E(|\mathbf{v}^H \mathbf{x}_N(t)|^2) \Leftrightarrow \max_{\|\tilde{\mathbf{v}}\|=1} (\tilde{\mathbf{v}}^H \mathbf{B} \tilde{\mathbf{v}}) \quad (4.3)$$

où \mathbf{B} est une matrice $p \times p$ définie par $\mathbf{B} \stackrel{\text{def}}{=} \mathbf{A}^H \mathbf{C}_N \mathbf{A}$, la matrice $\mathbf{A} \in \mathbb{C}^{n \times p}$ étant une base orthonormale de $\text{null}_r(\bar{\mathbf{C}})$ et $\tilde{\mathbf{v}}$ est un vecteur de dimension p .

En effet, sous la contrainte norme unité, la contribution du bruit dans (4.3) est à puissance constante, ainsi la maximisation dans (4.3) concerne seulement le terme signal. D'ailleurs les autres $p - 1$ directions de $\text{null}_r(\bar{\mathbf{C}})$, qui sont complémentaires de \mathbf{v}_0 , sont presque tout le temps dans le sous-espace bruit (du moins à fort SNR). Par conséquent, la maximisation dans (4.3) mène essentiellement à la sélection de la direction désirée du vecteur \mathbf{v}_0 . Une implémentation par bloc de l'algorithme est résumée par la TABLE 4.1. Le coût de calcul de cette implémentation est de l'ordre de $(K + p)n^2 + (n + 1)p^2 + np + O(n^3)$.

Comme le processus aveugle d'estimation du filtre MMSE induit une perte en performance comparé au cas non-aveugle (voir paragraphe 1.6 et chapitre 3), on introduit ici (comme pour l'algorithme du chapitre 3) une approche à *deux étapes*, pour compenser cette perte et du coup obtenir des égaliseurs avec des retards non-nuls d'où une amélioration des performances. L'algorithme d'égalisation MMSE aveugle à deux étapes est résumé par la TABLE 4.2. Il en résulte alors, un coût global de ce processus de l'ordre de $(K + p + 1)n^2 + (K + p^2 + p)n + p^2 + 2O(n^3)$.

\mathbf{C}_N	$= \frac{1}{K} \sum_{t=N-1}^{K+N-2} \mathbf{x}_N(t) \mathbf{x}_N^H(t)$, (K : Nombre d'échantillons)
$(\mathbf{U}, \Sigma, \mathbf{V})$	$= \text{svd}(\mathbf{C}_N(p+1:n, :))$
\mathbf{A}	$= \mathbf{V}(:, n-p+1:n)$
\mathbf{B}	$= \mathbf{A}^H \mathbf{C}_N \mathbf{A}$
$\tilde{\mathbf{v}}$	$=$ vecteur propre dominant de \mathbf{B}
\mathbf{v}	$= \mathbf{A} \tilde{\mathbf{v}}$

TAB. 4.1 – Algorithme d'égalisation aveugle.

Estimer $\hat{s}(t)$, $t = 0..K-1$, en utilisant \mathbf{v} donné par la TABLE 4.1, i.e. $\hat{s}(t) = \mathbf{v}^H \mathbf{x}_N(t)$, à une constante multiplicative scalaire non-nulle près.	
\mathbf{g}_τ	$= \frac{1}{K} \sum_{t=\tau}^{K+\tau-1} \mathbf{x}_N(t) \hat{s}(t-\tau)^*$
\mathbf{v}_τ	$= \mathbf{C}_N^{-1} \mathbf{g}_\tau$

TAB. 4.2 – Procédure d'égalisation en deux étapes.

4.3 Implémentation Adaptative

Dans le but de réduire la complexité de calcul globale de l'algorithme de $O(n^3)$ à $O(np)$, une implémentation adaptative est proposée. Les colonnes de la matrice \mathbf{A} correspondent aux p vecteurs propres associés aux p plus petites valeurs propres de la matrice \mathbf{Q} de dimension $n \times n$ définie par

$$\mathbf{Q} \stackrel{\text{def}}{=} \bar{\mathbf{C}}^H \bar{\mathbf{C}}.$$

Tenant compte de la mise à jour récursive de la matrice \mathbf{C}_N , donnée par l'équation (3.24), la matrice $\bar{\mathbf{C}}$ est alors remplacée par la relation récursive suivante

$$\bar{\mathbf{C}}(t) = \beta \bar{\mathbf{C}}(t-1) + \mathbf{x}_{N-1}(t-1) \mathbf{x}_N^H(t). \quad (4.4)$$

où $0 < \beta < 1$ est le facteur d'oubli. Ainsi, la matrice $\mathbf{Q}(t)$ est récursivement donnée par

$$\mathbf{Q}(t) = \beta^2 \mathbf{Q}(t-1) + \mathbf{q}_1(t) \mathbf{q}_1^H(t) + \mathbf{q}_2(t) \mathbf{q}_2^H(t), \quad (4.5)$$

où $\mathbf{q}_1(t)$ et $\mathbf{q}_2(t)$ sont deux vecteurs de dimension n donnés par

$$\begin{aligned} \mathbf{q}_1(t) &= \sigma(t-1) (\beta \mathbf{x}'(t-1) + \lambda(t-1) \mathbf{x}_N(t)), \\ \mathbf{q}_2(t) &= j\sigma(t-1) \left(\beta \mathbf{x}'(t-1) - \frac{1}{\lambda(t-1)} \mathbf{x}_N(t) \right), \\ \mathbf{x}'(t) &= \bar{\mathbf{C}}^H(t) \mathbf{x}_{N-1}(t), \\ \sigma(t) &= \sqrt{\frac{\lambda(t)}{1 + \lambda(t)^2}}, \\ \lambda(t) &= \frac{1}{2} (\|\mathbf{x}_{N-1}(t)\|^2 + \sqrt{4 + \|\mathbf{x}_{N-1}(t)\|^4}). \end{aligned} \quad (4.6)$$

Il s'en suit que la matrice $\mathbf{A}(t)$ peut être considérée comme une base orthonormale du sous-espace mineur généré par la matrice d'autocorrélation virtuelle $\mathbf{Q}(t)$, de la séquence de données $\{\mathbf{q}_1(t), \mathbf{q}_2(t)\}_{t \in \mathbb{Z}}$. A partir des vecteurs de données $\mathbf{q}_1(t)$ et $\mathbf{q}_2(t)$, on peut extraire, récursivement, les colonnes de la matrice $\mathbf{A}(t)$ en opérant par deux fois avec un algorithme d'estimation et de suivi de sous-espaces mineurs. Dans ce travail on utilise l'algorithme FDPMP [29] (*Fast Data Projection Method*) que résume la TABLE 4.3, où $\mathbf{e}_1 = [1 \ \mathbf{0}_{1 \times (p-1)}]^T$, $\nu > 0$ et la notation 'normalisation' signifie, normalisation des vecteur-colonnes de la matrice $\hat{\mathbf{T}}(t)$. Le choix de l'algorithme FDPMP est motivé par sa simplicité et ses remarquables performances en poursuite comparé à d'autres algorithmes existants de suivi de sous-espaces mineurs ayant la même complexité de calcul. En effet, les résultats de simulation (voir la section 4.4) ont montré que la plus part des algorithmes rapides (d'une complexité de calcul $O(np)$) d'estimation et de suivi de sous-espaces mineurs divergent immédiatement (e.g., OOJA [6]) ou présentent de mauvaises performances (e.g. PASTd [88], YAST [11], HFRANS [10],...), quand la dimension du sous-espace mineurs est très faible devant la dimension du sous-espace total i.e., $p \ll n$ (et c'est exactement la situation à laquelle on fait face ici). Ces imperfections sont dues essentiellement au problème des *minimums locaux*. Noter que l'algorithme FDPMP donne de meilleurs performances (voir Fig. 4.1).

$\begin{aligned} \mathbf{y}(t) &= \mathbf{A}^H(t-1)\mathbf{q}(t) \\ \mathbf{T}(t) &= \mathbf{A}(t-1) - \frac{\nu}{\ \mathbf{q}(t)\ ^2} \mathbf{q}(t)\mathbf{y}^H(t) \\ \mathbf{a}(t) &= \mathbf{y}(t) - \ \mathbf{y}(t)\ \mathbf{e}_1 \\ \hat{\mathbf{T}}(t) &= \mathbf{T}(t) - \frac{1}{\mathbf{q}^H(t)\mathbf{a}(t)} [\mathbf{T}(t)\mathbf{a}(t)]\mathbf{a}^H(t) \\ \mathbf{A}(t) &= \text{normalisation}(\hat{\mathbf{T}}(t)) \end{aligned}$
--

TAB. 4.3 – L'algorithme FDPMP.

En utilisant l'approximation de la projection $\mathbf{C}_N(t-1)\mathbf{A}(t) \approx \mathbf{C}_N(t-1)\mathbf{A}(t-1)$, qui est valide lorsque les variations de $\mathbf{A}(t)$ sont lentes [88], la matrice $\mathbf{B}(t)$ de dimension $p \times p$ est mise à jour en $O(np)$ opérations par la relation suivante

$$\mathbf{B}(t) = \beta \mathbf{B}(t-1) + \mathbf{b}(t)\mathbf{b}^H(t),$$

où $\mathbf{b}(t)$ est le vecteur de dimension p donné par

$$\mathbf{b}(t) = \mathbf{A}^H(t)\mathbf{x}_N(t).$$

Finalement, l'extraction du vecteur propre dominant de $\mathbf{B}(t)$ est obtenue en utilisant la méthode de puissance par

$$\tilde{\mathbf{v}}(t) = \frac{\mathbf{B}(t)\tilde{\mathbf{v}}(t-1)}{\|\mathbf{B}(t)\tilde{\mathbf{v}}(t-1)\|}. \quad (4.7)$$

Noter que le vecteur $\mathbf{x}'(t)$ de (4.6) est calculé en $O(np)$ opérations, en utilisant la propriété du décalage-invariant de la matrice d'autocorrélation (voir détails dans [52]). L'algorithme d'égalisation MMSE adaptative aveugle est donné par la TABLE 4.4, son coût de calcul est de l'ordre de $15np + n + 3p^2 + p$ opérations par itération.

$$\begin{aligned}
 \mathbf{q}_1(t) &= \sigma(t-1)(\beta \mathbf{x}'(t-1)) + \lambda(t-1)\mathbf{x}_N(t) \\
 \mathbf{q}_2(t) &= j\sigma(t-1)(\beta \mathbf{x}'(t-1) - \frac{1}{\lambda(t-1)}\mathbf{x}_N(t)) \\
 \mathbf{A}'(t) &= \text{FDPM}(\mathbf{A}(t-1), \mathbf{q}_1(t), \nu) \\
 \mathbf{A}(t) &= \text{FDPM}(\mathbf{A}'(t), \mathbf{q}_2(t), \nu) \\
 \mathbf{b}(t) &= \mathbf{A}^H(t)\mathbf{x}_N(t) \\
 \mathbf{B}(t) &= \beta \mathbf{B}(t-1) + \mathbf{b}(t)\mathbf{b}^H(t) \\
 \tilde{\mathbf{v}}(t) &= \frac{\mathbf{B}(t)\tilde{\mathbf{v}}(t-1)}{\|\mathbf{B}(t)\tilde{\mathbf{v}}(t-1)\|} \\
 \mathbf{x}'(t) &= \tilde{\mathbf{C}}^H(t)\mathbf{x}_{N-1}(t) \\
 \mathbf{v}(t) &= \mathbf{A}(t)\tilde{\mathbf{v}}(t) \\
 \lambda(t) &= \frac{1}{2}(\|\mathbf{x}_{N-1}(t)\|^2 + \sqrt{4 + \|\mathbf{x}_{N-1}(t)\|^4}) \\
 \sigma(t) &= \sqrt{\frac{\lambda(t)}{1+\lambda(t)^2}}
 \end{aligned}$$

TAB. 4.4 – Algorithme d'égalisation adaptative aveugle.

4.3.1 Procédure adaptative à deux étapes via l'algorithme du gradient stochastique normalisé

Comme (3.1) décrit un problème de minimisation sans contrainte, on pourra tout simplement appliquer la technique du gradient-descendant [46] pour obtenir ainsi une estimation récursive du vecteur égaliseur à retard τ , \mathbf{v}_τ donné par (3.1), à partir d'une estimation du signal émis $\hat{s}(t)$. Le gradient de $J(\mathbf{v}_\tau) = E(|s(t-\tau) - \mathbf{v}^H \mathbf{x}_N(t)|^2)$ est donné par $\nabla J(\mathbf{v}_\tau) = \mathbf{C}_N \mathbf{v}_\tau - \mathbf{g}_\tau$. D'où, le vecteur $\mathbf{v}_\tau(t)$ est mis à jour par

$$\mathbf{v}_\tau(t) = \mathbf{v}_\tau(t-1) - \mu(\hat{\mathbf{C}}_N(t)\mathbf{v}_\tau(t-1) - \hat{\mathbf{g}}_\tau(t)),$$

où $\hat{\mathbf{C}}_N(t)$ et $\hat{\mathbf{g}}_\tau(t)$ sont des estimés à l'instant t de la matrice d'autocorrélation \mathbf{C}_N donnée par (1.8) et le vecteur \mathbf{g}_τ donné par (3.2), respectivement. Afin de garantir la convergence de l'algorithme, le pas d'adaptation μ est choisi tel que $0 < \mu \ll \frac{1}{\lambda_1}$, où λ_1 est la plus grande valeur propre de la matrice \mathbf{C}_N . Par conséquent, on utilise le pas d'adaptation normalisé $\mu = \frac{\bar{\mu}}{\text{trace}(\mathbf{C}_N)}$. Ainsi, en choisissant $\bar{\mu}$ proche de l'unité on aura $\mu \ll \frac{1}{\lambda_1}$. Les estimés $\hat{\mathbf{C}}_N(t)$ et $\hat{\mathbf{g}}_\tau(t)$, seront remplacés par leur valeurs instantanés, i.e., $\hat{\mathbf{C}}_N(t) = \mathbf{x}_N(t)\mathbf{x}_N^H(t)$ et $\hat{\mathbf{g}}_\tau(t) = \mathbf{x}_N(t)\hat{s}^*(t-\tau)$, respectivement. Ce qui donne la mise à jour suivante du filtre égaliseur

$$\mathbf{v}_\tau(t) = \mathbf{v}_\tau(t-1) - \frac{\bar{\mu}\mathbf{x}_N(t)}{\|\mathbf{x}_N(t)\|^2}(\mathbf{x}_N^H(t)\mathbf{v}_\tau(t-1) - \hat{s}^*(t-\tau)).$$

Ainsi, la TABLE 4.5 résume la procédure d'égalisation adaptative à deux étapes via la technique du gradient stochastique normalisé (*Normalized Stochastic Gradient*, NSG), ce qui donne, dans ce cas, un coût de calcul globale de $15np + 4n + 3p^2 + p$.

4.3.2 Procédure adaptative à deux étapes via l'algorithme des moindres carrés récursif

L'avantage de l'approche décrite au paragraphe 4.3.1, est son faible coût de calcul (seulement $O(n)$ opérations par itération sont requises). Cependant, cette méthode souffre d'une convergence très lente.

Estimer $\hat{s}(t)$, en utilisant $\mathbf{v}(t)$ donné par la TABLE 4.4, i.e. $\hat{s}(t) = \mathbf{v}^H(t)\mathbf{x}_N(t)$, à une constante multiplicative scalaire non-nulle près. $\mathbf{x}''(t) = \bar{\mu} \frac{\mathbf{x}_N(t)}{\ \mathbf{x}_N(t)\ ^2}$ $\mathbf{v}_\tau(t) = \mathbf{v}_\tau(t-1) - \mathbf{x}''(t)(\mathbf{x}_N(t)^H \mathbf{v}_\tau(t-1) - \hat{s}^*(t-\tau))$
--

 TAB. 4.5 – *Procédure d'égalisation adaptative à deux étapes via la technique NSG.*

Une alternative qui permet d'améliorer la vitesse de convergence, consiste à calculer le vecteur \mathbf{v}_τ de l'équation (3.1) via la technique des moindres carrés récurrente (*Recursive Least Square*, RLS), dont le coût de calcul est évidemment plus élevé. En utilisant le lemme d'inversion matricielle [46], la matrice $\mathbf{F}(t) \stackrel{\text{def}}{=} \mathbf{C}_N^{-1}(t)$ est mise à jour par

$$\begin{aligned} \mathbf{F}(t) &= \frac{1}{\beta} (\mathbf{F}(t-1) - \mathbf{f}'(t)\mathbf{f}^H(t)), \\ \mathbf{f}(t) &= \mathbf{F}(t-1)\mathbf{x}_N(t), \\ \rho(t) &= \mathbf{f}^H(t)\mathbf{x}_N(t), \\ \mathbf{f}'(t) &= \frac{\mathbf{f}(t)}{\beta + \rho(t)}. \end{aligned}$$

Le vecteur \mathbf{g}_τ de l'équation (3.2) est alors remplacé par la relation récurrente suivante

$$\mathbf{g}_\tau(t) = \beta \mathbf{g}_\tau(t-1) + \mathbf{x}_N(t)\hat{s}^*(t-\tau).$$

Finalement, on abouti à l'expression récurrente du filtre égaliseur

$$\mathbf{v}_\tau(t) = \mathbf{v}_\tau(t-1) - \mathbf{f}'(t) (\mathbf{f}^H(t)\mathbf{g}_\tau(t-1) - \hat{s}^*(t-\tau)).$$

Ainsi, la TABLE 4.6 résume la procédure d'égalisation adaptative à deux étapes via la technique RLS. Noter que cette approche requiert $2n^2 + 15np + 5n + 3p^2 + p$ opérations par itération.

Estimer $\hat{s}(t)$, en utilisant $\mathbf{v}(t)$ donné par la TABLE 4.4, i.e. $\hat{s}(t) = \mathbf{v}^H(t)\mathbf{x}_N(t)$, à une constante multiplicative scalaire non-nulle près. $\mathbf{f}(t) = \mathbf{F}(t-1)\mathbf{x}_N(t)$ $\rho(t) = \mathbf{f}^H(t)\mathbf{x}_N(t)$ $\mathbf{f}'(t) = \frac{\mathbf{f}(t)}{\beta + \rho(t)}$ $\mathbf{v}_\tau(t) = \mathbf{v}_\tau(t-1) - \mathbf{f}'(t) (\mathbf{f}^H(t)\mathbf{g}_\tau(t-1) - \hat{s}^*(t-\tau))$ $\mathbf{F}(t) = \frac{1}{\beta} (\mathbf{F}(t-1) - \mathbf{f}'(t)\mathbf{f}^H(t))$ $\mathbf{g}_\tau(t) = \beta \mathbf{g}_\tau(t-1) + \mathbf{x}_N(t)\hat{s}^*(t-\tau)^*$

 TAB. 4.6 – *Procédure d'égalisation adaptative à deux étapes via la technique RLS.*

4.4 Résultats des simulations

Dans cette section nous illustrons les performances de l'algorithme d'égalisation aveugle proposé par quelques exemples de simulation. Comme pour les expérimentations présentées au paragraphe 1.6.5 et au paragraphe 2.10, nos tests sont basés sur des canaux SIMO ($q = 1$, $p = 3$ et $L = 4$), dont les coefficients sont générés aléatoirement selon une distribution gaussienne complexe. Le signal d'entrée est une séquence iid, appartenant à un alphabet fini de phase et d'amplitude à quatre états QAM4. La largeur de la fenêtre temporelle est $N = 6$.

4.4.1 Estimation du sous-espace mineur

Des expérimentations par simulation sont réalisées afin de comparer les performances de quelques algorithmes d'estimation et de suivi de sous-espaces mineurs. Il s'agit des algorithmes OOJA [6] (*Orthogonal OJA algorithm*), PASTd [88] (*Projection Approximation Subspace Tracking with deflation algorithm*), YAST [11] (*Yet Another Subspace Tracker algorithm*), HFRANS [10] (*Householder transformation based implementation of Fast Rayleigh's quotient-based Adaptive Noise Subspace algorithm*) et FDPM [29]. La dimension du sous-espace mineur est $p = 3$ et la dimension du sous-espace total est $n = Np = 18$. On mesure l'erreur d'estimation du sous-espace par le critère suivant

$$\gamma(t) = \frac{\text{trace}(\mathbf{A}(t)^H \mathbf{E}_1 \mathbf{E}_1^H \mathbf{A}(t))}{\text{trace}(\mathbf{A}(t)^H \mathbf{E}_2 \mathbf{E}_2^H \mathbf{A}(t))}, \quad (4.8)$$

moyenné sur 100 réalisations indépendantes, où \mathbf{E}_1 (resp. \mathbf{E}_2) est la valeur exacte du sous-espace principal de dimension $(n - p)$ (resp. sous-espace mineur de dimension p). Comme on peut le voir, en terme d'erreur d'estimation de sous-espace, la Fig. 4.1 montre que l'algorithme FDPM donne de meilleures performances que les autres, alors que l'algorithme OOJA diverge immédiatement.

4.4.2 Égalisation aveugle

Comme mesure de performance, on estime le MSE moyen par

$$\text{MSE} = \frac{1}{K} \sum_{t=\tau}^{K+\tau+1} (|s(t - \tau) - \hat{\mathbf{v}}_\tau^H \mathbf{x}_N(t)|^2)$$

dans le cas d'un traitement par bloc, et le MSE instantané par

$$\text{MSE}(t) = |s(t - \tau) - \hat{\mathbf{v}}_\tau(t)^H \mathbf{x}_N(t)|^2$$

dans le cas d'un traitement adaptative, le tout est moyenné sur 100 réalisations indépendantes. Le MSE est comparé au MSE théorique donné par

$$\text{MSE}_{opt} = 1 - \mathbf{g}_\tau^H \mathbf{C}_N^{-1} \mathbf{g}_\tau.$$

Le rapport signal à bruit (SNR : Signal to Noise Ratio) est défini par $\text{SNR} = -20 \log(\sigma_b)$.

Dans la Fig. 4.2 on trace le MSE (en dB) en fonction du SNR (en dB) pour un nombre d'échantillons

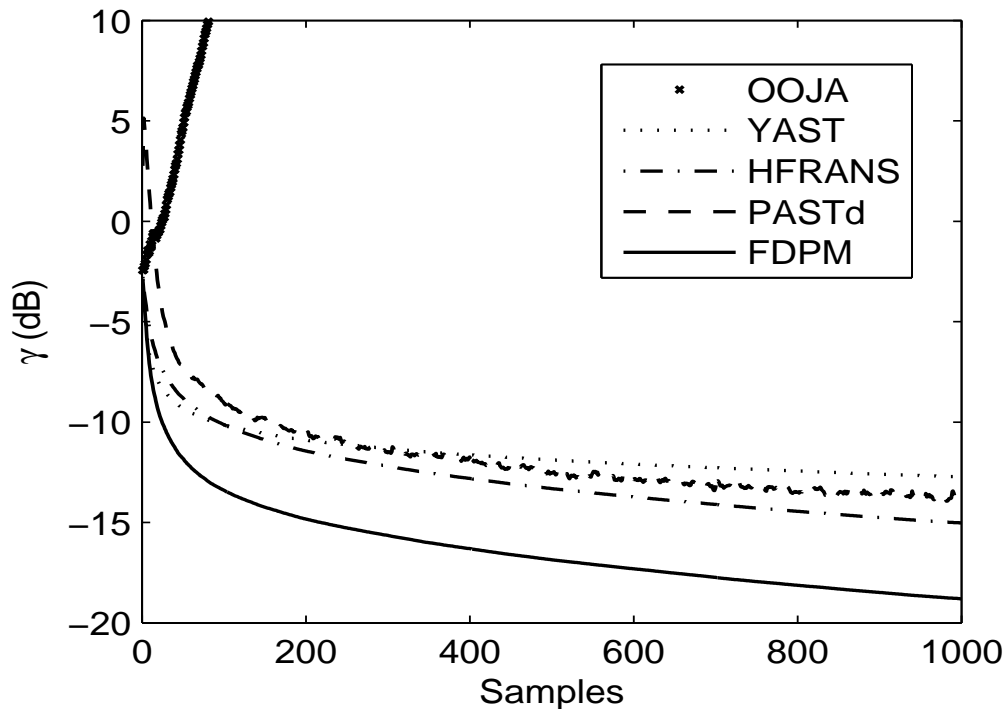


FIG. 4.1 – Performances des algorithmes de suivi de sous-espaces mineurs.

$K = 500$. On peut observer la perte en performances du filtre égaliseur MMSE à retard zéro comparé au filtre optimal (spécialement à fort SNR), chose qui est due (comme c'est expliquée plus haut) à la procédure d'estimation aveugle. Aussi, ça illustre l'efficacité de l'approche à deux-étapes, qui non seulement compense la perte en performance mais aussi elle permet de choisir des retards d'égalisation non-nuls, ce qui aide à améliorer les performances dans l'ensemble. La Fig. 4.3 illustre la vitesse de convergence de l'algorithme adaptative, pour un SNR= 15dB. La Fig. 4.4 est dédié à la robustesse aux erreurs de surestimation de l'ordre du canal. Les courbes comparent le MSE obtenu par les algorithmes donnés par TABLE 3.1, TABLE 3.2 et TABLE 4.1 pour un SNR=15 dB et un nombre d'échantillons $K = 500$. Il est claire que l'algorithme donné par TABLE 4.1 est insensible aux erreurs de surestimation de l'ordre du canal.

4.5 Conclusion

Dans ce chapitre, nous avons développé une nouvelle technique d'égalisation aveugle du second ordre, destinée à des système SIMO à réponse impulsionnelle finie et dont la mise en oeuvre n'a nul besoin d'estimer l'ordre du canal. Comme pour l'approche du chapitre 3, des implémentations en bloc et adaptative rapide sont proposées. Une procédure dite à *deux-étapes* a été développée et qui exploite la connaissance a priori de l'alphabet auquel appartiennent les symboles émis, afin d'obtenir des retards d'égalisation non-nuls et du coup améliorer les performances. Aussi nous avons présenté une étude comparative d'algorithmes rapides d'estimation et de suivi de sous-espaces mineurs. En fin, des

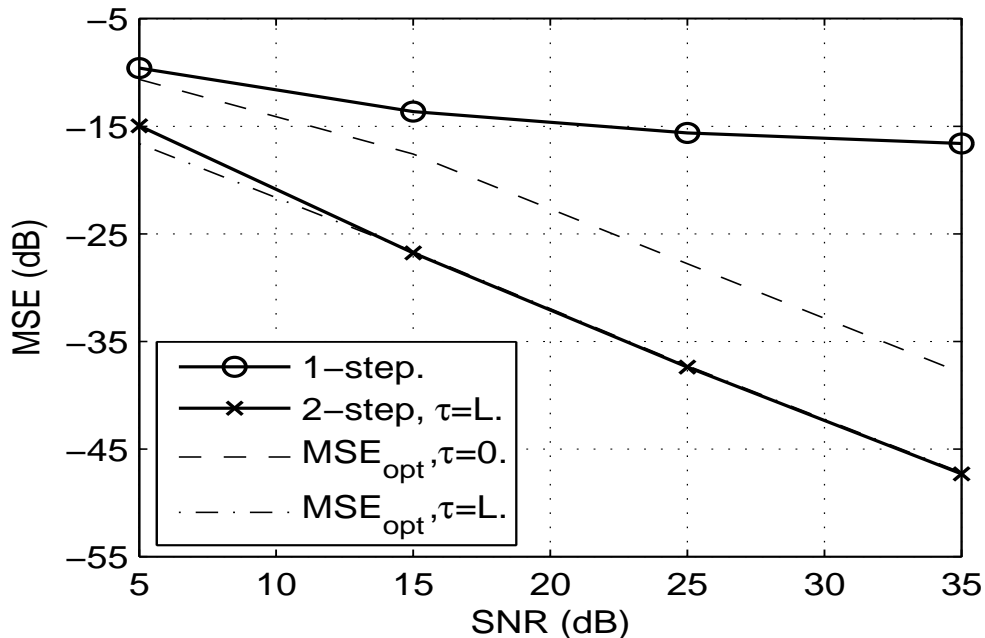
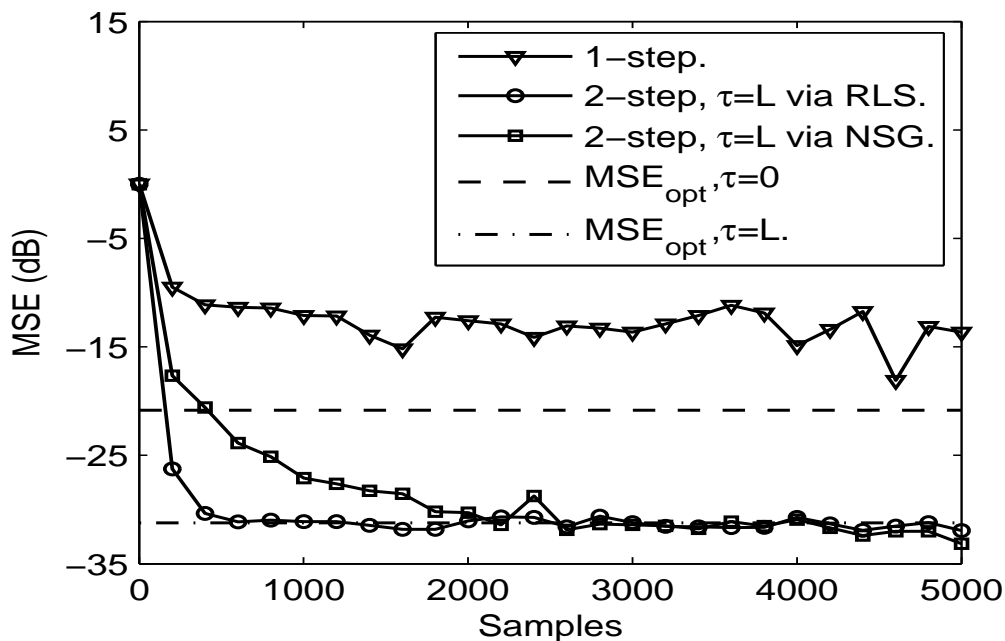

 FIG. 4.2 – Performance de l'égaliseur, $K = 500$.


FIG. 4.3 – Convergence de l'égaliseur adaptatif, SNR=15dB.

expérimentations par simulation sont données afin d'illustrer les performances de cette nouvelle approche. Noter que cette technique peut être étendue au cas MIMO.

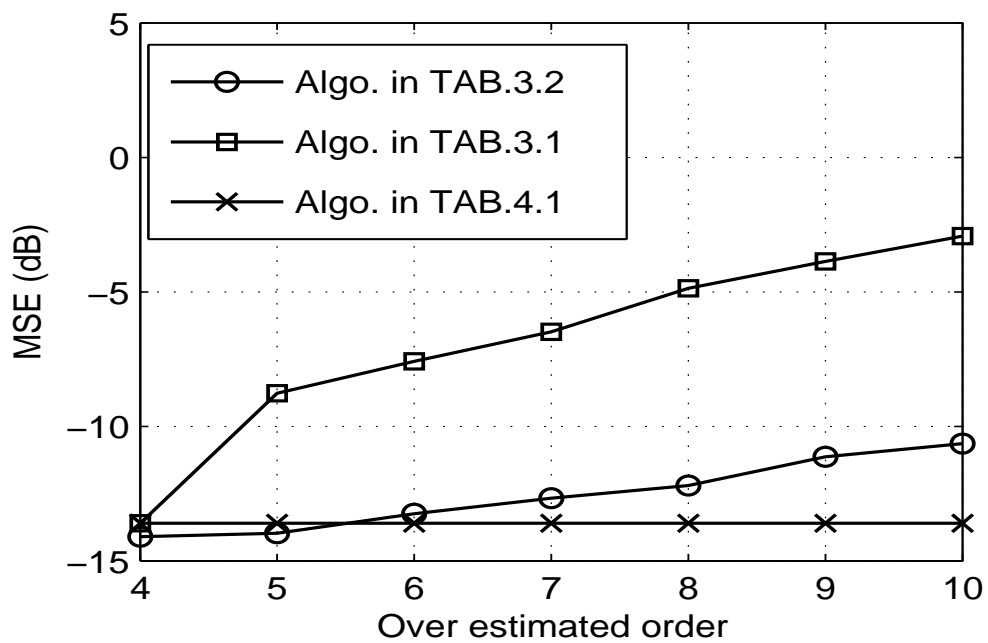


FIG. 4.4 – Robustesse aux erreurs de surestimation de l'ordre du canal, l'ordre exacte $L = 4$, $K = 500$ et $\text{SNR} = 15\text{dB}$.

Conclusion générale

Dans ce travail de thèse, nous avons considéré essentiellement le problème de sensibilité des égaliseurs aveugles aux erreurs de surestimation de l'ordre du canal, de même que le problème de leur coût de calcul, relativement élevé par rapport à celui des schémas classiques d'égalisation.

Le premier volet de ce travail est un état des lieux concernant des solutions déjà existantes pour ces problèmes. Nous avons présenté une description assez détaillée des principales approches ayant traités à ces questions. Tandis que dans le deuxième volet, nous avons présenté deux méthodes originales d'égalisation aveugle pour des systèmes multicanaux à réponse impulsionnelle finie, basées sur des statistiques du second ordre. Dans les deux approches, nous avons exploité le fait que le filtre égaliseur MMSE à retard zéro appartient à la fois au sous-espace signal et au noyau d'une matrice obtenue par une troncation appropriée de la matrice de covariance du vecteur des données reçues, pour ainsi développer des techniques simples d'estimation du filtre MMSE à retard zéro, en minimisant une certaine forme quadratique soumise à des contraintes bien appropriées. Dans les deux approches nous avons inclus une procédure dite à *deux étapes* qui, d'une part, permet de compenser la baisse en performances comparée au cas non-aveugle (cette baisse est due au processus d'estimation aveugle), d'autre part elle permet de choisir des retards d'égalisation non-nuls, d'où une amélioration significative des performances. De même, nous avons développé, pour les deux approches, des implémentations adaptatives rapides, ayant une complexité de calcul linéaire, avec une vitesse de convergence rapide comparée aux méthodes existantes.

La première approche que nous avons proposée (développée dans le chapitre 3) est plus générale. Elle est destinée à des systèmes MIMO. Sa mise en oeuvre dépend de l'ordre du canal. Néanmoins, une étude théorique et une analyse des résultats de simulation ont montré sa faible sensibilité aux erreurs de surestimation de l'ordre du canal. Cette sensibilité est d'autant plus faible, si la contrainte imposée est linéaire et/ou le nombre d'échantillons est assez large. Concernant toujours la première approche, nous avons effectué une analyse asymptotique des performances, dont les résultats théoriques sont en parfaite concordance avec ceux obtenus par simulation. Pour ce qui est de la seconde approche (celle présentée au chapitre 4), elle a l'avantage d'être totalement indépendante de l'ordre du canal dont l'estimation n'est donc plus nécessaire pour son éventuelle mise en oeuvre.

D'autres résultats, obtenus en marge de cette étude, concernent quelques algorithmes rapides de poursuite de sous-espaces sont présentés. En effet, dans le cas où les paramètres du canal sont variables dans

le temps, l'utilisation dans l'algorithme d'égalisation, de l'algorithme YAST [11] comme suiveur de sous-espace dominant (sous-espace signal) donne de bien meilleurs résultats que ceux obtenus en utilisant OPAST [5] (*Orthogonal Projection Approximation Subspace Tracking algorithm*). Par ailleurs, dans le cas d'une poursuite de sous-espace mineur, dont la dimension est faible devant celle du sous-espace total, les résultats de simulation donnent l'avantage à l'algorithme FDPM [29].

Cependant, des méthodes telles que celles décrites dans la partie État de l'art, ont l'avantage d'estimer directement (en une seule étape) le filtre égaliseur à retard non-nul, chose qui s'avérera importante dans certains cas limites, où l'égaliseur à retard zéro faillit à donner des performances satisfaisantes.

Comme perspectives à ce travail, nous proposons de combiner ces critères d'égalisation aveugle que nous avons établis, à des schémas d'égalisation classique (avec une séquence d'apprentissage très courte), pour ainsi aboutir à des schémas d'égalisation semi-aveugle, et qui auront pour avantages une simplicité dans leur mise en oeuvre pratique et des niveaux de performances très satisfaisants. De même, on peut étudier une éventuelle extension de ces techniques proposées au cas où les signaux émis sont corrélés temporellement, i.e., l'hypothèse qui stipule que les signaux d'entrée sont temporellement blancs peut être relaxer.

Bibliographie

- [1] K. Abed-Meraim, P. Loubaton and E. Moulines, "A subspace algorithm for certain blind identification problems," *IEEE Transactions on Information Theory*, vol. 43, no. 2, pp. 499-511, Mar. 1997.
- [2] K. Abed-Meraim, E. Moulines and P. Loubaton, "Prediction error method for second-order blind identification," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 694-705, Mar. 1997.
- [3] K. Abed-Meraim, W. Qiu and Y. Hua, "Blind system identification," *Proceedings of the IEEE*, vol. 85, no. 8, pp. 1310-1322, Aug. 1997.
- [4] K. Abed-Meraim and Y. Hua, "Blind equalization methods in colored noise field," *In Proc. of Information Decision and Control.*, pp. 477-481, Feb. 1999.
- [5] K. Abed-Meraim, A. Chkeif and Y. Hua, "Fast orthogonal PAST algorithm," *IEEE Signal Processing Letters*, vol. 7, no. 3, pp. 60-62, Mar. 2000.
- [6] K. Abed-Meraim, S. Attallah, A.Chkeif and Y. Hua, "Orthogonal OJA algorithm," *IEEE Signal Processing Letters*, vol. 7, no. 5, pp. 116-119, May 2000.
- [7] H. Akaike, "Information theory and an extension of the maximum likelihood principle," *In Proc. of 2nd Int. Symp. Infom. Theory*, suppl. *Problems of Control and Inform. Theory* pp. 267-281, 1973.
- [8] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. Automat. Contr.*, vol. AC-19, pp. 716-723, 1974.
- [9] S. Attallah and K. Abed-Meraim, "Low cost adaptive algorithm for noise subspace estimation," *Electronics Letters*, vol. 38, no. 12, pp. 609-611, June 2002.
- [10] S. Attallah, "The generalized rayleigh's quotient adaptive noise subspace algorithm : a householder transformation based implementation," *IEEE Transactions On Circuits And Systems II* :, vol. , no. 1, Jan. 2006.
- [11] R. Badeau, B. David and G. Richard, "Yet another subspace tracker," *In Proc. of ICASSP*, pp. 329-332, Philadelphia, USA, Mar. 2005.
- [12] C. A. Belfiore and J. H. Park, "Decision feedback equalization," *In Proc. of the IEEE*, vol. pp. , Aug. 1979.
- [13] S. Bellini, "Bussgang techniques for blind deconvolution and equalization," In S. Haykin, editor, *Blind deconvolution*, Englewood Cliffs, NJ : Prentice Hall, 1994.

-
- [14] A. Belouchrani and J. F. Cardoso, "Maximum likelihood source separation for discrete sources," *In Signal Processing VII : Theories and Applications (Proc. of the EUSIPCO-94)*, pp. 768-771, Edinburgh, Scotland, Sept. 13-16 1994, Elsevier.
- [15] A. Belouchrani and K. Abed-Meraim, "Constant modulus blind source separation technique : a new approach," *In Proc. of ISSPA*, pp. 232-235, Brisbane, Australia, Aug. 1996.
- [16] A. Belouchrani, K. Abed-Meraim, J. F. Cardoso and E. Moulines, "A blind source separation technique using second-order statistics," *IEEE Trans. Signal Processing*, vol. 45, no. 2, pp. 434-444, Feb. 1997.
- [17] A. Benvéniste, M. Goursat and G. Ruget, "Robust identification of a nonminimum phase system : blind adjustment of a linear equalizer in data communications," *IEEE Trans. Aut. Cont.*, vol. 25, pp. 385-399, June 1980.
- [18] A. Benvéniste and M. Goursat, "Blind equalizers," *IEEE Trans. on communicatios*, vol. 32, pp. 871-882, Aug. 1982.
- [19] J.W. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Trans. Circuits and systems*, vol. CAS-25, no. 9, pp. 772-781, Sep. 1978.
- [20] J.F. Cardoso and E. Moulines, "Asymptotic performance analysis of direction-finding algorithms based on fourth-order cumulants," *IEEE Trans. Signal Processing*, vol. 43, no. 1, pp. 214-224, Jan. 1995.
- [21] B. Champagne and Q.G. Liu, "Plane rotation-based EVD updating schemes for efficient subspace tracking," *IEEE Transactions on Signal Processing*, vol. 46, no. 7, pp. 1886-1900, July 1998.
- [22] A. Chkeif, K. Abed-Meraim, G.K. Kaleh and Y. Hua, "Spatio-temporal blind adaptive multiuser detection," *IEEE Trans. Communication*, vol. 48, no. 5, pp. 729-732, May 2000.
- [23] T. Chonavel, B. Champagne and C. Riou, "Fast adaptive eigenvalue decomposition : A maximum likelihood approach," *Signal Processing*, vol. 83, pp. 307-324, 2003.
- [24] C.E. Davila, "Efficient, high performance, subspace tracking for time-domain data," *IEEE Trans. Signal Processing*, vol. 48, no. 12, pp. 3307-3315, Dec. 2000.
- [25] E. de Carvalho and D. Slock, "Semi-blind methods for FIR multichannel estimation," In G.B. Giannakis *et al*, Eds., *Signal Processing Advances in wireless and mobile communications, Vol. 1 : Trends in Channel Estimation and Equalization*. Englewood Cliffs, NJ : Prentice Hall, 2001.
- [26] Z. Ding, "An outer-product decomposition algorithm for multichannel blind identification," *Proc. of the 8th IEEE Signal Processing Workshop on Statistical Signal and Array Processing*, pp. 132-135, Corfu, Greece, June 1996.
- [27] Z. Ding, "Matrix outer-product decomposition method for blind multiple channel identification," *IEEE Tr. Sig. Proc.*, vol. 45, no. 12, pp. 3054-3061, Dec. 1997.
- [28] Z. Ding and Y. Li, *Blind Equalization and Identification*. New York : Marcel Dekker, 2001.
- [29] X. G. Doukopoulos, "Power techniques for blind channel estimation in wireless communications," Ph.D. thesis, Univ. of Rennes, Oct. 2004.
-

-
- [30] G. D. Forney, Jr., "Maximum likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Transactions on Information Theory*, vol. 18, pp. 636-678, May 1972.
- [31] O. L. Frost, "An algorithm for Linearly Constrained Adaptive Array Processing," *Proc. of the IEEE*, vol. 60, no. 8, pp. 926-935, Aug. 1972.
- [32] W. A. Gardner, "A new method of channel identification," *IEEE Trans. Commun.*, vol. 39, pp. 1885-1911, 1985.
- [33] H. Gazzah and K. Abed-Meraim, "Blind equalization with controlled delay robust to order overestimation," *In Proc. of ISSPA*, vol. , pp. 268-271, Kuala Lumpur, Malaysia, Aug 2001.
- [34] H. Gazzah, P. A. Regalia, J. P. Delmas and K. Abed-Meraim, "A Blind multichannel identification algorithm robust to order overestimation," *IEEE Tr. Sig. Proc.*, vol. 50, no. 6, pp. 1449-1458, June 2002.
- [35] W. H. Gerstacker and D. P. Taylor, "Blind channel order estimation based on second-order statistics," *IEEE Sig. Proc. Letters*, vol. 10, no. 02, pp. 39-42, Feb. 2003.
- [36] D. Gesbert and P. Duhamel, "Unimodal blind adaptive channel equalization : An RLS implementation of the mutually referenced equalizers," *In Proc. of the conference*, pp. , city, country, month 1997.
- [37] D. Gesbert and P. Duhamel, "Robust blind channel identification and equalization based on multi-step predictors," *In Proc. of ICASSP*, pp. , Seattle, WA, April 1997.
- [38] D. Gesbert, P. Duhamel and S. Mayrargue, "On-line blind multichannel equalization based on mutually referenced filters," *IEEE Transactions on Signal Processing*, vol. 45, no. 9, pp. 2307-2317, Sep. 1997.
- [39] G. B. Giannakis and S. D. Halford, "Blind fractionally spaced equalization of noisy FIR channels : Direct and adaptive solutions," *IEEE Transactions on Signal Processing*, vol. 45, no. 9, pp. 2277-2292, Sep. 1997.
- [40] G.B. Giannakis *et al*, Eds., *Signal Processing Advances in wireless and mobile communications, Vol. 1 : Trends in Channel Estimation and Equalization*. Englewood Cliffs, NJ : Prentice Hall, 2001.
- [41] M. Goch, "Blind decision feedback equalization for terrestrial television receivers," *In Proc. of the IEEE*, vol. 39 pp. 669-682, 1991.
- [42] D.N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Trans. Commun.*, vol. COMM-28, pp. 1867-1875, Nov 1980.
- [43] A. Gorokov, M. Kristensson and B. Ottersten, "Robust blind second order deconvolution," *IEEE Signal Processing Letters*, vol. 6, no. 1, pp. 13-16, Jan. 1999.
- [44] A. Gorokov and P. Loubaton, "Blind identification of MIMO-FIR systems : A generalized linear prediction approach," *Signal Processing*, vol. 73, pp. 105-124, 1999.
-

-
- [45] P. J. Green and D. P. Taylor, "Dynamic channel order estimation algorithm," *In Proc. of the 2nd IEEE International Workshop on Electronic Design, Test and Applications*, pp. , city, country, month 2004.
- [46] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ : Prentice Hall, 1996.
- [47] M. Honig, U. Madhow and S. Verdu, "Blind Adaptive Multiuser Detection," *IEEE Transactions on Information Theory*, vol. 41, no. 4, pp. 944-960, July 1995.
- [48] D. H. Johnson and D. E. Dudgeon, *Array Signal Processing : Concepts and Techniques*. Englewood Cliffs, NJ : Prentice Hall, 1980.
- [49] I. Kacha, K. Abed-Meraim and A. Belouchrani, "A fast adaptive blind equalization algorithm robust to channel order over -estimation errors," *In Proc. of the 3rd IEEE SAM Signal Processing Workshop*, pp. 148-152, Barcelona, Spain, July 2004.
- [50] I. Kacha, K. Abed-Meraim and A. Belouchrani, "A new blind adaptive MMSE equalizer for MIMO systems," *In Proc. of the 16th Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications*, Berlin, Germany, September 2005.
- [51] I. Kacha, K. Abed-Meraim and A. Belouchrani, "Blind adaptive equalization without channel order estimation," *In Proc. ICASSP*, vol. 4, pp. 585-588, Toulouse, France, May 2006.
- [52] I. Kacha, K. Abed-Meraim and A. Belouchrani, "Fast adaptive blind MMSE equalizer for multichannel FIR systems," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 14827, 17 pages, 2006.
- [53] I. Karasalo, "Estimating the covariance matrix by signal subspace averaging," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 1, pp. 8-12, Feb. 1986.
- [54] X. Li and H. Fan, "QR factorization based blind channel identification and equalization with second-order statistics," *IEEE Transactions on Signal Processing*, vol. 48, no. 1, pp. 60-69, Jan. 2000.
- [55] ———, "Linear prediction methods for blind fractionally spaced equalization," *IEEE Transactions on Signal Processing*, vol. 48, no. 6, pp. 1667-1675, June. 2000.
- [56] ———, "Direct estimation of blind zero-forcing equalizers based on second-order statistics," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2211-2218, Aug. 2000.
- [57] ———, "Blind channel identification : Subspace tracking method without rank estimation," *IEEE Transactions on Signal Processing*, vol. 49, no. 10, pp. 2372-2382, Oct. 2001.
- [58] A. P. Liavas, P. A. Regalia and J. P. Delmas, "Robustness of least-squares and subspace methods for blind channel identification/equalization with respect to channel undermodeling," *IEEE Tr. Sig. Proc.*, vol. 47, no. 6, pp., June 1999.
- [59] A. P. Liavas, P. A. Regalia and, J. P. Delmas, "Blind channel approximation : Effective channel order determination," *IEEE Tr. Sig. Proc.*, vol. 47, no. 12, pp. 3336-3344, Dec. 1999.
- [60] R. W. Lucky, "Automatic equalization for digital communication," *Bell Syst. Tech. Journal*, vol. 44 pp. 547-588, April 1965.
-

-
- [61] R. W. Lucky, "Techniques for adaptive equalization of digital communication systems," *Bell Syst. Tech. Journal*, vol. 45 pp. 255-286, Feb. 1966.
- [62] O. Macchi and L. Guidou, "A new equalizer and double sampling equalizer," *Annales des télécommunications*, vol. 30, 1975.
- [63] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*. Upper Saddle Rivers, NJ : Prentice Hall, 2000.
- [64] E. Moulines, P. Duhamel, J.F. Cardoso and S. Mayrargue, "Subspace Methods for the Blind Identification of Multichannel FIR Filters," *IEEE Trans. Signal Processing*, vol. 43, no. 2, pp. 516-525, Feb. 1995.
- [65] D.T. Pham, "Blind separation of instantaneous mixture of sources via an independent component analysis," *IEEE Trans. Signal Processing*, vol. 44, no. 11, pp. 2768-2779, Nov. 1996.
- [66] J. G. Proakis, *Digital communications*, New York : MacGraw-Hill, 1995.
- [67] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 465-471, 1978.
- [68] Y. Sato, "A method of self-recovering equalization for multilevel amplitude-modulation," *IEEE Trans. Commun.*, vol. COMM-23, pp. 679-682, June 1975.
- [69] G. Schwartz, "Estimating the dimension of a model," *Ann. Stat.*, vol. 6, pp. 461-464, 1978.
- [70] J. Shen and Z. Ding, "Direct blind MMSE channel equalization based on second-order statistics," *IEEE Tr. Sig. Proc.*, vol. 48, no. 4, pp. 1015-1022, April 2000.
- [71] M. Sheng and H. Fan, "Blind MMSE equalization : A new direct method," *In Proc. ICASSP*, vol. 5, pp. 2457-2460, Istanbul, Turkey, June 2000.
- [72] D. Slock, "Blind fractionally-spaced equalization, perfect reconstruction filter banks and multi-channel linear prediction," *In Proc. ICASSP*, vol. 4, pp. 585-588, Adelaide, Australia, 1994.
- [73] P. Strobach, "Low-rank adaptive filters," *IEEE Transactions on Signal Processing*, vol. 44, no. 12, pp. 2932-2947, Dec. 1996.
- [74] L. Tong, G.Xu and T. Kailath, "A new approach to blind identification and equalization of multipaths channels," *In Proc. of 25th Asilomar Conference on Circuits, systems and computers*, pp. 856-860, Pacific Grove, CA, USA, 1991.
- [75] L. Tong and Q. Zhao, "Joint order detection and blind channel estimation by least-squares smoothing," *IEEE Transactions on Signal Processing*, vol. 47, no. 9, pp. 2345-2355, Sep. 1999.
- [76] L. Tong and Q. Zhao, "Adaptive blind channel estimation by least-squares smoothing," *IEEE Transactions on Signal Processing*, vol. 47, no. 11, pp. 3000-3012, Nov. 1999.
- [77] J. R. Treichler, I. Fijalkow and C. R. Johnson, "Fractionally spaced equalizers," *Signal processing magazine*, vol. 13, no. 3, pp. 65-81, June 1996.
- [78] M.K. Tsatsanis and G.B. Giannakis, "Modeling and equalization of rapidly fading channels," *Int. J. of Adaptive Control and Signal Processing*, vol. 10, nos. 2/3, pp. 159-176, Mar. 1996.

- [79] M. K. Tsatsanis and Z. Xu, "Performance Analysis of Minimum Variance CDMA Receivers," *In Proc. of Int. Conf. on Digital Signal Processing (DSP97)*, vol. 1, pp. 379-382, Santorini-Hellas, Greece, July. 1997.
- [80] M. K. Tsatsanis and Z. Xu, "Constrained optimization methods for direct blind equalization," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 3, pp. 424-433, Mar. 1999.
- [81] A.J. van der Veen and A. Paulraj, "An analytical constant modulus algorithm," *IEEE Trans. Signal Processing*, vol. 44, no. 5, pp. 1136-1154, May 1996.
- [82] M. Wax and T. Kailath, "Detection of signals by information theoretic criteria," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-33, no. 2, pp.387-392, April 1985.
- [83] S. B. Weinstein and R. D. Gitlin, "Fractionally spaced equalization : An improved digital transversal equalizer," *Bell system Technical Journal*, vol. 60, 1981.
- [84] J. Xavier and V. Barroso, "A channel order independent method for blind equalization of MIMO systems ," *In Proc. ICASSP*, vol. 5, pp. 2897-2900, Phoenix, Arizona, USA, Mar. 1999.
- [85] G. Xu, H. Liu, L. Tong and T. Kailath, "A least-squares approach to blind channel identification," *IEEE Transactions on Signal Processing*, vol. 43, no. 12, pp. 2982-2993, Dec. 1995.
- [86] Z. Xu and M. K. Tsatsanis, "Adaptive minimum variance methods for direct blind multichannel equalization," *In Proc. of the conference*, pp. 2105-2108, city, contry, month 1998.
- [87] J.F. Yang and M. Kaveh, "Adaptive eigensubspace algorithms for direction or frequency estimation and tracking," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 36, no. 2, pp. 241-251, Feb. 1988.
- [88] B. Yang, "Projection Approximation Subspace Tracking," *IEEE Trans. Signal Processing*, vol. 43, no. 1, pp. 95-107, Jan. 1995.

Annexes

Annexe A

Démonstrations des lemmes et théorèmes

A.1 Démonstration du *Théorème 2.1*

On décompose \mathbf{H}_N tel que

$$\begin{aligned}\mathbf{H}_N &= \begin{bmatrix} \underline{\mathbf{H}} \\ \overline{\mathbf{H}} \end{bmatrix}, \\ \underline{\mathbf{H}} &= \mathbf{H}_N(1:p, :) = [\mathbf{h}(0) \cdots \mathbf{h}(L) \mathbf{0}_{p \times (N-1)}], \\ \overline{\mathbf{H}} &= \mathbf{H}_N(p+1:n, :) = [\mathbf{0}_{(n-p) \times 1} \mathbf{H}_{N-1}].\end{aligned}\tag{A.1}$$

Dans ce cas

$$\begin{aligned}J(\mathbf{P}) &= \text{trace} \{ [\mathbf{I}_p - \mathbf{P}] \mathbf{H}_N \mathbf{H}_N^H [\mathbf{I}_p - \mathbf{P}]^H \} \\ &= \text{trace} \{ [\underline{\mathbf{H}} - \mathbf{P}\overline{\mathbf{H}}] [\underline{\mathbf{H}}^H - \overline{\mathbf{H}}^H \mathbf{P}^H] \} \\ &= \text{trace} \{ \underline{\mathbf{H}} \underline{\mathbf{H}}^H - \mathbf{P}\overline{\mathbf{H}} \underline{\mathbf{H}}^H - \underline{\mathbf{H}} \overline{\mathbf{H}}^H \mathbf{P}^H + \mathbf{P}\overline{\mathbf{H}} \overline{\mathbf{H}}^H \mathbf{P}^H \}.\end{aligned}$$

En annulant le gradient de $J(\mathbf{P})$, on obtient

$$\nabla J(\mathbf{P}) = \frac{\partial J(\mathbf{P})}{\partial \mathbf{P}^H} = -\overline{\mathbf{H}} \underline{\mathbf{H}}^H + \mathbf{P}\overline{\mathbf{H}} \overline{\mathbf{H}}^H = \mathbf{0} \Leftrightarrow [\underline{\mathbf{H}} - \mathbf{P}\overline{\mathbf{H}}] \overline{\mathbf{H}}^H = \mathbf{0}.\tag{A.2}$$

Tenant compte des relations (A.1), l'équation (A.2) équivaut à $\mathbf{H}' \mathbf{H}_{N-1}^H = \mathbf{0}$,

avec $\mathbf{H}' = [\mathbf{h}(1) \cdots \mathbf{h}(L) \mathbf{0}_{p \times (N-1)}] - \mathbf{P}\mathbf{H}_{N-1}$. Comme \mathbf{H}_{N-1} est à rang-colonne plein, il s'en suit que $\mathbf{H}' = \mathbf{0}$, d'où il convient d'écrire que si \mathbf{P}_{opt} est la solution optimale de (2.9), alors

$$\begin{aligned}[\mathbf{I}_p - \mathbf{P}_{opt}] \mathbf{H}_N &= \underline{\mathbf{H}} - \mathbf{P}_{opt} \overline{\mathbf{H}} = [\mathbf{h}(0) \mathbf{H}'] = [\mathbf{h}(0) \mathbf{0}_{p \times (d-1)}], \\ \tilde{\mathbf{e}}_{min}(t) &= [\mathbf{I}_p - \mathbf{P}_{opt}] \mathbf{H}_N \mathbf{s}_m(t) = \mathbf{h}(0) \mathbf{s}(t).\end{aligned}$$

□

A.2 Démonstration du Théorème 2.2

\mathbf{Q}_{opt} est obtenu en minimisant

$$J'(\mathbf{Q}) = \text{trace}(\tilde{\mathbf{C}}_N - \mathbf{Q}^H \tilde{\mathbf{C}}_{N,\tau} - \tilde{\mathbf{C}}_{N,\tau}^H \mathbf{Q} + \mathbf{Q}^H \tilde{\mathbf{C}}_N \mathbf{Q})$$

$$\Rightarrow \nabla J'(\mathbf{Q}) = \frac{\partial J'(\mathbf{Q})}{\partial \mathbf{Q}^H} = -\tilde{\mathbf{C}}_{N,\tau} + \tilde{\mathbf{C}}_N \mathbf{Q} = \mathbf{0} \Leftrightarrow \mathbf{Q}_{opt} = \tilde{\mathbf{C}}_N^\# \tilde{\mathbf{C}}_{N,\tau},$$

d'après (2.8) $\mathbf{v}_\tau^{zf} = \mathbf{Q}_{opt} \mathbf{v}_0^{zf}$. □

A.3 Démonstration du Théorème 2.3

i) Les relations (2.18) peuvent être réécrites sous la forme

$$[\mathbf{I}_{m-1} \ \mathbf{0}_{m-1,1}] \mathbf{V}^H \mathbf{x}_N(t) = [\mathbf{0}_{m-1,1} \ \mathbf{I}_{m-1}] \mathbf{V}^H \mathbf{x}_N(t+1),$$

en l'absence du bruit ($\mathbf{b}(t) = \mathbf{0}$) et d'après (1.4), on obtient

$$\begin{aligned} [\mathbf{I}_{m-1} \ \mathbf{0}_{m-1,1}] \mathbf{V}^H \mathbf{H}_N \mathbf{s}_m(t) &= [\mathbf{0}_{m-1,1} \ \mathbf{I}_{m-1}] \mathbf{V}^H \mathbf{H}_N \mathbf{s}_m(t+1) \\ &\Updownarrow \\ [\mathbf{I}_{m-1} \ \mathbf{0}_{m-1,1}] \mathbf{V}^H \mathbf{H}_N [\mathbf{0}_{m,1} \ \mathbf{I}_m] \mathbf{s}_{m+1}(t+1) &= [\mathbf{0}_{m-1,1} \ \mathbf{I}_{m-1}] \mathbf{V}^H \mathbf{H}_N [\mathbf{I}_m \ \mathbf{0}_{m,1}] \mathbf{s}_{m+1}(t+1), \end{aligned} \quad (\text{A.3})$$

comme la séquence $\{s(t)\}$ fournit une excitation persistante d'un ordre d'au moins $m+1$, l'équation (A.3) donne alors

$$\begin{aligned} [\mathbf{I}_{m-1} \ \mathbf{0}_{m-1,1}] \mathbf{V}^H \mathbf{H}_N [\mathbf{0}_{m,1} \ \mathbf{I}_m] &= [\mathbf{0}_{m-1,1} \ \mathbf{I}_{m-1}] \mathbf{V}^H \mathbf{H}_N [\mathbf{I}_m \ \mathbf{0}_{m,1}] \\ &\Updownarrow \\ \begin{bmatrix} 0 & \mathcal{I}_{1,1} & \cdots & \mathcal{I}_{1,m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \mathcal{I}_{m-1,1} & \cdots & \mathcal{I}_{m-1,m} \end{bmatrix} &= \begin{bmatrix} \mathcal{I}_{2,1} & \cdots & \mathcal{I}_{2,m} & 0 \\ \vdots & \ddots & \vdots & \vdots \\ \mathcal{I}_{m,1} & \cdots & \mathcal{I}_{m,m} & 0 \end{bmatrix} \end{aligned}$$

où $\mathcal{I}_{i,j}$ est le (i,j) ^{ème} terme de la matrice $\mathcal{I} = \mathbf{V}^H \mathbf{H}_N$. Il est alors facile de conclure que $\mathcal{I} = r \mathbf{I}_m$, où r est une constante scalaire.

ii) En l'absence du bruit ($\mathbf{b}(t) = \mathbf{0}$) et tenant compte de (1.4), on peut aisément montrer que i) et ii) sont équivalentes. □

A.4 Démonstration du Lemme 2.1

En utilisant le produit de Kronecker (voir [19] pour détails) le vecteur d'erreur $\varepsilon(t)$ est donné par

$$\begin{aligned}\varepsilon(t) &= [\mathbf{I}_{m-1} \mathbf{0}_{m-1,1}] \mathbf{V}^H \mathbf{x}_N(t) - [\mathbf{0}_{m-1,1} \mathbf{I}_{m-1}] \mathbf{V}^H \mathbf{x}_N(t+1), \\ \varepsilon^H(t) &= \mathbf{x}_N^H(t) \mathbf{V} [\mathbf{I}_{m-1} \mathbf{0}_{m-1,1}]^T - \mathbf{x}_N^H(t+1) \mathbf{V} [\mathbf{0}_{m-1,1} \mathbf{I}_{m-1}]^T, \\ \varepsilon^*(t) &= ([\mathbf{I}_{m-1} \mathbf{0}_{m-1,1}] \otimes \mathbf{x}_N^H(t) - [\mathbf{0}_{m-1,1} \mathbf{I}_{m-1}] \otimes \mathbf{x}_N^H(t+1)) \text{vec}(\mathbf{V}) \\ &= \mathbf{Z}(t) \mathbf{v},\end{aligned}$$

avec $\mathbf{Z}(t) \stackrel{\text{def}}{=} [\mathbf{I}_{m-1} \mathbf{0}_{m-1,1}] \otimes \mathbf{x}_N^H(t) - [\mathbf{0}_{m-1,1} \mathbf{I}_{m-1}] \otimes \mathbf{x}_N^H(t+1)$, ce qui donne $\|\varepsilon(t)\|^2 = \mathbf{v}^H \mathbf{Z}^H(t) \mathbf{Z}(t) \mathbf{v} \Rightarrow \mathcal{J}(\mathbf{v}) = \mathbf{v}^H \mathbf{E} (\mathbf{Z}^H(t) \mathbf{Z}(t)) \mathbf{v}$. Si on pose $\mathbf{C} \stackrel{\text{def}}{=} \mathbf{E} (\mathbf{Z}^H(t) \mathbf{Z}(t))$, alors et après quelques opérations algébriques, on obtient l'expression suivante de la matrice \mathbf{C}

$$\mathbf{C} = \text{diag}(1, 2, \dots, 2, 1) \otimes \mathbf{C}_{N,0} - \mathbf{J}_{m-1} \otimes \mathbf{C}_{N,1} - \mathbf{J}_{m-1}^{-1} \otimes \mathbf{C}_{N,1}^H,$$

\mathbf{J}_{m-1} étant la matrice donnée par (2.1). □

A.5 Démonstration du Théorème 2.4

Comme \mathbf{M}_1 et \mathbf{M}_2 sont des bases de $\text{null}_l(\tilde{\mathbf{C}}_{N,\tau})$ et $\text{null}_l(\tilde{\mathbf{C}}_{N,\tau+1})$ respectivement, on peut alors écrire

$$\begin{aligned}\mathbf{M}_1^H \mathbf{H}_N \mathbf{J}_m^\tau \mathbf{H}_N^H &= \mathbf{0}, \\ \mathbf{M}_2^H \mathbf{H}_N \mathbf{J}_m^{\tau+1} \mathbf{H}_N^H &= \mathbf{0},\end{aligned}$$

comme \mathbf{H}_N est une matrice à rang-colonne plein, cela implique

$$\begin{aligned}\mathbf{M}_1^H \mathbf{H}_N \mathbf{J}_m^\tau &= \mathbf{0}, \\ \mathbf{M}_2^H \mathbf{H}_N \mathbf{J}_m^{\tau+1} &= \mathbf{0}.\end{aligned}$$

Vu les structures particulières des matrices \mathbf{J}_m^τ et $\mathbf{J}_m^{\tau+1}$ (voir équations (2.1)), il existent deux matrices $\mathbf{A}_1 \in \mathbb{C}^{\tau \times (n-m+\tau)}$ et $\mathbf{A}_2 \in \mathbb{C}^{(\tau+1) \times (n-m+\tau+1)}$ telles que

$$\mathbf{H}_N^H \mathbf{M}_1 = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{0} \end{bmatrix}, \quad (\text{A.4})$$

$$\mathbf{H}_N^H \mathbf{M}_2 = \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{0} \end{bmatrix}. \quad (\text{A.5})$$

Supposons que (2.20) est vérifiée, il s'en suit que $\mathbf{M}_1^H \mathbf{H}_N \mathbf{H}_N^H \mathbf{M}_2 \mathbf{v}'_\tau = \mathbf{0}$, en utilisant (A.4) et (A.5) on obtient

$$\begin{bmatrix} \mathbf{A}_1^H & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{A}_2 \mathbf{v}'_\tau \\ \mathbf{0} \end{bmatrix} = \mathbf{0}, \quad (\text{A.6})$$

si on pose $\mathbf{A}_2 \mathbf{v}'_\tau = [\alpha_1, \dots, \alpha_\tau, \alpha_{\tau+1}]^T$, alors l'équation (A.6) implique

$$\mathbf{A}_1^H [\alpha_1, \dots, \alpha_\tau]^T = \mathbf{0}, \quad (\text{A.7})$$

comme \mathbf{A}_1^H est une matrice à rang-colonne plein¹ (\mathbf{A}_1^H a plus de lignes que de colonnes), alors de l'équation (A.7) on peut conclure que $[\alpha_1, \dots, \alpha_\tau]^T = \mathbf{0}$. Multipliant de part et d'autre (A.5) par le vecteur \mathbf{v}'_τ , on obtient alors

$$\mathbf{H}_N^H \mathbf{M}_2 \mathbf{v}'_\tau = \begin{bmatrix} \mathbf{A}_2 \mathbf{v}'_\tau \\ \mathbf{0}_{m-\tau-1,1} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{\tau,1} \\ \alpha_{\tau+1} \\ \mathbf{0}_{m-\tau-1,1} \end{bmatrix}.$$

Ce qui mène à conclure que le filtre $\mathbf{v}_\tau = \mathbf{M}_2 \mathbf{v}'_\tau$ est un égaliseur ZF à retard τ .

Inversement, supposons maintenant qu'un vecteur $\mathbf{v}_\tau \in \mathbb{C}^n$ soit un égaliseur ZF de retard τ du système (1.4), i.e., $\mathbf{v}_\tau^H \mathbf{H}_N = [\mathbf{0}_{1,\tau} \quad 1 \quad \mathbf{0}_{1,m-\tau-1}] \Rightarrow \mathbf{v}_\tau^H \tilde{\mathbf{C}}_{N,\tau+1} = \mathbf{v}_\tau^H \mathbf{H}_N \mathbf{J}_m^{\tau+1} \mathbf{H}_N^H = \mathbf{0}$, donc $\mathbf{v}_\tau \in \text{null}_l(\tilde{\mathbf{C}}_{N,\tau+1}) \Rightarrow \exists \mathbf{v}'_\tau \in \mathbb{C}^{n-m+\tau+1} : \mathbf{v}_\tau = \mathbf{M}_2 \mathbf{v}'_\tau$. Comme la sortie de l'égaliseur $\mathbf{v}_\tau^H \tilde{\mathbf{x}}_N(t)$ est non-corrélée avec le vecteur de données $\mathbf{M}_1^H \tilde{\mathbf{x}}_N(t)$, où $\tilde{\mathbf{x}}_N(t) = \mathbf{x}_N(t) - \mathbf{b}_N(t)$ (voir la définition dans l'équation (2.5)), alors $E((\mathbf{M}_1^H \tilde{\mathbf{x}}_N(t))^* \tilde{\mathbf{x}}_N(t)^T \mathbf{M}_2^* [\mathbf{v}'_\tau]^*) = \mathbf{0} \Rightarrow \mathbf{M}_1^T \tilde{\mathbf{C}}_{N,0}^* \mathbf{M}_2^* [\mathbf{v}'_\tau]^* = \mathbf{0} \Leftrightarrow \mathbf{M}_1^H \tilde{\mathbf{C}}_{N,0} \mathbf{M}_2 \mathbf{v}'_\tau = \mathbf{0}$. Donc (2.20) est vérifiée. \square

A.6 Démonstration du Théorème 3.1

D'après (1.10), la matrice de covariance s'exprime aussi par

$$\mathbf{C}_N = \mathbf{U}_s \text{diag}(\lambda_1, \dots, \lambda_d) \mathbf{U}_s^H + \sigma_b^2 \mathbf{U}_b \mathbf{U}_b^H.$$

Comme les colonnes de la matrice \mathbf{U}_s génèrent le sous-espace signal, alors il existe une matrice \mathbf{P}' de dimension $d \times d$, non-singulière telle que $\mathbf{U}_s = \mathbf{H}_N \mathbf{P}'$, tandis que les colonnes de \mathbf{U}_b génèrent son complément orthogonal, le *sous-espace bruit*, i.e., $\mathbf{U}_b^H \mathbf{U}_s = \mathbf{0}$. Comme \mathbf{W} est une base orthonormale du sous-espace signal, alors il existe deux matrices \mathbf{P} et \mathbf{P}'' de dimension $d \times d$, non-singulières telles que

$$\mathbf{W} = \mathbf{H}_N \mathbf{P} = \mathbf{U}_s \mathbf{P}'',$$

d'où

$$\mathbf{C}_N \mathbf{W} = (\mathbf{H}_N \mathbf{P}' \text{diag}(\lambda_1, \dots, \lambda_d) \mathbf{U}_s^H + \sigma_b^2 \mathbf{U}_b \mathbf{U}_b^H) \mathbf{U}_s \mathbf{P}'' = \mathbf{H}_N \mathbf{S},$$

avec $\mathbf{S} = \mathbf{P}' \text{diag}(\lambda_1, \dots, \lambda_d) \mathbf{P}''$ une matrice non-singulière. Par conséquent,

$$\mathbf{T} = \mathbf{C}_N(p+1 : n, :) \mathbf{W} = \mathbf{H}_N(p+1 : n, :) \mathbf{S}.$$

Comme \mathbf{H}_N (eq. (1.7)) est une matrice bloc-Toeplitz, il s'en suit que

$$\mathbf{H}_N(p+1 : n, :) = [\mathbf{0}_{(n-p) \times q} \quad \mathbf{H}_{N-1}].$$

¹En effet, on peut démontrer que la probabilité que \mathbf{A}_1^H soit une matrice de rang-colonne plein est égale à un.

La matrice \mathbf{H}_{N-1} étant à rang-colonne plein, cela implique alors

$$\dim(\text{null}_r(\mathbf{T})) = \dim(\text{null}_r([\mathbf{0}_{(n-p) \times q} \ \mathbf{H}_{N-1}])) = q.$$

Il s'en suit qu'une matrice $d \times q$ à rang colonne plein $\tilde{\mathbf{V}}$, solution de l'équation (3.9), peut être considérée comme une base du noyau droit de la matrice \mathbf{T} . Tenant compte de (3.6), les colonnes de la matrice $\tilde{\mathbf{V}}_0$, qui caractérise le filtre égaliseur MMSE donné par (3.5), appartiennent à $\text{null}_r(\mathbf{T})$ et sont linéairement indépendantes, il s'en suit que $\tilde{\mathbf{V}} = \tilde{\mathbf{V}}_0 \mathbf{R}$, où \mathbf{R} est une matrice $q \times q$ non-singulière. \square

A.7 Démonstration du Lemme 4.1

Noter d'abord que la matrice \mathbf{C}_{N-1} de dimension $(n-p) \times (n-p)$ est une sous-matrice à rang plein de $\bar{\mathbf{C}}$. Il s'en suit que $\dim(\text{null}_r(\bar{\mathbf{C}})) = p$.

Soit $\mathbf{W} \in \mathbb{C}^{n \times m}$ une base orthonormale du sous-espace signal. Comme $\text{range}(\mathbf{H}_N) = \text{range}(\mathbf{W}) = \text{range}(\mathbf{C}_N \mathbf{W})$, alors il existe une matrice $d \times d$ non-singulière \mathbf{R} telle que $\mathbf{C}_N \mathbf{W} = \mathbf{H}_N \mathbf{R}$. par conséquent $\bar{\mathbf{C}} \mathbf{W} = [\mathbf{0}_{(n-p) \times 1} \ \mathbf{H}_{N-1}] \mathbf{R}$. comme \mathbf{H}_{N-1} est à rang-colonne plein, alors $\dim(\text{null}_r(\bar{\mathbf{C}} \mathbf{W})) = \dim(\text{null}_r([\mathbf{0}_{(n-p) \times 1} \ \mathbf{H}_{N-1}])) = 1$, ce qui mène à conclure que seule une direction de $\text{null}_r(\bar{\mathbf{C}})$ appartient au sous-espace signal. \square

Annexe B

Publications

Liste des publications de l'auteur

1. I. Kacha, K. Abed-Meraim and A. Belouchrani, "A fast adaptive blind equalization algorithm robust to channel order over -estimation errors," *In Proc. of the 3rd IEEE SAM Signal Processing Workshop*, pp. 148-152, Barcelona, Spain, July 2004.
2. I. Kacha, K. Abed-Meraim and A. Belouchrani, "A new blind adaptive MMSE equalizer for MIMO systems," *In Proc. of the 16th Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications*, Berlin, Germany, September 2005.
3. I. Kacha, K. Abed-Meraim and A. Belouchrani, "Blind adaptive equalization without channel order estimation," *In Proc. ICASSP*, vol. 4, pp. 585-588, Toulouse, France, May 2006.
4. I. Kacha, K. Abed-Meraim and A. Belouchrani, "Fast adaptive blind MMSE equalizer for multichannel FIR systems," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 14827, 17 pages, 2006.

A FAST ADAPTIVE BLIND EQUALIZATION ALGORITHM ROBUST TO CHANNEL ORDER OVER-ESTIMATION ERRORS

I. Kacha*, K. Abed-Meraim** and A. Belouchrani*

* Electrical Engineering Department, École Nationale Polytechnique (ENP),
PO Box 182, El Harrach 16200, Algiers, Algeria.

** Signal and Image Processing Department, Télécom Paris (ENST),
46, rue Barrault, 75634 Paris Cedex 13, France.

ikacha@caramail.com, abelouch@ucsd.edu, abed@tsi.enst.fr

ABSTRACT

This paper presents a new algorithm for the blind equalization of FIR-SIMO systems. The proposed equalizer has several advantages and particular features that are studied in this work. More precisely, (i) we prove its robustness against channel order over-estimation errors, (ii) we propose an efficient fast adaptive implementation of our equalizer, (iii) we derive a statistical performance analysis to compare the equalization performance with that of the optimal MMSE equalizer and finally (iv) we develop a two-step procedure to further improve the performance gain and control the equalization delay. Simulation results are provided to illustrate the effectiveness of the proposed blind equalization algorithms.

1. INTRODUCTION

Adaptive channel equalization techniques have been widely used in communication systems, for their convenience in real time applications. Conventional equalizer algorithms require training sequences to update the parameters of the equalizer. This solution clearly reduces the effective channel throughput and alternative methods have been considered using blind equalization algorithms. In this work, we develop a blind adaptive equalization algorithm based on MMSE estimation, which presents a number of properties such as robustness to channel order over-estimation errors and low computational complexity. This algorithm includes a two step-estimation procedure, which allows us to compensate for the performance loss of the equalizer, compared to the non-blind one, and to obtain equalizers with controlled delays. Simulation examples are presented to demonstrate the performance of the proposed algorithm and assess the validity of our theoretical study.

2. DATA MODEL

Consider a SIMO system of p outputs given by:

$$\mathbf{x}(t) = \sum_{k=0}^L \mathbf{h}(k)s(t-k) + \mathbf{b}(t) \quad (1)$$

where $\mathbf{h}(z) = \sum_{k=0}^L \mathbf{h}(k)z^{-k}$ is an unknown causal FIR $p \times 1$ transfer function. We assume (A1) $\mathbf{h}(z) \neq 0, \forall z$. (A2) $s(t)$ a scalar (non-observable) iid zero-mean process of power σ_s^2 and (A3) $\mathbf{b}(t)$ is an additive spatially and temporally white noise of

power σ_b^2 and independent of the transmitted sequence $\{s(t)\}$. By stacking N successive samples of the received signal $\mathbf{x}(t)$, i.e.

$$\mathbf{x}_N(t) = [\mathbf{x}^T(t) \mathbf{x}^T(t-1) \dots \mathbf{x}^T(t-N+1)]^T$$

we obtain an $n \times 1$ data-vector ($n = Np$) that can be written as:

$$\mathbf{x}_N(t) = \mathbf{H}\mathbf{s}(t) + \mathbf{b}_N(t) \quad (2)$$

where $\mathbf{s}(t) = [s(t) \dots s(t-d+1)]^T$ ($d = N+L$) and \mathbf{H} is the channel convolution matrix of dimension $n \times d$, given by:

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}(0) & \dots & \mathbf{h}(L) & \dots & \mathbf{0} \\ & & & & \\ & & & & \\ \mathbf{0} & & \mathbf{h}(0) & \dots & \mathbf{h}(L) \end{bmatrix} \quad (3)$$

3. MMSE EQUALIZER

Consider a τ -delay MMSE equalizer ($\tau \in \{0, 1, \dots, d-1\}$). Under the above data model, one can easily show that the equalizer vector \mathbf{v}_τ corresponding to the desired solution is given by:

$$\mathbf{v}_\tau = \arg \min_{\mathbf{v}} (E(\|\mathbf{s}(t-\tau) - \mathbf{v}^H \mathbf{x}_N(t)\|^2)) \quad (4)$$

$$= \mathbf{C}^{-1} \mathbf{g}_\tau \quad (5)$$

where:

$$\mathbf{C} \stackrel{\text{def}}{=} E[\mathbf{x}_N(t)\mathbf{x}_N^H(t)] \quad (6)$$

is the data covariance matrix and \mathbf{g}_τ is an $n \times 1$ vector given by:

$$\mathbf{g}_\tau \stackrel{\text{def}}{=} E(\mathbf{x}_N(t)\mathbf{s}^*(t-\tau)) \quad (7)$$

$$= \sigma_s^2 \mathbf{H}(:, \tau+1) \quad (8)$$

where $\mathbf{H}(:, \tau+1)$ denotes the $(\tau+1)$ -th column vector of \mathbf{H} . Clearly, the MMSE filter \mathbf{v}_τ belongs to the signal subspace (i.e. $\text{Range}(\mathbf{H})$) and thus one can write

$$\mathbf{v}_\tau = \mathbf{W}\tilde{\mathbf{v}}_\tau \quad (9)$$

where \mathbf{W} is a $n \times d$ matrix which column vectors form a basis of the signal subspace and $\tilde{\mathbf{v}}_\tau$ is a d -dimensional vector.

Our objective here is to derive a blind estimate of the zero-delay MMSE equalizer \mathbf{v}_0 . From equations (5), (7) and (9), we can write

$$\mathbf{C}\mathbf{W}\tilde{\mathbf{v}}_0 = \sigma_s^2 \begin{bmatrix} \mathbf{h}(0) \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \quad (10)$$

Equation (10) implies that

$$\overline{\mathbf{C}}\mathbf{W}\tilde{\mathbf{v}}_0 = \mathbf{0} \quad (11)$$

where $\overline{\mathbf{C}}$ is a submatrix of \mathbf{C} given by its last $n - p$ rows. Reversely, we established that this equation characterizes uniquely the zero-delay MMSE equalizer. We have the following result:

Theorem 1 Under the above data assumptions and for $N \geq L + 1$, the solution of

$$\overline{\mathbf{C}}\mathbf{W}\tilde{\mathbf{v}} = \mathbf{0} \quad (12)$$

is unique (up to a scalar constant) and corresponds to the desired MMSE equalizer, i.e. $\tilde{\mathbf{v}} = \alpha\tilde{\mathbf{v}}_0$ for a given scalar constant α .

Proof: Since $\text{Range}(\mathbf{W}) = \text{Range}(\mathbf{C}\mathbf{W}) = \text{Range}(\mathbf{H})$, there exists a non-singular $d \times d$ matrix \mathbf{T} such that:

$$\mathbf{C}\mathbf{W} = \mathbf{H}\mathbf{T}.$$

Therefore

$$\overline{\mathbf{C}}\mathbf{W}\mathbf{v} = \mathbf{0} \Rightarrow \overline{\mathbf{H}}\mathbf{T}\mathbf{v} = \mathbf{0}$$

where $\overline{\mathbf{H}}$ is the $(n - p) \times n$ matrix given by:

$$\overline{\mathbf{H}} = \begin{bmatrix} \mathbf{0} & \mathbf{h}(0) & \dots & \mathbf{h}(L) & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{h}(0) & \dots & \mathbf{h}(L) \end{bmatrix} \quad (13)$$

Under assumption (A1), it is shown in [4] that \mathbf{H} is full column rank as soon as $N \geq L$. Therefore, the kernel of $\overline{\mathbf{H}}$ (and consequently, the kernel of $\overline{\mathbf{C}}\mathbf{W}$) is of rank 1. Since $\tilde{\mathbf{v}}_0$, the MMSE filter given by (10), belongs to $\text{Ker}(\overline{\mathbf{C}}\mathbf{W})$ and $\text{rank}(\text{Ker}(\overline{\mathbf{C}}\mathbf{W})) = 1$, it follows that any vector $\tilde{\mathbf{v}}$, solution of equation (12) is equal (up to a constant factor) to $\tilde{\mathbf{v}}_0$. \square

In practice, equation (12) is solved in the least squares sense according to:

$$\tilde{\mathbf{v}}_0 = \arg \min_{\|\mathbf{v}\|=1} (\mathbf{v}^H \mathbf{Q} \mathbf{v}) \quad (14)$$

where \mathbf{Q} is a $(d \times d)$ matrix define by:

$$\mathbf{Q} \stackrel{\text{def}}{=} \mathbf{W}^H \overline{\mathbf{C}}^H \overline{\mathbf{C}} \mathbf{W} \quad (15)$$

Thus, we obtain a batch-processing implementation of the blind MMSE equalization algorithm that is summarized in table 1 (we use here informal MATLAB notation and EVD represents the output of the eigen-decomposition with sorted (in decreasing order) eigenvalues).

\mathbf{C}	$= \frac{1}{T} \sum_{t=N}^T \mathbf{x}_N(t) \mathbf{x}_N^H(t)$
$(\mathbf{U}_1, \Lambda_1)$	$= \text{EVD}(\mathbf{C})$
\mathbf{W}	$= \mathbf{U}_1(:, 1 : d)$
$\overline{\mathbf{C}}$	$= \mathbf{C}(p + 1 : n, :)$
\mathbf{Q}	$= \mathbf{W}^H \overline{\mathbf{C}}^H \overline{\mathbf{C}} \mathbf{W}$
$(\mathbf{U}_2, \Lambda_2)$	$= \text{EVD}(\mathbf{Q})$
$\tilde{\mathbf{v}}_0$	$= \mathbf{U}_2(:, d)$
\mathbf{v}_0	$= \mathbf{W}\tilde{\mathbf{v}}_0$, (up to a scalar constant)

Table 1. Blind MMSE equalization algorithm

4. FAST ADAPTIVE IMPLEMENTATION

In tracking applications we are interested in estimating the equalizer vector recursively. We propose here a fast adaptive implementation of the proposed blind MMSE equalizer. The matrix \mathbf{C} is replacing by its recursive version

$$\mathbf{C}(t) = \beta \mathbf{C}(t-1) + \mathbf{x}_N(t) \mathbf{x}_N^H(t) \quad (16)$$

where $0 < \beta < 1$ is a forgetting factor. The weight matrix \mathbf{W} corresponding to the d dominant eigenvectors of \mathbf{C} can be estimated using a fast subspace estimation and tracking algorithm based on the power method. In this work, we use the OPAST (Orthogonal Projection Approximation Subspace Tracking) algorithm in [2] that is summarized in table 2. Note that only $O(nd)$ operations are required at each time step (instead of $O(n^3)$ for a full EVD), (see [2] and [5] for more details). The second step of our algorithm

initialization	
$\mathbf{W}(0) =$	$\begin{bmatrix} \mathbf{I}_d \\ \mathbf{0}_{(n-d) \times d} \end{bmatrix}, \quad \mathbf{Z}(0) = \mathbf{I}_d$
for each time step do	
$\mathbf{y}(t) =$	$\mathbf{W}^H(t-1) \mathbf{x}_N(t)$
$\mathbf{q}(t) =$	$\frac{1}{\beta} \mathbf{Z}(t-1) \mathbf{y}(t)$
$\gamma(t) =$	$\frac{1}{1 + \mathbf{y}^H(t) \mathbf{q}(t)}$
$\mathbf{p}(t) =$	$\gamma(t) (\mathbf{x}_N(t) - \mathbf{W}^H(t-1) \mathbf{y}(t))$
$\tau(t) =$	$\frac{1}{\ \mathbf{q}(t)\ ^2} \left(\frac{1}{\sqrt{1 + \ \mathbf{p}(t)\ ^2 \ \mathbf{q}(t)\ ^2}} - 1 \right)$
$\mathbf{p}'(t) =$	$\tau(t) \mathbf{W}(t-1) \mathbf{q}(t) + (1 + \tau(t) \ \mathbf{q}(t)\ ^2) \mathbf{p}(t)$
$\mathbf{Z}(t) =$	$\frac{1}{\beta} \mathbf{Z}(t-1) - \gamma(t) \mathbf{q}(t) \mathbf{q}^H(t)$
$\mathbf{W}(t) =$	$\mathbf{W}(t-1) + \mathbf{p}'(t) \mathbf{q}^H(t)$

Table 2. OPAST algorithm

consists of estimating recursively the d -dimensional vector $\tilde{\mathbf{v}}$ as the least eigenvector of matrix \mathbf{Q} or equivalently as the dominant eigenvector of its inverse¹.

First, equation (16) yields

$$\overline{\mathbf{C}}(t) = \beta \overline{\mathbf{C}}(t-1) + \overline{\mathbf{x}}_N(t) \mathbf{x}_N^H(t) \quad (17)$$

where $\overline{\mathbf{x}}_N(t)$ is a sub-vector of $\mathbf{X}(t)$ given by its last $n - p$ elements. Consider the matrix:

$$\mathbf{T}(t) \stackrel{\text{def}}{=} \overline{\mathbf{C}}(t) \mathbf{W}(t) \quad (18)$$

applying the projection approximation:

$$\overline{\mathbf{C}}(t) \mathbf{W}(t) \approx \overline{\mathbf{C}}(t) \mathbf{W}(t-1)$$

which is valid if matrix $\mathbf{W}(t)$ is slowly varying with t , and substituting equation (17) into (18) yields

$$\mathbf{T}(t) = \beta \mathbf{T}(t-1) + \overline{\mathbf{x}}_N(t) \mathbf{y}^H(t) \quad (19)$$

then equation (15) can be replaced by the following recursion

$$\mathbf{Q}(t) = \beta^2 \mathbf{Q}(t-1) - \mathbf{D}(t) \mathbf{J}^{-1}(t) \mathbf{D}^H(t) \quad (20)$$

¹ \mathbf{Q} is a singular matrix when dealing with the exact statistics. However, when considering the sample averaged estimate of \mathbf{C} and due to the estimation errors, the estimate of \mathbf{Q} is almost surely a non-singular matrix.

where $\mathbf{D}(t)$ is the $d \times 2$ matrix

$$\mathbf{D}(t) = \begin{bmatrix} \beta \mathbf{T}^H(t-1) \bar{\mathbf{x}}_N(t) & \mathbf{y}(t) \end{bmatrix} \quad (21)$$

and $\mathbf{J}(t)$ is the 2×2 non-singular matrix

$$\mathbf{J}(t) = \begin{bmatrix} \|\bar{\mathbf{x}}_N(t)\|^2 & -1 \\ -1 & 0 \end{bmatrix} \quad (22)$$

Consider the $d \times d$ hermitian matrix

$$\mathbf{R}(t) = \mathbf{Q}^{-1}(t) \quad (23)$$

using the matrix (Schur) inversion lemma [3], we obtain

$$\mathbf{R}(t) = \frac{1}{\beta^2} \mathbf{R}(t-1) + \mathbf{S}(t) \boldsymbol{\Sigma}(t) \mathbf{S}^H(t) \quad (24)$$

where $\mathbf{S}(t)$ is the $d \times 2$ matrix

$$\mathbf{S}(t) = \frac{1}{\beta^2} \mathbf{R}(t-1) \mathbf{D}(t) \quad (25)$$

and $\boldsymbol{\Sigma}(t)$ is the 2×2 matrix

$$\boldsymbol{\Sigma}(t) = (\mathbf{J}(t) - \mathbf{S}^H(t) \mathbf{D}(t))^{-1} \quad (26)$$

The extraction of the dominant eigenvector of $\mathbf{R}(t)$ is obtained by power iteration as:

$$\tilde{\mathbf{v}}_0(t) = \frac{\mathbf{R}(t) \tilde{\mathbf{v}}_0(t-1)}{\|\mathbf{R}(t) \tilde{\mathbf{v}}_0(t-1)\|} \quad (27)$$

Thus the MMSE equalizer vector is expressed (up to a scalar constant) by

$$\mathbf{v}_0(t) = \mathbf{W}(t) \tilde{\mathbf{v}}_0(t) \quad (28)$$

Note that the whole processing requires only $O(nd)$ operations. The complete pseudo-code for the adaptive blind MMSE equalization algorithm is given in table 3.

5. ROBUSTNESS

In this section, we consider the robustness of the proposed blind MMSE equalizer against channel order over-estimation errors. The channel order is used to determine the column dimension equal to $d = L + N$ of matrix \mathbf{W} (which corresponds, in practice, to the size of the dominant subspace of \mathbf{C}). Let $L' > L$ be the over-estimated channel order and hence $d' = L' + N$ is the column dimension of \mathbf{W} , i.e., we consider the subspace spanned by the d' dominant eigen-vector of \mathbf{C} . We argue here that, as long as the number of sensors p plus the over-estimation error order $L' - L$ is smaller than the noise subspace dimension, i.e. $p + L' - L < n - d$, the least squares solution of (11) provides a consistent estimate of the MMSE equalizer.

This observation comes from the following: Let $\mathbf{x}_1, \dots, \mathbf{x}_p$ be a basis of $\text{Ker}(\bar{\mathbf{C}})$ (the kernel of $\bar{\mathbf{C}}$ is of dimension p , i.e. $\bar{\mathbf{C}}\mathbf{x} = 0$ admits p linearly independent solutions) such that \mathbf{x}_1 belongs to the signal subspace, i.e. $\text{Range}(\mathbf{H})$. According to Theorem 1, \mathbf{x}_1 is the only basis vector that belongs to $\text{Range}(\mathbf{H})$. Now, the solution of (11) would be unique (up to a scalar constant) if:

$$\text{Range}(\mathbf{W}) \cap \text{Range}([\mathbf{x}_1, \dots, \mathbf{x}_p]) = \text{Range}(\mathbf{x}_1)$$

Initialization

- OPAST initialization (cf. Table 2)
- $\bar{\mathbf{C}}(0) = [\mathbf{0}_{(n-p) \times p} \quad \mathbf{I}_{(n-p) \times (n-p)}]$
- $\tilde{\mathbf{v}}_0(0)$ = a random unitary vector
- $\mathbf{T}(0) = \bar{\mathbf{C}}(0) \mathbf{W}(0)$
- $\mathbf{R}(0) = (\mathbf{T}^H(0) \mathbf{T}(0))^{-1}$

for each time step do

- update $\mathbf{W}(t)$ using OPAST (cf. Table 2)

$$\begin{aligned} \mathbf{J}(t) &= \begin{bmatrix} \|\bar{\mathbf{x}}_N(t)\|^2 & -1 \\ -1 & 0 \end{bmatrix} \\ \mathbf{f}(t) &= \beta \mathbf{T}^H(t-1) \bar{\mathbf{x}}_N(t) \\ \mathbf{D}(t) &= \begin{bmatrix} \mathbf{f}(t) & \mathbf{y}(t) \end{bmatrix} \\ \mathbf{S}(t) &= \frac{1}{\beta^2} \mathbf{R}(t-1) \mathbf{D}(t) \\ \boldsymbol{\Sigma}(t) &= (\mathbf{J}(t) - \mathbf{S}^H(t) \mathbf{D}(t))^{-1} \\ \mathbf{R}(t) &= \frac{1}{\beta^2} \mathbf{R}(t-1) + \mathbf{S}(t) \boldsymbol{\Sigma}(t) \mathbf{S}^H(t) \\ \tilde{\mathbf{v}}_0(t) &= \frac{\mathbf{R}(t) \tilde{\mathbf{v}}_0(t-1)}{\|\mathbf{R}(t) \tilde{\mathbf{v}}_0(t-1)\|} \\ \mathbf{v}_0(t) &= \mathbf{W}(t) \tilde{\mathbf{v}}_0(t) \\ \mathbf{T}(t) &= \beta \mathbf{T}(t-1) + \bar{\mathbf{x}}_N(t) \mathbf{y}^H(t) \end{aligned}$$

Table 3. Adaptive blind equalization algorithm

or equivalently

$$\text{Range}(\mathbf{W}) \cap \text{Range}([\mathbf{x}_2, \dots, \mathbf{x}_p]) = \{\mathbf{0}\}.$$

The above condition would be verified if the intersection of the subspace spanned by the projections of $\mathbf{x}_2, \dots, \mathbf{x}_p$ onto the noise subspace and the subspace spanned by the $L' - L$ noise vectors of \mathbf{W} introduced by the overestimation error is empty (except for the zero vector). As the latter are randomly introduced by the EVD of \mathbf{C} and since $p + L' - L < n - d$, then one can expect this subspace intersection to be empty almost surely.

This argument, even though not a rigorous proof of robustness, has been confirmed by our simulation results (see simulation example given below where one can see that the performance loss of the equalization due to the channel order over-estimation error remains relatively limited).

6. PERFORMANCE ANALYSIS

We study here the asymptotic expression of the blind equalization MSE defined by²

$$\min_{\alpha} E(|s(n) - \alpha \mathbf{v}^H \mathbf{x}_N(n)|^2). \quad (29)$$

where \mathbf{v} is the unit-norm blind MMSE equalizer presented previously. In particular, we show that the blind MMSE equalizer has a performance loss compared to the optimal (non-blind) one. This loss of estimation accuracy in terms of asymptotic MSE is asymptotically given by

$$\frac{\|\mathbf{v}_0\|^2}{T} \text{Trace}(\mathbf{C} \boldsymbol{\Sigma}_v)$$

²The scalar constant α comes from the fact that, in the blind context, the MMSE is estimated only up to an unknown scalar coefficient.

where \mathbf{v}_0 is the zero-delay MMSE equalizer given by (10) and Σ_v is the asymptotic error covariance matrix of the estimate of \mathbf{v} . The explicit expression of Σ_v is given by the following lemma:

Lemma 1 Under assumptions (A1), (A2), (A3) and $E(|s(k)|^4) < \infty$, the blind MMSE equalizer given by Table 1 is asymptotically gaussian distributed, i.e.

$$\sqrt{T}(\hat{\mathbf{v}} - \mathbf{v}) \xrightarrow{\mathcal{L}} \mathcal{N}(\mathbf{0}, \Sigma_v)$$

where \mathbf{v} represents the exact value of \mathbf{v} (i.e., when using exact statistics) and $\hat{\mathbf{v}}$ its estimated version from a T -sample observation. The expression of Σ_v is given by

$$\Sigma_v = \mathbf{M}_1 \Sigma_C \mathbf{M}_1^H$$

where Σ_C is the asymptotic covariance matrix of the sample estimate of $\text{vec}(\mathbf{C})$ given in [7] and

$$\begin{aligned} \mathbf{M}_1 &= (\mathbf{v}^T \otimes \mathbf{Q}^\#) \left[(\mathbf{J}^T \bar{\mathbf{C}} \mathbf{W})^T \otimes \mathbf{W}^H + \mathbf{W}^T \otimes (\mathbf{W}^H \bar{\mathbf{C}}^H \mathbf{J}) \right. \\ &\quad \left. + (\mathbf{M}_2^H \bar{\mathbf{C}}^H \bar{\mathbf{C}} \mathbf{W})^T \otimes \mathbf{I}_n + \mathbf{I}_n \otimes (\mathbf{W}^H \bar{\mathbf{C}}^H \bar{\mathbf{C}} \mathbf{M}_2) \right] \\ \mathbf{M}_2 &= \{\mathbf{W}(:, 1)^T \otimes \Gamma_1^\#, \dots, \mathbf{W}(:, d) \otimes \Gamma_d^\#\} (\mathbf{1} \otimes \mathbf{I}_n) \\ \Gamma_i^\# &= (\mathbf{C} - \lambda_i \mathbf{I}_n)^\# \\ \mathbf{1} &= [1, 1, \dots, 1]^T \\ \mathbf{J} &= [\mathbf{0}_{n \times p} \quad \mathbf{I}_{n-p}] \end{aligned}$$

\otimes being the Kronecker product, $^\#$ denotes the matrix pseudo-inversion operator and $\lambda_1 > \lambda_2 \geq \dots \geq \lambda_n$ are the eigenvalues of \mathbf{C} .

Due to space limitation, we omit here the proof and work-out of above expression.

7. TWO-STEP EQUALIZATION ALGORITHM

To compensate for the above performance loss and also to have a controlled non-zero equalization delay, we propose here a two-step approach to estimate the blind MMSE equalizer. In the first step, we estimate \mathbf{v}_0 according to the previous algorithm, while, in the second step, we refine this estimation by exploiting the a priori knowledge of the finite alphabet to which belongs the symbols $s(t)$. This is done by performing a hard decision on the symbols³ that are then used to re-estimate \mathbf{v}_τ according to equations (5) and (7).

Let $\mathbf{W} \in \mathcal{C}^{n \times d}$ be an orthonormal basis of the signal subspace. Since \mathbf{g}_τ belongs to the signal subspace, one can write (see [1]):

$$\mathbf{v}_\tau = \mathbf{W}(\mathbf{W}^H \mathbf{C} \mathbf{W})^{-1} \mathbf{W}^H \mathbf{g}_\tau \quad (30)$$

This expression of \mathbf{v}_τ is used for the fast adaptive implementation of this two-step algorithm since $\mathbf{Z} = (\mathbf{W}^H \mathbf{C} \mathbf{W})^{-1}$ is already computed by the OFAST. The recursive expression of vector \mathbf{g}_τ is given by

$$\mathbf{g}_\tau(t) = \beta \mathbf{g}_\tau(t-1) + \mathbf{x}_N(t) \hat{s}^*(t-\tau) \quad (31)$$

and thus equation (30) can be replaced by the following recursion

$$\mathbf{v}_\tau(t) = \mathbf{W}(t) \mathbf{Z}(t) \mathbf{W}^H(t) \mathbf{g}_\tau(t) \quad (32)$$

³We assume here the use of a differential modulation to get rid of the phase indeterminacy inherent to the blind equalization problem.

Note that, by choosing a non-zero equalizer delay τ , we improve the equalization performance as shown below. The two step adaptive blind MMSE equalization algorithm can be written as summarized in table 3 and table 4. The overall computational cost of this algorithm is $O(8nd)$ flops per iteration.

Estimate $\hat{s}(t)$ (using $\mathbf{v}_0(t)$) $\mathbf{g}_\tau(t) = \beta \mathbf{g}_\tau(t-1) + \mathbf{x}_N(t) \hat{s}^*(t-\tau)$ $\mathbf{v}_\tau(t) = \mathbf{W}(t) \mathbf{Z}(t) \mathbf{W}^H(t) \mathbf{g}_\tau(t)$
--

Table 4. Two-step equalization procedure

8. SIMULATION RESULTS

We provide in this section some simulation examples to illustrate the performance of the proposed blind equalizer. We consider a $p = 3$ channels system of degree $L = 4$ chosen randomly at each run according to a complex Gaussian distribution for each channel coefficient. The channels are excited by an iid QAM4 sequence. The MSE in (29) is evaluated over 100 Monte-Carlo runs. The figures below illustrate the convergence rate of the adaptive algorithm (Fig. 1 with SNR= 20dB), the performance gain of the 2-step approach (Fig. 2 with $T = 500$) and the robustness to channel order over-estimation errors (Fig. 3 with $T = 500$ and SNR= 5dB), respectively.

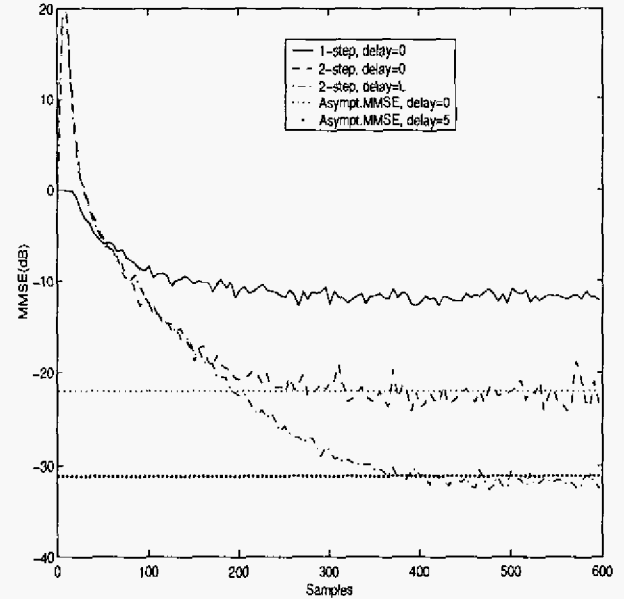


Fig. 1. Convergence of the adaptive MMSE equalization algorithm.

9. CONCLUSION

This paper introduces a new blind estimation method of the MMSE equalizer filter for SIMO-FIR systems. Batch and fast adaptive implementation algorithms are developed. A two-step version using the a priori knowledge of the source signal finite alphabet

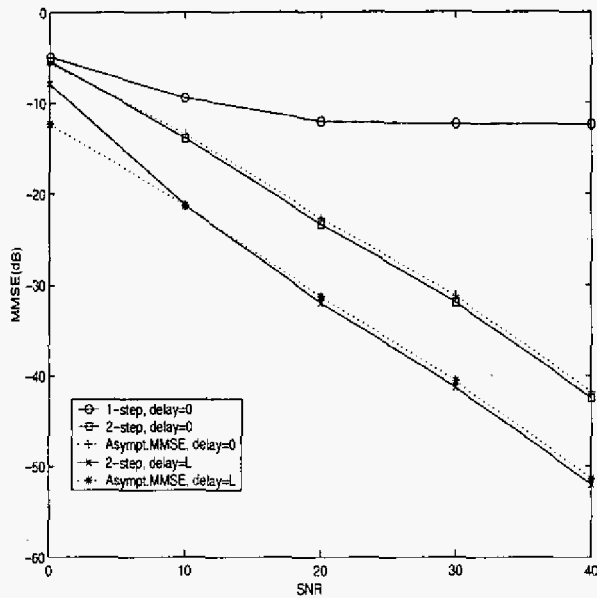


Fig. 2. Performance gain of the non-zero delay equalizer.

has been proposed in order to control the equalization delay and improve the estimation performance. Robustness against channel order over-estimation errors and (asymptotic) estimation performance of proposed equalization method are studied.

10. REFERENCES

- [1] A. Chkeif, K. Abed-Meraim, G. K. Kaleh, and Y. Hua, "Spatio-Temporal Blind Adaptive Multiuser Detection", *IEEE Trans. Communication*, vol. 48, no. 5, pp. 729-732, May 2000.
- [2] K. Abed-Meraim, A. Chkeif, and Y. Hua, "Fast orthogonal PAST algorithm", *IEEE Signal Processing Letters*, vol. 7, no. 3, pp. 60-62, Mar. 2000.
- [3] S. Haykin, "Adaptive Filter Theory", *Prentice Hall, International Eds.*, 1996.
- [4] E. Moulines, P. Duhamel, J.F. Cardoso, and S. Mayrargue, "Subspace Methodes for the Blind Identification of Multi-channel FIR Filters", *IEEE Trans. Signal Processing*, vol. 43, no. 2, pp. 516-525, Feb. 1995.
- [5] B. Yang, "Projection Approximation Subspace Tracking", *IEEE Trans. Signal Processing*, vol. 43, no. 1, pp. 95-107, Jan. 1995.
- [6] L. Tong, G. Xu, and T. Kailath, "Blind Identification and Equalization of Multipath Channels", *In Proc. Int. Conf. Commun.*, June 1992, pp. 1513-1517.
- [7] J.F. Cardoso and E. Moulines, "Asymptotic performance analysis of direction-finding algorithms based on fourth-order cumulants", *IEEE Tr. on Sig. Proc.*, vol. 43, no. 1, pp. 214-224, Jan. 1995.

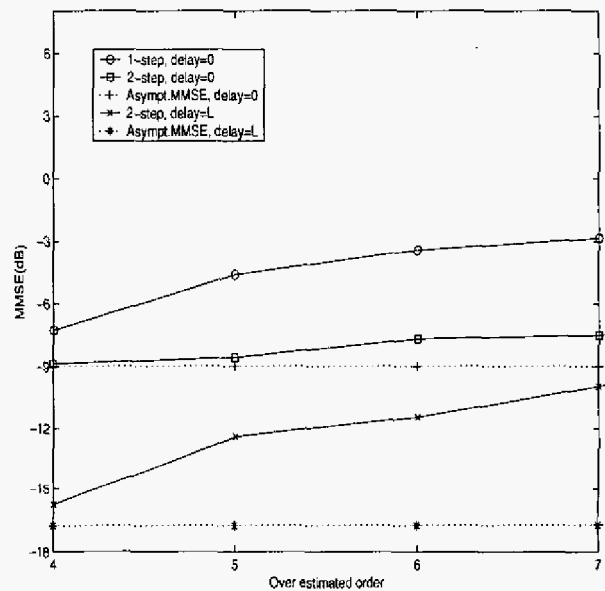


Fig. 3. Robustness against order over-estimation errors.

A new blind adaptive MMSE equalizer for MIMO systems

Ibrahim Kacha

Electrical Engineering Department
École Nationale Polytechnique (ENP)
PO Box 182, El Harrach 16200
Algiers, Algeria
Email: ibrahim.kacha@enp.edu.dz

Karim Abed-Meraim

Signal and Image Processing Department
Telecom Paris (ENST)
46, rue Barrault, 75634
Paris Cedex 13, France
Email: abed@tsi.enst.fr

Adel Belouchrani

Electrical Engineering Department
École Nationale Polytechnique (ENP)
PO Box 182, El Harrach 16200
Algiers, Algeria
Email: adel.belouchrani@enp.edu.dz

Abstract—In this paper, we consider the problem of blind equalization of MIMO systems. In a recent work a blind MMSE equalizer robust to channel order over-estimation errors was proposed for SIMO systems. In this work, we extend this method and propose a new MMSE algorithm that presents better robustness and better tracking performances. Moreover, the proposed MMSE technique has been extended here to the MIMO case. Simulation results are then given to illustrate the effectiveness of our MMSE algorithms.

I. INTRODUCTION

Adaptive channel equalization techniques have been widely used in communication systems for their convenience in real time applications. Conventional equalizer algorithms require training sequences to update the parameters of the equalizer. This solution clearly reduces the effective channel throughput and alternative methods have been considered using blind or semi-blind equalization algorithms [1][2].

Recently, a blind adaptive equalization algorithm based on MMSE estimation, which presents a number of nice properties such as robustness to channel order over-estimation errors and low computational complexity has been introduced in [3]. In this paper we extend this work and propose: (i) a modified block implementation of the MMSE equalizer that is more robust to channel order over-estimation errors than the one in [3], (ii) a modified adaptive implementation that has better tracking performances especially for fast varying channels and (iii) an extended version of the equalizer for MIMO systems.

II. DATA MODEL

Consider a MIMO system of q inputs, p outputs ($p > q$) given by:

$$\mathbf{x}(t) = \sum_{k=0}^L \mathbf{H}(k)\mathbf{s}(t-k) + \mathbf{b}(t) \quad (1)$$

where $\mathbf{H}(z) = \sum_{k=0}^L \mathbf{H}(k)z^{-k}$ is an unknown causal FIR $p \times q$ transfer function. We assume (A1) $\mathbf{H}(z) \neq 0, \forall z$. (A2) The input (non-observable) signal $\mathbf{s}(t)$ is a q -dimensional random vector assumed to be iid zero-mean process of power $\sigma_s^2 \mathbf{I}_q$ and (A3) $\mathbf{b}(t)$ is an additive spatially and temporally white noise of power $\sigma_b^2 \mathbf{I}_p$ and independent of the transmitted sequence $\{\mathbf{s}(t)\}$.

By stacking N successive samples of the received signal $\mathbf{x}(t)$ into a single vector, we obtain the n -dimensional ($n = Np$) vector:

$$\begin{aligned} \mathbf{x}_N(t) &= [\mathbf{x}^T(t) \mathbf{x}^T(t-1) \dots \mathbf{x}^T(t-N+1)]^T \quad (2) \\ &= \mathbf{H}\mathbf{s}_r(t) + \mathbf{b}_N(t) \quad (3) \end{aligned}$$

where $\mathbf{s}_r(t) = [\mathbf{s}^T(t) \dots \mathbf{s}^T(t-r+1)]^T$, ($r = N+L$) and \mathbf{H} is the channel convolution matrix of dimension $n \times d$, ($d = qr$), given by:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}(0) & \dots & \mathbf{H}(L) & \mathbf{0} \\ & \ddots & & \ddots \\ \mathbf{0} & & \mathbf{H}(0) & \dots & \mathbf{H}(L) \end{bmatrix}. \quad (4)$$

It is shown in [4], that if N is large enough and under assumption (A1), matrix \mathbf{H} is full column rank.

III. MMSE EQUALIZER

Consider a τ -delay MMSE equalizer ($\tau \in \{0, 1, \dots, r-1\}$). Under the above data model, one can easily show that the equalizer matrix \mathbf{V}_τ corresponding to the desired solution is given by:

$$\mathbf{V}_\tau = \arg \min_{\mathbf{V}} E(\|\mathbf{s}(t-\tau) - \mathbf{V}^H \mathbf{x}_N(t)\|^2) \quad (5)$$

$$= \mathbf{C}^{-1} \mathbf{G}_\tau \quad (6)$$

where:

$$\mathbf{C} \stackrel{\text{def}}{=} E[\mathbf{x}_N(t)\mathbf{x}_N^H(t)] \quad (7)$$

is the data covariance matrix and \mathbf{G}_τ is an $n \times q$ vector given by:

$$\mathbf{G}_\tau \stackrel{\text{def}}{=} E(\mathbf{x}_N(t)\mathbf{s}^H(t-\tau)) \quad (8)$$

$$= \sigma_s^2 \mathbf{H}(:, \tau q + 1 : (\tau + 1)q) \quad (9)$$

where $\mathbf{H}(:, \tau q + 1 : (\tau + 1)q)$ denotes the sub-matrix of \mathbf{H} given by the column vectors of indices varying in the range $[\tau q + 1, \dots, (\tau + 1)q]$ (In this paper, we use MATLAB notation to refer to sub-matrices selection). Clearly, the columns of MMSE matrix filter \mathbf{V}_τ belongs to the signal subspace (i.e. $\text{Range}(\mathbf{H})$) and thus one can write

$$\mathbf{V}_\tau = \mathbf{W}\tilde{\mathbf{V}}_\tau \quad (10)$$

where \mathbf{W} is a $n \times d$ matrix which column vectors form a basis of the signal subspace and $\tilde{\mathbf{V}}_\tau$ is a $d \times q$ dimensional matrix. Our objective here is to derive a blind estimate of the zero-delay MMSE equalizer \mathbf{V}_0 . From equations (6), (9) and (10), one can write

$$\mathbf{C}\mathbf{W}\tilde{\mathbf{V}}_0 = \sigma_s^2 \begin{bmatrix} \mathbf{H}(0) \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \quad (11)$$

equation (11) implies that

$$\overline{\mathbf{C}}\mathbf{W}\tilde{\mathbf{V}}_0 = \mathbf{0} \quad (12)$$

where $\overline{\mathbf{C}}$ is a submatrix of \mathbf{C} given by its last $(n-p)$ rows. Reversely, we established that this equation characterizes uniquely the zero-delay MMSE equalizer. We have the following result (the proof is omitted due to space limitation):

Theorem 1: Under the above data assumptions and for $N > qL$ the solution of

$$\overline{\mathbf{C}}\mathbf{W}\tilde{\mathbf{V}} = \mathbf{0} \quad (13)$$

subject to the constraint: $\text{rank}(\tilde{\mathbf{V}}) = q$, is unique (up to a constant $q \times q$ non-singular matrix) and corresponds to the desired MMSE equalizer, i.e. $\tilde{\mathbf{V}} = \tilde{\mathbf{V}}_0\mathbf{R}$ for a given constant $q \times q$ invertible matrix \mathbf{R} .

Note that the $q \times q$ constant matrix \mathbf{R} comes from the inherent indeterminacies of MIMO systems blind identification using second order statistics [4]. Usually, this indeterminacy is solved by applying HOS (Higher Order Statistics)-based blind source separation algorithms.

In the SIMO case ($q = 1$), equation (13) can be solved in the least squares sense subject to the unit norm constraint, i.e. the d -dimensional equalizer vector corresponds (up to a scalar factor) to the least eigen vector of the matrix $\mathbf{W}^H\overline{\mathbf{C}}^H\overline{\mathbf{C}}\mathbf{W}$. Batch and fast adaptive implementation algorithms are developed in [3]. Nevertheless, this approach can not be extended to MIMO case ($q > 1$), owing to the fact that the constraint in Theorem 1 is not satisfied. One possible solution is to impose a linear constraint (instead of the quadratic constraint) such as:

$$\tilde{\mathbf{V}}(1:q,:) = \begin{bmatrix} 1 & \cdots & 0 \\ \times & \ddots & \vdots \\ \times & \times & 1 \end{bmatrix} \quad (14)$$

what will guarantee that matrix $\tilde{\mathbf{V}}$ has a full columns rank q . Let

$$\mathbf{A} = \mathbf{I}_q \otimes (\overline{\mathbf{C}}\mathbf{W}) \quad (15)$$

$$\mathbf{v} = \text{Vec}(\tilde{\mathbf{V}}) \quad (16)$$

and for $k = 1, \dots, q$

$$\mathbf{a}(k) = \mathbf{A}(:, (k-1)d+k) \quad (17)$$

$$\bar{\mathbf{A}}(k) = \mathbf{A}(:, (k-1)d+k+1:kd) \quad (18)$$

$$\bar{\mathbf{v}}(k) = \mathbf{v}((k-1)d+k+1:kd) \quad (19)$$

where \otimes is the Kronecker product and $\text{Vec}(\cdot)$ is the column vectorization operator.

It follows from equations (13) and (14) that

$$\mathbf{a} + \bar{\mathbf{A}}\bar{\mathbf{v}} = \mathbf{0} \quad (20)$$

where

$$\mathbf{a} = \sum_{k=1}^q \mathbf{a}(k) \quad (21)$$

$$\bar{\mathbf{A}} = [\bar{\mathbf{A}}(1) \quad \cdots \quad \bar{\mathbf{A}}(q)] \quad (22)$$

$$\bar{\mathbf{v}} = [\bar{\mathbf{v}}(1)^T \quad \cdots \quad \bar{\mathbf{v}}(q)^T]^T. \quad (23)$$

The solution of equation (20) is then given by

$$\bar{\mathbf{v}}_0 = -\bar{\mathbf{A}}^\# \mathbf{a} \quad (24)$$

where $(\cdot)^\#$ represents the matrix pseudo-inversion operator.

Finally, by defining the $d \times q$ matrix $\tilde{\mathbf{V}}_0$ as

$$i_k = \frac{k(k+1)}{2} \quad (25)$$

$$\bar{\mathbf{v}}_0(k) = \bar{\mathbf{v}}_0((k-1)d - i_{k-1} + 1 : kd - i_k) \quad (26)$$

$$\tilde{\mathbf{V}}_0(:, k) = [\mathbf{0}_{1 \times (k-1)} \quad 1 \quad \bar{\mathbf{v}}_0(k)^T]^T \quad (27)$$

for $k = 1 \dots q$, we obtain the equalizer matrix (up to a constant invertible matrix)

$$\mathbf{V}_0 = \mathbf{W}\tilde{\mathbf{V}}_0. \quad (28)$$

Thus, we obtain a block-processing implementation of the blind MMSE equalization algorithm that is summarized in Table I (we use here informal MATLAB notation and EVD represents the output of the eigen-decomposition with sorted (in decreasing order) eigenvalues).

TABLE I
BLIND MMSE EQUALIZATION ALGORITHM

\mathbf{C}	$= \frac{1}{T-N+1} \sum_{t=N}^T \mathbf{x}_N(t)\mathbf{x}_N^H(t)$, (T : sample size)
(\mathbf{U}, Λ)	$= \text{EVD}(\mathbf{C})$
\mathbf{W}	$= \mathbf{U}(:, 1:d)$
$\overline{\mathbf{C}}$	$= \mathbf{C}(p+1:n, :)$
\mathbf{A}	$= \mathbf{I}_q \otimes (\overline{\mathbf{C}}\mathbf{W})$
$\mathbf{a}(k)$	$= \mathbf{A}(:, (k-1)d+k)$, $k = 1, \dots, q$
$\bar{\mathbf{A}}(k)$	$= \mathbf{A}(:, (k-1)d+k+1:kd)$, $k = 1, \dots, q$
\mathbf{a}	$= \sum_{k=1}^q \mathbf{a}(k)$
$\bar{\mathbf{A}}$	$= [\bar{\mathbf{A}}(1) \quad \cdots \quad \bar{\mathbf{A}}(q)]$
$\bar{\mathbf{v}}_0$	$= -\bar{\mathbf{A}}^\# \mathbf{a}$
i_k	$= \frac{k(k+1)}{2}$, $k = 1, \dots, q$
$\bar{\mathbf{v}}_0(k)$	$= \bar{\mathbf{v}}_0((k-1)d - i_{k-1} + 1 : kd - i_k)$, $k = 1, \dots, q$
$\tilde{\mathbf{V}}_0(:, k)$	$= [\mathbf{0}_{1 \times (k-1)} \quad 1 \quad \bar{\mathbf{v}}_0(k)^T]^T$, $k = 1, \dots, q$
\mathbf{V}_0	$= \mathbf{W}\tilde{\mathbf{V}}_0$, (up to a constant invertible matrix)

Remark: The proposed MMSE equalizer is more robust against channel order over-estimation errors than the one in [3]. Indeed solution of (13) is, in general, a linear combination of the desired solution \mathbf{V}_0 (that lives in the signal subspace) and noise subspace vectors (introduced by the channel order over-estimation errors). However, it is observed that, for a finite sample size and for moderate and high SNRs the contribution of the desired solution \mathbf{V}_0 in (13) is much higher

than that of the noise subspace vectors. Consequently, in that context, solving (13) in the least squares sense under unit norm constraint, leads to a solution that lives almost in the noise subspace (i.e. the part of \mathbf{V}_0 in the final solution becomes negligible). On the other hand, by solving (13) subject to linear constraint (equations (24) and (28)), one obtain a solution where the linear factor of \mathbf{V}_0 is more significant (which is due to the fact that vector \mathbf{a} in (24) belongs to the signal subspace).

IV. FAST ADAPTIVE IMPLEMENTATION

In tracking applications, we are interested in estimating the equalizer vector recursively. We introduce here a fast adaptive implementation of the proposed blind MMSE equalizer. For simplicity we present only the SIMO case, the extension to the MIMO case is straightforward but cumbersome. The matrix \mathbf{C} is replaced by its recursive estimate

$$\mathbf{C}(t) = \beta \mathbf{C}(t-1) + \mathbf{x}_N(t) \mathbf{x}_N^H(t) \quad (29)$$

where $0 < \beta < 1$ is a forgetting factor. The weight matrix \mathbf{W} corresponding to the d dominant eigenvectors of \mathbf{C} can be estimated using a fast subspace estimation and tracking algorithm. In this work, we use the YAST (Yet Another Subspace Tracker) algorithm in [5]. The choice of YAST algorithm is motivated by its remarkable tracking performance compared to other existing subspace tracking algorithms of similar computational complexity (PAST[6], OPAST[7], ..). The YAST algorithm is summarized in Table II. Note that only $O(nd)$ operations are required at each time instant (instead of $O(n^3)$ for a full EVD). The vector $\mathbf{x}'(t) = \mathbf{C}(t-1) \mathbf{x}_N(t)$ can be computed in $O(n)$ operations, by using the shift invariance property of the correlation matrix, similarly to [8].

The second step of our algorithm consists of estimating recursively the $(d-1)$ -dimensional vector $\bar{\mathbf{v}}_0$. From equations (15), (21) and (22) we obtain

$$\mathbf{a}(t) = \bar{\mathbf{C}}(t) \mathbf{w}(t) \quad (30)$$

$$\bar{\mathbf{A}}(t) = \bar{\mathbf{C}}(t) \bar{\mathbf{W}}(t) \quad (31)$$

where $\mathbf{w}(t)$ is the first column of matrix $\mathbf{W}(t)$ and $\bar{\mathbf{W}}(t)$ is a sub-matrix of $\mathbf{W}(t)$ given its last $(d-1)$ columns. Equation (29) yields

$$\bar{\mathbf{C}}(t) = \beta \bar{\mathbf{C}}(t-1) + \bar{\mathbf{x}}_N(t) \mathbf{x}_N^H(t) \quad (32)$$

where $\bar{\mathbf{x}}_N(t)$ is a sub-vector of $\mathbf{x}_N(t)$ given by its last $(n-p)$ elements. The projection approximation

$$\mathbf{C}(t) \mathbf{W}(t) \approx \mathbf{C}(t) \mathbf{W}(t-1)$$

which is valid if matrix $\mathbf{W}(t)$ is slowly varying with time [7], implies

$$\bar{\mathbf{C}}(t) \mathbf{w}(t) \approx \bar{\mathbf{C}}(t) \mathbf{w}(t-1) \quad (33)$$

$$\bar{\mathbf{C}}(t) \bar{\mathbf{W}}(t) \approx \bar{\mathbf{C}}(t) \bar{\mathbf{W}}(t-1). \quad (34)$$

TABLE II
YAST ALGORITHM

$\mathbf{y}(t)$	$= \mathbf{W}^H(t-1) \mathbf{x}_N(t)$
$\mathbf{x}'(t)$	$= \mathbf{C}(t-1) \mathbf{x}_N(t)$
$\mathbf{y}'(t)$	$= \mathbf{W}^H(t-1) \mathbf{x}'(t)$
$\sigma(t)$	$= (\mathbf{x}_N(t)^H \mathbf{x}_N(t) - \mathbf{y}(t)^H \mathbf{y}(t))^{\frac{1}{2}}$
$\mathbf{h}(t)$	$= \mathbf{Z}(t-1) \mathbf{y}(t)$
$\gamma(t)$	$= (\beta + \mathbf{y}(t)^H \mathbf{h}(t))^{-1}$
$\bar{\mathbf{Z}}(t)$	$= \frac{1}{\beta} (\mathbf{Z}(t-1) - \mathbf{h}(t) \gamma(t) \mathbf{h}(t)^H)$
$\alpha(t)$	$= \mathbf{x}_N(t)^H \mathbf{x}_N(t)$
$\mathbf{y}''(t)$	$= \beta \mathbf{y}'(t) + \mathbf{y}(t) \alpha(t)$
$c_{yy}(t)$	$= \beta \mathbf{x}_N(t)^H \mathbf{x}'(t) + \alpha^*(t) \alpha(t)$
$\mathbf{h}'(t)$	$= \bar{\mathbf{Z}}(t-1) \mathbf{y}''(t)$
$\gamma'(t)$	$= (c_{yy}(t) - \mathbf{y}''(t)^H \mathbf{h}'(t))^{-1}$
$\mathbf{h}''(t)$	$= \mathbf{h}'(t) - \mathbf{y}(t)$
$\bar{\mathbf{Z}}'(t)$	$= \bar{\mathbf{Z}}(t) + \mathbf{h}''(t) \gamma'(t) \mathbf{h}''(t)^H$
$\mathbf{g}(t)$	$= \mathbf{h}''(t) \gamma'(t) \sigma^*(t)$
$\gamma''(t)$	$= \sigma(t) \gamma'(t) \sigma^*(t)$
$\mathbf{Z}'(t)$	$= [\bar{\mathbf{Z}}'(t), -\mathbf{g}(t); -\mathbf{g}(t)^H, \gamma''(t)]$
$(\phi(t), \lambda(t))$	$= \text{eigs}(\mathbf{Z}'(t), 1)$
$\varphi(t)$	$= \phi_{(1,d)}(t)$
$z(t)$	$= \phi_{(d+1)}(t)$
$\rho(t)$	$= z(t) $
$\theta(t)$	$= e^{j \arg(z(t))}$
$\mathbf{f}(t)$	$= \varphi(t) \theta^*(t)$
$\mathbf{f}'(t)$	$= \mathbf{f}(t) (1 + \rho(t))^{-1}$
$\mathbf{y}'''(t)$	$= \mathbf{y}(t) \sigma^{-1}(t) - \mathbf{f}'(t)$
$\mathbf{e}(t)$	$= \mathbf{x}(t) \sigma^{-1}(t) - \mathbf{W}(t-1) \mathbf{y}'''(t)$
$\mathbf{W}(t)$	$= \mathbf{W}(t-1) - \mathbf{e}(t) \mathbf{f}^H(t)$
$\mathbf{g}'(t)$	$= \mathbf{g}(t) + \mathbf{f}'(t) (\gamma''(t) - \theta(t) \lambda(t) \theta^*(t))$
$\mathbf{Z}(t)$	$= \bar{\mathbf{Z}}'(t) + \mathbf{g}'(t) \mathbf{f}'(t)^H + \mathbf{f}'(t) \mathbf{g}(t)^H$

Hence, applying approximations (33), (34) and substituting equation (32) into (30) and (31) yields

$$\mathbf{a}(t) = \beta \mathbf{a}(t-1) + \mu(t) \bar{\mathbf{x}}_N(t) \quad (35)$$

$$\bar{\mathbf{A}}(t) = \beta \bar{\mathbf{A}}(t-1) + \bar{\mathbf{x}}_N(t) \bar{\mathbf{y}}(t)^H \quad (36)$$

where the scalar $\mu(t)$ and the $(d-1)$ -dimensional vector $\bar{\mathbf{y}}(t)$ are given by

$$\mu(t) = \mathbf{x}_N(t)^H \mathbf{w}(t-1) \quad (37)$$

$$\bar{\mathbf{y}}(t) = \bar{\mathbf{W}}(t-1)^H \mathbf{x}_N(t) \quad (38)$$

Note that $\bar{\mathbf{y}}(t)$ is considered sub-vector of $\mathbf{y}(t)$ computed in YAST (cf. TableII), given by its last $(d-1)$ elements.

Consider the $(d-1) \times (d-1)$ matrix

$$\mathbf{F}(t) \stackrel{\text{def}}{=} \bar{\mathbf{A}}(t)^H \bar{\mathbf{A}}(t) \quad (39)$$

substituting equation (36) into (39) yields

$$\mathbf{F}(t) = \beta^2 \mathbf{F}(t-1) - \mathbf{D}(t) \mathbf{J}^{-1}(t) \mathbf{D}^H(t) \quad (40)$$

where $\mathbf{D}(t)$ is the $(d-1) \times 2$ matrix

$$\mathbf{D}(t) = [\beta \bar{\mathbf{A}}^H(t-1) \bar{\mathbf{x}}_N(t) \quad \bar{\mathbf{y}}(t)] \quad (41)$$

and $\mathbf{J}(t)$ is the 2×2 non-singular matrix

$$\mathbf{J}(t) = \begin{bmatrix} \|\bar{\mathbf{x}}_N(t)\|^2 & -1 \\ -1 & 0 \end{bmatrix}. \quad (42)$$

Using the matrix (Schur) inversion lemma [9], we obtain

$$\mathbf{M}(t) \stackrel{\text{def}}{=} \mathbf{F}(t)^{-1} \quad (43)$$

$$= \frac{1}{\beta^2} \mathbf{M}(t-1) + \mathbf{S}(t) \boldsymbol{\Sigma}(t) \mathbf{S}^H(t) \quad (44)$$

where $\mathbf{S}(t)$ is the $(d-1) \times 2$ matrix

$$\mathbf{S}(t) = \frac{1}{\beta^2} \mathbf{M}(t-1) \mathbf{D}(t) \quad (45)$$

and $\boldsymbol{\Sigma}(t)$ is the 2×2 matrix

$$\boldsymbol{\Sigma}(t) = (\mathbf{J}(t) - \mathbf{S}^H(t) \mathbf{D}(t))^{-1}. \quad (46)$$

Finally, equation (24) can be replaced by

$$\bar{\mathbf{v}}_0(t) = -\bar{\mathbf{A}}(t)^\# \mathbf{a}(t) \quad (47)$$

$$= -(\bar{\mathbf{A}}(t)^H \bar{\mathbf{A}}(t))^{-1} \bar{\mathbf{A}}(t)^H \mathbf{a}(t) \quad (48)$$

$$= -\mathbf{M}(t) \mathbf{m}(t) \quad (49)$$

where

$$\mathbf{m}(t) = \bar{\mathbf{A}}(t)^H \mathbf{a}(t). \quad (50)$$

Thus the MMSE equalizer vector is expressed (up to a scalar constant) by

$$\mathbf{v}_0(t) = \mathbf{W}(t) [1 \quad \bar{\mathbf{v}}_0(t)^T]^T \quad (51)$$

Note that the whole processing requires only $O(nd)$ operations. The complete pseudo-code for the fast adaptive blind MMSE equalization algorithm is given in Table III.

TABLE III
ADAPTIVE BLIND MMSE EQUALIZATION ALGORITHM

• Update $\mathbf{W}(t)$ and $\mathbf{y}(t)$ using YAST (cf. Table II)
$\bar{\mathbf{y}}(t) = \mathbf{y}^{(2:d)}(t)$
$\mathbf{J}(t) = \begin{bmatrix} \ \bar{\mathbf{x}}_N(t)\ ^2 & -1 \\ -1 & 0 \end{bmatrix}$
$\mathbf{D}(t) = \beta \bar{\mathbf{A}}^H(t-1) \bar{\mathbf{x}}_N(t) \quad \bar{\mathbf{y}}(t)$
$\mathbf{S}(t) = \frac{1}{\beta^2} \mathbf{M}(t-1) \mathbf{D}(t)$
$\boldsymbol{\Sigma}(t) = (\mathbf{J}(t) - \mathbf{S}^H(t) \mathbf{D}(t))^{-1}$
$\mathbf{M}(t) = \frac{1}{\beta^2} \mathbf{M}(t-1) + \mathbf{S}(t) \boldsymbol{\Sigma}(t) \mathbf{S}^H(t)$
$\bar{\mathbf{A}}(t) = \beta \bar{\mathbf{A}}(t-1) + \bar{\mathbf{x}}_N(t) \bar{\mathbf{y}}^H(t)$
$\mu(t) = \mathbf{x}_N(t)^H \mathbf{w}(t-1)$
$\mathbf{a}(t) = \beta \mathbf{a}(t-1) + \mu(t) \bar{\mathbf{x}}_N(t)$
$\mathbf{m}(t) = \bar{\mathbf{A}}(t)^H \mathbf{a}(t)$
$\bar{\mathbf{v}}_0(t) = -\mathbf{M}(t) \mathbf{m}(t)$
$\mathbf{v}_0(t) = \mathbf{W}(t) [1 \quad \bar{\mathbf{v}}_0(t)^T]^T$ (up to a scalar constant)

V. TWO-STEP EQUALIZATION ALGORITHM

It is shown in [3] that the blind estimation of the MMSE filter results in a performance loss compared to the non blind one. To compensate for this performance loss and also to have a controlled non-zero equalization delay, we propose here a two-step approach to estimate the blind MMSE equalizer. In the first step, we estimate \mathbf{V}_0 according to the previous algorithms, while, in the second step, we refine this estimation by exploiting the a priori knowledge of the finite alphabet to

which belongs the symbols $s(t)$. This is done by performing (in batch or adaptive way) a blind source separation and a hard decision on the symbols that are then used to re-estimate \mathbf{V}_τ according to equations (6) and (9) (We assume here the use of a differential modulation to get rid of the phase indeterminacy inherent to the blind equalization problem).

Let $\mathbf{W} \in \mathbb{C}^{n \times d}$ be an orthonormal basis of the signal subspace. Since \mathbf{G}_τ belongs to the signal subspace, one can write (see [10]):

$$\mathbf{V}_\tau = \mathbf{W} (\mathbf{W}^H \mathbf{C} \mathbf{W})^{-1} \mathbf{W}^H \mathbf{G}_\tau \quad (52)$$

This expression of \mathbf{V}_τ is used for the fast adaptive implementation of this two-step algorithm since $\mathbf{Z} = (\mathbf{W}^H \mathbf{C} \mathbf{W})^{-1}$ is already computed by the YAST. The recursive expression of vector \mathbf{G}_τ is given by

$$\mathbf{G}_\tau(t) = \beta \mathbf{G}_\tau(t-1) + \mathbf{x}_N(t) s^H(t-\tau) \quad (53)$$

and thus equation (52) can be replaced by the following recursion

$$\mathbf{V}_\tau(t) = \mathbf{W}(t) \mathbf{Z}(t) \mathbf{W}^H(t) \mathbf{G}_\tau(t) \quad (54)$$

Note that, by choosing a non-zero equalizer delay τ , we improve the equalization performance as shown below.

VI. SIMULATION RESULTS

We provide in this section some simulation examples to illustrate the performance of the proposed blind equalizer. Our tests are based on SIMO ($q = 1$, $p = 3$ and $L = 4$) and MIMO ($q = 2$, $p = 3$ and $L = 4$) channels. The channel coefficients are chosen randomly at each run according to a complex Gaussian distribution. The input signals are iid QAM4 sequences. As a performance measure, we estimate the average MSE given by

$$\text{MSE} = \frac{1}{q} \min_{\mathbf{R}} E(\|\mathbf{s}(t-\tau) - \mathbf{R} \hat{\mathbf{V}}_\tau^H \mathbf{x}_N(t)\|^2) \quad (55)$$

over 100 Monte-Carlo runs. The MSE is compared to the theoretical MSE given by

$$\text{MSE}_{th} = \frac{1}{q} \text{Tr}(\mathbf{I}_q - \mathbf{G}_\tau^H \mathbf{C}^{-1} \mathbf{G}_\tau) \quad (56)$$

The signal-to-noise ration (SNR) is defined by $\text{SNR} = 20 \log(\frac{\sigma_s}{\sigma_n})$.

In Fig.1, we plot the MSE (in dB) against SNR (in dB) for $T = 500$. One can observe, the performance loss of our MMSE filter compared to the exact one due to the blind estimation procedure.

Fig.2, illustrates the performance gain of the two-step approach ($T = 500$), which allows to compensate the performance loss and choose a non-zero equalization delay.

Fig.3 compares the tracking performances of our algorithm with those of the algorithm in [3]. The latter uses the OPAST algorithm for subspace tracking instead of YAST. The channel variation model is the one given in [11] and the SNR is set to 15dB. As we can observe, our algorithm succeed to track the channel variations while the algorithm in [3] fails.

Fig.4 is dedicated to robustness study. The plots represent the

MSE obtained by our algorithm compared to those of the algorithm in [3]. Clearly, the use of linear constraint improves significantly the robustness of the blind MMSE filter.

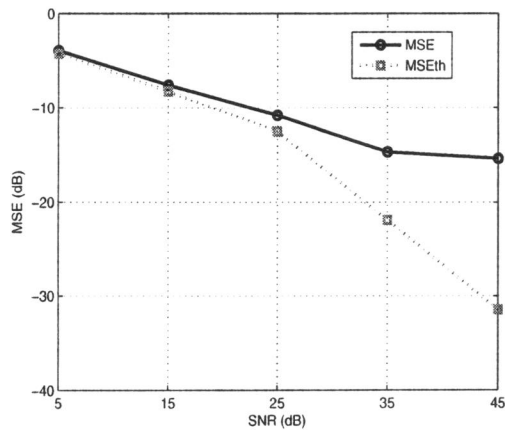


Fig. 1. Performance of the zero delay MIMO equalizer.

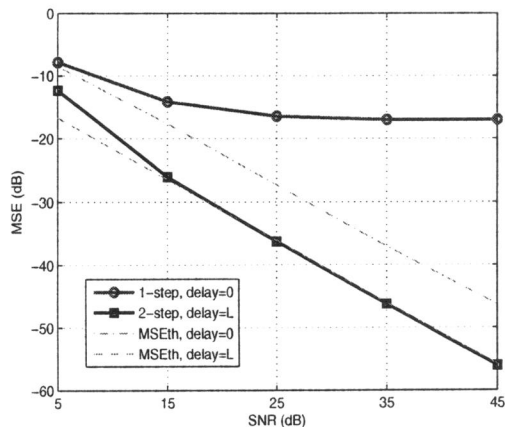


Fig. 2. Performance of the non-zero delay SIMO equalizer.

VII. CONCLUSION

This paper introduces a new blind estimation method of the MMSE equalizer filter for MIMO-FIR systems. Batch and fast adaptive implementation algorithms are developed. A two-step version using the *a priori* knowledge of the source signal finite alphabet has been proposed in order to control the equalization delay and improve the estimation performance. Robustness against channel order over-estimation errors and performance of proposed equalization method are studied.

REFERENCES

- [1] L. Tong and S. Perreau, "Multichannel blind identification: from subspace to maximum likelihood methods," *Proceedings of the IEEE*, vol. 86, no. 6, pp. 1951-1968, Oct. 1998.
- [2] E. de Carvalho and D. Slock, "Deterministic quadratic Semi-blind FIR multichannel estimation: algorithms and performance," *In Proc. ICASSP*, 2000, vol. 5, pp. 2553-2556.

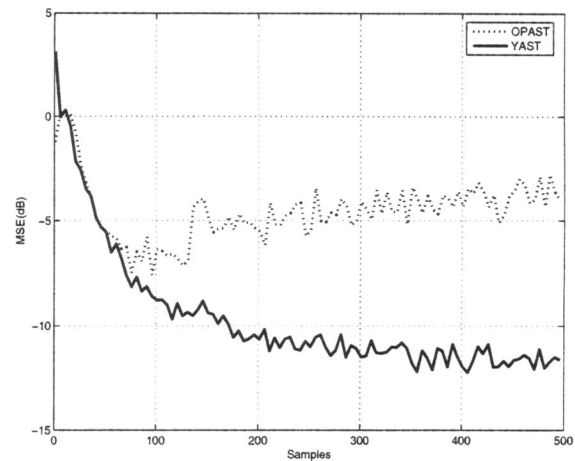


Fig. 3. Convergence of the adaptive MMSE equalization algorithm in the time varying channel case.

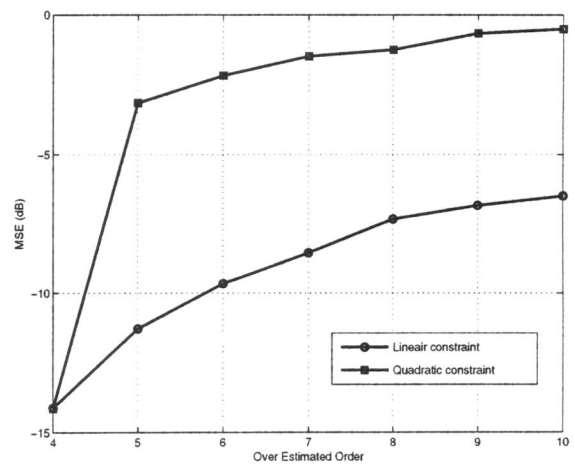


Fig. 4. Robustness against channel order over-estimation errors.

- [3] I. Kacha, K. Abed-Meraim, and A. Belouchrani, "A fast adaptive blind equalization algorithm robust to channel order over-estimation errors," *In Proc. of the 3rd IEEE SAM Signal Processing Workshop.*, July 2004.
- [4] K. Abed-Meraim, P. Loubaton, and E. Moulines, "A subspace algorithm for certain blind identification problems," *IEEE Trans. Signal Processing*, vol. 43, no. 2, pp. 499-511, Mar 1997.
- [5] R. Badeau, B. David, and G. Richard, "Yet another subspace tracker," *In Proc. ICASSP*, Mar 2005.
- [6] B. Yang, "Projection Approximation Subspace Tracking," *IEEE Trans. Signal Processing*, vol. 43, no. 1, pp. 95-107, Jan. 1995.
- [7] K. Abed-Meraim, A. Chkeif, and Y. Hua, "Fast orthogonal PAST algorithm," *IEEE Signal Processing Letters*, vol. 7, no. 3, pp. 60-62, Mar. 2000.
- [8] C.E. Davila, "Efficient, high performance, subspace tracking for time-domain data," *IEEE Trans. Signal Processing*, vol. 48, no. 12, pp. 3307-3315, Dec 2000.
- [9] S. Haykin, "Adaptive Filter Theory," *Prentice Hall, International Eds.*, 1996.
- [10] A. Chkeif, K. Abed-Meraim, and G.K. Kaleh, "Spatio-temporal blind adaptive multiuser detection," *IEEE Trans. Communication*, vol. 48, no. 5, pp. 729-732, May 2000.
- [11] M.K. Tsatsanis and G.B. Giannakis, "Modeling and equalization of rapidly fading channels," *Int. J. of Adaptive Control and Signal Processing*, vol. 10, nos. 2/3, pp. 159-176, Mar 1996.

BLIND ADAPTIVE EQUALIZATION METHOD WITHOUT CHANNEL ORDER ESTIMATION

I. Kacha^{1,2}, K. Abed-Meraim¹, A. Belouchrani²

¹ Télécom Paris (ENST) - Département TSI

37-39 rue Dareau, 75014 Paris, France

² École Nationale Polytechnique (ENP) - Département d'Électronique

10 avenue Hassan Badi El-Harrach, 16200 Alger, Algérie

e-mail: {kacha, abed}@tsi.enst.fr, adel.belouchrani@enp.edu.dz

ABSTRACT

In this paper, we propose a new blind minimum mean square error (MMSE) equalization algorithm of noisy single-input multiple-outputs finite impulse response (SIMO-FIR) systems, relying only on second order statistics. This algorithm offers an important advantage, a total independence of the channel order. Exploiting the fact that the equalizer filter belongs both, to the signal subspace and to the kernel of truncated data covariance matrix, the algorithm achieves blindly a direct estimation of the zero-delay MMSE equalizer parameters. The proposed approach has several features that are studied in this work. More precisely, we develop a two-step procedure to further improve the performance gain and to control the equalization delay. We present an efficient adaptive implementation of our equalizer, which reduces the computational complexity from $O(n^3)$ to $O(n^2p)$, where n is the data vector length and p is the number of sensors. Simulation results are provided to illustrate the effectiveness of the proposed blind equalization algorithm.

1. INTRODUCTION

Since the pioneer work of Tong *et al* [1], active research in blind Identification/Equalization area has led to a variety of second-order statistics-based algorithms (see the book [2], as well as the references therein). Many efficient solutions (e.g. [3]) suffer from the lack of robustness against channel order overestimation errors. A lot of research effort has been done to either develop efficient techniques for channel order estimation [4, 5] or to develop blind identification/equalization methods robust to channel order estimation errors. Several *robust* techniques have been proposed so far [6, 7, 8], but all of them depend explicitly or implicitly on the channel order and hence have only a relative robustness. In this paper, we describe a new technique for direct design of SIMO blind MMSE equalizer, completely independent of the channel order. We show first that the zero-delay equalizer filter belongs simultaneously, to the signal subspace and to the kernel of truncated data covariance matrix. This property is used, under some weak conditions, to estimate the parameters of the equalizer by maximizing a certain quadratic form subject to a properly chosen constraint. We present an efficient adaptive implementation of the novel algorithm, having only $O(n^2p)$ complexity. A two-step estimation procedure is included (in batch or adaptive way), which allows us to compensate for the performance loss of the equalizer compared to the non-blind one and to choose a non-zero equalization delay.

2. DATA MODEL

Consider a SIMO system of p outputs, given by

$$\mathbf{x}(t) = \sum_{k=0}^L \mathbf{h}(k)s(t-k) + \mathbf{b}(t), \quad (1)$$

where $\mathbf{h}(z) = \sum_{k=0}^L \mathbf{h}(k)z^{-k}$ is an unknown causal FIR $p \times 1$ transfer function. We assume **(A1)** $\mathbf{h}(z) \neq 0, \forall z$. **(A2)** The input (non-observable) signal $s(t)$ is a scalar iid zero-mean process of power σ_s^2 . **(A3)** $\mathbf{b}(t)$ is an additive spatially and temporally white noise of power $\sigma_b^2 \mathbf{I}_p$ and independent of the transmitted sequence $\{s(t)\}$. By stacking N successive samples of the received signal $\mathbf{x}(t)$ into a single vector, we obtain the n -dimensional ($n = Np$) vector

$$\begin{aligned} \mathbf{x}_N(t) &= [\mathbf{x}(t)^T \ \mathbf{x}(t-1)^T \ \dots \ \mathbf{x}(t-N+1)^T]^T \\ &= \mathbf{H}_N \mathbf{s}_m(t) + \mathbf{b}_N(t), \end{aligned} \quad (2)$$

where $\mathbf{s}_m(t) = [s(t) \ \dots \ s(t-m+1)]^T$, ($m = N+L$), $\mathbf{b}_N(t) = [\mathbf{b}(t)^T \ \dots \ \mathbf{b}(t-N+1)^T]^T$ and \mathbf{H}_N is the channel convolution matrix of dimension $n \times m$, given by

$$\mathbf{H}_N = \begin{bmatrix} \mathbf{h}(0) & \dots & \mathbf{h}(L) & & \mathbf{0} \\ & & & \ddots & \\ & & \mathbf{0} & \mathbf{h}(0) & \dots & \mathbf{h}(L) \end{bmatrix}. \quad (3)$$

It is shown in [9], that if $N > L$ and under assumption **(A1)**, matrix \mathbf{H}_N is full column rank.

3. ALGORITHM DERIVATION

3.1. MMSE equalizer

Consider a τ -delay MMSE equalizer ($\tau \in \{0, 1, \dots, m-1\}$). Under the above data model, one can show that the n -dimensional equalizer vector \mathbf{v}_τ corresponding to the desired solution is given by

$$\mathbf{v}_\tau = \arg \min_{\mathbf{v}} E(\|s(t-\tau) - \mathbf{v}^H \mathbf{x}_N(t)\|^2) = \mathbf{C}_N^{-1} \mathbf{g}_\tau, \quad (4)$$

where

$$\mathbf{C}_N \stackrel{\text{def}}{=} E[\mathbf{x}_N(t) \mathbf{x}_N(t)^H] = \sigma_s^2 \mathbf{H}_N \mathbf{H}_N^H + \sigma_b^2 \mathbf{I}_n, \quad (5)$$

is the data covariance matrix and \mathbf{g}_τ is an $n \times 1$ vector given by

$$\mathbf{g}_\tau \stackrel{\text{def}}{=} E(\mathbf{x}_N(t)s(t-\tau)^*) = \sigma_s^2 \mathbf{H}_N(:, \tau+1), \quad (6)$$

where $\mathbf{H}_N(:, \tau+1)$ denotes the $(\tau+1)$ -th column vector of \mathbf{H}_N^1 . Clearly, the MMSE filter \mathbf{v}_τ belongs to the signal subspace (i.e. $\text{range}(\mathbf{H}_N)$).

3.2. Blind equalization

Our objective here is to derive a blind estimate of the zero-delay MMSE equalizer \mathbf{v}_0 . From equations (4) and (6), \mathbf{v}_0 is given implicitly by

$$\mathbf{C}_N \mathbf{v}_0 = \sigma_s^2 \begin{bmatrix} \mathbf{h}(0) \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}. \quad (7)$$

If we truncate the first p rows of system (7), we obtain

$$\bar{\mathbf{C}} \mathbf{v}_0 = \mathbf{0}, \quad (8)$$

where $\bar{\mathbf{C}}$ is an $(n-p) \times n$ sub-matrix of \mathbf{C}_N given by its last $n-p$ rows, i.e.: $\bar{\mathbf{C}} = \mathbf{C}_N(p+1:n,:)$. It is obvious that \mathbf{v}_0 belongs to the right null space of $\bar{\mathbf{C}}$. Next, we establish a link between the structure of the right null space of $\bar{\mathbf{C}}$ and the zero-delay MMSE equalizer.

Lemma 1 *Under the above data assumptions and for $N > L+1$, the right null space of $\bar{\mathbf{C}}$: $\text{null}_r(\bar{\mathbf{C}}) = \{\mathbf{z} \in \mathbb{C}^n : \bar{\mathbf{C}}\mathbf{z} = \mathbf{0}\}$, is a p -dimensional subspace, where only one direction of it belongs to the signal subspace.*

Proof: First, notice that the $(n-p) \times (n-p)$ matrix \mathbf{C}_{N-1} is a full rank sub-matrix of $\bar{\mathbf{C}}$. It follows that $\dim(\text{null}_r(\bar{\mathbf{C}})) = p$. Let $\mathbf{W} \in \mathbb{C}^{n \times m}$ be an orthonormal basis of the signal subspace. Since $\text{range}(\mathbf{H}_N) = \text{range}(\mathbf{W}) = \text{range}(\mathbf{C}_N \mathbf{W})$, there exists a non-singular $m \times m$ matrix \mathbf{R} such that $\mathbf{C}_N \mathbf{W} = \mathbf{H}_N \mathbf{R}$. Therefore, $\bar{\mathbf{C}} \mathbf{W} = [\mathbf{0}_{(n-p) \times 1} \quad \mathbf{H}_{N-1}] \mathbf{R}$. As \mathbf{H}_{N-1} is full column rank, it implies that $\dim(\text{null}_r(\bar{\mathbf{C}} \mathbf{W})) = 1$, what carries out to conclude that only one direction of $\text{null}_r(\bar{\mathbf{C}})$ belongs to the signal subspace. ■

To extract the direction of $\text{null}_r(\bar{\mathbf{C}})$ which belongs to the signal subspace (the direction of \mathbf{v}_0), we use the technique proposed in [10]. The selection of the filter \mathbf{v} corresponding to the equalizer \mathbf{v}_0 is obtained by choosing the one with maximum power, i.e.

$$\max_{\mathbf{v} \in \text{null}_r(\bar{\mathbf{C}})} E(\|\mathbf{v}^H \mathbf{x}_N(t)\|^2) \Leftrightarrow \max_{\|\tilde{\mathbf{v}}\|=1} (\tilde{\mathbf{v}}^H \mathbf{B} \tilde{\mathbf{v}}), \quad (9)$$

where $\mathbf{B} \stackrel{\text{def}}{=} \mathbf{A}^H \mathbf{C}_N \mathbf{A}$, matrix $\mathbf{A} \in \mathbb{C}^{n \times p}$ is an orthonormal basis of $\text{null}_r(\bar{\mathbf{C}})$ and $\tilde{\mathbf{v}}$ is a p -dimensional vector such as $\mathbf{v} = \mathbf{A} \tilde{\mathbf{v}}$. Indeed, under the unit-norm constraint, the noise contribution in (9) is of constant power and thus the maximization in (9) concerns only the signal term. Besides, the other $p-1$ directions of $\text{null}_r(\bar{\mathbf{C}})$, that are complementary to \mathbf{v}_0 , are almost in the noise subspace (at least for high signal-to-noise ratio (SNR)). Therefore, the maximization in (9) leads essentially to the selection of the desired direction of vector \mathbf{v}_0 . A batch-processing implementation of the algorithm is summarized in table 1.

¹We use here some informal MATLAB notations.

$\begin{aligned} \mathbf{C}_N &= \frac{1}{K} \sum_{t=0}^{K-1} \mathbf{x}_N(t) \mathbf{x}_N(t)^H, \quad (K: \text{sample size}) \\ (\mathbf{U}, \Sigma, \mathbf{V}) &= \text{svd}(\mathbf{C}_N(p+1:n,:)) \\ \mathbf{A} &= \mathbf{V}(:, n-p+1:n) \\ \mathbf{B} &= \mathbf{A}^H \mathbf{C}_N \mathbf{A} \\ \tilde{\mathbf{v}} &= \text{the dominant eigenvector of } \mathbf{B} \\ \mathbf{v} &= \mathbf{A} \tilde{\mathbf{v}} \end{aligned}$

Table 1. Blind MMSE equalization algorithm.

3.3. Selection of the equalizer delay

It is shown in [7] that the blind estimation of the zero-delay MMSE filter results in a performance loss compared to the non-blind one. To compensate for this performance loss and also to have a controlled non-zero delay which helps to improve the performance of the equalizer, we propose here a two-step approach to estimate the blind MMSE equalizer. In the first step, we estimate \mathbf{v}_0 according to the previous algorithm, while, in the second step, we refine this estimation by exploiting the a priori knowledge of the finite alphabet to which belongs the symbols $s(t)$. This is done by performing a hard decision on the symbols² that are then used to re-estimate \mathbf{v}_τ according to equations (4) and (6). The two-step blind MMSE equalization algorithm is summarized in Table 1 and Table 2.

$\begin{aligned} &\text{Estimate } s(t), t = 0..K-1, \text{ (using } \mathbf{v} \text{ given by Table 1)} \\ \mathbf{g}_\tau &= \frac{1}{K} \sum_{t=\tau}^{K+\tau-1} \mathbf{x}_N(t) \hat{s}(t-\tau)^* \\ \mathbf{v}_\tau &= \mathbf{C}_N^{-1} \mathbf{g}_\tau \end{aligned}$

Table 2. Two-step equalization procedure.

4. ADAPTIVE IMPLEMENTATION

In order to reduce the global computational complexity of the algorithm, from $O(n^3)$ to $O(n^2 p)$, an adaptive implementation is proposed. The columns of matrix \mathbf{A} correspond to the p -least eigenvectors of matrix $\mathbf{Q} \stackrel{\text{def}}{=} \bar{\mathbf{C}}^H \bar{\mathbf{C}}$. Thus \mathbf{A} can be obtained as an orthogonal complement of the $(n-p)$ -dimensional major subspace of \mathbf{Q} . If $\mathbf{A}^\perp \in \mathbb{C}^{n \times (n-p)}$ is an orthonormal basis of the major subspace of \mathbf{Q} then $\mathbf{A} \mathbf{A}^H = \mathbf{I}_n - \mathbf{A}^\perp \mathbf{A}^{\perp H}$. As matrix \mathbf{C}_N is recursively updated by $\mathbf{C}_N(t) = \beta \mathbf{C}_N(t-1) + \mathbf{x}_N(t) \mathbf{x}_N(t)^H$, where $0 < \beta < 1$ is a forgetting factor, matrix $\bar{\mathbf{C}}$ is then replaced by the recursion $\bar{\mathbf{C}}(t) = \beta \bar{\mathbf{C}}(t-1) + \mathbf{x}_{N-1}(t-1) \mathbf{x}_N(t)^H$. Thus, matrix $\mathbf{Q}(t)$ is recursively given by

$$\mathbf{Q}(t) = \beta^2 \mathbf{Q}(t-1) + \mathbf{y}_1(t) \mathbf{y}_1(t)^H + \mathbf{y}_2(t) \mathbf{y}_2(t)^H, \quad (10)$$

²We assume here the use of a differential modulation to get rid of the phase indeterminacy inherent to the blind equalization problem.

where $\mathbf{y}_1(t)$ and $\mathbf{y}_2(t)$ are two n -dimensional vectors given by

$$\begin{aligned}\mathbf{y}_1(t) &= \sigma(t-1)(\beta\mathbf{x}'(t-1) + \lambda(t-1)\mathbf{x}_N(t)), \\ \mathbf{y}_2(t) &= j\sigma(t-1)(\beta\mathbf{x}'(t-1) - \frac{1}{\lambda(t-1)}\mathbf{x}_N(t)), \\ \lambda(t) &= \frac{1}{2}(\|\mathbf{x}_{N-1}(t)\|^2 + \sqrt{4 + \|\mathbf{x}_{N-1}(t)\|^4}), \\ \sigma(t) &= \sqrt{\frac{\lambda(t)}{1 + \lambda(t)^2}}, \quad \mathbf{x}'(t) = \bar{\mathbf{C}}(t)^H \mathbf{x}_{N-1}(t).\end{aligned}\quad (11)$$

It follows that matrix $\mathbf{A}^\perp(t)$ can be considered as an orthonormal basis of a major subspace of the *virtual* covariance matrix $\mathbf{Q}(t)$ of data sequence $\{\mathbf{y}_1(t), \mathbf{y}_2(t)\}_{t \in \mathbb{Z}}$. From data vectors $\mathbf{y}_1(t)$ and $\mathbf{y}_2(t)$, the columns of $\mathbf{A}^\perp(t)$ can be extracted recursively when performing twice a fast major subspace estimating and tracking algorithm. In this work, we use the OPAST (Orthogonal Projection Approximation Subspace Tracking) algorithm [11] that is summarized in Table 3, where $\mathbf{y}'(t)$ stands alternatively for $\mathbf{y}_1(t)$ and $\mathbf{y}_2(t)$. The choice of OPAST algorithm is motivated by its simplicity and its remarkable performance compared to other existing subspace tracking algorithms of similar computational complexity. Finally, the extraction of the dominant eigenvector of $\mathbf{B}(t)$ is

$$\begin{aligned}\mathbf{y}(t) &= \mathbf{A}^\perp(t-1)^H \mathbf{y}'(t) \\ \mathbf{q}(t) &= \frac{1}{\beta} \mathbf{Z}(t-1) \mathbf{y}(t) \\ \gamma(t) &= \frac{1}{1 + \mathbf{y}(t)^H \mathbf{q}(t)} \\ \mathbf{p}(t) &= \gamma(t) (\mathbf{y}'(t) - \mathbf{A}^\perp(t-1) \mathbf{y}(t)) \\ r(t) &= \frac{1}{\|\mathbf{q}(t)\|^2} \left(\frac{1}{\sqrt{1 + \|\mathbf{p}(t)\|^2 \|\mathbf{q}(t)\|^2}} - 1 \right) \\ \mathbf{p}'(t) &= r(t) \mathbf{A}^\perp(t-1) \mathbf{q}(t) + (1 + r(t)) \|\mathbf{q}(t)\|^2 \mathbf{p}(t) \\ \mathbf{Z}(t) &= \frac{1}{\beta} \mathbf{Z}(t-1) - \gamma(t) \mathbf{q}(t) \mathbf{q}(t)^H \\ \mathbf{A}^\perp(t) &= \mathbf{A}^\perp(t-1) + \mathbf{p}'(t) \mathbf{q}(t)^H\end{aligned}$$

Table 3. OPAST algorithm.

obtained by power iteration as

$$\tilde{\mathbf{v}}(t) = \frac{\mathbf{B}(t) \tilde{\mathbf{v}}(t-1)}{\|\mathbf{B}(t) \tilde{\mathbf{v}}(t-1)\|}. \quad (12)$$

The outline of the adaptive blind MMSE equalization algorithm is given in Table 4. Using matrix inversion lemma, matrix $\mathbf{F}(t) \stackrel{\text{def}}{=} \mathbf{C}_N(t)^{-1}$ is updated recursively as

$$\begin{aligned}\mathbf{F}(t) &= \frac{1}{\beta} (\mathbf{F}(t-1) - \mathbf{f}'(t) \mathbf{f}(t)^H), \\ \mathbf{f}(t) &= \mathbf{F}(t-1) \mathbf{x}_N(t), \\ \rho(t) &= \mathbf{f}(t)^H \mathbf{x}_N(t), \quad \mathbf{f}'(t) = \frac{\mathbf{f}(t)}{\beta + \rho(t)}.\end{aligned}\quad (13)$$

Thus, the adaptive version of the two-step procedure is summarized in Table 5. Note that the whole processing requires $O(n^2 p)$ operations per iteration.

Remark: It is possible to further reduce the computational complexity to $O(np)$. Matrix $\mathbf{A}(t)$ can be estimated directly in $O(np)$ operations using a fast minor subspace tracker such as FDPMP (Fast Data Projection Method) algorithm [12]. The computation of the vector $\mathbf{x}'(t) = \bar{\mathbf{C}}(t)^H \mathbf{x}_{N-1}(t)$ in (11) can be reduced from $O(n^2)$ to $O(np)$ by means of the technique described in [13], which exploits the shift invariance property of the correlation matrix. Using the projection approximation $\mathbf{C}_N(t) \mathbf{A}(t) \approx \mathbf{C}_N(t) \mathbf{A}(t-1)$,

$$\begin{aligned}\mathbf{y}_1(t) &= \sigma(t-1)(\beta\mathbf{x}'(t-1) + \lambda(t-1)\mathbf{x}_N(t)) \\ \mathbf{y}_2(t) &= j\sigma(t-1)(\beta\mathbf{x}'(t-1) - \frac{1}{\lambda(t-1)}\mathbf{x}_N(t)) \\ (\mathbf{A}'(t), \mathbf{Z}'(t)) &= \text{OPAST}(\mathbf{A}^\perp(t-1), \mathbf{Z}(t-1), \mathbf{y}_1(t), \beta) \\ (\mathbf{A}^\perp(t), \mathbf{Z}(t)) &= \text{OPAST}(\mathbf{A}'(t), \mathbf{Z}'(t), \mathbf{y}_2(t), \beta) \\ \mathbf{\Pi}(t) &= \mathbf{I}_n - \mathbf{A}^\perp(t) \mathbf{A}^\perp(t)^H \\ \mathbf{A}(t) &= \text{eigs}(\mathbf{\Pi}(t), p) \\ \mathbf{C}_N(t) &= \beta \mathbf{C}_N(t-1) + \mathbf{x}_N(t) \mathbf{x}_N(t)^H \\ \mathbf{B}(t) &= \mathbf{A}(t)^H \mathbf{C}_N(t) \mathbf{A}(t) \\ \tilde{\mathbf{v}}(t) &= \frac{\mathbf{B}(t) \tilde{\mathbf{v}}(t-1)}{\|\mathbf{B}(t) \tilde{\mathbf{v}}(t-1)\|} \\ \mathbf{v}(t) &= \mathbf{A}(t) \tilde{\mathbf{v}}(t) \\ \bar{\mathbf{C}}(t) &= \mathbf{C}_N(t)_{(p+1:n,:)} \\ \mathbf{x}'(t) &= \bar{\mathbf{C}}(t)^H \mathbf{x}_{N-1}(t) \\ \lambda(t) &= \frac{1}{2} (\|\mathbf{x}_{N-1}(t)\|^2 + \sqrt{4 + \|\mathbf{x}_{N-1}(t)\|^4}) \\ \sigma(t) &= \sqrt{\frac{\lambda(t)}{1 + \lambda(t)^2}}\end{aligned}$$

Table 4. Adaptive blind MMSE equalization algorithm.

Table 5. Adaptive two-step equalization procedure.

$$\begin{aligned}\text{Estimate } \hat{s}(t), \text{ (using } \mathbf{v}(t) \text{ given by Table 4)} \\ \mathbf{f}(t) &= \mathbf{F}(t-1) \mathbf{x}_N(t) \\ \rho(t) &= \mathbf{f}(t)^H \mathbf{x}_N(t) \\ \mathbf{f}'(t) &= \frac{\mathbf{f}(t)}{\beta + \rho(t)} \\ \mathbf{v}_\tau(t) &= \mathbf{v}_\tau(t-1) - \mathbf{f}'(t) (\mathbf{f}(t)^H \mathbf{g}_\tau(t-1) - \hat{s}(t-\tau)^*) \\ \mathbf{F}(t) &= \frac{1}{\beta} (\mathbf{F}(t-1) - \mathbf{f}'(t) \mathbf{f}(t)^H) \\ \mathbf{g}_\tau(t) &= \beta \mathbf{g}_\tau(t-1) + \mathbf{x}_N(t) \hat{s}(t-\tau)^*\end{aligned}$$

which is valid if matrix $\mathbf{A}(t)$ is slowly varying with time [11], matrix $\mathbf{B}(t)$ can be updated in $O(np)$ operations. Finally the equalizer filter $\mathbf{v}_\tau(t)$ in the two-step procedure can be computed via normalized least mean square (NLMS) algorithm only in $O(np)$ operations. However, the major disadvantage of this approach comes from the fact that fast $O(np)$ minor subspace tracker algorithms suffer from *local minima* problem when the minor subspace's dimension is much lower than that of the total subspace, i.e.: $p \ll n$ (which is exactly, the situation we are facing herein).

5. SIMULATION RESULTS

We provide in this section some simulation examples to illustrate the performance of the proposed blind equalizer. Our tests are based on SIMO channels ($p = 3$ and $L = 4$). The channel coefficients are chosen randomly at each run according to a complex Gaussian distribution. The input signals are iid unit-power QAM4 sequences. The width of the temporal window is $N = 6$. As a performance measure, we estimate the average MSE given by $\text{MSE} = \frac{1}{K} \sum_{t=\tau}^{K+\tau+1} (|s(t-\tau) - \hat{\mathbf{v}}_\tau^H \mathbf{x}_N(t)|^2)$ (resp. $\text{MSE}(t) = |s(t-\tau) - \hat{\mathbf{v}}_\tau(t)^H \mathbf{x}_N(t)|^2$) in batch processing case (resp. adaptive case), over 100 Monte-Carlo runs. The MSE is compared to the theoretical MSE given by $\text{MSE}_{\text{th}} = 1 - \mathbf{g}_\tau^H \mathbf{C}_N^{-1} \mathbf{g}_\tau$. SNR is defined by $\text{SNR} = -20 \log(\sigma_b)$. In Fig.1, we plot the MSE (in dB) against SNR (in dB) for $K = 500$. One can observe the performance loss of the zero-delay MMSE filter compared to the optimal one (especially at high SNRs) due, as shown in [7], to

the blind estimation procedure. Also, it illustrates the effectiveness of the two-step approach, which allows us to compensate for the performances loss and to choose a non-zero equalization delay. Fig.2 illustrates the convergence rate of the adaptive algorithm with SNR= 15dB. Fig.3 is dedicated to robustness against overestimation errors. The plots compare the MSE obtained by the algorithms in [7, 8] to those obtained by our algorithm (exact order $L = 4$, SNR= 15dB, $K = 500$). Clearly, our method is insensitive to channel order over-estimation errors.

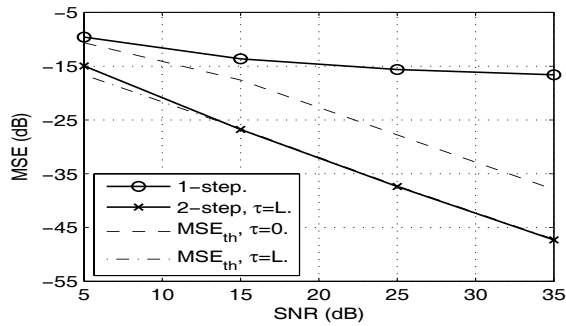


Fig. 1. Performance of the equalizer.

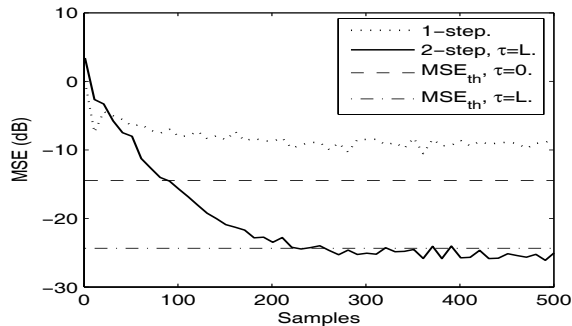


Fig. 2. Convergence of the adaptive equalizer.

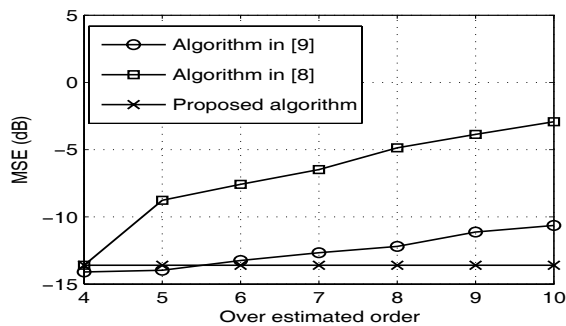


Fig. 3. Robustness against channel order over-estimation errors.

6. CONCLUSION

In this paper we have proposed a blind equalization technique for SIMO-FIR systems that does not need an estimate of the channel order. Batch and adaptive implementation are developed. A two-step approach, using the *a priori* knowledge of the source signal finite alphabet, has been proposed to compensate for the performance loss and to choose a non-zero equalization delay. Note that, the extension of the proposed technique to MIMO case is straightforward.

7. REFERENCES

- [1] L. Tong, G.Xu and T. Kailath, "A new approach to blind identification and equalization of multipaths channels," *Asilomar Conf. on Sig., Syst. and Comp.*, vol. 01, pp. 856-860, Morgantown, WV, 1991.
- [2] G.B. Giannakis, Y. Hua, P. Stoica and L. Tong, Eds., *Signal Processing Advances in wireless and mobile communications, Vol. 1: Trends in Channel Estimation and Equalization*. Englewood Cliffs, NJ: Prentice Hall, 2001.
- [3] E. Moulines, P. Duhamel, J.F. Cardoso and S. Mayrargue, "Subspace Methods for the Blind Identification of Multichannel FIR Filters," *IEEE Tr. Sig. Proc.*, vol. 43, no. 2, pp. 516-525, Feb. 1995.
- [4] A. P. Liavas, P. A. Regalia and J. P. Delmas, "Blind channel approximation: Effective channel order determination," *IEEE Tr. Sig. Proc.*, vol. 47, no. 12, pp. 3336-3344, Dec. 1999.
- [5] W. H. Gerstacker and D. P. Taylor, "Blind channel order estimation based on second-order statistics," *IEEE Sig. Proc. Letters*, vol. 10, no. 02, pp. 39-42, Feb. 2003.
- [6] H. Gazzah, P. A. Regalia, J. P. Delmas and K. Abed-Meraim, "A Blind multichannel identification algorithm robust to order overestimation," *IEEE Tr. Sig. Proc.*, vol. 50, no. 06, pp. 1449-1458, June 2002.
- [7] I. Kacha, K. Abed-Meraim and A. Belouchrani, "A fast adaptive blind equalization algorithm robust to channel order over-estimation errors," *SAM Sig. Proc. Work.*, pp. 148-152, Barcelona, July 2004.
- [8] I. Kacha, K. Abed-Meraim and A. Belouchrani, "A new blind adaptive MMSE equalizer for MIMO systems," *Int. Sym. on PIMRC*, Berlin, Sep. 2005.
- [9] K. Abed-Meraim, P. Loubaton and E. Moulines, "A subspace algorithm for certain blind identification problems," *IEEE Tr. Sig. Proc.*, vol. 43, no. 2, pp. 499-511, Mar. 1997.
- [10] K. Abed-Meraim and Y. Hua, "Blind equalization methods in colored noise field," *Inf. Dec. and Cont.*, pp. 477-481, Adelaide, Feb. 1999.
- [11] K. Abed-Meraim, A. Chkeif and Y. Hua, "Fast orthogonal PAST algorithm," *IEEE Sig. Proc. Letters*, vol. 7, no. 3, pp. 60-62, Mar. 2000.
- [12] X. G. Doukopoulos, "Power techniques for blind channel estimation in wireless communications," Ph.D. dissertation, University of Rennes 1, Rennes, France, Oct. 2004.
- [13] C.E. Davila, "Efficient, high performance, subspace tracking for time-domain data," *IEEE Tr. Sig. Proc.*, vol. 48, no. 12, pp. 3307-3315, Dec. 2000.

Fast Adaptive Blind MMSE Equalizer for Multichannel FIR Systems

Ibrahim Kacha,^{1,2} Karim Abed-Meraim,² and Adel Belouchrani¹

¹Département d'Électronique, École Nationale Polytechnique (ENP), 10 avenue Hassen Badi El-Harrach, 16200 Algiers, Algeria

²Département Traitement du Signal et de l'Image, École Nationale Supérieure des Télécommunications (ENST), 37-39 rue Dareau, 75014 Paris, France

Received 30 December 2005; Revised 14 June 2006; Accepted 22 June 2006

We propose a new blind minimum mean square error (MMSE) equalization algorithm of noisy multichannel finite impulse response (FIR) systems, that relies only on second-order statistics. The proposed algorithm offers two important advantages: a low computational complexity and a relative robustness against channel order overestimation errors. Exploiting the fact that the columns of the equalizer matrix filter belong both to the signal subspace and to the kernel of truncated data covariance matrix, the proposed algorithm achieves blindly a direct estimation of the zero-delay MMSE equalizer parameters. We develop a two-step procedure to further improve the performance gain and control the equalization delay. An efficient fast adaptive implementation of our equalizer, based on the projection approximation and the shift invariance property of temporal data covariance matrix, is proposed for reducing the computational complexity from $O(n^3)$ to $O(qnd)$, where q is the number of emitted signals, n the data vector length, and d the dimension of the signal subspace. We then derive a statistical performance analysis to compare the equalization performance with that of the optimal MMSE equalizer. Finally, simulation results are provided to illustrate the effectiveness of the proposed blind equalization algorithm.

Copyright © 2006 Ibrahim Kacha et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

1.1. Blind equalization

An elementary problem in the area of digital communications is that of intersymbol interference (ISI). ISI results from linear amplitude and phase dispersion in the transmission channel, mainly due to multipath propagation. To achieve reliable communications, channel equalization is necessary to deal with ISI.

Conventional nonblind equalization algorithms require training sequence or a priori knowledge of the channel [1]. In the case of wireless communications these solutions are often inappropriate, since a training sequence is usually sent periodically, thus the effective channel throughput is considerably reduced. It follows that the blind and semiblind equalization of transmission channels represent a suitable alternative to traditional equalization, because they do not fully rely on training sequence or a priori channel knowledge.

In the first contributions [2, 3], blind identification/equalization (BIE) schemes were based, implicitly or explicitly on higher- (than second-) order statistics of the observation. However, the shortcoming of these methods is the high error variances often exhibited by higher-order statistical esti-

mates. This often translates into slow convergence for on-line methods or unreasonable data length requirements for off-line methods. In the pioneering work of Tong et al. [4], it has been shown that the second-order statistics contain sufficient information for BIE of multichannel FIR systems. Later, active research in BIE area has led to a variety of second-order statistics-based algorithms (see the survey paper [5], as well as the references therein). Many efficient solutions (e.g., [6]) suffer from the lack of robustness against channel order overestimation errors and are also computationally expensive. A lot of research effort has been done to either develop efficient techniques for channel order estimation (e.g., [7, 8]) or to develop BIE methods robust to channel order estimation errors. Several *robust* techniques have been proposed so far [9–13], but all of them depend explicitly or implicitly on the channel order and hence have only a limited robustness, in the sense that their performance degrades significantly when the channel overestimation error is large.

1.2. Contributions

In this work, we develop a blind adaptive equalization algorithm based on MMSE estimation, which presents a number of nice properties such as robustness to channel order

overestimation errors and low computational complexity. More precisely, this paper describes a new technique for direct design of MIMO blind adaptive MMSE equalizer, having $O(qnd)$ complexity and relative robustness against channel order overestimation errors. We show that the columns of the zero-delay equalizer matrix filter belongs simultaneously to the signal subspace and to the kernel of truncated data covariance matrix. This property leads to a simple estimation method of the equalizer filter by minimizing a certain quadratic form subject to a properly chosen constraint. We present an efficient fast adaptive implementation of the novel algorithm, including a two-step estimation procedure, which allows us to compensate for the performance loss of the equalizer, compared to the nonblind one, and to choose a nonzero equalization delay. Also, we derive the asymptotic performance analysis of our method which leads to a closed form expression of the performance loss (compared to the optimal one) due to the considered blind processing.

The rest of the paper is organized as follows. In Section 2 the system model and problem statement are developed. Batch and adaptive implementations of the algorithm, using respectively, linear and quadratic constraints are introduced in Sections 3 and 4. Section 5 is devoted to the asymptotic performance analysis of the proposed blind MMSE filter. Simulation examples and performances evaluation are provided in Section 6. Finally, conclusions are drawn in Section 7.

1.3. Notations

Most notations are standard: vectors and matrices are represented by boldface small and capital letters, respectively. The matrix transpose, the complex conjugate, the hermitian, and the Moore-Penrose pseudo-inverse are denoted by $(\cdot)^T$, $(\cdot)^*$, $(\cdot)^H$, and $(\cdot)^\#$, respectively. \mathbf{I}_n is the $n \times n$ identity matrix and $\mathbf{0}$ (resp., $\mathbf{0}_{i \times k}$) denotes the zero matrix of appropriate dimension (resp., the zero matrix of dimension $i \times k$). The symbol \otimes stands for the Kronecker product; $\text{vec}(\cdot)$ and $\text{vec}^{-1}(\cdot)$ denote the column vectorization operator and its inverse, respectively. $E(\cdot)$ is the mathematical expectation. Also, we use some informal MATLAB notations, such as $\mathbf{A}(k, :)$, $\mathbf{A}(:, k)$, $\mathbf{A}(i, k)$, \dots , for the k th row, the k th column, the (i, k) th entry of matrix \mathbf{A} , respectively.

2. DATA MODEL

Consider a discrete time MIMO system of q inputs, p outputs ($p > q$) given by

$$\mathbf{x}(t) = \sum_{k=0}^L \mathbf{H}(k)\mathbf{s}(t-k) + \mathbf{b}(t), \quad (1)$$

where $H(z) = \sum_{k=0}^L \mathbf{H}(k)z^{-k}$ is an unknown causal FIR $p \times q$ transfer function. We assume (A1) $H(z)$ is irreducible and column reduced, that is, $\text{rank}(H(z)) = q$, for all z and $\mathbf{H}(L)$ is full column rank. (A2) The input (nonobservable) signal $\mathbf{s}(t)$ is a q -dimensional random vector assumed to be an iid (independently and identically distributed) zero-mean unit power

complex circular process [14], with finite fourth-order moments, that is, $E(\mathbf{s}(t+\tau)\mathbf{s}^H(t)) = \delta(\tau)\mathbf{I}_q$, $E(\mathbf{s}(t+\tau)\mathbf{s}^T(t)) = \mathbf{0}$, $E(|s_i(t)|^4) < \infty$, $i = 1, \dots, q$. (A3) $\mathbf{b}(t)$ is an additive spatially and temporally white Gaussian noise of power $\sigma_b^2\mathbf{I}_p$ and independent of the transmitted sequence $\{\mathbf{s}(t)\}$.¹

By stacking N successive samples of the received signal $\mathbf{x}(t)$ into a single vector, we obtain the n -dimensional ($n = Np$) vector

$$\begin{aligned} \mathbf{x}_N(t) &= [\mathbf{x}^T(t) \ \mathbf{x}^T(t-1) \ \dots \ \mathbf{x}^T(t-N+1)]^T \\ &= \mathbf{H}_N \mathbf{s}_m(t) + \mathbf{b}_N(t), \end{aligned} \quad (2)$$

where $\mathbf{s}_m(t) = [\mathbf{s}^T(t) \ \dots \ \mathbf{s}^T(t-m+1)]^T$, $\mathbf{b}_N(t) = [\mathbf{b}^T(t) \ \dots \ \mathbf{b}^T(t-N+1)]^T$, $m = N+L$ and \mathbf{H}_N is the channel convolution matrix of dimension $n \times d$, ($d = qm$), given by

$$\mathbf{H}_N = \begin{bmatrix} \mathbf{H}(0) & \dots & \mathbf{H}(L) & \mathbf{0} \\ & \ddots & & \ddots \\ \mathbf{0} & & \mathbf{H}(0) & \dots & \mathbf{H}(L) \end{bmatrix}. \quad (3)$$

It is shown in [15] that if N is large enough and under assumption (A1), matrix \mathbf{H}_N is full column rank.

3. ALGORITHM DERIVATION

3.1. MMSE equalizer

Consider a τ -delay MMSE equalizer ($\tau \in \{0, 1, \dots, m-1\}$). Under the above data model, one can easily show that the equalizer matrix \mathbf{V}_τ corresponding to the desired solution is given by

$$\mathbf{V}_\tau = \arg \min_{\mathbf{V}} E(\|\mathbf{s}(t-\tau) - \mathbf{V}^H \mathbf{x}_N(t)\|^2) = \mathbf{C}^{-1} \mathbf{G}_\tau, \quad (4)$$

where

$$\mathbf{C} \stackrel{\text{def}}{=} E(\mathbf{x}_N(t)\mathbf{x}_N^H(t)) = \mathbf{H}_N \mathbf{H}_N^H + \sigma_b^2 \mathbf{I}_n \quad (5)$$

is the data covariance matrix and \mathbf{G}_τ is an $n \times q$ matrix given by

$$\mathbf{G}_\tau \stackrel{\text{def}}{=} E(\mathbf{x}_N(t)\mathbf{s}^H(t-\tau)) = \mathbf{H}_N \mathbf{J}_{q\tau, q, q(m-\tau-1)}, \quad (6)$$

$\mathbf{J}_{j,k,l}$ is a truncation matrix defined as follow:

$$\mathbf{J}_{j,k,l} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{0}_{j \times k} \\ \mathbf{I}_k \\ \mathbf{0}_{l \times k} \end{bmatrix}. \quad (7)$$

Note that $\mathbf{H}_N \mathbf{J}_{q\tau, q, q(m-\tau-1)}$ denotes the submatrix of \mathbf{H}_N given by the column vectors of indices varying in the range $[\tau q +$

¹ Note that the column reduced condition in assumption (A1) can be relaxed, but that would lead to more complex notations. Similarly, the circularity and the finite value of the fourth-order moments of the input signal in assumption (A2) and the Gaussianity of additive noise in assumption (A3) are not necessary for the derivation of our algorithm, but used only for the asymptotic performance analysis.

$1, \dots, (\tau+1)q]$. From (4), (5), (6) and using matrix inversion lemma, matrix \mathbf{V}_τ is also expressed as $\mathbf{V}_\tau = \mathbf{H}_N \bar{\mathbf{V}}_\tau$, where $\bar{\mathbf{V}}_\tau$ is a $d \times q$ -dimensional matrix given by

$$\bar{\mathbf{V}}_\tau = \frac{1}{\sigma_b^2} \left(\mathbf{I}_d - \frac{1}{\sigma_b^2} (\sigma_b^2 \mathbf{I}_d + \mathbf{H}_N^H \mathbf{H}_N)^{-1} \mathbf{H}_N^H \mathbf{H}_N \right) \mathbf{J}_{q\tau, q, q(m-\tau-1)}. \quad (8)$$

Clearly, the columns of MMSE matrix filter \mathbf{V}_τ belong to the *signal subspace* (i.e., $\text{range}(\mathbf{H}_N)$) and thus one can write

$$\mathbf{V}_\tau = \mathbf{W} \tilde{\mathbf{V}}_\tau, \quad (9)$$

where \mathbf{W} is an $n \times d$ matrix whose column vectors form an orthonormal basis of the signal subspace (there exist a nonsingular $d \times d$ matrix \mathbf{P} such that $\mathbf{W} = \mathbf{H}_N \mathbf{P}$) and $\tilde{\mathbf{V}}_\tau$ is a $d \times q$ -dimensional matrix.

3.2. Blind equalization

Our objective here is to derive a blind estimate of the zero-delay MMSE equalizer \mathbf{V}_0 . From (4), (6), (7), and (9), one can write $\mathbf{V}_0 = \mathbf{W} \tilde{\mathbf{V}}_0$, with

$$\mathbf{C} \mathbf{W} \tilde{\mathbf{V}}_0 = \begin{bmatrix} \mathbf{H}(0) \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}. \quad (10)$$

If we truncate the first p rows of system (10), we obtain

$$\mathbf{T} \tilde{\mathbf{V}}_0 = \mathbf{0}, \quad (11)$$

where \mathbf{T} is an $(n-p) \times d$ matrix given by

$$\mathbf{T} \stackrel{\text{def}}{=} \bar{\mathbf{C}} \mathbf{W}, \quad (12)$$

$$\bar{\mathbf{C}} = \mathbf{C}(p+1:n, :) = \mathbf{J}_{p, n-p, 0}^T \mathbf{C}. \quad (13)$$

Matrix $\bar{\mathbf{C}}$ is a submatrix of \mathbf{C} given by its $n-p$ rows. Equation (11) shows that the columns of $\tilde{\mathbf{V}}_0$ belong to the right null space of \mathbf{T} ($\text{null}_r(\mathbf{T}) = \{\mathbf{z} \in \mathbb{C}^d : \mathbf{T}\mathbf{z} = \mathbf{0}\}$). Reversely, we can establish that (11) characterizes uniquely the zero-delay MMSE equalizer. We have the following result.

Theorem 1. *Under the above data assumptions and for $N > qL + 1$ the solution of*

$$\mathbf{T} \tilde{\mathbf{V}} = \mathbf{0}, \quad (14)$$

subject to the constraint

$$\text{rank}(\tilde{\mathbf{V}}) = q, \quad (15)$$

is unique (up to a constant $q \times q$ nonsingular matrix) and corresponds to the desired MMSE equalizer, that is,

$$\tilde{\mathbf{V}} = \tilde{\mathbf{V}}_0 \mathbf{R}, \quad (16)$$

for a given constant $q \times q$ invertible matrix \mathbf{R} .

Proof. Let $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ denote the eigenvalues of \mathbf{C} . Since \mathbf{H}_N is full column rank, the *signal part* of the covariance matrix \mathbf{C} , that is, $\mathbf{H}_N \mathbf{H}_N^H$ has rank d , hence $\lambda_k > \sigma_b^2$, $k = 1, \dots, d$ and $\lambda_k = \sigma_b^2$, $k = d+1, \dots, n$. Denote the unit-norm eigenvectors associated with the eigenvalues $\lambda_1, \dots, \lambda_d$ by $\mathbf{u}_s(1), \dots, \mathbf{u}_s(d)$, and those corresponding to $\lambda_{d+1}, \dots, \lambda_n$ by $\mathbf{u}_b(1), \dots, \mathbf{u}_b(n-d)$. Also define $\mathbf{U}_s = [\mathbf{u}_s(1) \dots \mathbf{u}_s(d)]$ and $\mathbf{U}_b = [\mathbf{u}_b(1) \dots \mathbf{u}_b(n-d)]$. The covariance matrix is thus also expressed as $\mathbf{C} = \mathbf{U}_s \text{diag}(\lambda_1, \dots, \lambda_d) \mathbf{U}_s^H + \sigma_b^2 \mathbf{U}_b \mathbf{U}_b^H$. The columns of matrix \mathbf{U}_s span the signal subspace, that is, $\text{range}(\mathbf{H}_N \mathbf{H}_N^H) = \text{range}(\mathbf{H}_N)$, there exist a nonsingular $d \times d$ matrix \mathbf{P}' such that $\mathbf{U}_s = \mathbf{H}_N \mathbf{P}'$, while the columns of \mathbf{U}_b span its orthogonal complement, the *noise subspace*, that is, $\mathbf{U}_b^H \mathbf{U}_s = \mathbf{0}$. As \mathbf{W} is an orthonormal basis of the signal subspace, there exists nonsingular $d \times d$ matrices \mathbf{P} and \mathbf{P}'' such that $\mathbf{W} = \mathbf{H}_N \mathbf{P} = \mathbf{U}_s \mathbf{P}''$, hence $\mathbf{C} \mathbf{W} = (\mathbf{H}_N \mathbf{P}' \text{diag}(\lambda_1, \dots, \lambda_d) \mathbf{U}_s^H + \sigma_b^2 \mathbf{U}_b \mathbf{U}_b^H) \mathbf{U}_s \mathbf{P}'' = \mathbf{H}_N \mathbf{S}$, where $\mathbf{S} = \mathbf{P}' \text{diag}(\lambda_1, \dots, \lambda_d) \mathbf{P}''$ is nonsingular. Consequently, $\mathbf{T} = \mathbf{C}(p+1:n, :) \mathbf{W} = \mathbf{H}_N(p+1:n, :) \mathbf{S}$. Since \mathbf{H}_N is block-Toeplitz matrix (see equation (3)), $\mathbf{H}_N(p+1:n, :) = [\mathbf{0}_{(n-p) \times q} \ \mathbf{H}_{N-1}]$. As \mathbf{H}_{N-1} is full column rank, it implies that $\dim(\text{null}_r(\mathbf{T})) = \dim(\text{null}_r([\mathbf{0}_{(n-p) \times q} \ \mathbf{H}_{N-1}])) = q$. It follows that any full column rank $d \times q$ matrix $\tilde{\mathbf{V}}$, solution of (14), can be considered as a basis of the right null space of matrix \mathbf{T} . According to (11) the columns of matrix $\tilde{\mathbf{V}}_0$, which characterize the MMSE filter given by (10), belong to $\text{null}_r(\mathbf{T})$ and are linearly independent, it follows that $\tilde{\mathbf{V}} = \tilde{\mathbf{V}}_0 \mathbf{R}$, where \mathbf{R} is a nonsingular $q \times q$ matrix. \square

3.3. Implementation

3.3.1. The SIMO case

In the SIMO case ($q = 1$) matrix $\tilde{\mathbf{V}}$ is replaced by the d -dimensional vector $\tilde{\mathbf{v}}$ and (14) can be solved, simply, in the least squares sense subject to the unit norm constraint:

$$\tilde{\mathbf{v}} = \arg \min_{\|\mathbf{z}\|=1} (\mathbf{z}^H \mathbf{Q} \mathbf{z}), \quad (17)$$

where \mathbf{Q} is a $(d \times d)$ matrix defined by

$$\mathbf{Q} \stackrel{\text{def}}{=} \mathbf{T}^H \mathbf{T}. \quad (18)$$

Then, according to (9) and (16), we obtain the MMSE equalizer vector $\mathbf{v}_0 = r \mathbf{v}$, where r is a given nonzero scalar and \mathbf{v} is the n -dimensional vector given by

$$\mathbf{v} = \mathbf{W} \tilde{\mathbf{v}}. \quad (19)$$

A batch-processing implementation of the SIMO blind MMSE equalization algorithm is summarized in Algorithm 1.

3.3.2. The MIMO case

In this situation, the quadratic constraint on $\tilde{\mathbf{V}}$ does not guarantee condition (15) in Theorem 1. One possible solution is to choose a linear constraint (instead of the quadratic one)

$$\begin{aligned}
\mathbf{C} &= \frac{1}{K} \sum_{t=0}^{K-1} \mathbf{x}_N(t) \mathbf{x}_N^H(t), \quad (K: \text{sample size}) \\
(\mathbf{W}, \Lambda_1) &= \text{eigs}(\mathbf{C}, d), \quad (\text{extracts the } d \text{ principal eigenvectors of } \mathbf{C}) \\
\mathbf{T} &= \mathbf{C}(p+1 : n, :)\mathbf{W} \\
\mathbf{Q} &= \mathbf{T}^H \mathbf{T} \\
\tilde{\mathbf{v}} &= \text{the least eigenvector of } \mathbf{Q} \\
\mathbf{v} &= \mathbf{W}\tilde{\mathbf{v}}
\end{aligned}$$

ALGORITHM 1: SIMO blind MMSE equalization algorithm.

such as the $q \times q$ first block of matrix $\tilde{\mathbf{V}}$ is lower triangular

$$\tilde{\mathbf{V}}(1 : q, 1 : q) = \begin{bmatrix} 1 & \cdots & 0 \\ \times & \ddots & \vdots \\ \times & \times & 1 \end{bmatrix}, \quad (20)$$

which will guarantee that matrix $\tilde{\mathbf{V}}$ has a full column rank q .

It is clear that (14) is equivalent to (see [16] for more details)

$$(\mathbf{I}_q \otimes \mathbf{T}) \text{vec}(\tilde{\mathbf{V}}) = \mathbf{0}. \quad (21)$$

Taking into account the lower triangular constraint in (20), (21) becomes

$$\mathbf{a} + \mathbf{A}\bar{\mathbf{v}} = \mathbf{0}, \quad (22)$$

where

$$\begin{aligned}
\bar{\mathbf{v}} &= \mathbf{J}^T \text{vec}(\tilde{\mathbf{V}}), \\
\mathbf{a} &= \text{vec}(\mathbf{T}\mathbf{J}_{0,q,d-q}), \\
\mathbf{A} &= (\mathbf{I}_q \otimes \mathbf{T})\mathbf{J}, \\
\mathbf{J} &= \text{diag}(\mathbf{J}_1, \mathbf{J}_2, \dots, \mathbf{J}_q), \\
\mathbf{J}_k &= \mathbf{J}_{k,d-k,0}, \quad k = 1, \dots, q.
\end{aligned} \quad (23)$$

The solution of (22) is given by

$$\bar{\mathbf{v}} = -\mathbf{A}^\# \mathbf{a}. \quad (24)$$

Matrix $\tilde{\mathbf{V}}$, solution of (14), is then given by $\tilde{\mathbf{V}} = \text{vec}^{-1}(\tilde{\mathbf{v}})$ where $\tilde{\mathbf{v}}$ is obtained from $\bar{\mathbf{v}}$ by adding ones and zeros at the appropriate entries according to

$$\tilde{\mathbf{v}} = \mathbf{J}\bar{\mathbf{v}} + \text{vec}(\mathbf{J}_{0,q,d-q}). \quad (25)$$

From (9) and (16), we obtain the MMSE equalizer matrix $\mathbf{V}_0 = \mathbf{V}\mathbf{R}^{-1}$, where \mathbf{R} is a constant invertible $q \times q$ matrix and \mathbf{V} is an $(n \times q)$ matrix given by

$$\mathbf{V} = \mathbf{W}\tilde{\mathbf{v}}. \quad (26)$$

Thus, we obtain a block-processing implementation of the MIMO blind MMSE equalization algorithm that is summarized in Algorithm 2. Note that the $q \times q$ constant matrix

\mathbf{R} comes from the inherent indeterminacies of MIMO blind identification systems using second-order statistics [15]. Usually, this indeterminacy is solved by applying some blind source separation algorithms.

3.4. Selection of the equalizer delay

It is known that the choice of the equalizer delay may affect significantly the equalization performance in SIMO and MIMO systems. In particular, nonzero-delay equalizers can have much improved performance compared to the zero-delay ones [10]. Indeed, one can write the spatiotemporal vector in (2) as follows:

$$\mathbf{x}_N(t) = \sum_{k=0}^{m-1} \mathbf{G}_k \mathbf{s}(t-k) + \mathbf{b}_N(t), \quad (27)$$

where \mathbf{G}_k is defined in (6) and represents a submatrix of \mathbf{H}_N given by the column vectors of indices varying in the range $[kq+1, \dots, (k+1)q]$. One can observe that $\|\mathbf{G}_0\| \leq \|\mathbf{G}_1\| \leq \dots \leq \|\mathbf{G}_L\| = \|\mathbf{G}_{L+1}\| = \dots = \|\mathbf{G}_{N-1}\|$ and $\|\mathbf{G}_{N-1}\| \geq \|\mathbf{G}_N\| \geq \dots \geq \|\mathbf{G}_{d-1}\|$. In other words, the input symbols with delays τ , $L \leq \tau \leq N-1$ are multiplied in (27) by (matrix) factors of maximum norm. Consequently, the best equalizer delay belongs, in general, to the range $[L, \dots, N-1]$. One can observe also that, the performance gain of the nonzero equalizer with delay in the range $[L, \dots, N-1]$ can be large compared to that of equalizers with extreme delays, that is, $\tau = 0$ or $\tau = d-1$. The gain difference becomes, in general, negligible when we consider two equalizers with delays belonging to the interval $[L, \dots, N-1]$ (see [10]). Hence, in practice, the search for the optimal equalizer delay is computationally expensive and worthless and it is often sufficient to choose a *good* delay in the range $[L, \dots, N-1]$, for example, $\tau = L$ as we did in this paper.

Moreover, it is shown in Section 5 that the blind estimation of the MMSE filter results in a performance loss compared to the nonblind one. To compensate for this performance loss and also to have a controlled nonzero equalization delay which helps to improve performance of the equalizer, we propose here a two-step approach to estimate the blind MMSE equalizer. In the first step, we estimate \mathbf{V}_0 according to the previous algorithms, while, in the second step, we refine this estimation by exploiting the a priori knowledge of the finite alphabet to which belongs the symbols $\mathbf{s}(t)$. This

$$\begin{aligned}
\mathbf{C} &= \frac{1}{K} \sum_{t=0}^{K-1} \mathbf{x}_N(t) \mathbf{x}_N^H(t), \quad (K: \text{sample size}) \\
(\mathbf{W}, \Lambda) &= \text{eigs}(\mathbf{C}, d), \quad (\text{extracts the } d \text{ principal eigenvectors of } \mathbf{C}) \\
\mathbf{T} &= \mathbf{C}(p+1:n, :)\mathbf{W} \\
\mathbf{a} &= \text{vec}(\mathbf{T}(:, 1:q)) \\
\mathbf{A} &= (\mathbf{I}_q \otimes \mathbf{T})\mathbf{J} \\
\bar{\mathbf{v}} &= -\mathbf{A}^{\#}\mathbf{a} \\
\tilde{\mathbf{V}} &= \text{vec}^{-1}(\mathbf{J}\bar{\mathbf{v}}) + \mathbf{J}_{0,q,d-q} \\
\mathbf{V} &= \mathbf{W}\tilde{\mathbf{V}}
\end{aligned}$$

ALGORITHM 2: MIMO blind MMSE equalization algorithm.

Estimate $\hat{\mathbf{s}}(t)$, $t = 0 \dots K-1$, using \mathbf{V} given by Algorithm 1 or Algorithm 2 followed by BSS (e.g., ACMA in [17]).

$$\begin{aligned}
\mathbf{G}_\tau &= \frac{1}{K} \sum_{t=\tau}^{K+\tau-1} \mathbf{x}_N(t) \hat{\mathbf{s}}^H(t-\tau) \\
\mathbf{V}_\tau &= \mathbf{C}^{-1} \mathbf{G}_\tau
\end{aligned}$$

ALGORITHM 3: Two-step equalization procedure.

is done by performing a hard decision on the symbols that are then used to reestimate \mathbf{V}_τ according to (4) and (6).²

More precisely, operating with equalizer filter \mathbf{V} in (26) (or in (19) for the SIMO case) on the received data vector $\mathbf{x}_N(t)$ in (2), we obtain, according to (9) and (16), an estimation of the emitted signal $\tilde{\mathbf{s}}(t) = \mathbf{V}^H \mathbf{x}_N(t) = \mathbf{R}^H \mathbf{V}_0^H \mathbf{x}_N(t)$, as $\mathbf{V}_0^H \mathbf{x}_N(t) = \mathbf{s}(t) + \boldsymbol{\epsilon}(t)$, where $\boldsymbol{\epsilon}(t)$ represents the residual estimation error (of minimum variance) of $\mathbf{s}(t)$, it follows that

$$\tilde{\mathbf{s}}(t) = \mathbf{R}^H \mathbf{s}(t) + \tilde{\boldsymbol{\epsilon}}(t), \quad (28)$$

where $\tilde{\boldsymbol{\epsilon}}(t) = \mathbf{R}^H \boldsymbol{\epsilon}(t)$. It is clear from (28), that the estimated signal $\tilde{\mathbf{s}}(t)$ is an instantaneous mixture of the emitted signal $\mathbf{s}(t)$ corrupted by an additive colored noise $\tilde{\boldsymbol{\epsilon}}(t)$. Thus, an identification of \mathbf{R} (i.e., resolving the ambiguity) is then necessary to extract the original signal and to decrease the mean square error (MSE) towards zero. This is achieved by applying (in batch or adaptive way) a blind source separation (BSS) algorithm to the equalizer output (28), followed by a hard decision on the symbols. In this paper, we have used the ACMA algorithm (analytical constant modulus algorithm) in [17] for batch processing implementation and the A-CMS algorithm (adaptive constant modulus separation) in [18] for adaptive implementation. Indeed, constant modulus algorithms (CMA)-like algorithms (ACMA and A-CMS) have relatively low cost and are very efficient in separating (finite alphabet) communication signals. The two-step

blind MMSE equalization algorithms are summarized in Algorithms 1, 2, and 3.

3.5. Robustness

We study here the robustness of the proposed blind MMSE equalizer against channel order overestimation errors. Let us consider, for simplicity, the SIMO case where the channel order is used to determine the column dimension equal to $d = L + N$ of matrix \mathbf{W} (which corresponds, in practice, to the size of the dominant subspace of \mathbf{C}). Let $L' > L$ be the over-estimated channel order and hence $d' = L' + N$ is the column dimension of \mathbf{W} , that is, we consider the subspace spanned by the d' dominant eigenvector of \mathbf{C} . We argue here that, as long as the number of sensors p plus the overestimation error order $L' - L$ is smaller than the noise subspace dimension, that is, $p + L' - L < n - d$, the least squares solution of (14) provides a consistent estimate of the MMSE equalizer. This observation comes from the following.

Note that, using (5), matrix $\bar{\mathbf{C}}$ defined in (13) is expressed as $\bar{\mathbf{C}} = [\mathbf{H}' \ \mathbf{C}']$, where \mathbf{H}' is an $(n-p) \times p$ -dimensional matrix and $\mathbf{C}' = \mathbf{H}_{N-1} \mathbf{H}_{N-1}^H + \sigma_b^2 \mathbf{I}_{n-p}$ an $(n-p) \times (n-p)$ full-rank matrix. It follows that the right null space of $\bar{\mathbf{C}}$, $\text{null}_r(\bar{\mathbf{C}}) = \{\mathbf{z} \in \mathbb{C}^n : \bar{\mathbf{C}}\mathbf{z} = \mathbf{0}\}$, is a p -dimensional subspace. Now, one can observe that only one direction of $\text{null}_r(\bar{\mathbf{C}})$ belongs to the signal subspace since $\text{null}_r(\bar{\mathbf{C}}) \cap \text{range}(\mathbf{H}_N) = \text{null}_r(\bar{\mathbf{C}}\mathbf{H}_N) = \text{null}_r(\bar{\mathbf{C}}\mathbf{W})$ (the last equality comes from the fact that \mathbf{H}_N and \mathbf{W} span both the same (signal) subspace). According to the proof of Theorem 1, $\dim(\text{null}_r(\bar{\mathbf{C}}\mathbf{W})) = 1$.

Let $\mathbf{b}_1, \dots, \mathbf{b}_p$ be a basis of $\text{null}_r(\bar{\mathbf{C}})$ such that \mathbf{b}_1 belongs to the signal subspace (i.e., $\text{range}(\mathbf{H}_N)$). Now, the solution of

² We assume here the use of a differential modulation to get rid of the phase indeterminacy inherent to the blind equalization problem.

(14) would be unique (up to a scalar constant) if

$$\text{range}(\mathbf{W}) \cap \text{range}\left(\begin{bmatrix} \mathbf{b}_1 & \cdots & \mathbf{b}_p \end{bmatrix}\right) = \text{range}(\mathbf{b}_1), \quad (29)$$

or equivalently

$$\text{range}(\mathbf{W}) \cap \text{range}\left(\begin{bmatrix} \mathbf{b}_2 & \cdots & \mathbf{b}_p \end{bmatrix}\right) = \{\mathbf{0}\}. \quad (30)$$

The above condition would be verified if the intersection of the subspace spanned by the projections of $\mathbf{b}_2, \dots, \mathbf{b}_p$ onto the noise subspace and the subspace spanned by the $L' - L$ noise vectors of \mathbf{W} introduced by the overestimation error is empty (except for the zero vector). As the latter are randomly introduced by the eigenvalue decomposition (EVD) of \mathbf{C} and since $p + L' - L < n - d$, then one can expect this subspace intersection to be empty almost surely.

Note also that, by using linear constraint, one obtains better robustness than with quadratic constraint. The reason is that the solution of (14) is, in general, a linear combination of the desired solution \mathbf{v}_0 (that lives in the signal subspace) and noise subspace vectors (introduced by the channel order overestimation errors). However, it is observed that, for a finite sample size and for moderate and high SNRs the contribution of the desired solution \mathbf{v}_0 in (14) is much higher than that of the noise subspace vectors. This is due to the fact that the low energy output of the noise subspace vectors comes from their orthogonality with the system matrix \mathbf{H}_N (this is a structural property, independent of the sample size), while the desired solution \mathbf{v}_0 belongs to the kernel of $\bar{\mathbf{C}}$ due to the decorrelation (whiteness) property of the input signal which is valid asymptotically for large sample size. Indeed, one can observe (see Figure 6) that when increasing K (the sample size), the robustness of the quadratically constrained equalizer improves significantly. Consequently, in the context of small or moderate sample sizes, solving (14) in the least squares sense under unit norm constraint leads to a solution that lives almost in the noise subspace (i.e., the part of \mathbf{v}_0 in the final solution becomes very small). On the other hand, by solving (14) subject to linear constraints (24) and (25), one obtains a solution where the linear factor of \mathbf{v}_0 is more significant (which is due to the fact that vector \mathbf{a} in (24) belongs to the range subspace of \mathbf{A}).

This argument, eventhough not a rigorous proof of robustness, has been confirmed by our simulation results (see simulation example given below where one can see that the performance loss of the equalization due to the channel order overestimation error remains relatively limited).

4. FAST ADAPTIVE IMPLEMENTATION

In tracking applications, we are interested in estimating the equalizer vector recursively with low computational complexity. We introduce here a fast adaptive implementation of the proposed blind MMSE equalization algorithms. The computational reduction is achieved by exploiting the idea of the projection approximation [19] and the shift-invariance property of the temporal data covariance matrices [20].

Matrix \mathbf{C} is replaced by its recursive estimate

$$\mathbf{C}(t) = \sum_{k=0}^t \beta^{t-k} \mathbf{x}_N(k) \mathbf{x}_N^H(k) = \beta \mathbf{C}(t-1) + \mathbf{x}_N(t) \mathbf{x}_N^H(t), \quad (31)$$

where $0 < \beta < 1$ is a forgetting factor. The weight matrix \mathbf{W} corresponding to the d dominant eigenvectors of \mathbf{C} can be estimated using a fast subspace estimation and tracking algorithm. In this paper, we use the YAST algorithm (yet another subspace tracker) [21]. The choice of YAST algorithm is motivated by its remarkable tracking performance compared to other existing subspace tracking algorithms of similar computational complexity (PAST [19], OPAST [22], etc.). The YAST algorithm is summarized in Algorithm 4. Note that only $O(nd)$ operations are required at each time instant (instead of $O(n^3)$ for a full EVD). Vector $\mathbf{x}'(t) = \mathbf{C}(t-1) \mathbf{x}_N(t)$ in Algorithm 4 can be computed in $O(n)$ operations, by using the shift-invariance property of the correlation matrix, as seen in Appendix A.

Applying, to (12), the projection approximation

$$\mathbf{C}(t) \mathbf{W}(t) \approx \mathbf{C}(t) \mathbf{W}(t-1), \quad (32)$$

which is valid if matrix $\mathbf{W}(t)$ is slowly varying with time [22], yields

$$\mathbf{T}(t) = \beta \mathbf{T}(t-1) + \mathbf{J}_{p,n-p,0}^T \mathbf{x}_N(t) \mathbf{y}^H(t), \quad (33)$$

where vector $\mathbf{J}_{p,n-p,0}^T \mathbf{x}_N(t)$ is a subvector of $\mathbf{x}_N(t)$ given by its last $(n-p)$ elements and vector $\mathbf{y}(t) = \mathbf{W}^H(t-1) \mathbf{x}_N(t)$ is computed by YAST (cf. Algorithm 4).

4.1. The SIMO case

In this case, our objective is to estimate recursively the d -dimensional vector $\tilde{\mathbf{v}}$ in (17) as the least eigenvector of matrix \mathbf{Q} or equivalently as the dominant eigenvector of its inverse.³ Using (18), (33) can be replaced by the following recursion:

$$\mathbf{Q}(t) = \beta^2 \mathbf{Q}(t-1) - \mathbf{D}_Q(t) \mathbf{\Gamma}_Q^{-1}(t) \mathbf{D}_Q^H(t), \quad (34)$$

where $\mathbf{D}_Q(t)$ is the $d \times 2$ matrix

$$\mathbf{D}_Q(t) = \begin{bmatrix} \beta \mathbf{T}^H(t-1) \mathbf{J}_{p,n-p,0}^T \mathbf{x}_N(t) & \mathbf{y}(t) \end{bmatrix}, \quad (35)$$

and $\mathbf{\Gamma}_Q(t)$ is the 2×2 nonsingular matrix

$$\mathbf{\Gamma}_Q(t) = \begin{bmatrix} \|\mathbf{J}_{p,n-p,0}^T \mathbf{x}_N(t)\|^2 & -1 \\ -1 & 0 \end{bmatrix}. \quad (36)$$

Consider the $d \times d$ Hermitian matrix $\mathbf{F}(t) \stackrel{\text{def}}{=} \mathbf{Q}^{-1}(t)$, using the matrix (Schur) inversion lemma [1], we obtain

$$\mathbf{F}(t) = \frac{1}{\beta^2} \mathbf{F}(t-1) + \mathbf{D}_F(t) \mathbf{\Gamma}_F(t) \mathbf{D}_F^H(t), \quad (37)$$

³ \mathbf{Q} is a singular matrix when dealing with the exact statistics. However, when considering the sample averaged estimate of \mathbf{C} , due to the estimation errors and the projection approximation, the estimate of \mathbf{Q} is almost surely a nonsingular matrix.

$$\begin{aligned}
\mathbf{y}(t) &= \mathbf{W}^H(t-1)\mathbf{x}_N(t) \\
\mathbf{x}'(t) &= \mathbf{C}(t-1)\mathbf{x}_N(t) \\
\mathbf{y}'(t) &= \mathbf{W}^H(t-1)\mathbf{x}'(t) \\
\sigma(t) &= (\mathbf{x}_N^H(t)\mathbf{x}_N(t) - \mathbf{y}^H(t)\mathbf{y}(t))^{1/2} \\
\mathbf{h}(t) &= \mathbf{Z}(t-1)\mathbf{y}(t) \\
\gamma(t) &= (\beta + \mathbf{y}^H(t)\mathbf{h}(t))^{-1} \\
\tilde{\mathbf{Z}}(t) &= \frac{1}{\beta}(\mathbf{Z}(t-1) - \mathbf{h}(t)\gamma(t)\mathbf{h}^H(t)) \\
\alpha(t) &= \mathbf{x}_N^H(t)\mathbf{x}_N(t) \\
\mathbf{y}''(t) &= \beta\mathbf{y}'(t) + \mathbf{y}(t)\alpha(t) \\
c_{yy}(t) &= \beta\mathbf{x}_N^H(t)\mathbf{x}'(t) + \alpha^*(t)\alpha(t) \\
\mathbf{h}'(t) &= \tilde{\mathbf{Z}}(t-1)\mathbf{y}''(t) \\
\gamma'(t) &= (c_{yy}(t) - [\mathbf{y}''(t)]^H\mathbf{h}'(t))^{-1} \\
\mathbf{h}''(t) &= \mathbf{h}'(t) - \mathbf{y}(t) \\
\tilde{\mathbf{Z}}'(t) &= \tilde{\mathbf{Z}}(t) + \mathbf{h}''(t)\gamma'(t)[\mathbf{h}''(t)]^H \\
\mathbf{g}(t) &= \mathbf{h}''(t)\gamma'(t)\sigma^*(t) \\
\mathbf{y}'''(t) &= \sigma(t)\gamma'(t)\sigma^*(t) \\
\mathbf{Z}'(t) &= [\tilde{\mathbf{Z}}'(t), -\mathbf{g}(t); -\mathbf{g}^H(t), \mathbf{y}'''(t)] \\
(\phi(t), \lambda(t)) &= \text{eigs}(\mathbf{Z}'(t), 1) \\
\boldsymbol{\varphi}(t) &= \boldsymbol{\phi}_{(1;d)}(t) \\
z(t) &= \boldsymbol{\phi}_{(d+1)}(t) \\
\rho(t) &= |z(t)| \\
\theta(t) &= e^{j\arg(z(t))}, \text{ (arg stands for the phase argument)} \\
\mathbf{f}(t) &= \boldsymbol{\varphi}(t)\theta^*(t) \\
\mathbf{f}'(t) &= \mathbf{f}(t)(1 + \rho(t))^{-1} \\
\mathbf{y}''''(t) &= \mathbf{y}(t)\sigma^{-1}(t) - \mathbf{f}'(t) \\
\mathbf{e}(t) &= \mathbf{x}(t)\sigma^{-1}(t) - \mathbf{W}(t-1)\mathbf{y}''''(t) \\
\mathbf{W}(t) &= \mathbf{W}(t-1) - \mathbf{e}(t)\mathbf{f}^H(t) \\
\mathbf{g}'(t) &= \mathbf{g}(t) + \mathbf{f}'(t)(\mathbf{y}''(t) - \theta(t)\lambda(t)\theta^*(t)) \\
\mathbf{Z}(t) &= \tilde{\mathbf{Z}}'(t) + \mathbf{g}'(t)[\mathbf{f}'(t)]^H + \mathbf{f}'(t)\mathbf{g}'^H(t)
\end{aligned}$$

ALGORITHM 4: YAST algorithm.

where $\mathbf{D}_F(t)$ is the $d \times 2$ matrix

$$\mathbf{D}_F(t) = \frac{1}{\beta^2}\mathbf{F}(t-1)\mathbf{D}_Q(t), \quad (38)$$

and $\Gamma_F(t)$ is the 2×2 matrix

$$\Gamma_F(t) = (\Gamma_Q(t) - \mathbf{D}_F^H(t)\mathbf{D}_Q(t))^{-1}. \quad (39)$$

The extraction of the dominant eigenvector of $\mathbf{F}(t)$ is obtained by power iteration as

$$\tilde{\mathbf{v}}(t) = \frac{\mathbf{F}(t)\tilde{\mathbf{v}}(t-1)}{\|\mathbf{F}(t)\tilde{\mathbf{v}}(t-1)\|}. \quad (40)$$

The complete pseudocode for the SIMO adaptive blind MMSE equalization algorithm is given in Algorithm 5. Note that the whole processing requires only $O(nd)$ flops per iteration.

Update $\mathbf{W}(t)$ and $\mathbf{y}(t)$ using YAST (cf. Algorithm 4)

$$\begin{aligned}
\bar{\mathbf{x}}(t) &= \mathbf{x}_N(t)_{(p+1:n)} \\
\Gamma_Q(t) &= \begin{bmatrix} \|\bar{\mathbf{x}}(t)\|^2 & -1 \\ -1 & 0 \end{bmatrix} \\
\mathbf{D}_Q(t) &= [\beta\mathbf{T}^H(t-1)\bar{\mathbf{x}}(t) \quad \mathbf{y}(t)] \\
\mathbf{D}_F(t) &= \frac{1}{\beta^2}\mathbf{F}(t-1)\mathbf{D}_Q(t) \\
\Gamma_F(t) &= (\Gamma_Q(t) - \mathbf{D}_F^H(t)\mathbf{D}_Q(t))^{-1} \\
\mathbf{F}(t) &= \frac{1}{\beta^2}\mathbf{F}(t-1) + \mathbf{D}_F(t)\Gamma_F(t)\mathbf{D}_F^H(t) \\
\tilde{\mathbf{v}}(t) &= \frac{\mathbf{F}(t)\tilde{\mathbf{v}}(t-1)}{\|\mathbf{F}(t)\tilde{\mathbf{v}}(t-1)\|} \\
\mathbf{v}(t) &= \mathbf{W}(t)\tilde{\mathbf{v}}(t) \\
\mathbf{T}(t) &= \beta\mathbf{T}(t-1) + \bar{\mathbf{x}}(t)\mathbf{y}^H(t)
\end{aligned}$$

ALGORITHM 5: SIMO adaptive blind equalization algorithm.

4.2. The MIMO case

Here, we introduce a fast adaptive version of the MIMO blind MMSE equalization algorithm given in Algorithm 2. First note that, due to the projection approximation and the finite sample size effect, matrix \mathbf{A} is almost surely full column rank and hence

$$\mathbf{A}^\# = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H. \quad (41)$$

Therefore vector $\bar{\mathbf{v}}$ in (24) can be expressed as

$$\bar{\mathbf{v}}(t) = [\bar{\mathbf{v}}_1^T(t) \quad \bar{\mathbf{v}}_2^T(t) \quad \cdots \quad \bar{\mathbf{v}}_q^T(t)]^T, \quad (42)$$

where vectors $\bar{\mathbf{v}}_k(t)$, for $k = 1, \dots, q$, are given by

$$\begin{aligned}
\bar{\mathbf{v}}_k(t) &= -\mathbf{F}_k(t)\mathbf{f}_k(t), \\
\mathbf{F}_k(t) &= (\mathbf{J}_k^T\mathbf{Q}(t)\mathbf{J}_k)^{-1}, \\
\mathbf{f}_k(t) &= \mathbf{J}_k^T\mathbf{Q}(t)\mathbf{J}_{k-1,1,d-k}.
\end{aligned} \quad (43)$$

Using (34) and the matrix (Schur) inversion lemma [1], matrix $\mathbf{F}_k(t)$ can be updated by the following recursion:

$$\begin{aligned}
\mathbf{F}_k(t) &= \frac{1}{\beta^2}\mathbf{F}_k(t-1) + \mathbf{D}_{F_k}(t)\Gamma_{F_k}(t)\mathbf{D}_{F_k}^H(t), \\
\mathbf{D}_{F_k}(t) &= \frac{1}{\beta^2}\mathbf{F}_k(t-1)\mathbf{J}_k^T\mathbf{D}_Q(t), \\
\Gamma_{F_k}(t) &= (\Gamma_Q(t) - \mathbf{D}_{F_k}^H(t)\mathbf{J}_k^T\mathbf{D}_Q(t))^{-1},
\end{aligned} \quad (44)$$

where matrices $\mathbf{D}_Q(t)$ and $\Gamma_Q(t)$ are given by (35) and (36).

Algorithm 6 summarizes the fast adaptive version of the MIMO blind MMSE equalization algorithm. Note that the whole processing requires only $O(qnd)$ flops per iteration.

4.3. Two-step procedure

Let $\mathbf{W} \in \mathbb{C}^{n \times d}$ be an orthonormal basis of the signal subspace. Since \mathbf{G}_T belongs to the signal subspace, one can write

Update $\mathbf{W}(t)$ and $\mathbf{y}(t)$ using YAST (cf. Algorithm 4)

$$\bar{\mathbf{x}}(t) = \mathbf{x}_N(t)_{(p+1:n)}$$

$$\mathbf{\Gamma}_Q(t) = \begin{bmatrix} \|\bar{\mathbf{x}}(t)\|^2 & -1 \\ -1 & 0 \end{bmatrix}$$

$$\mathbf{D}_Q(t) = \begin{bmatrix} \beta \mathbf{T}^H(t-1) \bar{\mathbf{x}}(t) & \mathbf{y}(t) \end{bmatrix}$$

$$\mathbf{Q}(t) = \beta^2 \mathbf{Q}(t-1) - \mathbf{D}_Q(t) \mathbf{\Gamma}_Q^{-1}(t) \mathbf{D}_Q^H(t)$$

For $k = 1, \dots, q$:

$$\mathbf{f}_k(t) = \mathbf{Q}(t)_{(k+1:d,k)}$$

$$\mathbf{D}_{F_k}(t) = \frac{1}{\beta^2} \mathbf{F}_k(t-1) \mathbf{D}_Q(t)_{(k+1:d,:)}$$

$$\mathbf{\Gamma}_{F_k}(t) = (\mathbf{\Gamma}_Q(t) - \mathbf{D}_{F_k}^H(t) \mathbf{D}_Q(t)_{(k+1:d,:)})^{-1}$$

$$\mathbf{F}_k(t) = \frac{1}{\beta^2} \mathbf{F}_k(t-1) + \mathbf{D}_{F_k}(t) \mathbf{\Gamma}_{F_k}(t) \mathbf{D}_{F_k}^H(t)$$

$$\bar{\mathbf{V}}_k(t) = -\mathbf{F}_k(t) \mathbf{f}_k(t)$$

end

$$\bar{\mathbf{V}}(t) = [\bar{\mathbf{V}}_1^T(t) \quad \bar{\mathbf{V}}_2^T(t) \quad \dots \quad \bar{\mathbf{V}}_q^T(t)]^T$$

$$\tilde{\mathbf{V}}(t) = \text{vec}^{-1}(\mathbf{J} \bar{\mathbf{V}}(t)) + \mathbf{J}_{0,q,d-q}$$

$$\mathbf{V}(t) = \mathbf{W}(t) \tilde{\mathbf{V}}(t)$$

$$\mathbf{T}(t) = \beta \mathbf{T}(t-1) + \bar{\mathbf{x}}(t) \mathbf{y}^H(t)$$

ALGORITHM 6: MIMO adaptive blind MMSE equalization algorithm.

(see [23])

$$\mathbf{V}_\tau = \mathbf{W}(\mathbf{W}^H \mathbf{C} \mathbf{W})^{-1} \mathbf{W}^H \mathbf{G}_\tau. \quad (45)$$

This expression of \mathbf{V}_τ is used for the fast adaptive implementation of the two-step algorithm since $\mathbf{Z} = (\mathbf{W}^H \mathbf{C} \mathbf{W})^{-1}$ is already computed by the YAST. The recursive expression of vector \mathbf{G}_τ is given by

$$\mathbf{G}_\tau(t) = \beta \mathbf{G}_\tau(t-1) + \mathbf{x}_N(t) \hat{\mathbf{s}}^H(t-\tau), \quad (46)$$

where $\hat{\mathbf{s}}(t)$ is an estimate of $\mathbf{s}(t)$ given by applying a BSS to $\tilde{\mathbf{s}}(t)$ in (28). In our simulation, we used the A-CMS algorithm in [18]. Thus, (45) can be replaced by the following recursion:

$$\begin{aligned} \mathbf{V}_\tau(t) &= \beta \mathbf{V}_\tau(t-1) + \mathbf{z}(t) \hat{\mathbf{s}}^H(t-\tau), \\ \mathbf{z}(t) &= \mathbf{W}(t) \mathbf{Z}(t) \mathbf{W}^H(t) \mathbf{x}_N(t). \end{aligned} \quad (47)$$

Note that, by choosing a nonzero equalizer delay τ , we improve the equalization performance as shown below. The adaptive two-step blind MMSE equalization algorithm is summarized in Algorithms 5, 6, and 7. The overall computational cost of this algorithm is $(q+8)nd + O(qn + qd^2)$ flops per iteration.

5. PERFORMANCE ANALYSIS

As mentioned above, the extraction of the equalizer matrix needs some blind source separation algorithms to solve the indeterminacy problem which is inherent to second-order

Estimate $\hat{\mathbf{s}}(t)$, using $\mathbf{V}(t)$ given by Algorithm 5 or Algorithm 6 followed by BSS (e.g., A-CMS in [18]).

$$\mathbf{z}(t) = \mathbf{W}(t) \mathbf{Z}(t) \mathbf{W}^H(t) \mathbf{x}_N(t)$$

$$\mathbf{V}_\tau(t) = \beta \mathbf{V}_\tau(t-1) + \mathbf{z}(t) \hat{\mathbf{s}}^H(t-\tau)$$

ALGORITHM 7: Adaptive two-step equalization procedure.

MIMO blind identification methods. Thus, the performance of our MIMO equalization algorithms depends, in part, on the choice of the blind source separation algorithm which leads to a very cumbersome asymptotic convergence analysis. For simplicity, we study the asymptotic expression of the estimated zero-delay blind equalization MSE in the SIMO case only, where, the equalizer vector is given up to an unknown nonzero scalar constant. To evaluate the performance of our algorithm, this constant is estimated according to

$$r = \arg \min_{\alpha} \|\mathbf{v}_0 - \alpha \mathbf{v}\|^2 = \frac{\mathbf{v}^H \mathbf{v}_0}{\|\mathbf{v}\|^2}, \quad (48)$$

where \mathbf{v}_0 represents the exact value of the zero-delay MMSE equalizer and \mathbf{v} the blind MMSE equalizer presented previously.

5.1. Asymptotic performance loss

Theoretically, the optimal MSE is given by

$$\text{MSE}_{\text{opt}} = E(|s(t) - \mathbf{v}_0^H \mathbf{x}_N(t)|^2) = 1 - \mathbf{g}_0^H \mathbf{C}^{-1} \mathbf{g}_0, \quad (49)$$

where vector \mathbf{g}_0 is given by (6) (for $q = 1, \tau = 0$). Let $\widehat{\text{MSE}}_{\text{opt}}$ denotes the MSE reached by $\hat{\mathbf{v}}_0$ the estimate of \mathbf{v}_0 :

$$\widehat{\text{MSE}}_{\text{opt}} \stackrel{\text{def}}{=} E(|s(t) - \hat{\mathbf{v}}_0^H \mathbf{x}_N(t)|^2). \quad (50)$$

In terms of MSE, the blind estimation leads to a performance loss equal to

$$\widehat{\text{MSE}}_{\text{opt}} - \text{MSE}_{\text{opt}} = \text{trace}(\mathbf{C}(\hat{\mathbf{v}}_0 - \mathbf{v}_0)(\hat{\mathbf{v}}_0 - \mathbf{v}_0)^H). \quad (51)$$

Asymptotically (i.e., for large sample sizes K), this performance loss is given by

$$\varepsilon \stackrel{\text{def}}{=} \lim_{K \rightarrow +\infty} KE(\widehat{\text{MSE}}_{\text{opt}} - \text{MSE}_{\text{opt}}) = \text{trace}(\mathbf{C} \boldsymbol{\Sigma}_{\mathbf{v}}), \quad (52)$$

where $\boldsymbol{\Sigma}_{\mathbf{v}}$ is the asymptotic covariance matrix of vector $\hat{\mathbf{v}}_0$. As $\hat{\mathbf{v}}_0$ is a "function" of the sample covariance matrix of the observed signal $\mathbf{x}_N(t)$, denoted here by $\hat{\mathbf{C}}$ and given, from K -sample observation, by

$$\hat{\mathbf{C}} = \frac{1}{K} \sum_{t=0}^{K-1} \mathbf{x}_N(t) \mathbf{x}_N^H(t), \quad (53)$$

it is clear that $\boldsymbol{\Sigma}_{\mathbf{v}}$ depends on the asymptotic covariance matrix of $\hat{\mathbf{C}}$. The following lemma gives the explicit expression of the asymptotic covariance matrix of the random vector $\hat{\mathbf{C}} = \text{vec}(\hat{\mathbf{C}})$.

Lemma 1. Let \mathbf{C}_τ be the τ -lag covariance matrix of the signal $\mathbf{x}_N(t)$ defined by

$$\mathbf{C}_\tau \stackrel{\text{def}}{=} E(\mathbf{x}_N(t + \tau)\mathbf{x}_N^H(t)) \quad (54)$$

and let $\text{cum}(x_1, x_2, \dots, x_k)$ be the k th-order cumulant of the random variables (x_1, x_2, \dots, x_k) .

Under the above data assumptions, the sequence of estimates $\hat{\mathbf{C}} = \text{vec}(\hat{\mathbf{C}})$ is asymptotically normal with mean $\mathbf{c} = \text{vec}(\mathbf{C})$ and covariance $\Sigma_{\mathbf{c}}$. That is,

$$\sqrt{K}(\hat{\mathbf{c}} - \mathbf{c}) \xrightarrow{\mathcal{L}} \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{c}}). \quad (55)$$

The covariance $\Sigma_{\mathbf{c}}$ is given by

$$\Sigma_{\mathbf{c}} = \kappa \tilde{\mathbf{c}} \tilde{\mathbf{c}}^H + \sum_{\tau=-(m-1)}^{m-1} \mathbf{C}_\tau^T \otimes \mathbf{C}_\tau^H, \quad (56)$$

$$\tilde{\mathbf{c}} = \text{vec}(\mathbf{C} - \sigma_b^2 \mathbf{I}_n),$$

$$\kappa = \text{cum}(s(t), s^*(t), s(t), s^*(t)),$$

where κ is the kurtosis of the input signal $s(t)$.

Proof. see Appendix B. \square

Now, to establish the asymptotic normality of vector estimate $\hat{\mathbf{v}}_0$, we use the so-called “continuity theorem,” which states that an asymptotically normal statistic transmits its asymptotic normality to any parameter vector estimated from it, as long as the mapping linking the statistic to the parameter vector is sufficiently regular in a neighborhood of the true (asymptotic) value of the statistic. More specifically, we have the following theorem [24].

Theorem 2. Let $\boldsymbol{\theta}_K$ be an asymptotically normal sequence of random vectors, with asymptotic mean $\boldsymbol{\theta}$ and asymptotic covariance $\Sigma_{\boldsymbol{\theta}}$. Let $\boldsymbol{\omega} = [\omega_1 \ \dots \ \omega_{n_\omega}]^T$ be a real-valued vector function defined on a neighborhood of $\boldsymbol{\theta}$ such that each component function ω_k has nonzero differential at point $\boldsymbol{\theta}$, that is, $\mathbf{D}\omega_k(\boldsymbol{\theta}) \neq \mathbf{0}$, $k = 1, \dots, n_\omega$. Then, $\boldsymbol{\omega}(\boldsymbol{\theta}_K)$ is an asymptotically normal sequence of n_ω -dimensional random vectors with mean $\boldsymbol{\omega}(\boldsymbol{\theta})$ and covariance $\Sigma = [\Sigma_{i,j}]_{1 \leq i,j \leq n_\omega}$ given by

$$\Sigma_{i,j} = \mathbf{D}\omega_i^T(\boldsymbol{\theta}) \Sigma_{\boldsymbol{\theta}} \mathbf{D}\omega_j(\boldsymbol{\theta}). \quad (57)$$

Applying the previous theorem to the estimate of \mathbf{v}_0 leads to the following theorem.

Theorem 3. Under the above data assumptions and in the SIMO case ($q = 1$), the random vector $\hat{\mathbf{v}}_0$ is asymptotically Gaussian distributed with mean \mathbf{v}_0 and covariance $\Sigma_{\mathbf{v}}$, that is,

$$\sqrt{K}(\hat{\mathbf{v}}_0 - \mathbf{v}_0) \xrightarrow{\mathcal{L}} \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{v}}). \quad (58)$$

The expression of $\Sigma_{\mathbf{v}}$ is given by

$$\Sigma_{\mathbf{v}} = \mathbf{M} \Sigma_{\mathbf{c}} \mathbf{M}^H, \quad (59)$$

where $\Sigma_{\mathbf{c}}$ is the asymptotic covariance matrix of the sample estimate of vector $\mathbf{c} = \text{vec}(\mathbf{C})$ given in Lemma 1 and matrix \mathbf{M} is given by

$$\mathbf{M} = r \left(\mathbf{I}_n - \frac{\mathbf{v}\mathbf{v}^H}{\|\mathbf{v}\|^2} \right) [(\tilde{\mathbf{v}}^T \otimes \mathbf{I}_n) \boldsymbol{\Gamma} - \mathbf{W} \mathbf{M}_2 \mathbf{M}_1],$$

$$\boldsymbol{\Gamma} = \begin{bmatrix} \mathbf{W}^T(:, 1) \otimes (\lambda_1 \mathbf{I}_n - \mathbf{C})^\# \\ \vdots \\ \mathbf{W}^T(:, d) \otimes (\lambda_d \mathbf{I}_n - \mathbf{C})^\# \end{bmatrix},$$

$$\mathbf{M}_1 = [(\mathbf{C} \mathbf{J}_{p,n-p,0} \mathbf{T})^T \otimes \mathbf{I}_d] \mathbf{U}_{n,d} \boldsymbol{\Gamma}^* \mathbf{U}_{n,n} + [\mathbf{I}_d \otimes (\mathbf{T}^H \mathbf{J}_{p,n-p,0}^T \mathbf{C})] \boldsymbol{\Gamma} + (\mathbf{J}_{p,n-p,0} \mathbf{T})^T \otimes \mathbf{W}^H + \mathbf{W}^T \otimes (\mathbf{T}^H \mathbf{J}_{p,n-p,0}^T),$$

$$\mathbf{M}_2 = \tilde{\mathbf{v}}^T \otimes \mathbf{Q}',$$

$$\mathbf{U}_{\alpha,\beta} = \sum_{i=1}^{\alpha} \sum_{j=1}^{\beta} \left(\mathbf{e}_i^\alpha [\mathbf{e}_j^\beta]^T \right) \otimes \left(\mathbf{e}_j^\beta [\mathbf{e}_i^\alpha]^T \right),$$

$$\mathbf{Q}' = \begin{cases} \mathbf{Q}^\#, & \text{in the quadratic constraint case} \\ \mathbf{J}_1 (\mathbf{J}_1^T \mathbf{Q} \mathbf{J}_1)^{-1} \mathbf{J}_1^T, & \text{in the linear constraint case,} \end{cases} \quad (60)$$

where $\mathbf{U}_{\alpha,\beta}$ is a permutation matrix, \mathbf{e}_k^l denotes the k th column vector of matrix \mathbf{I}_l and $\lambda_1 > \lambda_2 \geq \dots \geq \lambda_d$ are the d principal eigenvalues of \mathbf{C} associated to the eigenvectors $\mathbf{W}(:, 1), \dots, \mathbf{W}(:, d)$, respectively.

Proof. see Appendix C. \square

5.2. Validation of the asymptotic covariance expressions and performance evaluation

In this section, we assess the performance of the blind equalization algorithm by Monte-Carlo experiments. We consider a SIMO channel ($q = 1$, $p = 3$, and $L = 4$), chosen randomly using Rayleigh distribution for each tap. The input signal is an iid QAM4 sequence. The width of the temporal window is $N = 6$. The theoretical expressions are compared with empirical estimates, obtained by Monte-Carlo simulations (100 independent Monte-Carlo simulations are performed in each experiment). The performance criterion used here is the relative mean square error (RMSE), defined as the sample average, over the Monte-Carlo simulations, of the total estimation of MSE loss, that is, $\widehat{\text{MSE}}_{\text{opt}} - \text{MSE}_{\text{opt}}$. This quantity is compared with its exact asymptotic expression divided by the sample size K , $\varepsilon_K = (1/K)\varepsilon = (1/K)\text{trace}(\mathbf{C}\Sigma_{\mathbf{v}})$. The signal-to-noise ratio (SNR) is defined (in dB) by $\text{SNR} = -20 \log(\sigma_b)$.

Figure 1(a) compares, in the quadratic constraint case, the empirical RMSE (solid line) with the theoretical one ε_K (dashed line) as a function of the sample size K . The SNR is set to 15 dB. It is seen that the theoretical expression of

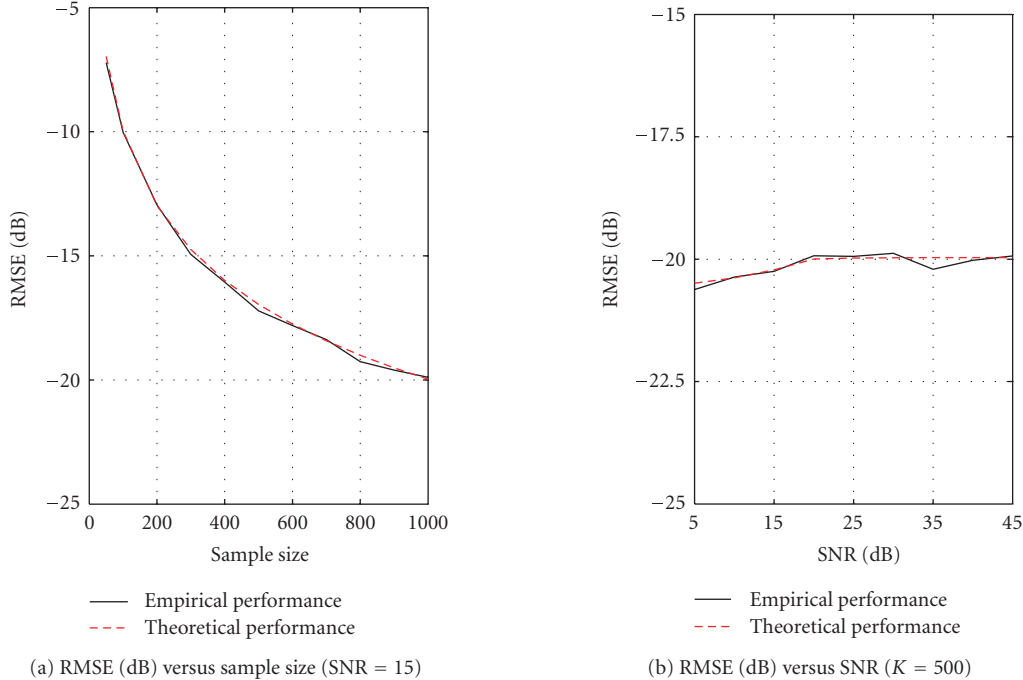


FIGURE 1: Asymptotic loss of performance: quadratic constraint.

the RMSE is valid from snapshot length as short as 50 samples, this means that the asymptotic conditions are reached for short sample size. In Figure 1(b) the empirical (solid line) and the theoretical (dashed line) RMSEs are plotted against the SNR. The sample size is set to $K = 500$ samples. This figure demonstrates that there is a close agreement between theoretical and experimental values. Similar results are obtained when the linear constraint is used.

6. SIMULATION RESULTS AND DISCUSSION

We provide in this section some simulation examples to illustrate the performance of the proposed blind equalizer. Our tests are based on SIMO and MIMO channels. The channel coefficients are chosen randomly at each run according to a complex Gaussian distribution. The input signals are iid QAM4 sequences. As a performance measure, we estimate the average MSE given by

$$\text{MSE} = \frac{1}{q} E(\|\mathbf{s}(t - \tau) - \hat{\mathbf{V}}_{\tau}^H \mathbf{x}_N(t)\|^2), \quad (61)$$

over 100 Monte-Carlo runs. The MSE is compared to the optimal MSE given by

$$\text{MSE}_{\text{opt}} = \frac{1}{q} \text{trace}(\mathbf{I}_q - \mathbf{G}_{\tau}^H \mathbf{C}^{-1} \mathbf{G}_{\tau}). \quad (62)$$

6.1. Performance evaluation

In this experiment, we investigate the performance of our algorithm. In Figure 2(a) (SIMO case with quadratic con-

straint) and Figure 2(b) (MIMO case) we plot the MSE (in dB) against SNR (in dB) for $K = 500$. One can observe the performance loss of the zero-delay MMSE filter compared to the optimal one, due (as shown above) to the blind estimation procedure. Also, it illustrates the effectiveness of the two-step approach, which allows us to compensate for the performance loss and to choose a nonzero equalization delay, that improves the overall performance.

Figure 3(a) (SIMO case with quadratic constraint) and Figure 3(b) (MIMO case) represent the convergence rate of the adaptive algorithm with SNR = 15 dB. Given the low computational cost of the algorithm, a relatively fast convergence rate is observed. Figure 4 compares, in fast time varying channel case, the tracking performance of the adaptive algorithm using respectively, YAST and OPASt as a subspace trackers. The channel variation model is the one given in [25] and the SNR is set to 15 dB. As we can observe, the adaptive equalization algorithm using YAST succeeds to track the channel variation, while it fails when using OPASt. Figure 5 compares the performance of our zero-delay MMSE equalizer with those given by the algorithms in [10, 11], respectively. The plot represents the estimated signal MSE versus the SNR for $K = 500$. As we can observe, our method outperforms the methods in [10, 11] for low SNRs.

6.2. Robustness to channel order overestimation errors

This experiment is dedicated to the study of the robustness against channel order overestimation errors. Figure 6(a) (resp., Figure 6(b)) represents the MSE versus the overestimated channel order for SNR = 15 and $K = 500$ (resp.,

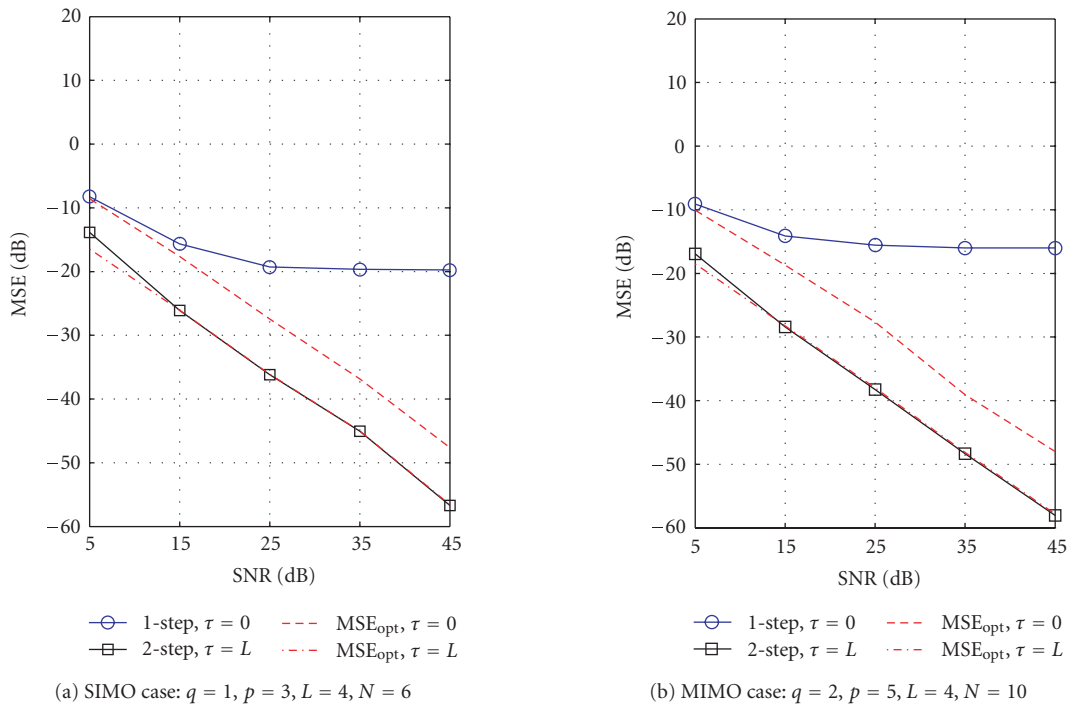


FIGURE 2: Performance of the equalizer.

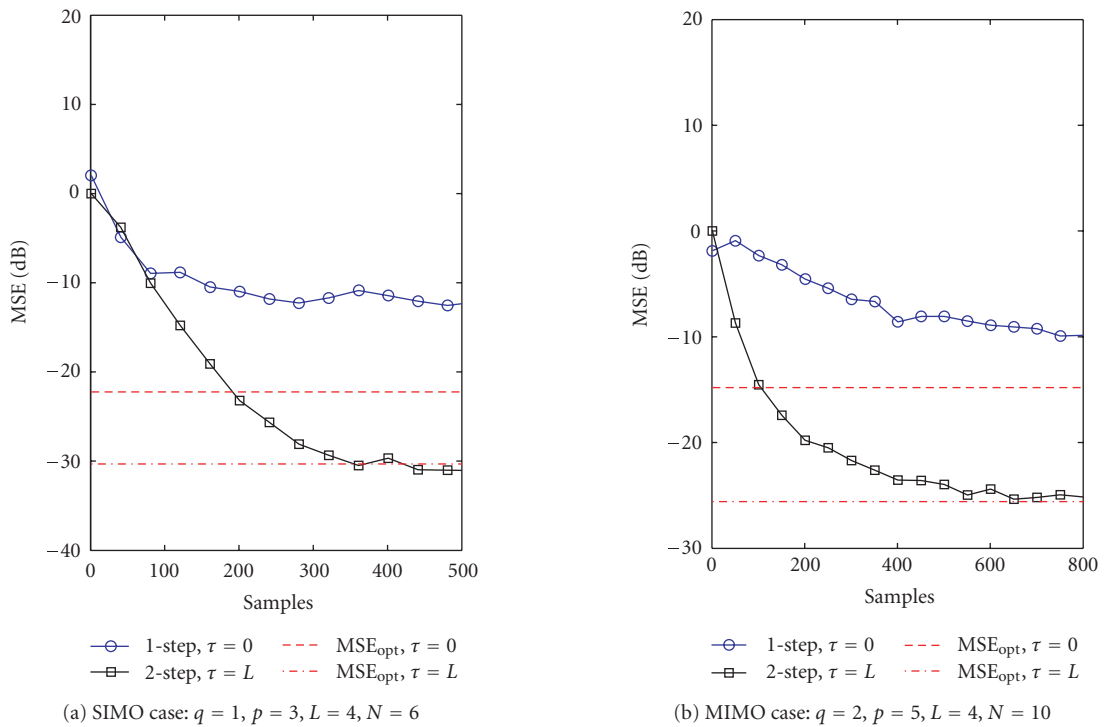


FIGURE 3: Convergence of the adaptive equalizer.

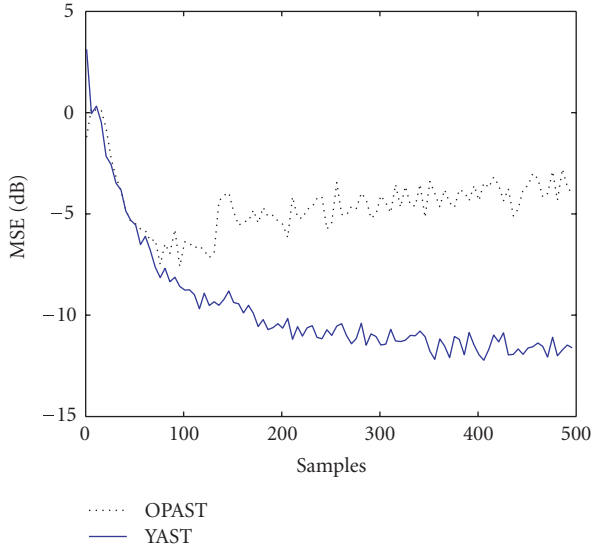


FIGURE 4: Convergence of the adaptive equalization algorithm in the time varying channel case.

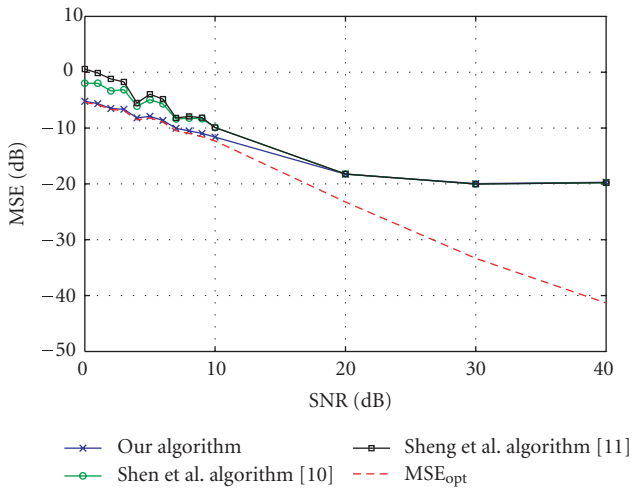


FIGURE 5: Performance comparison of batch-type SIMO equalizers ($q = 1, p = 3, L = 4, N = 6$).

$K = 1000$). The plot compares, in the SIMO case, the MSE obtained by our algorithm using linear constraint (l.c.) and quadratic constraint (q.c.), respectively, to that obtained by algorithm in [10] (identical results are obtained with algorithm in [11]). Clearly, the use of linear constraint improves significantly the robustness against channel order overestimation errors of the blind MMSE filter. Note that, as explained in Section 3.5, improved results are obtained with the proposed algorithm using quadratic constraint, when the sample size increases. This is observed by comparing the

results of the quadratic constraint method of Figure 6(b) with those of Figure 6(a).

6.3. Robustness against small values of $\mathbf{H}(0)$

In general, the main weakness of a zero-delay equalizer is its sensitivity to small values of the first channel coefficient $\mathbf{H}(0)$. In Figure 7, we illustrate the robustness of the proposed algorithm, when $\mathbf{H}(0)$ takes small value. More precisely, we plot the MSE versus the variance of $\mathbf{H}(0)$: $\sigma_{\mathbf{H}(0)}^2 \stackrel{\text{def}}{=} E(\|\mathbf{H}(0)\|^2)$, for $q = 1, p = 3, K = 500$, and SNR = 15 in Figure 7(a) (resp., SNR = 30 in Figure 7(b)). It is clear that for low and moderate SNRs a minimum variance of $\mathbf{H}(0)$ is needed (in the plot $\sigma_{\mathbf{H}(0)}^2 \geq 0.2$ is required) for the algorithm to provide satisfactory results. However, this threshold value can be quite small for high SNR as shown by Figure 7(b).

6.4. Influence of the number of sensors

Figure 8 represents the evolution of the MSE versus the number of sensors for $q = 1, K = 500$, and SNR = 5 in Figure 8(a) (resp., SNR = 15 in Figure 8(b)). One can observe that for low SNR, the algorithm requires a minimum degree of freedom in terms of number of sensors (typically $p - q$ should be larger than 2 or 3), while at moderate and large SNRs, p can be as small as $q + 1$. Eventhough not included here, due to space limitation, similar results have been observed in the MIMO case.

6.5. Discussion

These results highlight one of the main advantages of our method which is the improved robustness against channel order overestimation errors. Also, even when the channel order is known, the proposed algorithm outperforms the algorithms in [10, 11] for low SNR. Another strong advantages of the proposed algorithm is its low computational cost and higher convergence rate (in its adaptive version) compared to those in [10–12]. However, the methods in [10–12] have the advantages of allowing direct estimation (in one step) of the nonzero-delay equalizer which is important in certain limit cases, where the zero-delay equalizer fails to provide satisfactory performance (see Figure 7).

7. CONCLUSION

In this contribution, we have presented an original method for blind equalization of multichannel FIR filters. Batch and fast adaptive implementation algorithms are developed. A two-step version using the a priori knowledge of the source signal finite alphabet has been proposed in order to control the equalization delay and improve the estimation performance. An asymptotic performance analysis of the proposed algorithm has been carried out in the single input case (SIMO case). Robustness against channel

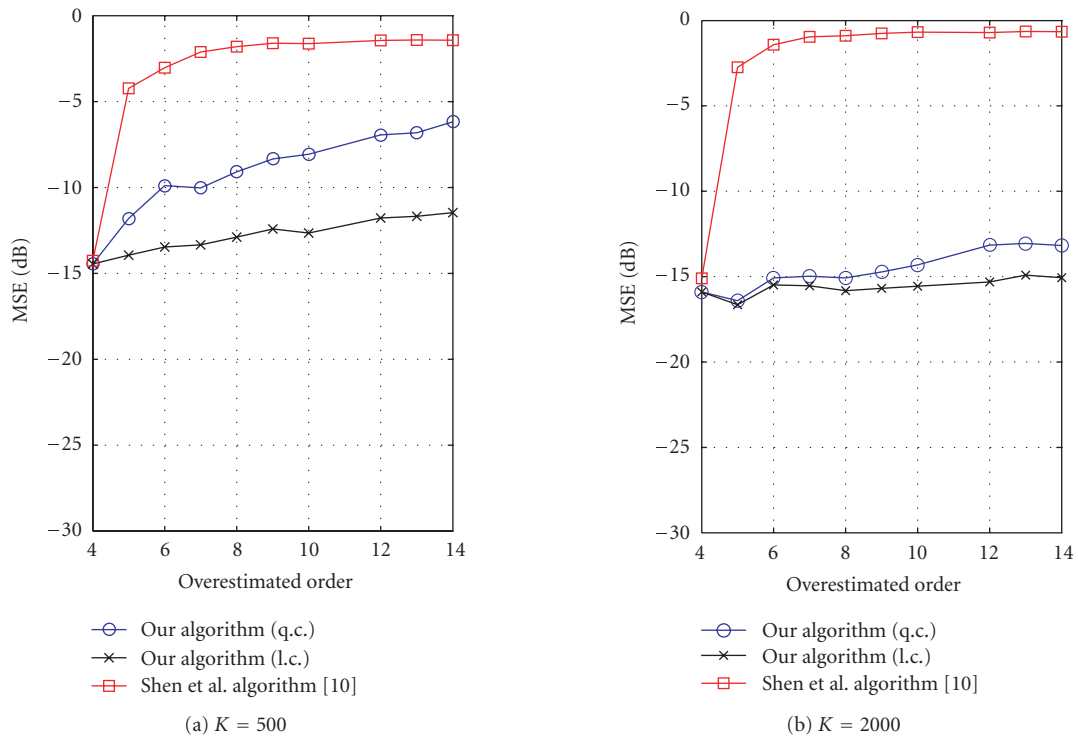


FIGURE 6: Robustness comparison (against channel order overestimation errors). The exact order is $L = 4$.

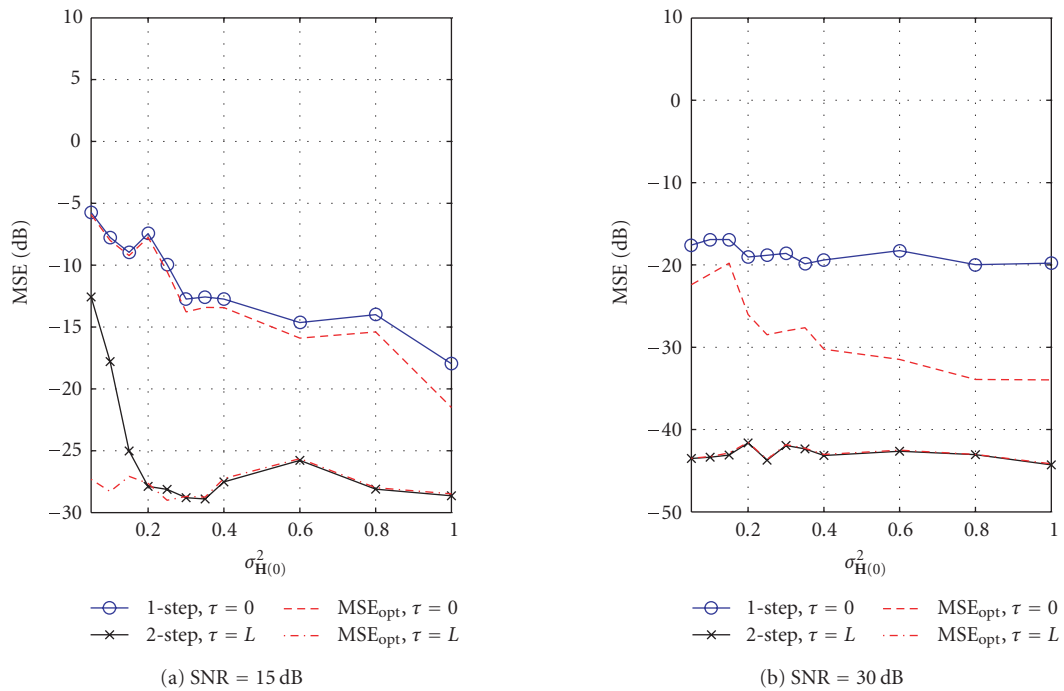


FIGURE 7: Robustness against small values of $\mathbf{H}(0)$.

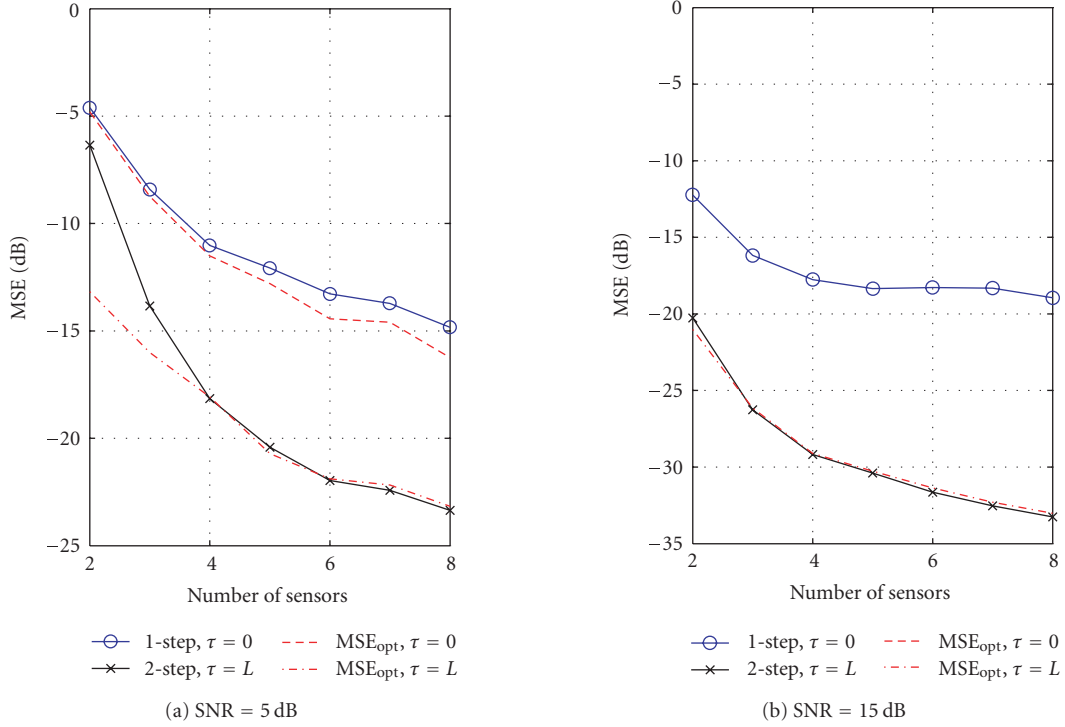


FIGURE 8: Mean square error versus the number of sensors.

order overestimation errors and performance of the proposed equalization method are studied.

APPENDICES

A. $O(n)$ COMPUTATION OF $\mathbf{x}'(t) = \mathbf{C}(t-1)\mathbf{x}_N(t)$

A technique to reduce the computation of the vector $\mathbf{x}'(t) = \mathbf{C}(t-1)\mathbf{x}_N(t)$ from $O(n^2)$ to $O(n)$ operations, is presented herein. This technique was proposed in [20] for time series data, and here we generalize it for multivariate data.

We begin by defining the $(n+p)$ -dimensional vector

$$\underline{\mathbf{g}}(t) \stackrel{\text{def}}{=} \underline{\mathbf{C}}(t-1)\mathbf{x}_{N+1}(t), \quad (\text{A.1})$$

where $\underline{\mathbf{C}}(t)$ is the extended covariance matrix given by

$$\underline{\mathbf{C}}(t) = \sum_{k=1}^t \beta^{t-k} \mathbf{x}_{N+1}(k)\mathbf{x}_{N+1}^H(k). \quad (\text{A.2})$$

Taking into account the fact that

$$\mathbf{x}_{N+1}(t) = [\mathbf{x}^T(t) \quad \mathbf{x}_N^T(t-1)]^T = [\mathbf{x}_N^T(t) \quad \mathbf{x}^T(t-N)]^T, \quad (\text{A.3})$$

one can write

$$\underline{\mathbf{C}}(t) = \begin{bmatrix} \mathbf{C}^1(t) & \mathbf{C}^2(t) \\ [\mathbf{C}^2(t)]^H & \mathbf{C}(t-1) \end{bmatrix} = \begin{bmatrix} \mathbf{C}(t) & \mathbf{C}^3(t) \\ [\mathbf{C}^3(t)]^H & \mathbf{C}^1(t-N) \end{bmatrix}, \quad (\text{A.4})$$

where

$$\mathbf{C}^1(t) = \sum_{k=1}^t \beta^{t-k} \mathbf{x}(k)\mathbf{x}^H(k),$$

$$\mathbf{C}^2(t) = \sum_{k=1}^t \beta^{t-k} \mathbf{x}(k)\mathbf{x}_N^H(k-1), \quad (\text{A.5})$$

$$\mathbf{C}^3(t) = \sum_{k=1}^t \beta^{t-k} \mathbf{x}_N(k)\mathbf{x}^H(k-N).$$

Using (A.3) and (A.4), we have

$$\underline{\mathbf{g}}(t) = \begin{bmatrix} \mathbf{C}^1(t-1)\mathbf{x}(t) + \mathbf{C}^2(t-1)\mathbf{x}_N(t-1) \\ [\mathbf{C}^2(t-1)]^H \mathbf{x}(t) + \mathbf{x}'(t-1) \end{bmatrix} \quad (\text{A.6})$$

$$= \begin{bmatrix} \mathbf{x}'(t) + \mathbf{C}^3(t-1)\mathbf{x}(t-N) \\ [\mathbf{C}^3(t-1)]^H \mathbf{x}_N(t) + \mathbf{C}^1(t-N-1)\mathbf{x}(t-N) \end{bmatrix}. \quad (\text{A.7})$$

Equation (A.6) is used to compute $\underline{\mathbf{g}}(t)$ and, from (A.7), $\mathbf{x}'(t)$ is updated as follows:

$$\mathbf{x}'(t) = \underline{\mathbf{g}}(t)_{(1:n)} - \mathbf{C}^3(t-1)\mathbf{x}(t-N). \quad (\text{A.8})$$

The only other quantities that need updating are the matrices

$$\begin{aligned}
\mathbf{g}(t)_{(1:p)} &= \mathbf{C}^1(t-1)\mathbf{x}(t) + \mathbf{C}^2(t-1)\mathbf{x}_N(t-1) \\
\mathbf{g}(t)_{(p+1:n+p)} &= [\mathbf{C}^2(t-1)]^H \mathbf{x}(t) + \mathbf{x}'(t-1) \\
\mathbf{x}'(t) &= \mathbf{g}(t)_{(1:n)} - \mathbf{C}^3(t)\mathbf{x}(t-N) \\
\mathbf{C}^1(t) &= \beta \mathbf{C}^1(t-1) + \mathbf{x}(t)\mathbf{x}^H(t) \\
\mathbf{C}^2(t) &= \beta \mathbf{C}^2(t-1) + \mathbf{x}(t)\mathbf{x}_N^H(t-1) \\
\mathbf{C}^3(t) &= \beta \mathbf{C}^3(t-1) + \mathbf{x}_N(t)\mathbf{x}^H(t-N)
\end{aligned}$$

ALGORITHM 8: Algorithm for updating $\mathbf{x}'(t) = \mathbf{C}(t-1)\mathbf{x}_N(t)$ in $O(n)$ operations.

in (A.4), which can be efficiently computed as

$$\begin{aligned}
\mathbf{C}^1(t) &= \beta \mathbf{C}^1(t-1) + \mathbf{x}(t)\mathbf{x}^H(t), \\
\mathbf{C}^2(t) &= \beta \mathbf{C}^2(t-1) + \mathbf{x}(t)\mathbf{x}_N^H(t-1), \\
\mathbf{C}^3(t) &= \beta \mathbf{C}^3(t-1) + \mathbf{x}_N(t)\mathbf{x}^H(t-N).
\end{aligned} \tag{A.9}$$

The algorithm listing is found in Algorithm 8.

B. PROOF OF LEMMA 1

Matrix Σ_c is defined by

$$\Sigma_c = [\Sigma_{c,k,l}]_{1 \leq k,l \leq n^2} \stackrel{\text{def}}{=} \lim_{K \rightarrow +\infty} KE((\hat{c} - c)(\hat{c} - c)^H), \tag{B.1}$$

it follows that

$$\begin{aligned}
\Sigma_{c,k,l} &= \lim_{K \rightarrow +\infty} KE((\hat{c}_k - c_k)(\hat{c}_l - c_l)^*) \\
&= \lim_{K \rightarrow +\infty} KE((\hat{C}_{a,b} - C_{a,b})(\hat{C}_{c,d} - C_{c,d})^*),
\end{aligned} \tag{B.2}$$

where c_i (resp., $C_{\alpha,\beta}$) and \hat{c}_i (resp., $\hat{C}_{\alpha,\beta}$) denote the i th (resp., the (α, β) th) entry of \mathbf{c} (resp., \mathbf{C}) and $\hat{\mathbf{c}}$ (resp., $\hat{\mathbf{C}}$), respectively, which are given by

$$\begin{aligned}
c_i &= C_{\alpha,\beta} = E(x_{N,\alpha}(t)x_{N,\beta}^*(t)), \\
\hat{c}_i &= \hat{C}_{\alpha,\beta} = \frac{1}{K} \sum_{t=0}^{K-1} x_{N,\alpha}(t)x_{N,\beta}^*(t), \\
\alpha &= \alpha' + n\delta(\alpha'), \quad \beta = \beta' + 1 - \delta(\alpha'), \quad 1 \leq \alpha, \beta \leq n,
\end{aligned} \tag{B.3}$$

where $x_{N,i}(t)$ is the i th entry of vector $\mathbf{x}_N(t)$, α' and β' denote, respectively, the rest and the quotient of the Euclidian division of i by n , and δ is the Kronecker symbol. (a, b) and (c, d) are obtained in a similar way for k and l , respectively.

Then, after some calculation (see [15] for more details) and using the relationship between cumulants and moments, we obtain the following expression of $\Sigma_{c,k,l}$:

$$\Sigma_{c,k,l} = \kappa_{k,l} + \sum_{\tau \in \mathbb{Z}} C_{\tau,a,c} C_{-\tau,d,b}, \tag{B.4}$$

where

$$\kappa_{k,l} \stackrel{\text{def}}{=} \sum_{\tau \in \mathbb{Z}} \text{cum}(x_{N,a}(\tau), x_{N,b}^*(\tau), x_{N,d}(0), x_{N,c}^*(0)), \tag{B.5}$$

taking into account the particular structure of the data model (2) and applying some standard properties of cumulants, the fourth-order cumulant in (B.5) is then expressed as

$$\begin{aligned}
&\text{cum}(x_{N,a}(\tau), x_{N,b}^*(\tau), x_{N,d}(0), x_{N,c}^*(0)) \\
&= \kappa \sum_{i \in \mathbb{Z}} \mathbf{H}_N(a, i + \tau) \mathbf{H}_N^*(b, i + \tau) \mathbf{H}_N(d, i) \mathbf{H}_N^*(c, i),
\end{aligned} \tag{B.6}$$

where $\kappa \stackrel{\text{def}}{=} \text{cum}(s(t), s^*(t), s(t), s^*(t))$ is the kurtosis of the input signal $s(t)$, and $\mathbf{H}_N(i, j)$ is the (i, j) th entry of \mathbf{H}_N . Plugging this expression into (B.5) yields

$$\begin{aligned}
\kappa_{k,l} &= \kappa \sum_j \mathbf{H}_N(a, j) \mathbf{H}_N^*(b, j) \sum_i \mathbf{H}_N(d, i) \mathbf{H}_N^*(c, i) \\
&= \kappa (C_{a,b} - \sigma_b^2 \delta(a-b)) (C_{c,d} - \sigma_c^2 \delta(c-d))^*.
\end{aligned} \tag{B.7}$$

Finally, it is easy to verify from (B.4) and (B.7) that Σ_c is expressed by (56).

C. PROOF OF THEOREM 3

Before proceeding, we first need to recall some basic properties of column vectorizing operator and, matrices and vectors differentiation (see [16] for more details). If \mathbf{A} , \mathbf{B} , and \mathbf{C} are given matrices, then $\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A}) \text{vec}(\mathbf{B})$ and $\delta \text{vec}(\mathbf{A}) = \text{vec}(\delta \mathbf{A})$ where δ denotes the differentiation operator. If \mathbf{A} is an $\alpha \times \beta$ matrix, then $\text{vec}(\mathbf{A}^T) = \mathbf{U}_{\alpha,\beta} \text{vec}(\mathbf{A})$ where $\mathbf{U}_{\alpha,\beta}$ is the permutation matrix defined in Theorem 3. Let λ be an eigenvalue of an $\alpha \times \alpha$ matrix \mathbf{A} , \mathbf{w} the eigenvector corresponding to λ , the differential $\delta \mathbf{w}$ of \mathbf{w} is given by $\delta \mathbf{w} = (\lambda \mathbf{I}_\alpha - \mathbf{A})^\# \delta \mathbf{A} \mathbf{w} = [\mathbf{w}^T \otimes (\lambda \mathbf{I}_\alpha - \mathbf{A})^\#] \delta \text{vec}(\mathbf{A})$. If \mathbf{A} is invertible, then $\delta \mathbf{A}^{-1} = -\mathbf{A}^{-1} \delta \mathbf{A} \mathbf{A}^{-1}$.

Let $\hat{\mathbf{v}}$ be an estimate of the blind MMSE equalizer vector given from K -sample observations, then $\hat{\mathbf{v}}_0$ is given according to $\hat{\mathbf{v}}_0 = \hat{r} \hat{\mathbf{v}}$, where $\hat{r} = \hat{\mathbf{v}}^H \mathbf{v}_0 / \|\hat{\mathbf{v}}\|^2$. Replacing $\hat{\mathbf{v}}_0$, $\hat{\mathbf{v}}$, and \hat{r} by $\mathbf{v}_0 + \delta \mathbf{v}_0$, $\mathbf{v} + \delta \mathbf{v}$, and $r + \delta r$, respectively, we obtain

$$\delta \mathbf{v}_0 = r \left(\mathbf{I}_n - \frac{\mathbf{v} \mathbf{v}^H}{\|\mathbf{v}\|^2} \right) \delta \mathbf{v}. \tag{C.1}$$

As $\mathbf{v} = \mathbf{W} \tilde{\mathbf{v}}$, it follows that

$$\delta \mathbf{v} = (\tilde{\mathbf{v}}^T \otimes \mathbf{I}_n) \delta \text{vec}(\mathbf{W}) + \mathbf{W} \delta \tilde{\mathbf{v}}. \tag{C.2}$$

Quadratic constraint case

In this case, $\tilde{\mathbf{v}}$ is the least eigenvector (which correspond to zero-eigenvalue) of matrix \mathbf{Q} given by (12) and (18). The differentiation of $\tilde{\mathbf{v}}$ gives

$$\delta \tilde{\mathbf{v}} = -(\tilde{\mathbf{v}}^T \otimes \mathbf{Q}^\#) \delta \text{vec}(\mathbf{Q}) = -\mathbf{M}_2 \delta \text{vec}(\mathbf{Q}). \tag{C.3}$$

From (12) and (18), matrix \mathbf{Q} is written as

$$\mathbf{Q} = \mathbf{W}^H \mathbf{C} \mathbf{J}_{p,n-p,0} \mathbf{J}_{p,n-p,0}^T \mathbf{C} \mathbf{W}. \quad (\text{C.4})$$

The differentiation of \mathbf{Q} gives

$$\begin{aligned} \delta \text{vec}(\mathbf{Q}) &= [(\mathbf{C} \mathbf{J}_{p,n-p,0} \mathbf{T})^T \otimes \mathbf{I}_d] \delta \text{vec}(\mathbf{W}^H) \\ &+ [\mathbf{I}_d \otimes (\mathbf{T}^H \mathbf{J}_{p,n-p,0}^T \mathbf{C})] \delta \text{vec}(\mathbf{W}) \\ &+ [(\mathbf{J}_{p,n-p,0} \mathbf{T})^T \otimes \mathbf{W}^H + \mathbf{W}^T \otimes (\mathbf{T}^H \mathbf{J}_{p,n-p,0}^T)] \delta \mathbf{c}. \end{aligned} \quad (\text{C.5})$$

As the columns of \mathbf{W} correspond to the d dominant eigenvectors of \mathbf{C} , thus $\delta \mathbf{W} = [\delta \mathbf{W}(:,1) \cdots \delta \mathbf{W}(:,d)]$, where $\delta \mathbf{W}(:,i) = (\mathbf{W}^T(:,i) \otimes (\lambda \mathbf{I}_n - \mathbf{C})^\#) \delta \mathbf{c}$, $i = 1, \dots, d$. This implies

$$\delta \text{vec}(\mathbf{W}) = \mathbf{\Gamma} \delta \mathbf{c}, \quad (\text{C.6})$$

where $\mathbf{\Gamma}$ is defined in Theorem 3. It follows that $\delta \text{vec}(\mathbf{W}^H) = \mathbf{U}_{n,d} \delta \text{vec}(\mathbf{W}^*) = \mathbf{U}_{n,d} \mathbf{\Gamma}^* \delta \text{vec}(\mathbf{C}^*)$, as $\mathbf{C}^* = \mathbf{C}^T$, we obtain

$$\delta \text{vec}(\mathbf{W}^H) = \mathbf{U}_{n,d} \mathbf{\Gamma}^* \mathbf{U}_{n,n} \delta \mathbf{c}. \quad (\text{C.7})$$

Plugging (C.6) and (C.7) in (C.5) yields

$$\delta \text{vec}(\mathbf{Q}) = \mathbf{M}_1 \delta \mathbf{c}, \quad (\text{C.8})$$

where \mathbf{M}_1 is given in Theorem 3. Finally, from (C.1), (C.2), (C.3), (C.6), and (C.8), we obtain

$$\delta \mathbf{v}_0 = \mathbf{M} \delta \mathbf{c}. \quad (\text{C.9})$$

Linear constraint case

In this case, we use the expression of $\tilde{\mathbf{v}}$ given by (25)

$$\tilde{\mathbf{v}} = \mathbf{J}_1 \bar{\mathbf{v}} + \mathbf{J}_{0,1,d-1}. \quad (\text{C.10})$$

From (23), (24), and (41), vector $\bar{\mathbf{v}}$ is expressed as

$$\bar{\mathbf{v}} = -(\mathbf{J}_1^T \mathbf{Q} \mathbf{J}_1)^{-1} \mathbf{J}_1^T \mathbf{Q} \mathbf{J}_{0,1,d-1}. \quad (\text{C.11})$$

Differentiating $\tilde{\mathbf{v}}$ yields

$$\begin{aligned} \delta \tilde{\mathbf{v}} &= \mathbf{J}_1 \delta \bar{\mathbf{v}} = \mathbf{J}_1 (\mathbf{J}_1^T \mathbf{Q} \mathbf{J}_1)^{-1} \mathbf{J}_1^T \delta \mathbf{Q} \mathbf{J}_1 (\mathbf{J}_1^T \mathbf{Q} \mathbf{J}_1)^{-1} \mathbf{J}_1^T \mathbf{Q} \mathbf{J}_{0,1,d-1} \\ &- \mathbf{J}_1 (\mathbf{J}_1^T \mathbf{Q} \mathbf{J}_1)^{-1} \mathbf{J}_1^T \delta \mathbf{Q} \mathbf{J}_{0,1,d-1} \\ &= -(\tilde{\mathbf{v}}^T \otimes \mathbf{Q}') \delta \text{vec}(\mathbf{Q}) = -\mathbf{M}_2 \mathbf{M}_1 \delta \mathbf{c}, \end{aligned} \quad (\text{C.12})$$

where \mathbf{Q}' is given as in Theorem 3. From (C.1), (C.2), (C.6), and (C.12), we obtain finally

$$\delta \mathbf{v}_0 = \mathbf{M} \delta \mathbf{c}. \quad (\text{C.13})$$

Using (C.9) (resp., (C.13)) in the quadratic constraint case (resp., in the linear constraint case) and Theorem 2 result leads to the expression of Σ_v given in Theorem 3.

ACKNOWLEDGMENT

Part of this work has been published in conferences [26, 27].

REFERENCES

- [1] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, Englewood Cliffs, NJ, USA, 3rd edition, 1996.
- [2] Y. Sato, "A method of self-recovering equalization for multi-level amplitude-modulation," *IEEE Transactions on Communications*, vol. 23, no. 6, pp. 679–682, 1975.
- [3] D. N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Transactions on Communications*, vol. 28, no. 11, pp. 1867–1875, 1980.
- [4] L. Tong, G. Xu, and T. Kailath, "A new approach to blind identification and equalization of multipaths channels," in *Proceedings of 25th Asilomar Conference on Circuits, Systems and Computers*, pp. 856–860, Pacific Grove, Calif, USA, November 1991.
- [5] K. Abed-Meraim, W. Qiu, and Y. Hua, "Blind system identification," *Proceedings of the IEEE*, vol. 85, no. 8, pp. 1310–1322, 1997.
- [6] E. Moulines, P. Duhamel, J.-F. Cardoso, and S. Mayrargue, "Subspace methods for the blind identification of multichannel FIR filters," *IEEE Transactions on Signal Processing*, vol. 43, no. 2, pp. 516–525, 1995.
- [7] A. P. Liavas, P. A. Regalia, and J.-P. Delmas, "Blind channel approximation: effective channel order determination," *IEEE Transactions on Signal Processing*, vol. 47, no. 12, pp. 3336–3344, 1999.
- [8] W. H. Gerstacker and D. P. Taylor, "Blind channel order estimation based on second-order statistics," *IEEE Signal Processing Letters*, vol. 10, no. 2, pp. 39–42, 2003.
- [9] J. Xavier and V. Barroso, "A channel order independent method for blind equalization of MIMO systems," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '99)*, vol. 5, pp. 2897–2900, Phoenix, Ariz, USA, March 1999.
- [10] J. Shen and Z. Ding, "Direct blind MMSE channel equalization based on second-order statistics," *IEEE Transactions on Signal Processing*, vol. 48, no. 4, pp. 1015–1022, 2000.
- [11] M. Sheng and H. Fan, "Blind MMSE equalization: a new direct method," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '00)*, vol. 5, pp. 2457–2460, Istanbul, Turkey, June 2000.
- [12] X. Li and H. Fan, "Direct estimation of blind zero-forcing equalizers based on second-order statistics," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2211–2218, 2000.
- [13] H. Gazzah, P. A. Regalia, J.-P. Delmas, and K. Abed-Meraim, "A blind multichannel identification algorithm robust to order overestimation," *IEEE Transactions on Signal Processing*, vol. 50, no. 6, pp. 1449–1458, 2002.
- [14] F. D. Neeser and J. L. Massey, "Proper complex random processes with applications to information theory," *IEEE Transactions on Information Theory*, vol. 39, no. 4, pp. 1293–1303, 1993.

- [15] K. Abed-Meraim, P. Loubaton, and E. Moulines, "A subspace algorithm for certain blind identification problems," *IEEE Transactions on Information Theory*, vol. 43, no. 2, pp. 499–511, 1997.
- [16] J. W. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Transactions on Circuits and Systems*, vol. 25, no. 9, pp. 772–781, 1978.
- [17] A.-J. van der Veen and A. Paulraj, "An analytical constant modulus algorithm," *IEEE Transactions on Signal Processing*, vol. 44, no. 5, pp. 1136–1155, 1996.
- [18] A. Belouchrani and K. Abed-Meraim, "Constant modulus blind source separation technique: a new approach," in *Proceedings of the International Symposium on Signal Processing and Its Applications (ISSPA '96)*, vol. 1, pp. 232–235, Gold Coast, Australia, August 1996.
- [19] B. Yang, "Projection approximation subspace tracking," *IEEE Transactions on Signal Processing*, vol. 43, no. 1, pp. 95–107, 1995.
- [20] C. E. Davila, "Efficient, high performance, subspace tracking for time-domain data," *IEEE Transactions on Signal Processing*, vol. 48, no. 12, pp. 3307–3315, 2000.
- [21] R. Badeau, B. David, and G. Richard, "Yet another subspace tracker," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '05)*, vol. 4, pp. 329–332, Philadelphia, Pa, USA, March 2005.
- [22] K. Abed-Meraim, A. Chkeif, and Y. Hua, "Fast orthogonal PAST algorithm," *IEEE Signal Processing Letters*, vol. 7, no. 3, pp. 60–62, 2000.
- [23] A. Chkeif, K. Abed-Meraim, G. Kawas-Kaleh, and Y. Hua, "Spatio-temporal blind adaptive multiuser detection," *IEEE Transactions on Communications*, vol. 48, no. 5, pp. 729–732, 2000.
- [24] J.-F. Cardoso and E. Moulines, "Asymptotic performance analysis of direction-finding algorithms based on fourth-order cumulants," *IEEE Transactions on Signal Processing*, vol. 43, no. 1, pp. 214–224, 1995.
- [25] M. K. Tsatsanis and G. B. Giannakis, "Modelling and equalization of rapidly fading channels," *International Journal of Adaptive Control and Signal Processing*, vol. 10, no. 2-3, pp. 159–176, 1996.
- [26] I. Kacha, K. Abed-Meraim, and A. Belouchrani, "A fast adaptive blind equalization algorithm robust to channel order overestimation errors," in *Proceedings of the 3rd IEEE Sensor Array and Multichannel Signal Processing Workshop*, pp. 148–152, Barcelona, Spain, July 2004.
- [27] I. Kacha, K. Abed-Meraim, and A. Belouchrani, "A new blind adaptive MMSE equalizer for MIMO systems," in *Proceedings of the 16th Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications*, Berlin, Germany, September 2005.

Ibrahim Kacha received the State Engineering and M.S. degrees both in electrical engineering from the École Nationale Polytechnique (ENP), Algiers, Algeria, in 1990 and 1993, respectively. He was a Lecturer at the Department of Electrical Engineering of ENP from 1993 to 2005. He is currently a Ph.D. student in the area of signal processing at the Department of Signal and Image Processing, École Nationale Supérieure des Télécommunications (ENST), Paris, France. His research interests



are statistical signal processing and blind system identification and equalization for digital communications.

Karim Abed-Meraim was born in 1967. He received the State Engineering degree from the École Polytechnique, Paris, France, in 1990, as well as from École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1992, the M.S. degree from Paris XI University, Orsay, France, in 1992, and the Ph.D. degree from École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1995 (in the field of signal processing and communications). From 1995 to 1998, he was a Research Staff Member at the Electrical Engineering Department of the University of Melbourne where he worked on several research projects related to blind system identification for wireless communications, blind source separation, and array processing for communications, respectively. He is currently an Associate Professor (since 1998) at the Signal and Image Processing Department of ENST. His research interests are in signal processing for communications and include system identification, multiuser detection, space-time coding, adaptive filtering and tracking, array processing, and performance analysis. He is an IEEE Senior Member and a past Associate Editor for the IEEE Transactions on Signal Processing.



Adel Belouchrani received the State Engineering degree in 1991 from École Nationale Polytechnique (ENP), Algiers, Algeria, the M.S. degree in signal processing from the Institut National Polytechnique de Grenoble (INPG), France, in 1992, and the Ph.D. degree in signal and image processing from Télécom Paris (ENST), France, in 1995. He was a Visiting Scholar at the Electrical Engineering and Computer Sciences Department, University of California, Berkeley, from 1995 to 1996. He was with the Department of Electrical and Computer Engineering, Villanova University, Villanova, PA, as a Research Associate from 1996 to 1997. He is currently and since 1998 with the Electrical Engineering Department of ENP as a Full Professor. His research interests are in statistical signal processing and (blind) array signal processing with applications in biomedical and communications, time-frequency analysis, time-frequency array signal processing, and wireless and spread spectrum communications.

