Optimization framework for large-scale sparse blind source separation

Christophe Kervazo

NNT : 2019SACLS354

Thèse de doctorat

Optimization framework for large-scale sparse blind source separation

Stratégies d'optimisation pour la séparation aveugle de sources parcimonieuses grande échelle

Thèse de doctorat de l'Université Paris-Saclay
préparée à Université Paris-Sud

Ecole doctorale n°580 Sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat : Traitement du signal et des images

Thèse présentée et soutenue à Saclay, le 04/10/2019, par

CHRISTOPHE KERVAZO

Composition du Jury :

Christian Jutten Professeur émérite, Institut Universitaire de France	Président
Nicolas Gillis Associate Professor, Université de Mons	Rapporteur
Nicolas Dobigeon Professeur, INP-ENSEEIHT Toulouse	Rapporteur
Émilie Chouzenoux Maître de conférence, Université Paris-Est Marne-La-Vallée	Examinateur
Pascal Larzabal Professeur, École Normale Supérieure	Examinateur
Jérôme Bobin Chercheur, CEA Saclay	Directeur de thèse
Florence Tupin Professeur, Télécom ParisTech	Invitée

Remerciements

Je souhaiterais tout d'abord remercier les membres du jury, ainsi que mes deux rapporteurs Nicolas Gillis et Nicolas Dobigeon, pour leur questions pertinentes et leur lecture attentive de ce manuscrit.

Mes remerciements vont ensuite à mon directeur de thèse, Jérôme Bobin, pour son aide précieuse et efficace, ainsi que pour les nombreuses discussions qui en ont découlées, me permettant notamment de mieux appréhender le monde de la recherche. Pour ses conseils avisés, je suis également reconnaissant à Florent Sureau, qui m'a permis d'approfondir de manière SURE certains aspects de ce doctorat.

J'ai également beaucoup apprécié le temps passé au sein du groupe CosmoStat. Je remercie son fondateur, Jean-Luc Starck, ainsi que tous ses membres. Je pense plus particulièrement à ceux avec qui j'ai partagé le bureau, avec (par ordre d'arrivée) : Cécile, Joana, Imane et Jiaxin. Parmi les membres de l'équipe, je souhaiterais également particulièrement remercier les deux stagiaires que j'ai eu le plaisir de co-encadrer, à savoir Tobias Liaudat et Yarui Zhang, pour leur contribution ainsi que l'apport pédagogique. Enfin, merci à tous ceux avec qui j'ai partagé encore davantage de bons moments : les membres du *Cheeseday* et les autres doctorants parmi autres.

J'aimerais également remercier les nombreux amis externes au laboratoire qui m'ont soutenu durant ces trois ans : que ce soit ceux d'école primaire, ceux du collège, du lycée, des classes préparatoires ou d'école d'ingénieur, leur amitié qui a su perdurer depuis compte beaucoup à mes yeux.

Enfin, j'aimerais dédier ce travail à ma famille et notamment à mes parents. Je pense que le contexte personnel qu'ils m'ont fourni tout du long de mon enfance a pour beaucoup joué dans la possibilité d'arriver jusqu'à une soutenance de doctorat.

« La science a-t-elle promis le bonheur ? Je ne le crois pas.
Elle a promis la vérité, et la question est de savoir si l'on fera jamais du bonheur avec la vérité »
Émile Zola

Contents

Notations and abbreviations

General notations

— u : a scalar. $u_{1..n}$ is used as a shortcut for writing the whole set of scalars u_1, u_2, ..., u_n ;

— $\mathbf{u}$: a vector ;

— $\mathbf{U}$: a matrix. Vectors are considered as a matrix with only one column ;

— U : a set ;

— $\mathbf{U}_i$: i$^{\text{th}}$ line of matrix $\mathbf{U}$ (when $\mathbf{U} = \mathbf{S}$ corresponds to the sources, also called an observation). The notation is extended to subsets of lines : if J is a finite subset of [1,n], $\mathbf{U}_J$ denotes the lines indexed by J ;

— $\mathbf{U}^j$: j$^{\text{th}}$ column of matrix $\mathbf{U}$ (when $\mathbf{U} = \mathbf{S}$ corresponds to the sources, also called a sample). The notation is extended to subsets of columns : if J is a subset of [1,t], $\mathbf{U}^J$ denotes the columns indexed by J ;

— $\mathbf{U}_{ij}$: (i,j)$^{\text{th}}$ entry of matrix $\mathbf{U}$ (also called coefficient) ;

— $\mathbf{U}^T$: transpose of $\mathbf{U}$;

— $\mathbf{U}^\dagger$: Moore-Penrose pseudo-inverse of $\mathbf{U}$;

— $\mathbf{U}^*$: true underlying matrix (to be estimated) ;

— $\tilde{\mathbf{U}}$: in proximal algorithms, estimate of a variable $\mathbf{U}^*$ before the application of the proximal operator ;

— $\hat{\mathbf{U}}$: estimate of $\mathbf{U}^*$ by an algorithm ;

— $\hat{\mathbf{U}}^{(l)}$: in iterative algorithms, denotes the estimation of $\mathbf{U}^*$ at iteration l ; $\hat{\mathbf{U}}^{(1..l)}$ is used as a shorthand for the set of all the estimates until iteration l.

— $\mathbf{U}[J]$: used to denote an estimation of $\mathbf{U}^*$ using the mini-batch indexed by J. Note that we do not use a hat here, as it is used for the estimate obtained from the aggregation of the various $\mathbf{U}[J]$;

— $f(.)$ function with scalar output ;

— $\mathbf{f}(.)$ function with matrix output ;

— $\mathbf{1}_{n\times t}$: matrix of size $n \times t$ filled with ones ;

— $\mathbf{Id}$: identity matrix ;

— J^C complementary set of J ;

— Diag$(\lambda_1, ..., \lambda_n)$: diagonal matrix having as diagonal elements $\lambda_1, ..., \lambda_n$.

Operators and norms

— $\|\mathbf{u}\|_{\ell_p}, p \in \mathbb{R}_+$: ℓ_p (quasi)-norm of $\mathbf{u}$, that is for $\mathbf{u} \in \mathbb{R}^u$, $\|\mathbf{u}\|_{\ell_p} = (\sum_{i=1}^{u} \mathbf{u}_i^p)^{\frac{1}{p}}$;

— $\|\mathbf{u}\|_0$: number of non-zeros elements in $\mathbf{u}$;

— $\|\mathbf{U}\|_F$: Frobenius norm of $\mathbf{U}$;

— $\|\mathbf{U}\|_p$ matrix norm of $\mathbf{U}$ induced by the ℓ_p-norm on vectors ;

— $\|\mathbf{U}\|_\infty$: maximum absolute value of the coefficients of $\mathbf{U}$. The notation is the same for vectors : $\|\mathbf{u}\|_\infty$;

— $\odot$: Hadamart product (e.g. elementwise product of two matrices) ;

— $\mathbf{U} \succeq 0$ means that all the coefficients of $\mathbf{U}$ are non-negative : $\mathbf{U}_{i,j} \geq 0$;

— $< .|. >$: scalar product ;

— $\Delta_f(.)$: gradient of f ;

— $\text{prox}_f(u)$: proximal operator of f in u. See Appendix A for definition and special case of soft-thresholding $\mathcal{S}_\lambda(u)$ and projection on set U, $\Pi_U(u)$;

— $M_f(.)$: Moreau envelope of f ;

— $\iota_U(.)$ is the characteristic function of a set U :

$$\forall u \in \mathbb{R}, \iota_U(u) = \begin{cases} 0 \texttt{ if } u \in U \\ \infty \texttt{ otherwise} \end{cases}$$

The notation is extended for matrices $\mathbf{U} \in \mathbb{R}^{m \times n} : \iota_U(\mathbf{U}) = \sum_{i=1}^{m} \sum_{j=1}^{n} \iota_U(\mathbf{U}_{ij})$.

— mad($\mathbf{u}$) : Median Absolute Deviation of $\mathbf{u}$. MAD($\mathbf{U}$) : vector, which elements are the mad of each line of $\mathbf{U}$;

— $\#U$: number of elements in set U ;

— $\exp_{\mathbf{u}}$ and $\log_{\mathbf{u}}$: exponential and logarithmic map used for the Fréchetmean ;

Specific naming

— $\mathbf{X} \in \mathbb{R}^{m \times t}$: dataset ;

— $\mathbf{A} \in \mathbb{R}^{m \times n}$: mixing matrix ;

— $\mathbf{S} \in \mathbb{R}^{n \times t}$: source matrix ;

— $\mathbf{N} \in \mathbb{R}^{m \times t}$: noise matrix ;

— $\mathbf{P}$: permutation matrix ;

— $\mathbf{W} = \mathbf{A}^\dagger$;

— $\mathbf{U}_{(1)}, \mathbf{U}_{(2)}, \mathbf{U}_{(3)}$... : some generic matrices ;

— $\Theta = \{\mathbf{A}, \mathbf{S}\}$

— n : number of sources ;

— m : number of observations ;

— t : number of samples ;

— T : number of samples in a transformed domain ;

— σ : a standard deviation ;

— h : differentiable term in a cost function (most of the time, h is also multi-convex) ;

— $h_{(i)}$: convex function corresponding to the multi-convex function h with all but one block fixed ;

— $\mathcal{J}(.)$, $\mathcal{G}(.)$ and $\mathcal{J}_{(i)}$: functions used to enforce constraints on $\mathbf{A}$ and $\mathbf{S}$ respectively, while $\mathcal{J}_{(i)}$ is used for the more general case of more than two $\mathbf{A}$ and $\mathbf{S}$ matrices. Examples of constraints : positive orthant $K^+ = \{\mathbf{S} \in \mathbb{R}^{n\times t}; \forall j \in [1,n], k \in [1,t], \mathbf{S}_j^k \geq 0\}$ and oblique constraint $\mathcal{O} = \{\mathbf{A} \in \mathbb{R}^{m\times n}; \forall j \in [1,n], \|\mathbf{A}^j\|_2^2 = 1\}$;

— K : number of constraints $\mathcal{J}_{(i)}$;

— $\mathbf{\Phi}$ or $\mathbf{\Phi_S}$: a sparsifying transform (size $T \times t$) ;

— $\mathbf{R_S}$ (size $n \times T$) control the trade-off between the data fidelity and the sparsity terms. It can be decomposed into $\mathbf{R_S} = \mathbf{\Lambda_S G}$ where $\mathbf{\Lambda_S}$ $(n \times n)$ is a diagonal matrix of the regularization parameters $\lambda_1, \lambda_2, ..., \lambda_n$ and $\mathbf{G}$ $(n \times T)$ is a matrix used to introduce individual penalization coefficients in the context of reweighted ℓ_1 ;

— $\mathbf{M_S}$: GMCA regularization parameters. $\mathbf{M_S}^{(l)} = \text{Diag}(\mu_1^{(l)}, \mu_2^{(l)}, ..., \mu_n^{(l)})\mathbf{1}_{n\times t}$;

— γ, δ : parameters in $(0,1)$ used for the step size ;

— η : gradient step size, or learning rate in the machine learning terminology ;

— $\mathcal{L}$: Lispchitz constant of a gradient ;

— $\omega_i, \forall i \in [1,B]$: weights : $\omega_i \geq 0$ and $\sum_{i=1}^{B} \omega_i = 1$. The ω_i can be written within a vector $\mathbf{w}$: $\mathbf{w}_i = \omega_i$;

— κ : constant used as a multiplicative factor of the mad. Usually, $\kappa = 3$;

— l : in iterative algorithms, number of the current iteration (l_f is specifically used for the Fréchetalgorithm) ;

— L : number of iterations of an iterative algorithm (L_f is specifically used for the Fréchetalgorithm) ;

— Δ : stopping criterion in iterative algorithms ;

— C_A : mixing matrix criterion (see Appendix B) ;

— C_{med}, C_{mean}, C_{angle} : metrics for separation quality. See Appendix B ;

— $\mathbf{s} = \mathbf{S}^* - \hat{\mathbf{S}}$: error on sources ;

— $\mathcal{E}$: error on the source estimation introduced by the use of blocks ;

— $\mathbf{R}$: residual ;

— ε : small constant (e.g. 10^{-3}) ;

— α : parameter of a function (i.e. of a generalized Gaussian distribution, sine or cosine, exponential decay...) ;

— σ_i : standard deviation of the noise in the source indexed by i ;

— $C_d(\mathbf{S})$: condition number of $\mathbf{S}^*$;

— C_d : condition number of the mixing matrix $\mathbf{A}^*$;

— p : sparsity level (in [0,1]) ;

— k : number of non-zeros coefficients ;

— r : block size ;

— B : number of mini-batches used ;

— $b \in [1, B]$: index of a mini-batch ;

— t_b : size of the mini-batch indexed by b ;

— J_b : indices of the columns in the mini-batch indexed by b ;

— $\phi(\mathbf{U}_{(1)}, \mathbf{U}_{(2)})$: distance between $\mathbf{U}_{(1)}$ and $\mathbf{U}_{(2)}$ (e.g. on a geodesic) ;

— $\mathcal{S}_m$: m-dimensional hypersphere ;

— ν : parameter of Nesterov smoothing technique ;

— ρ : step size used for Fréchetmean ;

— $\mathbf{f}^*$: non-linear mixing function ;

— $\mathbf{X}_u \in \mathbb{R}^{n \times t}$: unfolded manifolds ;

— $\tau \in \mathbb{R}_+$: a threshold ;

— $\mathbf{h}$: non-linear indeterminacy function appearing in non-linear BSS (note : in the context of linear BSS, $\mathbf{h}$ is a scaling factor) ;

— $\mathcal{P}$: polynomial function, supposed to be estimate $\mathbf{h}$;

Abreviations

— BSS : Blind Source Separation ;

— ICA : Independent Component Analysis ;

— NMF : Non-negative Matrix Factorization ;

— SMF : Sparse Matrix Factorization ;

— BCD : Block Coordinate Descent ;

— PALM : Proximal Alternating Linearized Minimization ;

— (p)ALS : (projected) Alternating Least-Squares ;

— PBC : Proximal Block Coordinate ;

— GD : Gradient Descent ;

— SGD : Stochastic Gradient Descent ;
— FBS : Forward Backward Splitting ;
— GFBS : Generalized Forward Backward Splitting ;

— PNL : Post Non-Linear ;
— LQ : Linear Quadratic ;

— GMCA : Generalized Morphological Component Analysis ;
— bGMCA : block Generalized Morphological Component Analysis ;
— dGMCA : distributed Generalized Morphological Component Analysis ;
— AMCA : Adaptative Morphological Component Analysis ;
— StackedAMCA : Stacked Adaptative Morphological Component Analysis ;

— PCA : Principal Component Analysis ;
— EFICA : Efficient FastICA ;
— HALS : Hierarchical Alternating Least Squares ;
— RNA : Relative Newton Algorithm ;
— MISEP : algorithm of [Almeida 2003] ;
— NFA : Nonlinear Factor Analysis ;
— ANICA : Adversarial Non-linear Independent Component Analysis ;

— SDR, SAR, SIR, SNR : Signal to Distortion-Artifact-Interference-Noise Ratio ;
— ME : Mean absolute Error ;
— MSE : Mean Square Error ;

— LC / MS : Liquid Chromatography Mass Spectroscopy ;
— LC / ^{1}H NMR : Liquid Chromatography - ^{1}H Nuclear Magnetic Resonance ;

— i.i.d. : Independent Identically Distributed
— 1/2/3/.../nD : 1/2/3/.../D-Dimensional ;

— ReLU : Rectified Linear Unit ;

Chapitre I

Résumé

La séparation aveugle de sources (BSS [1] – [Comon & Jutten 2010]) est une méthode de premier plan pour apprendre des décompositions physiques de données multi-valuées. Celle-ci a fait ses preuves dans de nombreux domaines, tels que par exemple le traitement du signal audio [Vincent *et al.* 2003, Vincent *et al.* 2011, Ozerov & Févotte 2010, Duong *et al.* 2010, Févotte *et al.* 2009], le biomédical [Jung *et al.* 2000, Negro *et al.* 2016, Poh *et al.* 2010] et l'astrophysique [Bobin *et al.* 2014].

Les jeux de données utilisés en BSS sont obtenus à partir de mélanges de signaux élémentaires appelés sources (*cf.* Fig. I.1 pour un exemple illustratif tiré de données astrophysiques). De manière générale, les observations, regroupées en tant que lignes d'une matrice $\mathbf{X}$ (de taille $m \times t$), peuvent donc s'écrire comme :

$$\mathbf{X} = \mathbf{f}^*(\mathbf{S}^*) + \mathbf{N}$$

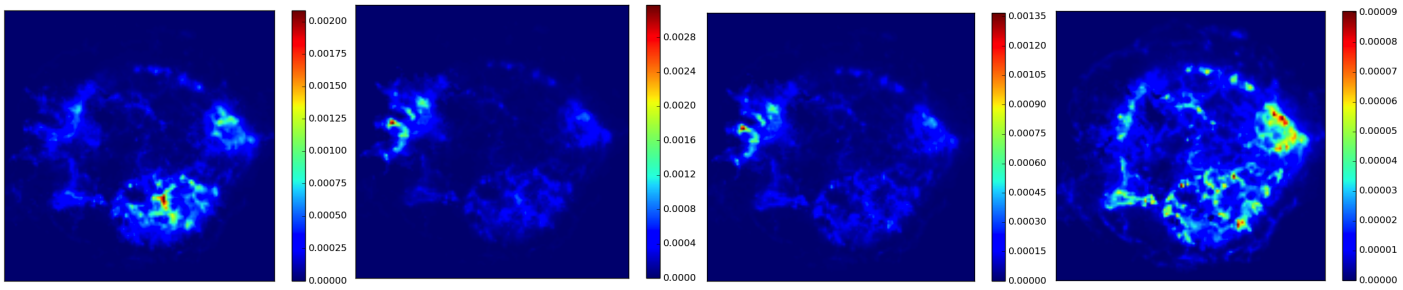
où les sources sont les lignes de la matrice $\mathbf{S}^*$ (de taille $n \times t$), qui sont mélangées par la fonction $\mathbf{f}$. La matrice $\mathbf{N}$ correspond aux imperfections du modèle et est appelée bruit. Une instance spécifique de ce type de problèmes est le cas où le mélange est *linéaire* :

$$\mathbf{X} = \mathbf{A}^*\mathbf{S}^* + \mathbf{N} \tag{I.1}$$

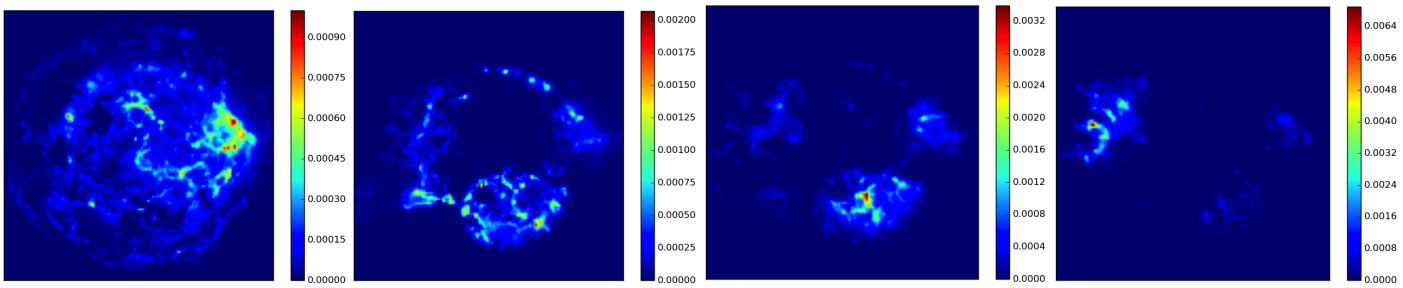
Où $\mathbf{A}^*$ (taille $m \times n$) est une matrice contenant les coefficients linéaires de mélange. De manière très simplifiée, l'objectif de la séparation de sources est de démélanger les sources $\mathbf{S}^*$ à l'origine des données (compte tenu de certaines indéterminations qui seront détaillées dans le reste du manuscrit). Ceci est d'autant plus délicat que la séparation aveugle ne suppose (quasiment) aucune connaissance a priori sur le processus de mélange $\mathbf{f}^*$, si ce n'est dans le cas où le mélange est supposé linéaire, et où celui-ci revient donc à une multiplication matricielle.

En l'état, le problème de BSS formulé comme précédemment admet une infinité de solutions, parmi lesquelles seul un petit nombre correspond aux signaux physiques, c'est à dire qui sont réellement à l'origine des données observées. Dit autrement, la BSS est un problème mal posé, pour lequel il est courant de rajouter une information a priori dans le but de restreindre l'espace des solutions possibles.

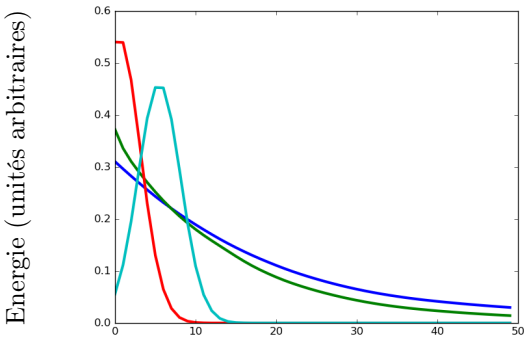
1. Les abréviations utilisées dans cette introduction correspondent, par soucis de cohérence avec le reste du manuscrit, aux acronymes anglais.



(a) $\mathbf{X}_1$ (b) $\mathbf{X}_5$ (c) $\mathbf{X}_9$ (d) $\mathbf{X}_{30}$



(e) Synchrotron (f) Thermique (g) Fer 1 (h) Fer 2



(i) $\mathbf{A}^*$

Figure I.1 – Exemple de problème de BSS en astrophysique, correspondant ici à l'étude d'un rémanent de supernova. *Haut :* Quelques données / mélanges observés par le satellite Chandra. Chaque image correspond au même rémanent, mais capturé à une longueur d'onde différente. Le mélange peut ici être considéré comme linéaire : chaque image est alors aplanie pour devenir une ligne de $\mathbf{X}^*$ dans (I.1). *Milieu :* sources physiques à l'origine des données / des mélanges observés : chaque image (inconnue en pratique et à retrouver par BSS) correspond à une émission spécifique provenant du rémanent. Ici, il s'agit de gauche à droite de : l'émission synchrotron, l'émission thermique et l'émission du fer, à deux décalages vers le rouge différent. Dans (I.1), chaque image est une ligne de $\mathbf{S}^*$; *Bas :* Chaque courbe est le spectre d'une des 4 émissions précédentes. Ceux-ci (chacun étant une colonne de $\mathbf{A}^*$) ne sont pas disjoints : ainsi, la prise d'images $\mathbf{X}$ à différentes longueurs d'ondes ne permet pas d'isoler chaque émission parfaitement, d'où le besoin d'une méthode de BSS.

Par exemple, les sources peuvent être supposées statistiquement indépendantes, ou avoir des coefficients positifs. Dans ce manuscrit, nous nous intéressons spécifiquement au cas où les sources sont *parcimonieuses*, puisque cette approche a permis d'obtenir d'excellents résultats lors de la dernière décennie [Zibulevsky & Pearlmutter 2001, Bobin *et al.* 2007, Bobin *et al.* 2015]. Pour résumer très succinctement, l'hypothèse de parcimonie présuppose que les sources comportent un grand nombre de coefficients nuls (potentiellement dans un espace transformé, par exemple le domaine de Fourier pour donner un cas simple).

En tant que tel, le problème de BSS parcimonieuse peut s'écrire comme un problème d'optimisation multi-convexe de la forme (nous prendrons ici le cas le plus simple) :

$$\underset{\mathbf{A}\in\mathbb{R}^{m\times n},\mathbf{S}\in\mathbb{R}^{n\times t}}{\operatorname{argmin}} \frac{1}{2}\left\|\mathbf{X}-\mathbf{AS}\right\|_F^2 + \left\|\mathbf{R_S}\odot\mathbf{S}\right\|_1 + \iota_{\{\forall i\in[1,n];\|\mathbf{A}^i\|_2^2=1\}}(\mathbf{A}) \tag{I.2}$$

Dans lequel :

- Le terme $\frac{1}{2}\left\|\mathbf{X}-\mathbf{AS}\right\|_F^2$, avec $\|.\|_F$ la norme de Forbenius, promeut une reconstruction fidèle des données.
- Le terme $\left\|\mathbf{R_S}\odot\mathbf{S}\right\|_1$ encourage la parcimonie des sources, où $\odot$ est utilisé pour le produit d'Hadamard. La matrice de paramètres $\mathbf{R_S}$ (de taille $n\times t$) permet de contrôler le compromis effectué entre les termes d'attache aux données et de parcimonie.
- Pour éviter des solutions $\mathbf{A}$ and $\mathbf{S}$ dégénérées dans lesquelles $\|\mathbf{A}\|_F\to\infty$ et $\|\mathbf{S}\|_F\to 0$ à cause du terme de parcimonie, un dernier terme appelé contrainte oblique est introduit. Il impose que toutes les colonnes de $\mathbf{A}$ se trouvent sur l'hypersphère unité ℓ_2. La fonction charactéristique est notée ι.

Les algorithmes classiques de BSS parcimonieuse s'attachent donc généralement à minimiser le problème (I.2), soit de manière exacte[2] (BCD [Tseng 2001] ou PALM [Bolte *et al.* 2014]), soit de manière approximative mais en introduisant des heuristiques[3] permettant d'augmenter empiriquement la robustesse grâce à un choix automatique des paramètres de régularisation $\mathbf{R_S}$ (GMCA [Bobin *et al.* 2007]).

Toutefois, en dépit de leurs nombreux succès, la plupart des méthodes ont été utilisées sur des problèmes de petites tailles. Par conséquent, le déluge de données actuel représente un important défi pour les méthodes de BSS actuelles. En astrophysique par exemple, de nouveaux instruments tels que le Square Kilometer Array

2. Exacte est à comprendre ici dans le sens où ces algorithmes garantissent de converger vers un point critique du problème (I.2). En revanche, ce point n'est pas nécessairement un minimum global du problème.

3. Par heuristique, nous entendons ici une méthode approximative simplifiant un des aspects du problème initial (réduction du temps de calcul, choix d'hyper-paramètres ou d'initialisation...), et permettant d'obtenir des résultats corrects mais non nécessairement optimaux.

(SKA) seront à même de fournir des volumes de données colossaux. Dans le contexte de la BSS, et pour donner un ordre d'idée des tailles à considérer, des données comprenant jusqu'à $t = 10^9$ échantillons et $m = 10^4$ observations pourraient faire leur apparition. De manière similaire, en spectroscopie les mélanges peuvent comporter plusieurs dizaines de sources. **L'objectif de ce doctorat est donc de proposer de nouvelles méthodes de séparation aveugle de sources parcimonieuses permettant de traiter des problèmes** *grande échelle***.**

Plus précisément, les travaux s'articulent autour de quatre problématiques majeures :

1 - **L'introduction d'une méthode de choix automatique des paramètres de régularisation $\mathbf{R_S}$ du problème (I.2) lors de sa minimisation par l'algorithme PALM [Bolte *et al.* 2014].** Cette problématique est primordiale, puisque qu'un tel choix est généralement effectué en testant différentes valeurs de $\mathbf{R_S}$ et en regardant la qualité de la solution obtenue par PALM pour chaque jeu de paramètres (approche de type *grid-search* en Anglais). Néanmoins, dans le cas grande échelle, les temps de calculs prohibitifs induits par cette approche sont souvent rédhibitoires, nécessitant l'introduction d'une méthode automatique permettant de ne lancer qu'une unique fois l'algorithme ;

2 - **L'introduction d'une méthode permettant de traiter un nombre de sources n important.** Le défi présente ici deux facettes, puisque les algorithmes classiques de BSS, lorsqu'ils sont confrontés à un nombre de sources croissant i) nécessitent des temps de calcul accrus ; ii) voient la qualité des solutions qu'ils proposent décliner ;

3 - **L'introduction d'une méthode permettant de traiter des jeux de données de grandes tailles, et en particulier présentant un grand nombre d'échantillons t.** Le problème a ici principalement trait à aux difficultés calculatoires, car un grand nombre d'échantillons i) provoque un temps de calcul fortement accru, rendant très couteuses des méthodes telles GMCA [Bobin *et al.* 2007] puisque celles-ci nécessitent, à cause de leurs structures itératives, de multiples inversions de matrices gigantesques ; ii) peut impliquer des tailles de données ne permettant même pas de stocker les matrices en jeu en mémoire, prohibant l'utilisation des méthodes de BSS ;

4 - **Une extension à la séparation aveugle de sources non-linéaire.** Peu de travaux ont été proposés en BSS non-linéaire parcimonieuse. La méthode introduite durant ce doctorat nécessite de résoudre plusieurs sous-problèmes de BSS linéaire, ce qui peut dans des cas complexes potentiellement nécessiter de résoudre auparavant les points précédents.

Après une introduction plus détaillée du problème et une présentation des stratégies d'optimisation utilisées dans les chapitres II et III respectivement, le chapitre IV s'attaque à la résolution du problème 1). Plus spécifiquement, la première partie consiste à déterminer les difficultés induites par l'utilisation de PALM dans le cadre de la BSS parcimonieuse. Une étude empirique nous permet d'avancer que dans ce contexte, utiliser PALM avec une approche de type *grid-search* pour la recherche de bons [4] paramètres de régularisation $\mathbf{R_S}$ souffre d'une faible *efficacité* et *versatilité*. Pour résumer succinctement, une faible efficacité signifie qu'il est délicat, pour une expérience donnée, de trouver de bons paramètres de régularisation $\mathbf{R_S}$. Une faible versatilité implique qu'il est difficile de généraliser un bon choix de $\mathbf{R_S}$ d'un jeu de données à un autre. Ainsi, faibles efficacité et versatilité rendent le choix des paramètres de régularisation très complexes lors de l'utilisation de PALM pour des problèmes grande échelle [5]. Cependant, l'étude proposée montre qu'une fois de bons paramètres $\mathbf{R_S}$ trouvés, l'estimation des matrices $\mathbf{A}^*$ et $\mathbf{S}^*$ par PALM peut être de très bonne qualité, surpassant celle d'algorithmes heuristiques tels que GMCA.
C'est pourquoi nous proposons, dans une deuxième partie, une méthode permettant de contourner les difficultés précédentes. Plus spécifiquement, une approche en deux étapes est introduite, comprenant :

— Une étape d'initialisation se basant sur l'algorithme GMCA : celui-ci, bien qu'approximatif car basé sur des moindres carrés alternés projetés (pALS), propose une méthode automatique de choix de $\mathbf{R_S}$, qui donne en pratique une première estimation décente $\hat{\mathbf{A}}$ et $\hat{\mathbf{S}}$;

— Une étape de raffinement se basant sur PALM. L'initialisation correcte fournie par GMCA permet de déterminer un bon jeu de paramètres $\mathbf{R_S}$, utilisé dans cette deuxième étape. PALM permet alors potentiellement d'améliorer la solution fournie par GMCA, tout en fournissant des garanties mathématiques, telle la convergence de l'algorithme.

La qualité de l'approche est démontrée grâce à une expérience de BSS sur des données réalistes d'astrophysique. Une discussion est également proposée quant aux limitations de la méthode.

4. Le terme "bon" est à comprendre au sens donnant des estimations $\hat{\mathbf{A}}$ et $\hat{\mathbf{S}}$ proches des vrais facteurs physiques $\mathbf{A}^*$ et $\mathbf{S}^*$. Cette notion est donc distincte de simplement trouver un minimum local du problème (I.2) : il s'agit ici de trouver un minimum particulier correspondant à des facteurs ayant un sens *physique*.

5. Et ce d'autant plus que PALM souffre en plus d'une *fiabilité* limitée, signifiant sa sensibilité à l'initialisation : autrement dit, il faut également potentiellement relancer plusieurs fois l'algorithme avec des initialisations différentes jusqu'à l'obtention de résultats corrects.

Dans le chapitre V, nous nous attaquons au problème 2). Nous montrons sur une expérience simple la difficulté de traiter des cas de BSS comprenant un grand nombre de sources n. Pour traiter ce problème, la solution proposée se concentre sur la stratégie d'optimisation. Plus spécifiquement, nous utilisons une méthode de minimisation par blocs de coordonnées, ce qui est couramment utilisé en BSS pour contourner l'aspect non-convexe de la fonction de coût (I.2) en utilisant sa structure multi-convexe. Les approches classiques peuvent être catégorisées en deux familles :

— Les méthodes de *déflations* ou *hiérarchiques* : celles-ci utilisent des blocs de taille 1 : à chaque itération, une seule source est mise à jour, ainsi que la colonne de $\mathbf{A}$ correspondante ;

— Les méthodes utilisant *l'intégralité* des matrices $\mathbf{A}$ et $\mathbf{S}$. Dans ces méthodes, telles le GMCA usuel, seulement deux blocs sont utilisés, le premier correspondant à la matrice de mélange et le deuxième à la matrice source.

Par opposition, la méthode proposée, appelée block-GMCA (bGMCA), introduit des blocs de tailles *intermédiaires* r au sein de l'algorithme GMCA [6]. Ainsi, alors que les approches par déflation ou hiérarchique correspondaient au cas $r = 1$, et les méthodes utilisant l'intégralité des matrices au cas $r = n$, bGMCA utilise des tailles $r \in [1, n]$.
Outre un gain substantiel en termes de temps de calcul, il est montré expérimentalement que l'algorithme permet une forte amélioration de la qualité de la séparation pour des tailles de blocs modérées. Dans des cas simples, bGMCA permet aussi de retrouver de manière quasi-exacte (aux incertitudes numériques près) les facteurs $\mathbf{A}^*$ et $\mathbf{S}^*$. L'explication avancée pour ces résultats est qu'il existe un compromis dans la taille des blocs : pour r proche de 1, l'algorithme souffre d'erreurs de propagation entre les itérations et le fait d'utiliser des blocs crée des erreurs, assimilables à un bruit supplémentaire. Pour $r = n$, le problème de séparation devient plus complexe. Prendre des valeurs de r intermédiaires semble donc permettre de limiter les erreurs de propagation, tout en bénéficiant d'un problème plus simple.
Pour conclure ce chapitre et montrer que bGMCA peut être utilisé dans le cadre de la BSS avec des a priori plus complexes que la seule parcimonie, un problème réaliste de LC / ^{1}H NMR est proposé, dans lequel la positivité des sources et de la matrice de mélange est de surcroit imposée.

Dans le chapitre VI, le problème 3) est pris en considération. Plus spécifiquement, bGMCA permettait de traiter des problèmes avec un grand nombre de sources mais nécessitait l'utilisation de l'ensemble de la matrice de données $\mathbf{X}$ à chaque itération. L'algorithme proposé ici, nommé distributed-GMCA (dGMCA), élimine cette

6. Plus exactement, des blocs sont introduits dans l'algorithme en deux étapes décrit précédemment.

limitation. Le principe général consiste en l'introduction de B mini-batchs [Xing *et al.* 2018] dans le schéma de minimisation par projections alternées des moindres carrés, et plus spécifiquement dans GMCA, ce qui permet de continuer à bénéficier de la grande robustesse de cet algorithme, ainsi que de son choix simple de $\mathbf{R_S}$ [7]. Cependant, l'utilisation du pALS implique qu'à chaque itération, un nombre B d'estimations de la matrice de mélange $\mathbf{A}^*$ soit calculé, soit une par mini-batch. Ces estimations ne sont pas toutes d'égales qualités, puisque réalisées sur des parties différentes, de petite taille, des données $\mathbf{X}$, qui peuvent être plus ou moins simples à démélanger. Une question naturelle est donc comment les *agréger*, pour obtenir à chaque itération une unique bonne estimation de $\mathbf{A}^*$. Une méthode naturelle serait de réaliser l'agrégation en effectuant une simple moyenne euclidienne. Cependant, l'estimée de $\mathbf{A}^*$ doit également respecter la contrainte oblique présente dans (I.2), ce qui n'est pas garanti par l'utilisation de la moyenne euclidienne. Nous proposons donc plutôt d'utiliser une moyenne de Fréchet, c'est à dire une moyenne sur l'hypersphère. Pour robustifier l'approche, une deuxième version de l'agrégation est proposée, similaire à une médiane sur l'hypersphère. De plus, une pondération est utilisée, permettant de prendre en compte la qualité de l'estimation des sources dans chaque mini-batch. Les expériences réalisées permettent de démontrer le gain de temps (en plus du gain en mémoire, puisque $\mathbf{X}$ est potentiellement découpé sur différents nœuds d'un cluster) obtenu par la méthode. L'étude de la qualité de la séparation permet par ailleurs de distinguer deux régimes :

— Dans le cas de sources modérément parcimonieuses, la séparation obtenue lors de l'utilisation de mini-batch est quasiment identique à celle en utilisant l'ensemble des données, si tant est qu'une taille de mini-batch raisonnable soit utilisée. Introduire des mini-batchs permet donc, à coût quasiment nul en termes de qualité de séparation, un gain en temps de calcul important ;

— Dans le cas de sources très parcimonieuses, les résultats sont plus surprenants : pour des mini-batchs de faibles tailles, la séparation est même meilleure que lorsque l'ensemble des données est utilisée. L'explication proposée se base sur des liens avec certains travaux récents sur la descente de gradient stochastique (SGD). Notre hypothèse est que, similairement à la SGD, l'introduction de mini-batch favorise certains minima du paysage d'optimisation de la fonction de coût (I.2) qui généralisent bien. Dans le contexte de la BSS, de tels minima correspondraient à des solutions peu sensibles à une réalisation donnée des sources.

Ces résultats sont confirmés sur une expérience réaliste en spectroscopie gamma.

Le chapitre VII est une extension des travaux précédents, et s'attaque au pro-

7. En effet, bien que le chapitre IV ait permis de trouver une méthode pour fixer les paramètres de régularisation dans PALM, il est à noter que l'approche nécessite au moins quelques itérations de GMCA pour initialiser l'étape de raffinement. Par conséquent, faire fonctionner le pALS dans le cas grande échelle reste nécessaire.

blème 4). Peu d'études se sont intéressées à la séparation aveugle de sources *non-linéaire*. Ceci est probablement dû à des indéterminations beaucoup plus importantes que dans le cas linéaire. Notamment, l'indépendance des sources n'est par exemple plus suffisante pour garantir leur séparation. Qui plus est, même dans le cas où les sources seraient séparées, le reconstruction n'est pas garantie puisque celles-ci sont retrouvées à une fonction non-linéaire $\mathbf{h}$ près, qui ne les remélange pas. L'algorithme proposé, nommé StackedAMCA, bénéficie d'une interprétation géométrique du problème de BSS non-linéaire : graphiquement, chaque source est transformée en une variété de dimension 1 (1D). Nous proposons d'estimer les non-linéarités en ajustant un modèle linéaire par morceaux à ces variétés grâce à un algorithme itératif. A chaque itération, un nouveau morceau est estimé grâce à un algorithme de BSS parcimonieuse robuste aux non-linéarités de grandes amplitudes [Bobin *et al.* 2015]. Pour passer d'un morceau à l'autre, et ainsi préparer l'itération suivante de l'algorithme, un résidu est utilisé, correspondant aux données $\mathbf{X}$ desquelles sont soustraites les contributions des modèles linéaires déjà estimés. Pour limiter les erreurs de propagation, à chaque itération l'algorithme repart des données originales $\mathbf{X}$ et déploie les variétés en appliquant l'ensemble des inverses des modèles linéaires calculés auparavant. L'algorithme admet de plus une interprétation en termes de réseau de neurones, qui est détaillée.
La partie expérimentale comporte plusieurs expériences : un mélange linéaire par morceaux, permettant ainsi d'étudier les mécanismes de StackedAMCA dans un cas où le processus de démélange correspond exactement à celui du mélange ; un mélange compliqué dans lesquelles sont présentes de nombreuses sources : StackedAMCA obtient une meilleure qualité de séparation que les autres méthodes de l'état de l'art. De manière intéressante, la méthode est aussi ici capable de reconstruire les sources, malgré l'indétermination par la fonction $\mathbf{h}$: dit autrement, la structure de l'algorithme crée une régularisation implicite qui permet la reconstruction. Nous proposons une dernière expérience dans laquelle la reconstruction n'est plus garantie, dans l'objectif de montrer que l'algorithme sépare encore toutefois bien les sources. Enfin, dans une dernière partie, les hypothèses requises pour StackedAMCA sont étudiées, et en particulier un sous-ensemble de mélanges pour lesquels l'algorithme est supposé être capable de reconstruire les sources est caractérisé.

Chapitre II

Sparse Modelling and Large-Scale BSS

A Multi-valued data analysis and BSS

The overwhelming quantities of data collected everyday have lead to the current so-called deluge of data. Such a phenomenon finds its roots in many origins : one can for instance cite the development of social media and search engines, or new industrial equipment, cameras, sensors... As a specific example, in astronomy instruments such as the Large-Synoptic Survey Telescope (LSST – [Ivezic *et al.* 2008]) or the Square Kilometer Array (SKA – [Blake *et al.* 2004]), to only name two of them, will produce data of several terabytes and even petabytes of memory [Longo *et al.* 2017].
Such unprecedented quantities require to develop scalable data processing tools being able to operate in the *large-scale* regime. In this work, we will focus on a specific method called Blind Source Separation (BSS), which is a key analysis tool to learn meaningful decompositions of multivalued data and which has already been successful in a wide variety of scientific fields such as audio processing [Vincent *et al.* 2011, Vincent *et al.* 2003, Ozerov & Févotte 2010, Duong *et al.* 2010, Févotte *et al.* 2009], biomedical data processing [Jung *et al.* 2000, Negro *et al.* 2016, Poh *et al.* 2010] or astrophysics [Bobin *et al.* 2014], to only cite three of them. More specifically, **the objective of this thesis is to extend a sub-domain of BSS, namely sparse BSS, to various large-scale contexts.**
In the next section, we start by giving a few examples of practical BSS problems, which will be followed by a formalization of the problem. Finally, the last part of this chapter will introduce the problems faced in the large-scale setting and the structure of this work.

A.1 Four BSS examples

A.1.1 Cocktail party problem

The most well-known BSS example might be the cocktail party problem. During a party, n persons are speaking with each other. A given number m of microphones are located in the room, registering the conversations. As the guests are speaking at the same time, they create a brouhaha and the signals $\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_m$ are corresponding to different mixings of their original sentences $\mathbf{S}_1^*, \mathbf{S}_2^*, ..., \mathbf{S}_n^*$. The signals captured by each microphone are different, depending on several physical effects

(such as for instance, the distance of each guest to a given microphone) which are not necessarily known beforehand. As such, the mixing process leading to the mixings $\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_m$ is supposed to be unknown. The challenge of BSS is then to recover the sounds pronounced by *each* individual from the mixings in a blind fashion : from $\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_m$ only, we want to retrieve $\mathbf{S}_1^*, \mathbf{S}_2^*, ..., \mathbf{S}_n^*$.

A.1.2 Spectroscopy : LC/MS data

The Liquid Chromatography – Mass Spectrometry (LC/MS) [Rapin *et al.* 2014] enables to study a fluid in order to identify and quantify its constitutive chemicals. This fluid could correspond to a drink and the goal of BSS would then be to identify the spectra $\mathbf{S}_1^*, \mathbf{S}_2^*, ..., \mathbf{S}_n^*$ of each chemical (*e.g.* caffeine, sucrose, menthone...) from the LC/MS data $\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_m$. Furthermore, if the mixing is supposed to be linear, the mixing coefficients (which are proportional to their concentrations) can also be of interest.
The LC/MS data are collected within a matrix $\mathbf{X} \in \mathbb{R}^{m \times t}$ in the following way :

— A first physical imperfect separation is performed, during which the fluid goes through a chromatography column and its chemicals are separated according to their speeds (which themselves depend on their physical properties). At each time moment $1/f_e$ (determined beforehand by a given rate f_e), the output of the chromatography column is then sampled, giving a first imperfect time separation.

— At each time $1/f_e$, the samples are analysed by a mass spectrometer. Since the different chemicals have different masses, this yields a second imperfect separation in mass.

This double imperfect separation both in time and in mass gives data similar to the ones of Fig. II.1 : the time axis corresponds to the different columns of $\mathbf{X}$, while the mass to charge ratio corresponds to the rows of $\mathbf{X}$. It has to be emphasized that at this point, due to the imperfectness of the separations, the chemical are still mixed within the $\mathbf{X}$ matrix. The goal of BSS is then to enhance the separation to precisely recover the spectra of the different chemicals as rows $\mathbf{S}_1^*, \mathbf{S}_2^*, ..., \mathbf{S}_n^*$ of a matrix $\mathbf{S}^*$ (*cf.* Fig. II.2a). This enables to identify the chemicals.

A.1.3 Astronomical data : Chandra and Square Kilometer Array

We will develop two examples of astrophysical data :

— *Chandra*
Chandra satellite [1] is a X-ray observatory taking pictures of the sky in different wavelength bands. Such observed images, picturing a supernovae remnant, are displayed in the upper row of Figure II.3. Each of them corresponds to a

1. http ://chandra.harvard.edu/

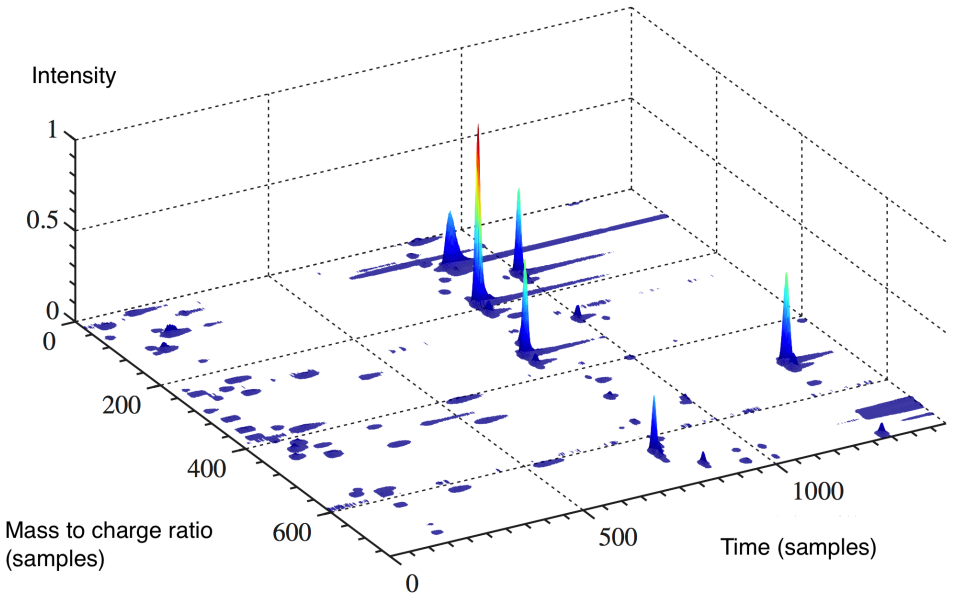
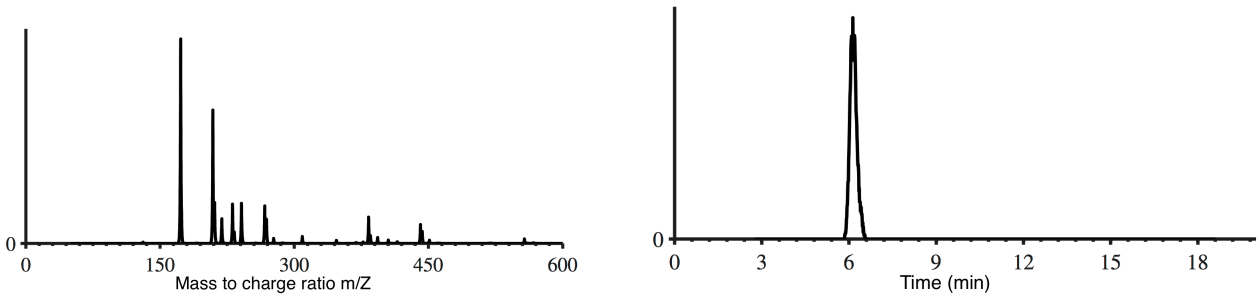


Figure II.1 – Example of LC/MS data. Figure taken from [Rapin 2014].

picture of the *same* supernovae remnant, but taken at different wavelength values. However, the different observations $\mathbf{X}_i, i \in [1, m]$ do not result each from separated physical emissions, but rather from a mixture of the elementary emissions $\mathbf{S}_i^*, i \in [1, 4]$ displayed in the middle row of Fig. II.3 (the synchrotron and thermal emissions, as well as the one originating from the iron present in the remnant, at two different redshift values). Indeed, all these emissions overlap in the wavelength domain (*cf.* lowest plot of Fig. II.3, in which it can be seen that several sources emit around 1, 5, 9 and 30), making that merely taking pictures at different wavelength values does not enable to directly separate them well.
Resorting to BSS enables to estimate the elementary emissions $\mathbf{S}_i^*, i \in [1, 4]$, which in turns allows to study the physical mechanisms at stake in the supernovae remnant. Furthermore, BSS methods can also estimate the corresponding spectra $\mathbf{A}_i^*, i \in [1, 4]$.

— *SKA*
The SKA is a continental-size radio-telescope that should be built in Australia and South America. It has several objectives, such as testing the general relativity, studying the large-scale structure of the cosmos, the epoch of reionization...
The SKA will be the world's largest telescope, eventually comprehending thousands of dishes and up to a million low-frequency antennas [2]. However, due to the wide range of frequencies in which it will operate and its size (approximately one square kilometer of collecting area), the quantities of data will

2. https ://www.skatelescope.org/the-ska-project/



(a) One line of $\mathbf{S}^*$ (b) One column of $\mathbf{A}^*$

Figure II.2 – Example of BSS problem on the LC/MS data : *Left :* spectrum of the DL-arginine ; *Right :* time elution of the DL-arginine. Figures taken from [Rapin 2014].

be tremendous : it could indeed produce more data per day than the entire content of internet in 2011 [3].
Therefore, while the faced BSS problems are similar in principle to the ones of the previous Chandra example, the datasets are much larger, calling for corresponding adequate BSS methods that will enable to draw as much information as possible from the data. As an example, datasets with up to $m = 10000$ wavelength bands and up to $t = 10^9$ pixels will potentially have to be considered.

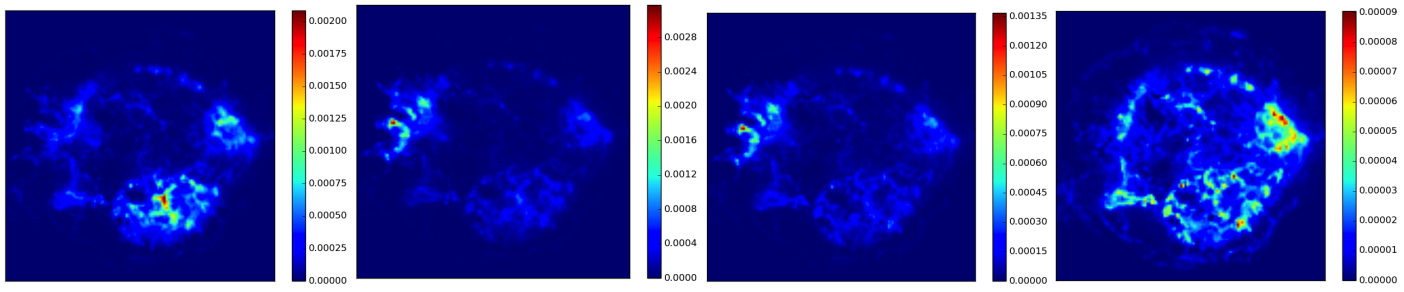
A.1.4 Show-through removal

The show-through effect appears during the scanning process of documents. The issue at hand is that the backside of the document is visible on the front side, for instance due to a too high transparency of the paper [Merrikh-Bayat *et al.* 2011]. Thus, when reading the content of one side of the paper, the reader also sees the writing on the back.
This can be written has a BSS problem with two observations $\mathbf{X}_1$ and $\mathbf{X}_2$, each corresponding to the scan of one side of the paper on which the show-through effect is visible, and two sources $\mathbf{S}_1^*$ and $\mathbf{S}_2^*$, each corresponding to an image of what is actually written on each side of the paper. The goal of BSS is to retrieve the sources, that is to remove the contribution of the other side of the paper that deteriorates the reading. See Figure II.4 for a concrete example of show-through.

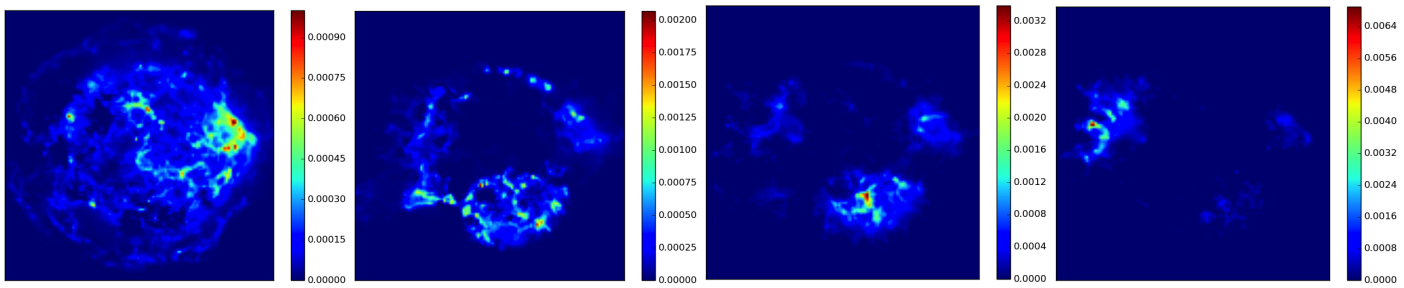
A.2 Mixing model

It is now time to introduce slightly more mathematically the BSS problem, by focusing on two different possible kinds of mixings : the linear and the non-linear models.

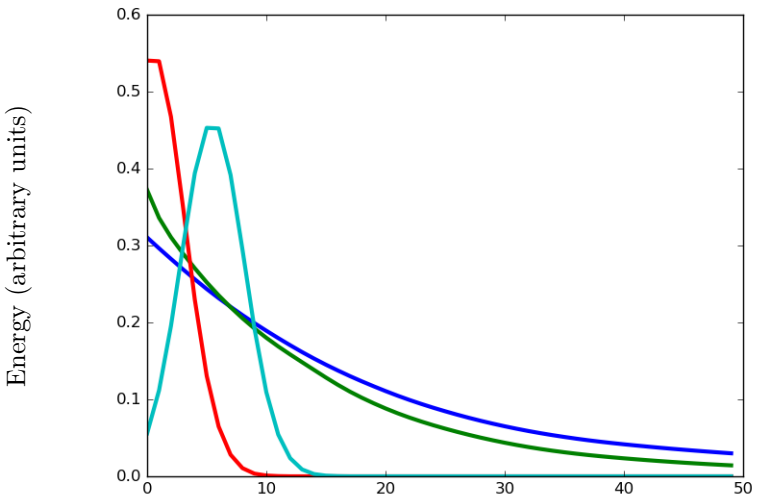
3. https ://www.computerworld.com.au/article/392735/ska_telescope_generate_more_data_than_entire_internet_2020/



(a) $\mathbf{X}_1$ (b) $\mathbf{X}_5$ (c) $\mathbf{X}_9$ (d) $\mathbf{X}_{30}$

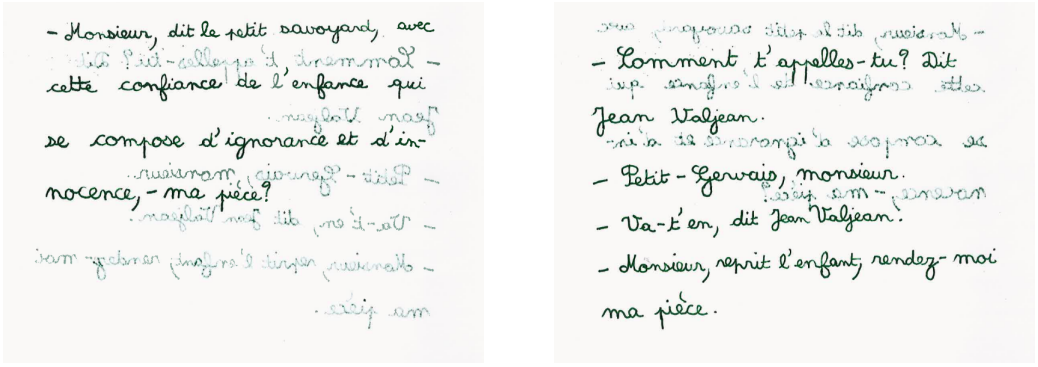


(e) Synchrotron (f) Thermal (g) Fe 1 (h) Fe 2



(i) $\mathbf{A}^*$

Figure II.3 – Example of BSS with the Chandra satellite : *Up :* some observed data (each flattened image of 128 x 128 pixels corresponds to one row of $\mathbf{X}$) ; *Middle :* true physical sources (each flattened image corresponds to one row of $\mathbf{S}^*$), corresponding to several kinds of emissions ; *Down :* true mixing matrix (each curve corresponds to the spectrum of an emission and is a column of $\mathbf{A}^*$)



(a) (b)

Figure II.4 – Example of show-through effect. Text from *Les Misérables*, Victor Hugo. *Left :* Recto ; *Right :* Verso.

A.2.1 Linear model

The linear model is largely the most widespread one and has led to a large variety of works [Comon & Jutten 2010]. In these, the m observations $\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_m$ are supposed to be the linear combinations of n sources $\mathbf{S}_1^*, \mathbf{S}_2^*, ..., \mathbf{S}_n^*$, each of them having t samples :

$$\forall i \in [1, m], \mathbf{X}_i = \sum_{j=1}^{n} a_{ij}^* \mathbf{S}_j^* \tag{II.1}$$

The linear BSS model can be written in matrix form : the observations $\mathbf{X}_i \in \mathbb{R}^t$ (with $i \in [1, m]$) are grouped as rows of a matrix $\mathbf{X}$ (of size $m \times t$) and similarly the sources are stacked as a matrix $\mathbf{S}^*$ (of size $n \times t$). The model then becomes :

$$\mathbf{X} = \mathbf{A}^* \mathbf{S}^* + \mathbf{N} \tag{II.2}$$

with $\mathbf{A}^*$ (of size $m \times n$) the mixing matrix containing the linear mixing coefficients a_{ij}^*. The matrix $\mathbf{N}$ (of size $m \times t$) enables to take into account some slight deviations from the ideal linear model, such as the ones induced for instance by some noise in the measurements. Thus, $\mathbf{N}$ is called the noise matrix. In the current work, we will further focus exclusively on the over-determined case [4] in which $n \leq m$.
The objective of linear source separation is to retrieve from the sole knowledge of $\mathbf{X}$ the sources $\mathbf{S}^*$ (up to limited indeterminacies, *cf.* below), as well as in the present *blind* setting the linear mixing coefficients $\mathbf{A}^*$. As such, BSS is a matrix factorization problem in which we aim to recover matrices having *physical* meanings, in contrast for instance to dictionary learning [Mairal *et al.* 2014].
Written in this form, BSS is however an ill-posed problem as we are looking for

4. While the under-determined setting in which there are more sources than observations is interesting, it leads to challenges that are beyond our scope. Nevertheless, in future work, Chapter V could be of interest for such a setting.

a small subset of physical solutions among the infinity of possible ones. Therefore, several families of algorithms have emerged, introducing some regularization. Each of them imposes a different kind of prior knowledge on the sources, aiming to help to discriminate the desired solutions. One can cite the Independent Component Analysis (ICA – [Comon & Jutten 2010]), the Non-negative Matrix Factorization (NMF – [Gillis & Glineur 2012]) and the Sparse Matrix Factorization (SMF – [Zibulevsky 2003]) families. Each one will be detailed in section A.3.
A natural question is then the identifiability issue : using the previous priors on the sources, can we really hope to recover them ? Starting with the ICA methods, such questions have been well studied in the past and it has been for instance shown that the independence of the sources – in the absence of noise and Gaussian sources – enables to recover them *up to a mere scaling and permutation indeterminacy* (Darmois theorem [Darmois 1953]). Concerning the sparsity prior, conditions for recovery up to the same indeterminacies are studied in [Gribonval & Schnass 2010, Gribonval *et al.* 2015]. In the last work, the authors furthermore studied the non-asymptotical noisy setting in the presence of outliers, with potentially over-complete dictionaries (*i.e.* mixing matrices in our setting). In brief, they show that $\mathbf{A}^*$ can be recovered using the cost function II.8, which is shown to have a local minimum around $\mathbf{A}^*$ with high probability. Among others, some of the required hypotheses are that $\mathbf{S}^*$ must be sparse enough (and follow other assumptions), $\mathbf{A}^*$ must be sufficiently incoherent (depending on $\mathbf{S}^*$ sparsity level), the noise level as well as the outlier energy are limited, and the number of non-outlier samples large enough (there are also assumptions on the regularization parameters). Note that these conditions might however be slightly restrictive in realistic experiments, and might not thus be always respected during this work.

A.2.2 Non-linear model

While convenient for many problems, the linear mixing model is only an approximation which might not always hold : it is not anymore valid when using sensors with saturations or non-linearities (for instance gas [Madrolle *et al.* 2018] or chemical [Jimenez 2006, Duarte & Jutten 2014] sensors), or in some specific applications (show-through removal [Merrikh-Bayat *et al.* 2011], hyperspectral imaging [Dobigeon *et al.* 2014]). It can therefore be relevant to change the BSS model to a *non-linear* one :

$$\mathbf{X} = \mathbf{f}^*(\mathbf{S}^*) + \mathbf{N} \tag{II.3}$$

Where $\mathbf{f}^*$ is an unknown *non-linear* function from $\mathbb{R}^{n\times t}$ to $\mathbb{R}^{m\times t}$ (where again here $n \leq m$). In this work, we will consider general functions $\mathbf{f}^*$, by mostly (*cf.* Sec. E) assuming that $\mathbf{f}^*$ is invertible and symmetrical around the origin, as well as regular enough. More detailed hypotheses, both on the mixing and the sources, are discussed in Chapter. VII.
At this point, it is important to mention that non-linear BSS is much more difficult than its linear counterpart and that it might not be possible to find both $\mathbf{f}^*$ and $\mathbf{S}^*$ up to a simple permutation and scaling indeterminacy. Therefore, and in contrast

to usual linear BSS, the *separation* of the sources must be distinguished from their *reconstruction* :

— The sources $\mathbf{S}$ are said to be well *separated* if they are estimated up to a permutation and an unknown non-linear function $\mathbf{h}$ that does not remix them ;

— The sources $\mathbf{S}$ are said to be well *reconstructed* if they are estimated up to a permutation and an unknown scaling factor (thus, $\mathbf{h}$ is here specifically a scaling function).

It is interesting to note that in ICA, and in contrast to the linear case, the independence of the sources is not anymore sufficient to guarantee their separation. In the case of sparse sources, [Ehsandoust *et al.* 2016] claimed the possibility to separate the sources if only one source is active for each sample. Nevertheless, sparsity does not guarantee the reconstruction.
Thus, the goal of sparse non-linear BSS is only to separate the sources by estimating the underlying non-linearities, but the source reconstruction is generally not straightforward as the problem is too ill-posed.

A.3 ICA, NMF and SMF

As said above, BSS requires tackling an ill-posed unsupervised matrix factorization problem. We here detail three additional priors (corresponding ICA, NMF, SMF) that have been introduced for enabling the identification of the mixture parameters. The discussion will be restricted to the most classical linear mixing model, the non-linear one being specifically addressed in Chapter VII. All the remainder of this work will then focus on the sparsity of the sources, that already lead to enhanced separation quality in various matrix factorization problems [Bobin *et al.* 2008, Zibulevsky & Pearlmutter 2001, Li *et al.* 2006, Le Roux *et al.* 2015].

A.3.1 ICA

The Independent Component Analysis is probably the family of BSS algorithms for which the most extensive literature exists. It is a statistical approach that assumes the sources to be independently identically distributed and that there is at most one non-Gaussian source. Under these assumptions, it can be shown that the sources can be recovered up to a mere permutation and scaling indeterminacy [Darmois 1953, Comon 1994] provided that the mixing matrix is invertible. The main idea is that mixing the independent sources will lead towards non independent signals, since the mixtures share the same source signals. An extensive review can be found in [Comon & Jutten 2010].
Typical ICA algorithms uses whitening and dimensionality reduction as preprocessing in order to simplify the problem. This can be for instance achieved using Principal Component Analysis (PCA – [Hotelling 1933, Eckart & Young 1936, Jolliffe 1986]), which enables to come back to the case $m = n$ and to restrict the search of the mixing matrix to the group of unitary matrices – while potentially reducing the noise impact. Then, the problem boils down to finding a demixing (rotation)

matrix $\mathbf{W}$ such that the estimated sources $\hat{\mathbf{S}} = \mathbf{WX}$ are independent. However, enforcing directly independence is not a trivial issue.
As such, several proxys have emerged. Among them, one can cite independence measures based :

— On the mutual information (using second characteristic function or the Kullback-Leibler divergence) : this is very general, and the mutual information is zero if and only if there is independence. The drawback is however that this criterion requires the joint density of the sources, which are unknown in practice and must be replaced by an estimate, which can be relatively costly in practice.

— On contrast functions : such a function is an optimization criterion such that its global maxima correspond to a separation of all the sources. However, one then need to check that all the cumulant are zero, while in practice only a few number of them are zero.

Despite its sound mathematical foundations, the ICA family suffers from several drawbacks. First, it (implicitly) requires the knowledge of the probability density functions of the sources. While approximations can be used, these can have an impact on the final solution. Second, the mutual independence of the sources might not be a valid hypothesis in several applications (*cf.* Planck data [Bobin *et al.* 2014], or hyperspectral data : as the sum of abundance fractions associated to each pixel is constant, the sources cannot be independent [Nascimento & Dias 2005]). Finally, its performances decrease much in the presence of noise, since most the approaches are assuming noiseless mixings (and since the addition of noise create some identifiability issues of the sources [Davies 2004]).

A.3.2 NMF

The second family of algorithms builds on the special case where it is known that $\mathbf{A}^*$ and $\mathbf{S}^*$ have only non-negative coefficients [Paatero & Tapper 1994, Lee & Seung 1999], which is often the case on real world data [Gillis & Glineur 2012]. For instance, the LC/MS data and the Chandra data of Section A.1 both fulfill this condition.
The non-negativity condition states that there are vectors $\mathbf{S}^*_j, j \in [1, n]$ such that all the data points in $\mathbf{X}$ can be written as non-negative linear combinations of the $\mathbf{S}^*_j$. In contrast to ICA, it makes that no subtraction can occur, which has led to the intuitive notion that NMF finds a data representation by patch, combining parts to form a whole [Lee & Seung 1999]. Furthermore, as the $\mathbf{S}^*_j$ are non-negative, a geometric interpretation of NMF is that it looks for a simplicial cone [5] in the positive orthant containing all the data points. However, if the data values are strictly positive, there is many such simplicial cones, raising the identifiability issue. In [Donoho & Stodden 2004], the authors however show that uniqueness of the simplicial cone can hold even if the data do not fill out the positive orthant, but

5. The simplicial cone generated by vectors $\mathbf{S}_j, j \in [1, n]$ is defined as $\Gamma_{\mathbf{S}} = \{\mathbf{x} \in \mathbb{R}^t | \mathbf{x} = \sum_{i=1}^{n} \alpha_i \mathbf{S}_i, 0 \leq \alpha_i\}$

rather a proper subset of the positive orthant. In particular, they show that the model is identifiable under the pure pixel assumption, stating that each source is active alone at least once.
As this condition might not always holds, NMF has also been combined with sparsity [Hoyer 2004, Kim & Park 2008, Rapin 2014].

A.3.3 SMF

In this thesis, we will specifically focus on sparse BSS, in which the sources are assumed to be sparse (roughly speaking, they are supposed to have a large number of zero coefficients, *cf.* Section B.1). This family has attracted much interest during the last two decades [Zibulevsky & Pearlmutter 2001, Bronstein *et al.* 2005, Li *et al.* 2006], which has mainly been motivated by the success of sparse signal modeling for solving very large classes of inverse problems [Starck *et al.* 2010].
Sparse BSS will be more extensively presented in Section B. In brief, it has lead to enhanced separation quality, in particular with the Generalized Morphological Component Analysis (GMCA – [Bobin *et al.* 2007]) algorithm. In the framework of ICA, Efficient FastICA (EFICA) [Koldovsky *et al.* 2006] is a FastICA-based algorithm that is especially adapted to retrieve sources with generalized Gaussian distributions, which includes sparse sources. In the seminal paper [Zibulevsky 2003], the author also proposed a Newton-like method for ICA called Relative Newton Algorithm (RNA), which uses quasi-maximum likelihood estimation to estimate sparse sources.
In the next section, we develop the concept of sparsity and sparse BSS.

B Sparsity and sparse BSS

B.1 Sparsity

Over the last twenty years, sparse modelling has had an increased impact in various signal processing areas, such as denoising [Elad & Aharon 2006] (or more generally signal restoration), feature extraction [Hyvarinen *et al.* 1998], compression [Le Pennec & Mallat 2000] and, in our case, source separation [Zibulevsky & Pearlmutter 2001, Bobin *et al.* 2007]. This interest has been mainly driven by the compressed sensing theory [Candes & Tao 2004, Donoho *et al.* 2006], which can be seen as an alternative to the Shannon sampling theory (in which the signals are assumed to be frequency band-limited and not sparse). In brief, compressed sensing theory gives some support to methods assuming the sparsity to make better-posed ill-posed inverse problems. Indeed, it can be shown that (under conditions) methods that look for a sparse solution might find the exact solution to the problem at hand.
The goal of this subsection is thus to briefly present the concept of sparsity (while it is of course widely used in this work, we do not aim at presenting in depth theoretical results). The presentation will mainly follow the one of [Starck *et al.* 2010, Mairal *et al.* 2014], but we also recommend [Mallat 1999, Elad 2010], while [Rapin 2014, Che-

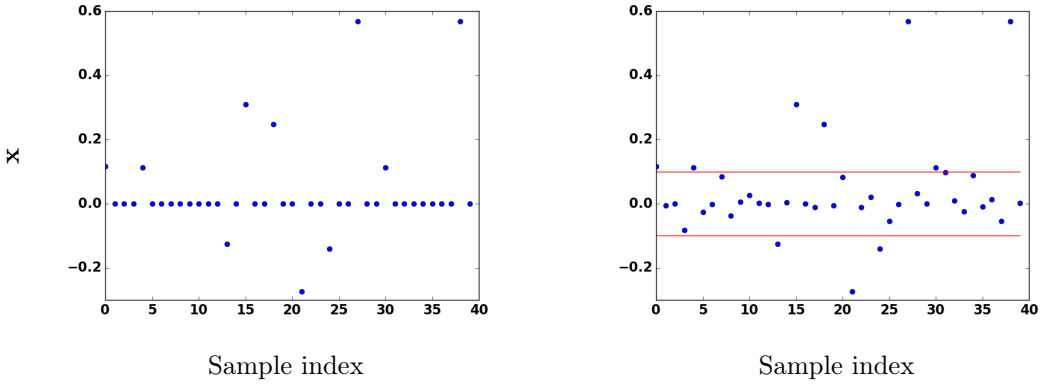


Figure II.5 – Examples of : *Left :* exactly sparse signal : most of the coefficients are 0 ; *Right :* approximately sparse signal : while most of the coefficients are non-zeros, many of them are *close* to 0, making that the signal can be well approximated by its highest amplitude coefficients. Thus, if all the coefficients between the two red lines are set to 0, the remaining samples comprehend most of the signal energy, which makes that the approximately sparse signal is decently approximated by the exactly sparse signal of the left figure.

not 2017] give a more specific overview of sparsity for BSS.

B.1.1 Definition of sparsity and sparsity in the direct domain

Generally speaking, sparsity amounts to represent a signal with as few variables as possible. More specifically, a signal $\mathbf{s} \in \mathbb{R}^{1\times t}$ is said to be exactly sparse if only few of its coefficients, let say $k \ll t$, are non-zeros : $\|\mathbf{s}\|_0 = k$. An example of such a signal is displayed in Figure II.5a.
Nevertheless, in real-life the exact sparsity is a too restrictive assumption and a more interesting case is the one of approximately sparse signals (also called weakly sparse or compressible signals). In these, while $\|\mathbf{s}\|_0 \simeq t$, only a small number k of samples have a high amplitude, making that only keeping them already enables a good approximation of the signal. In this sense the signal can be well approximated as k-sparse. This is for instance the case if the sorted magnitude of the samples $\mathbf{s}_i, i \in [1, t]$ decays according to a power low. For an example of such an approximately sparse signal, see Figure II.5b.
A good example of sparse signals in the direct domain is the one of the LC/MS data of Fig. II.2.

B.1.2 Sparsity in a transformed domain

However, most of the signals of interest are not (exactly or approximately) sparse in the *direct* domain, but rather admit a sparse representation $\mathbf{s}_{\mathbf{\Phi}} \in \mathbb{R}^{1\times T}$ in a *trans-*

formed domain $\mathbf{\Phi} \in \mathbb{R}^{t \times T}$ (see Fig. II.6 for a concrete example). Said differently, using a transform $\mathbf{\Phi}$ which captures well the morphology of the signal $\mathbf{s}$ enables to concentrate the energy of $\mathbf{s}$ within a few number of coefficients.
Many such domains have been proposed in the signal processing community, starting from the broadly known Fourier basis [Bracewell & Bracewell 1986]. However, the Fourier basis is not localized in space, making its use restricted to stationary signals. To bypass this issue, wavelet transforms have been introduced. Compared to the Fourier basis, the main advantage is that wavelets are localized both in space and frequency domains. A wavelet basis is a set of functions $\Phi_1, \Phi_2 ..., \Phi_n$ that are essentially dilated and shifted version of each other. Generally speaking, wavelets enable to sparsify signals that are polynomial by parts. Starting from the ones of Haar [Haar 1910], various wavelets have been designed that correspond to different signal or image geometric contents. Among them, one can cite [6] : Daubechies wavelets [Daubechies 1988], the curvelets [Candes & Donoho 2002], the contourlets [Do & Vetterli 2005], the bandlets [Le Pennec & Mallat 2005], the starlet [Starck *et al.* 2010], the shearlets [Easley *et al.* 2008]... It is important to emphasize that among all these proposed transforms, a distinction must be made between i) orthogonal (in which $\mathbf{\Phi}^T\mathbf{\Phi} = \mathbf{\Phi}\mathbf{\Phi}^T = \mathbf{Id}$) versus other bases of wavelets ; ii) redundant ($T > t$) versus nonredundant transforms. While redundant transforms enables interesting properties, they are more difficult to handle since the decomposition in the transformed domain is not unique and $\mathbf{\Phi^T\Phi} \neq \mathbf{Id}$ – see [Rapin 2014] for a detailed account of the use of redundant representations in the case of sparse BSS.
Once a transformed domain $\mathbf{\Phi}$ has been chosen, we still have to find a sparse representation of $\mathbf{s}$ in this domain. For redundant transform, such a representation is not unique and it is then possible to find among all the possible representations the one that will follow a given criterion, namely here sparsity. More specifically, for a given $\mathbf{s} \in \mathbb{R}^{1 \times t}$ the goal is to learn a representation $\mathbf{s} \simeq \mathbf{s_\Phi \Phi^T}$ such that $\mathbf{s_\Phi} \in \mathbb{R}^{1 \times T}$ is sparse. An option is to learn the representation $\mathbf{s_\Phi}$ from the minimization of a cost function using the *synthesis* formulation :

$$\underset{\mathbf{s_\Phi} \in \mathbb{R}^{1 \times T}}{\operatorname{argmin}} \; h(\mathbf{s_\Phi \Phi^T}) + g(\mathbf{s_\Phi}) \tag{II.4}$$

with a data fidelity term $h(\mathbf{s_\Phi \Phi^T}) \in \mathbb{R}^+$ expressing the discrepency of the model with regards to the observations $\mathbf{s}$, and a regularization term $g(\mathbf{s_\Phi}) \in \mathbb{R}^+$, which is used to enforce the sparsity of the representation. From this quite general model, several specific ones can be derived, depending on the choice of h and g. For instance, a natural choice of g is to choose it using the ℓ_0 norm, to count the number of non-zeros elements : $g(\mathbf{s_\Phi}) = \lambda \|\mathbf{s_\Phi}\|_0$. However, in practice using the ℓ_1 norm instead of the ℓ_0 one is common, since the ℓ_1 norm is the closest convex norm to the ℓ_0 one [Chen *et al.* 2001]. Concerning h, generally a squared Euclidian ℓ_2 distance is used,

6. As a side remark, instead of imposing beforehand the transform, it is further possible to learn it from the data through dictionary learning [Olshausen & Field 1996, Mairal *et al.* 2014, Mensch *et al.* 2018], making it much more adapted to the data at hand. This is however not the main focus of this thesis.

which stems from the assumption of a Gaussian noise : $h(s_\Phi \mathbf{\Phi^T}) = \frac{1}{2} \left\| \mathbf{s} - \mathbf{s_\Phi \Phi^T} \right\|_{\ell_2}^2$. These choices lead to the well known basis pursuit formulation :

$$\underset{s_\Phi \in \mathbb{R}^{1\times T}}{\operatorname{argmin}} \frac{1}{2} \left\| \mathbf{s} - \mathbf{s_\Phi \Phi^T} \right\|_{\ell_2}^2 + \lambda \left\| \mathbf{s_\Phi} \right\|_1 \tag{II.5}$$

For other usual formulations, we refer the reader to [Mairal *et al.* 2014]. We would like to recall – while this is not the main focus of this thesis – that another formulation than the synthesis one of II.4 is possible, namely the *analysis* formulation :

$$\underset{\mathbf{s}\in \mathbb{R}^{1\times t}}{\operatorname{argmin}} \; h(\mathbf{s}) + g(\mathbf{s\Phi}) \tag{II.6}$$

While for orthonormal transforms $\mathbf{\Phi}$, synthesis and analysis formulations are equivalent, this is not anymore the case for redundant transforms [Rapin 2014].

B.2 Sparse BSS as an optimization problem

The goal of this section is to give a quick overview of the optimization problem we will have to tackle to solve the BSS problem. More details will be given in the next chapter.

B.3 General problem

Generally speaking, in this work we will aim at performing BSS through the minimization of a non-convex cost function of the form :

$$\underset{\mathbf{A}\in \mathbb{R}^{m\times n}, \mathbf{S}\in \mathbb{R}^{n\times t}}{\operatorname{argmin}} \frac{1}{2} h(\mathbf{A}, \mathbf{S}) + \mathcal{J}(\mathbf{A}) + \mathcal{G}(\mathbf{S}) \tag{II.7}$$

Where :

— h is a data fidelity term measuring the discrepancy between the data and the mixture model. In particular, h will be assumed to be differentiable with respect to $\mathbf{A}$ and $\mathbf{S}$, to have Lipschitz gradients and to be block multi-convex [Xu & Yin 2013], *cf.* Chapter III-C.

— The penalizations $\mathcal{J}$ and $\mathcal{G}$ enforce some desired properties on $\mathbf{A}$ and $\mathbf{S}$. The conditions on such functions will be developed in Chapter III-C. As a quick example, $\mathcal{J}$ and $\mathcal{G}$ can be used in NMF to impose the non-negativity of the sources and the mixing matrix respectively. In this situation, $\mathcal{J}(\mathbf{A}) = \iota_{.\succeq 0}(\mathbf{A})$ and $\mathcal{G}(\mathbf{S}) = \iota_{.\succeq 0}(\mathbf{S})$, where ι_U is the indicator function of the set U : here, it is equal to infinity if at least one coefficient of the corresponding matrix is negative, and zero otherwise.

The minimization schemes used for Eq. II.7, and in particular the use of the multi-convex structure of h will be detailed in Chapter A.

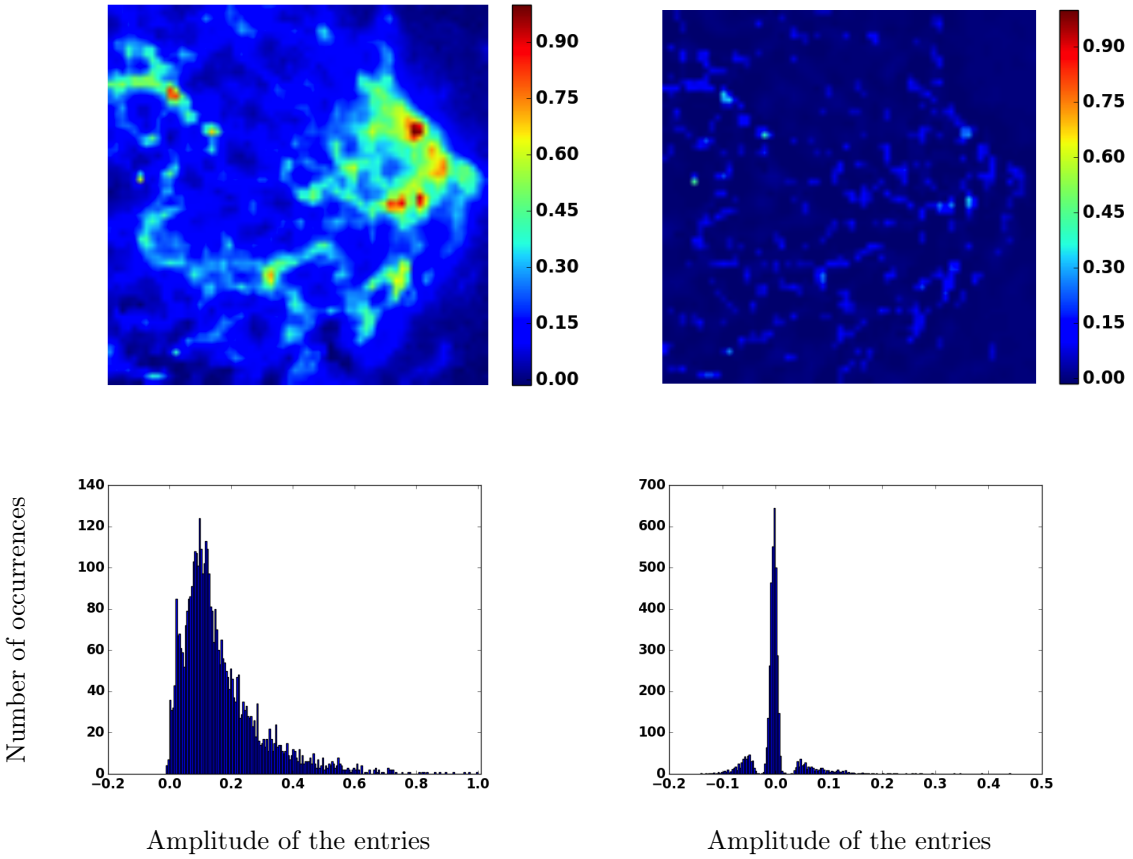


Figure II.6 – Example of use of a wavelet transform, here the starlets, to sparsify an image. Here, the image corresponds to a zoom on a noisy synchrotron emission in a supernovae remnant. *Upper left :* original image $\mathbf{s}^*$; *Upper right :* image in the starlet domain $\mathbf{S}_{\mathbf{\Phi}}$ (or more exactly, image corresponding to the first detail scale in this decomposition) : much more pixels are close to 0 ; *Down left :* histogram of the original image ; *Down right :* histogram of the image in the starlet domain : the amplitudes are much closer to 0 and the histogram is much less flat than the one in the original domain. In the starlet domain, the data can be considered as approximately sparse.

B.4 Concrete example : sparsity prior

To make things more concrete, we now derive a more specific version of Eq. II.7 in the context of sparse BSS, in which case it can be written using the following form :

$$\underset{\mathbf{A}\in\mathbb{R}^{m\times n},\mathbf{S}\in\mathbb{R}^{n\times t}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{X}-\mathbf{AS}\|_F^2 + \left\|\mathbf{R_S}\odot(\mathbf{S}\mathbf{\Phi_S^T})\right\|_{\ell_1} + \iota_{\{\forall i\in[1,n];\|\mathbf{A}^i\|_{\ell_2}^2=1\}}(\mathbf{A}) \tag{II.8}$$

— The $h(\mathbf{A},\mathbf{S}) = \frac{1}{2}\|\mathbf{X}-\mathbf{AS}\|_F^2$ term promotes a faithful reconstruction of the data under a white Gaussian noise $\mathbf{N}$ assumption inducing the use of the Frobenius norm $\|.\|_F$ (see *e.g.* [Hamzaoui & Bobin 2018, Bobin *et al.* 2019] for the case of Poisson noise).

— The $\mathcal{G}(\mathbf{S}) = \left\|\mathbf{R_S}\odot(\mathbf{S}\mathbf{\Phi_S^T})\right\|_1$ term promotes a ℓ_1 sparsity of the sources, with $\odot$ denoting the Hadamard product. This work will focus on the ℓ_1-norm, which has been shown to help the separation of the sources in the context of sparse matrix factorization [Rapin *et al.* 2013]. The $\mathbf{\Phi_S}$ matrix (of size $T\times t$, with $T \geqslant t$) corresponds to the transform enforcing the sparsity. In this thesis, it will be taken equal to either the identity (in which case the sparsity is enforced in the direct domain) or the starlet transform [Starck *et al.* 2007]. The regularization parameter $\mathbf{R_S}$ matrix (size $n\times T$) controls the trade-off between the data fidelity and the sparsity terms. It can be decomposed into $\mathbf{R_S} = \mathbf{\Lambda_S G}$ where $\mathbf{\Lambda_S}$ $(n\times n)$ is a diagonal matrix of the regularization parameters $\lambda_1, \lambda_2, ..., \lambda_n$ and $\mathbf{G}$ $(n\times T)$ is a matrix used to introduce individual penalization coefficients in the context of reweighted ℓ_1 [Candes *et al.* 2008] (when no reweighting is used, $\mathbf{G} = \mathbf{1}_{n\times T}$ is a matrix with all coefficients equal to 1).

— To avoid degenerated $\mathbf{A}$ and $\mathbf{S}$ matrices where $\|\mathbf{A}\|_F \to \infty$ and $\|\mathbf{S}\|_F \to 0$ due to the sparsity penalty, the last term enforces the oblique constraint, ensuring that for all i, the i^{th} column $\mathbf{A}^i$ of $\mathbf{A}$ is onto the ℓ_2 hypersphere : $\mathcal{J}(\mathbf{A}) = \iota_{\{\forall i\in[1,n];\|\mathbf{A}^i\|_2^2=1\}}(\mathbf{A})$, with ι being the characteristic function of the corresponding convex set.

B.5 Sparse BSS : hypotheses made throughout this work

In the following and without other indication, we will assume that :

— **Both the matrices $\mathbf{A}^*$ and $\mathbf{S}^*$ are full rank** : this is a common assumption, especially as in ICA the matrix $\mathbf{A}^*$ is supposed to be orthogonal ;

— **The number of sources n is known**.

— **We are in the over-determined setting with** $n \leq m$: the under-determined setting could however probably be tackled using the method of Chapter V, but this is out of the scope of this work. **Furthermore,** $m \ll t$: high values of t are required for identifiability issues [Gribonval & Schnass 2010].

— **The matrix $\mathbf{A}^*$ has unitary columns** : more specifically, each column lies on the ℓ_2 unit sphere, which enables to leverage the scale indeterminacy issue (*cf.* Sec. A.3.3).

— **The sources are sparse in a transform domain $\mathbf{\Phi_S}$ and respect the morphological diversity assumption** : see Chapter A-C.4.2 for a more detailed explanation, and [Gribonval & Schnass 2010] for identifiability in sparse matrix factorization.

C Large-scale sparse BSS and organization of the manuscript

This thesis is subdivided into four sub-problems taking their roots around the large-scale sparse BSS issue. We here detail them and shortly present some of the main results to give a quick overview of this work. The methods, results and interpretations will be more thoroughly presented in the following Chapters, after an introduction in Chapter III of the background required for understanding the sparse BSS optimization framework.

C.1 Preliminary question : avoiding relaunchs in sparse BSS and increasing robustness

This first problem is to be linked the difficulty of finding a (local or global) minimum of the cost function II.7 corresponding to a *physical* factorization of the data $\mathbf{X}$ [7]. To perform such a task, it is possible to tune the regularization parameters $\mathbf{R_S}$ to try to highlight the most interesting critical points. Another way is to find a good initialization of the algorithm, that is if possible an initialization within the basin of attraction of a minimum corresponding to good estimate $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$.
To perform such a task, it is usual to try several initializations and $\mathbf{R_S}$ values, and to retain the ones corresponding to the best estimates. This grid-search approach can however become costly in the large-scale context we aim at tackling in this thesis, as the computational cost of re-launching the algorithms can become prohibitive [8]. As such, methods that are robust to the initialization and enabling an automatic regularization parameter choice without any relaunch are of uttermost importance for working on practical datasets. Therefore, in Chapter IV we will aim at trying to generalize the automatic regularization parameter choice of GMCA [Bobin *et al.* 2007] to the more recent and mathematically grounded Proximal Alternating Linearized

7. Thus, we highlight that we are not interested by finding *any* critical point of Eq. II.7, but rather *a specific* one corresponding to the physical true underlying factor $\mathbf{A}^*$ and $\mathbf{S}^*$. Note also that there are several global minima, which can however potentially correspond to estimations $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$ of unequal qualities (see *e.g.* [Neyshabur 2017] in the context of over-parametrized machine learning).

8. Note that this study is nevertheless not fully restricted to the large-scale setting, since in some experiments practitioners might have no clue concerning the true $\mathbf{A}^*$ and $\mathbf{S}^*$, making it hard to assess the quality of the estimates $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$ obtained for given parameters and initializations.

Minimization (PALM) algorithm [Bolte *et al.* 2014]. The proposed strategy is empirically shown to be robust to the initialization.
More precisely, Chapter IV is subdivided in two main parts :

— We first conduct an empirical study showing that, for sparse BSS, a minimization of Eq. II.8 using PALM jointly with a grid-search approach for $\mathbf{R_S}$ choice is a strategy suffering from a low *efficiency* and *versatility*. In brief, this highlights that performing sparse BSS using PALM requires a very careful and sensitive tuning of $\mathbf{R_S}$. Similar conclusions can be (to a more limited extent) shown concerning PALM initialization : this approach also undergo a low *reliability* ;

— To alleviate the cumbersome hyper-parameter tuning described above and increase the robustness to the initialization, we then rationalize a two-step strategy, with i) a *warm-up* stage comprehending a GMCA, yielding in practice a descent first guess of $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$; ii) a *refinement* stage using a PALM : the regularization parameters $\mathbf{R_S}$ are chosen based the warm-up stage first guess.

The quality of the method is shown on a realistic astrophysical experiment and the limitations of the proposed two-step strategy are discussed. We refer the reader to Chapter IV.

C.2 Large number of sources n

Chapter V extends the above two-step strategy to tackle mixings with a high number of sources n. A classic example of such a problem is the one of the LC/MS data developed in Section A.1.2 : if the fluid to be analysed has a high number of constitutive elements, the corresponding number of spectra and thus of sources can reach for instance up to $n = 100$. No sparse BSS algorithm is currently tailored for such problems, as the issue is twofold :

— *Computational issue :* As the number of sources increases, so do the size of the $\mathbf{A}^*$ and $\mathbf{S}^*$ matrices. Since state-of-art methods such as [Bobin *et al.* 2007] requires several inversion of such matrices, the computational burden can become high ;

— *Deteriorated performances :* Beyond the computational aspects, the separation quality of most BSS algorithms tends to dramatically deteriorate in the presence of a high number of sources (*cf.* Fig. V.1 of Chapter V). Therefore, being able to maintain high separation performances is an open challenge.

The approach proposed in Chapter V re-uses the two-step strategy and introduces coordinate blocks of intermediate sizes r. This is in contrast to the other state-of-art methods that either use sizes $r = 1$ (hierarchical of deflations methods) or full-size blocks $r = n$ (*e.g.* GMCA). Beyond a huge gain in computation time, the proposed block-GMCA (bGMCA) algorithm is also empirically shown to *improve the separation quality* for relatively small block sizes. An interpretation of such a phenomenon, as well as extensive numerical experiments, are proposed in Chapter V.

C.3 Large datasets X

The most tangible large-scale issue might the one of large datasets $\mathbf{X} \in \mathbb{R}^{m \times t}$. As evoked in introduction, these might for instance become the new standard in astronomy, with space telescopes such as Euclid [9], huge radio-interferometers such as the SKA [10] (*cf.* Sec. A.1.3), or the telescope LSST [11]. Such devices will produce tremendous amounts of data : in order to give a rough idea of the involved quantities, one could speak of images of resolution $t = 10^9$ pixels and about $m = 10^4$ channels. In this context, sparse BSS algorithms that yielded high quality results [Bobin *et al.* 2007] cannot be used, due to :

— *Memory limitations :* The datasets are so huge that they cannot be stored at once within memory. Therefore, one of the few current solutions is to scan submatrices of $\mathbf{X}$ and work on each of them in an isolated way, which might deteriorate much the separation quality ;

— *Time limitations :* Even not considering memory issues, the heavy computations involved might largely slow down the processing time. As such, parallelized algorithms should be preferred.

Chapter VI proposes to introduce mini-batches, that is submatrices of $\mathbf{X}$, within the GMCA algorithm [12]. Note that this is in contrast to the use of intermediate-size block coordinate methods described above, as these split the factors $\mathbf{A}$ and $\mathbf{S}$ but use the whole $\mathbf{X}$. However, the use of mini-batches with the projected Alternating Least-Square (pALS) scheme of GMCA raises open questions, as each of the B mini-batches yields a different estimate of the *full* $\mathbf{A}^*$. Therefore, we need to *aggregate* these estimates at each iteration of the proposed distributed-GMCA (dGMCA). In brief, we propose to benefit from the structure implied by the oblique constraint in Eq. II.8 to enhance such an aggregation through the use of a *Fréchet mean*. To further robustify the process, we also propose to i) use weights taking into account the assumed quality of the estimation of each source in each mini-batch ; ii) use a more robust aggregation method, reminiscent of a median on the unit hypersphere. Beyond the expected gain in terms of computation time and memory (as each mini-batch can be sent to a different node of a cluster), a striking result is that when the sources $\mathbf{S}^*$ are very sparse, the use of the robust Fréchet mean enables to obtain better results than with full batch methods. We propose an explanation of this phenomenon in Chapter VI.

C.4 Non-linear BSS

Non-linear BSS is a difficult problem, although such a model can be required when the usual linear one is too simplistic for correctly modelling the mixing. An

9. http ://sci.esa.int/euclid/
10. https ://www.skatelescope.org
11. https ://www.lsst.org
12. Indeed, the two-step approach requires as a warm-up stage at least a few iterations of GMCA. As such, it still needs to be able to use GMCA on large $\mathbf{X}$.

example is the show-through removal presented in Section A.1.4, for which the physical processes imply a linear quadratic model rather than a mere linear model. The StackedAMCA algorithm we propose in Chapter VII to tackle non-linear sparse BSS can be envisioned as an extension of large-scale BSS. Indeed, it estimates the underlying non-linearities through a linear-by-part model comprehending L parts. As this approach requires to solve a potentially large number L of linear subproblems, it calls for efficient linear BSS algorithms.
More specifically, StackedAMCA is an iterative algorithm. Each iteration alternates between a linear step and a non-linear one :

— The linear step consists in the application on the current dataset of a sparse linear BSS algorithm robust to high amplitude non-linearities, which enables to find a new part of the linear-by-part approximation ;

— The non-linear step consists in creating a new dataset in which the contribution of all the previously found linear models is canceled. This paves the way for the next iteration, in which a new linear model will be fitted on this new dataset.

The algorithm obtains better separation results than other state-of-art methods. Interestingly enough, it can further reconstruct the sources in some settings despite increased indeterminacies in non-linear BSS (*cf.* Sec. A.2.2). The required hypotheses for StackedAMCA to work, as well as a subset of mixings for which it can further be expected to reconstruct the sources, are discussed in Chapter VII.

Chapitre III

Optimization frameworks for sparse BSS

A Reminder about sparse BSS cost function and outline of the chapter

As mentioned in the previous chapter, we will aim at performing BSS through the minimization of a cost function of the form :

$$\underset{\mathbf{A}\in\mathbb{R}^{m\times n},\mathbf{S}\in\mathbb{R}^{n\times t}}{\operatorname{argmin}} \frac{1}{2}h(\mathbf{A},\mathbf{S}) + \mathcal{J}(\mathbf{A}) + \mathcal{G}(\mathbf{S}) \tag{III.1}$$

Such a minimization is not easy to perform due to several reasons :

— While the term $h(\mathbf{A},\mathbf{S})$ is assumed to be smooth with a Lipschitz gradient, calling for simple optimization methods such as gradient descent, it is not necessarily the case of the $\mathcal{J}(\mathbf{A})$ and $\mathcal{G}(\mathbf{S})$ terms (*cf. e.g.* Sec. B.4 explaining the case of sparse sources and the use of the ℓ_1-norm, which is not differentiable in 0). Therefore, we will need to resort to *non-smooth* optimization algorithms and more specifically to proximal algorithms, which will be explained in section B.

— The problem is non-convex in $\mathbf{A}$ and $\mathbf{S}$. Nevertheless, as mentioned in Section B.3, it is *multi-convex* [Xu & Yin 2013], calling for the use of specific algorithms that will be detailed in sections C.3 (algorithms aiming at truly minimizing Eq. II.7), and C.4 (heuristics enabling enhanced practical results).

We assume the reader to have a basic knowledge of optimization tools, and refer to [Boyd *et al.* 2011] for generic notions.

B Proximal operators and proximal algorithms

Proximal algorithms enable to solve convex non-smooth optimization problems. They have been extensively used over the last decades in the signal processing community [Combettes & Wajs 2005, Jenatton *et al.* 2010, Bolte *et al.* 2010, Jezierska *et al.* 2012, Bolte *et al.* 2014, Chouzenoux *et al.* 2016]. Roughly speaking, they operate at a high level of abstraction, since their base operation uses the *proximal operator* of a function, which computation involves itself a (potentially simple) convex optimization problem. In this section, we will first explain the notion of proximal operator and then their use within proximal algorithms. Our presentation will mainly follow the ones from [Combettes & Pesquet 2011, Parikh *et al.* 2014].

B.1 Proximal operators

B.1.1 Definition

We will call a function $f : \mathbb{R}^t \to \mathbb{R} \cup \{\infty\}$ proximable if it is a closed proper convex function. For such functions, the proximal operator of f is defined as :

$$\text{prox}_f(\mathbf{u}) = \underset{\mathbf{y}\in\mathbb{R}^t}{\text{argmin}}\ f(\mathbf{y}) + \frac{1}{2}\|\mathbf{u}-\mathbf{y}\|_{\ell_2}^2 \tag{III.2}$$

with $\|.\|_{\ell_2}$ the usual Euclidian norm for vectors. Since the minimized function is strongly convex and not infinite everywhere, it has a unique minimizer.
Several interpretations of the proximal operators are interesting for their understanding :

— A natural one is to see proximal operators as a local minimizer of f : the function minimized within the operator is the sum of f and a term that penalizes high distances between the current point $\mathbf{u}$ and $\mathbf{y}$. More specifically, if $\mathbf{u_m}$ is a minimum of f, then $\text{prox}_f(\mathbf{u_m}) = \mathbf{u_m}$, which is called the fixed point property. This is one of the bases for understanding proximal algorithms and, combined to the firm non-expansiveness of proximal operators, leads to one of the most basic method, namely the proximal point algorithm (*cf.* Sec. B.3.1).

— Proximal operators can be interpreted as gradient steps for minimizing f or a function related to f. For instance, introducing the Moreau envelope of f : $M_f(\mathbf{u}) = \inf_{\mathbf{y}\in\mathbb{R}^t}(f(\mathbf{y})+\frac{1}{2}\|\mathbf{u}-\mathbf{y}\|_{\ell_2}^2)$, it can be shown that the proximal operator of f can be written as :

$$\text{prox}_f(\mathbf{u}) = \mathbf{u} - \nabla M_f(\mathbf{u}) \tag{III.3}$$

Therefore, prox_f can be viewed as a gradient step with step 1 (generalization to other steps are possible) for minimizing M_f, which has the same minimizers as f. This leads to a natural interpretation for minimization algorithms.

— Proximal operators can also be seen as a generalization of projections. This is explicit if f is the indicator function of a closed nonempty convex set U : $f(\mathbf{y}) = i_{.\in U}(\mathbf{y})$. Then the proximal operator of f reduces to the Euclidian projection onto U :

$$\text{prox}_f(\mathbf{u}) = \Pi_U(\mathbf{u}) = \underset{\mathbf{y}\in U}{\text{argmin}}\ \|\mathbf{u}-\mathbf{y}\|_{\ell_2} \tag{III.4}$$

This interpretation enables to understand the use of some proximal operators in proximal algorithms as a projection onto some constraint set.

While the requirement of computing the proximal operators through a minimization problem might seem intricate, in practice many usual functions have explicit or easy-to-compute proximal operators. The role of the following subsection is to present some of them which will be of use in the remaining of this work.
But before, we highlight a last generic property of proximal operators that will

be implicitly used throughout this whole work. Indeed, we will in practice be applying proximal operators on matrices, and not on vectors as defined above. Fortunately, this is not an issue since our constraints will be separable, enabling to use the separable sum property. More specifically, if f is separable such that $f(\mathbf{y}) = \sum_{i=1}^{K} f_i(\mathbf{y}_i), K \in \mathbb{N}^*$, then :

$$(\text{prox}_f(\mathbf{u}))_i = \text{prox}_{f_i}(\mathbf{u}_i) \tag{III.5}$$

Where $(\text{prox}_f(\mathbf{u}))_i$ is the i^{th} line of the vector $\text{prox}_f(\mathbf{u})$. Said differently, instead of working directly on matrices, we can come back to the vector case (*e.g.* with the oblique constraint on $\mathbf{A}$, *cf.* below) or even to the scalar case (*e.g.* with the sparsity constraint on $\mathbf{S}$).

B.2 Examples of proximal operators used in this work

This subsection gives a few examples of proximal operators, by focusing on the constraints $\mathcal{J}$ and $\mathcal{G}$ that will be used in this work.

1 - Penalizations $\mathcal{G}$ for the sources $\mathbf{S}$ and corresponding proximal operators :

- ℓ_1 *sparsity constraint in some transformed domain* : In this case, the sparsity constraint on $\mathbf{S}$ is enforced with a ℓ_1-norm penalization (which has led to enhanced separation quality in BSS [Rapin *et al.* 2013]) :

$$\mathcal{G}(\mathbf{S}) = \left\| \mathbf{R_S} \odot (\mathbf{S}\mathbf{\Phi_S^T}) \right\|_1 \tag{III.6}$$

  where the $\odot$ sign denotes the Hadamard product. The matrix $\mathbf{R_S}$ (of same size as $\mathbf{S}\mathbf{\Phi_S^T}$) contains individual regularization parameters for each coefficient. The $\mathbf{\Phi_S}$ is a transform, supposed orthogonal in the following, into a domain in which $\mathbf{S}$ can be sparsely represented. The proximal operator of $\mathcal{G}$ is then explicit and corresponds to the soft-thresholding operator with threshold $\mathbf{R_S}$ – which we shall denote $\mathcal{S}_{\mathbf{R_S}}$, *cf.* Appendix A for the definition – applied in the transformed domain : $\text{prox}_{\mathcal{G}}(\mathbf{S}) = \mathcal{S}_{\mathbf{R_S}}(\mathbf{S}\mathbf{\Phi_S}^T)\mathbf{\Phi_S}$

- *Non-negativity in the direct domain* : here, all coefficients in $\mathbf{S}$ must be non-negative, which can be written using the characteristic function of the positive orthant $K^+ = \{\mathbf{S} \in \mathbb{R}^{n \times t}; \forall i \in [1,n], j \in [1,t], \mathbf{S}_i^j \geq 0\}$:

$$\mathcal{G}(\mathbf{S}) = \iota_{K^+}(\mathbf{S}) \tag{III.7}$$

  Following the previous subsection B.1.1, the corresponding proximal operator is then the projection on the positive orthant Π_{K^+}, which is the identity for non-negative coefficients and 0 for the negative ones (*cf.* Appendix A for the exact definition).

- *Non-negativity in the direct domain and* ℓ_1 *sparsity constraint in some transformed domain* : due to the non-negativity constraint, all coefficients

in $\mathbf{S}$ must be non-negative in the direct domain in addition to the sparsity constraint in a transformed domain $\mathbf{\Phi_S}$. It can be formulated as :

$$\mathcal{G}(\mathbf{S}) = \|\mathbf{R_S} \odot \left(\mathbf{S}{\mathbf{\Phi_S}}^T\right)\|_1 + \iota_{K^+}(\mathbf{S}) \tag{III.8}$$

The difficulty is to enforce at the same time two constraints in two different domains, and the corresponding proximal operator of $\mathcal{G}$ is not explicit. It can either be roughly approximated by composing the proximal operators of the individual penalizations to produce a cheap update : $\text{prox}_{\mathcal{G}}(\mathbf{S}) = \mathcal{S}_{\mathbf{R_S}}(\Pi_{K^+}(\mathbf{S}){\mathbf{\Phi_S}}^T)\mathbf{\Phi_S}$, or computed accurately using the Generalized Forward-Backward splitting algorithm [Raguet *et al.* 2013] (*cf.* Sec. B.3.3).

2 - Penalizations $\mathcal{J}$ for the mixing matrix $\mathbf{A}$:

- *Oblique constraint* : the columns of $\mathbf{A}$ are constrained to lie on the ℓ_2 hyper-sphere. The corresponding set is $\mathcal{O} = \{\mathbf{A} \in \mathbb{R}^{m\times n}; \forall j \in [1,n], \|\mathbf{A}^j\|_{\ell_2}^2 = 1\}$. This constraint is used to avoid degenerated $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$ results.
  More specifically, $\mathcal{J}$ can be written as :

$$\mathcal{J}(\mathbf{A}) = \iota_{\mathcal{O}}(\mathbf{A}) \tag{III.9}$$

  Following this constraint, the proximal operator is $\text{prox}_{\mathcal{J}}(\mathbf{A}) = \Pi_{\mathcal{O}}(\mathbf{A})$, the projection on the ℓ_2 unit hypersphere of each column of $\mathbf{A}$ (*cf.* Appendix A).

- *Non-negativity and oblique constraint* : Adding the non-negativity constraint to the oblique constraint on $\mathbf{A}$ reads :

$$\mathcal{J}(\mathbf{A}) = \iota_{\mathcal{O}}(\mathbf{A}) + \iota_{K^+}(\mathbf{A}) \tag{III.10}$$

  The proximal operator can be shown to be the composition of the proximal operator corresponding to non-negativity $\Pi_{\mathcal{K}^+}$ followed by $\Pi_{\mathcal{O}}$: $\text{prox}_{\mathcal{J}}(\mathbf{A}) = \Pi_{\mathcal{O}}(\Pi_{K^+}(\mathbf{A}))$.

B.3 Proximal algorithms

Now that the proximal operators have been introduced, we can explain how to use them for solving convex (non-smooth) optimization problems. To do that, we will detail several so-called proximal algorithms. While the first one is mostly described for the sake of clarity and introducing the concept of proximal algorithms, the two following ones will be cornerstones of this work [1].

1. Note that we only detail the two proximal splitting methods that will be used in this work. Nevertheless, a vast literature related to this topic exists, with many well-known algorithms : Alternating Direction Method of Multipliers (ADMM - [Boyd *et al.* 2011]), Douglas-Rachford [Douglas & Rachford 1956], Chambolle-Pock [Chambolle & Pock 2011], Dykstra-like splitting [Combettes & Pesquet 2011]...

B.3.1 Proximal point algorithm

Let $f : \mathbb{R}^t \to \mathbb{R}$ be a closed proper convex function. The problem here is to find a minimizer of f :

$$\underset{\mathbf{u}\in\mathbb{R}^t}{\operatorname{argmin}}\ f(\mathbf{u}) \tag{III.11}$$

To perform such a task, the most simple proximal algorithm is the proximal point algorithm, which iterations are described by :

$$\hat{\mathbf{u}}^{(l+1)} = \operatorname{prox}_{\lambda f}(\hat{\mathbf{u}}^{(l)}) \tag{III.12}$$

where $\hat{\mathbf{u}}^{(l)}$ it the l^{th} iteration of the algorithm and $\lambda > 0$. Then, if f has a minimum, the algorithm converges to it, which is linked to the fact that the proximal operator is a non-expansive operator and a local minimizer of f. However, this simple method does not have many applications, since it is restricted to the cases where f is difficult to minimize but f plus a quadratic is simple to minimize. Therefore, more advanced minimization schemes using splitting methods have been introduced.

B.3.2 Forward backward splitting

The problem at hand is here to minimize :

$$\underset{\mathbf{u}\in\mathbb{R}^t}{\operatorname{argmin}}\ h(\mathbf{u}) + \mathcal{J}(\mathbf{u}) \tag{III.13}$$

with $h : \mathbb{R}^t \to \mathbb{R}$ and $\mathcal{J} : \mathbb{R}^t \to \mathbb{R} \cup \{\infty\}$ closed proper convex and h differentiable. As an example of such a problem, note that the BSS problem of Eq. II.8 *when performing the minimization over only one* variable (*e.g.* fixing $\mathbf{A}$ and performing the minimization over $\mathbf{S}$ only), is an instance (generalizing the discussion to matrices) of such a minimization :

$$\underset{\mathbf{S}\in\mathbb{R}^{n\times t}}{\operatorname{argmin}}\ \frac{1}{2}\left\|\mathbf{X} - \mathbf{A}\mathbf{S}\right\|_F^2 + \left\|\mathbf{R}_\mathbf{S} \odot (\mathbf{S}\mathbf{\Phi}_\mathbf{S}^\mathbf{T})\right\|_1 \tag{III.14}$$

In accordance to its name, the Forward-Backward Splitting (FBS – [Combettes & Wajs 2005]) method splits the objective into two terms, using the differentiability of h. The corresponding iterations write as :

$$\hat{\mathbf{u}}^{(l+1)} = \operatorname{prox}_{\eta\mathcal{J}}(\hat{\mathbf{u}}^{(l)} - \eta\nabla h(\hat{\mathbf{u}}^{(l)})) \tag{III.15}$$

where $\eta > 0$ is a step size and ∇h is the gradient of h. The name of the algorithm thus stems from the fact that it alternates between a forward / explicit gradient step and a backward / implicit step, namely the application of the proximal operator (special cases include : i) when $\mathcal{J} = 0$, the usual gradient descent ; ii) when $h = 0$, the previous proximal point method).
Provided that h is $\mathcal{L}$-Lipchitz, this methods converges when $\eta \in (0, 2/\mathcal{L})$. It has also to be emphasized that while the FBS algorithm has a convergence rate of $\mathcal{O}(1/l)$, it can be accelerated to $\mathcal{O}(1/l^2)$ using a linear combination of the previous estimates,

which is known as the FISTA algorithm [Beck & Teboulle 2009].
While the FBS already found many applications and will be extensively used throughout this thesis, it is computationaly efficient only if the proximal operator of $\mathcal{J}$ is easy to compute (or, even better, explicit). If it is not the case, but if $\mathcal{J}$ is the sum of proximable terms which proximal operators are easy to compute, the following Generalized Forward Backward Splitting (GFBS) algorithm can be used.

B.3.3 Generalized forward backward splitting

While the FBS algorithm is restricted to the sum of one differentiable and one non-differentiable term, the GFBS [Raguet *et al.* 2013] generalizes to a higher number of non-differentiable terms :

$$\underset{\mathbf{u}\in\mathbb{R}^t}{\operatorname{argmin}}\ h(\mathbf{u}) + \sum_{i=1}^{K} \mathcal{J}_{(i)}(\mathbf{u}) \tag{III.16}$$

with $K \geq 1$ and $\mathcal{J}_{(1)}, ..., \mathcal{J}_{(K)}$ some lower semi-continuous proper and convex function from $\mathbb{R}^t$ to $\mathbb{R} \cup \{\infty\}$. As an example of such a problem, looking for the sources with a fixed $\mathbf{A}$ in BSS when a sparsity and a non-negativity prior are both enforced leads to the following problem :

$$\underset{\mathbf{S}\in\mathbb{R}^{n\times t}}{\operatorname{argmin}}\ \frac{1}{2}\left\|\mathbf{X}-\mathbf{AS}\right\|_F^2 + \left\|\mathbf{R_S}\odot(\mathbf{S}\mathbf{\Phi_S^T})\right\|_1 + \iota_{K^+}(\mathbf{S}) \tag{III.17}$$

where thus $K = 2$ and $\mathcal{J}_{(1)}$ and $\mathcal{J}_{(2)}$ would correspond to the second and third terms respectively. In this example, while the proximal operator of $\mathcal{J}_{(1)}$ and $\mathcal{J}_{(2)}$ can be computed easily (at least, provided that the transform $\mathbf{\Phi_S}$ is orthogonal), it is not the case of the proximal operator of the sum $\mathcal{J}_{(1)} + \mathcal{J}_{(2)}$, which is why a GFBS algorithm is useful.
In brief, the GFBS algorithm uses at each iteration the proximal operator of each individual constraint $\mathcal{J}_{(1)}, ..., \mathcal{J}_{(K)}$. Each iteration can be written as :

for i = 1..K do
$$\mathbf{z}_{(i)} = \mathbf{z}_{(i)} + \mu\left(\operatorname{prox}_{\frac{\eta}{\omega_i}\mathcal{J}_{(i)}}(2\hat{\mathbf{u}}^{(l)} - \mathbf{z}_{(i)} - \eta\nabla h(\hat{\mathbf{u}}^{(l)})) - \hat{\mathbf{u}}^{(l)}\right)$$
$$\hat{\mathbf{u}}^{(l+1)} \leftarrow \textstyle\sum_{i=1}^{b} \omega_i \mathbf{z}_{(i)}$$

where the ω_i are weights in the sum (thus, $\forall i \in [1, K]$, $\omega_i \in (0, 1)$ and $\sum_{i=1}^{K} \omega_i = 1$), $\eta \in (0, 2\mathcal{L})$ is the gradient step, bounded by the Lipschitz constant $\mathcal{L}$ of h, and $\mu \in (0, \min(1.5, 0.5 + \frac{1}{\mathcal{L}\eta}))$. Thus, from the sole knowledge of the proximal operator of the $\mathcal{J}_{(i)}$'s, the GFBS enables to minimize problem III.16.

C Multi-convex optimization for sparse BSS

C.1 Problem statement

The previous section has enlighten how to handle the non-smoothness of matrix factorization problems by using proximal algorithms. We now aim at reviewing some

classical sparse BSS optimization methods. Such algorithms however need to tackle the remaining non-convexity issue of the cost function II.7. Generally speaking, in this part we aim at minimizing cost functions of the following form :

$$\underset{\mathbf{U}_{(1)}\in\mathbb{R}^{n_1\times t_1},...,\mathbf{U}_{(K)}\in\mathbb{R}^{n_K\times t_K}}{\operatorname{argmin}} h(\mathbf{U}_{(1)},...,\mathbf{U}_{(K)}) + \sum_{i=1}^{K} \mathcal{J}_{(i)}(\mathbf{U}_{(i)}) \tag{III.18}$$

with the following assumptions :

— The function $h : \mathbb{R}^{n_1\times t_1}\times\mathbb{R}^{n_2\times t_2}...\times\mathbb{R}^{n_K\times t_K} \to \mathbb{R}$ is assumed to be block multi-convex, that is for all $i \in [1, K]$, h is a convex function of $\mathbf{U}_{(i)}$ while all the other blocks are fixed. We will further write h_i the corresponding convex function over one block : $h_{(i)} : \mathbb{R}^{n_i\times t_i}, \mathbf{U} \to h(\mathbf{U}_{(1)},...,\mathbf{U}_{(i-1)},\mathbf{U},\mathbf{U}_{(i+1)}...,\mathbf{U}_{(K)})$ for any arbitrary $\mathbf{U}_{(1)} \in \mathbb{R}^{n_1\times t_1},...,\mathbf{U}_{(i-1)} \in \mathbb{R}^{n_{i-1}\times t_{i-1}}, \mathbf{U}_{(i+1)} \in \mathbb{R}^{n_{i+1}\times t_{i+1}},...,\mathbf{U}_{(K)} \in \mathbb{R}^{n_K\times t_K}$. In the following, each $h_{(i)}$ is further assumed to be differentiable and its gradient to be $\mathcal{L}_i$-Lipschitz ;

— For all $i \in [1, K]$, the extended-valued functions $\mathcal{J}_{(i)} : \mathbb{R}^{n_i\times t_i} \to \mathbb{R}\cup\{\infty\}$ are assumed to be convex, proper and lower-continuous. While these functions are used to enforce only individual penalizations over the $\mathbf{U}_{(i)}$, we however point out that they can be non-smooth.

Thus, the algorithms we will present in this section encompass a wider setting that the actual BSS problem of II.7, which is a specific case of Eq. III.18 with $K = 2$ and $\mathbf{U}_{(1)} = \mathbf{A}$, $\mathbf{U}_{(2)} = \mathbf{S}$.

C.2 Overview

All the algorithms we present in this section are part of the Gauss-Seidel family. More specifically, they use the multi-convex structure of the cost function (III.18) by performing an (approximate) minimization over each block $\mathbf{U}_{(1)},...,\mathbf{U}_{(K)}$. We will furthermore distinguish two sub-families of algorithms :

— Algorithms trying to minimize *exactly* problem (III.18) :
The multi-convex problem described by Eq. (III.18) can be tackled in its exact form using one of the following three main algorithms : Block-Coordinate Descent (BCD - [Tseng 2001]), Proximal Alternating Linearized Minimization (PALM – [Bolte *et al.* 2014]) and Proximal Block Coordinate (PBC – [Attouch *et al.* 2010]). Interestingly, these algorithms can precisely converge to a critical point of (III.18).

— Algorithms minimizing an *approximation* of problem (III.18) :
Among them, the Projected Alternating Least Squares (pALS) was first introduced in the context of NMF [Paatero & Tapper 1994]. Contrary to the previous algorithms, it does not truly minimize (III.18), but rather an approximation of it. In the context of sparse BSS, we will further discuss one of pALS extensions which has known a wild success : the Generalized Morphological Component Analysis (GMCA [Bobin *et al.* 2007]). GMCA is based

on the pALS framework, but uses an automatic decreasing hyper-parameter strategy along the iterations that yields much more robustness in practice. In addition, while the convergence of GMCA is not guaranteed because it is built on pALS scheme, in practice this heuristic [2] helps it to numerically stabilize.

C.3 Algorithms aiming at finding a (local) minimum

We first review the algorithms that aim at minimizing exactly problem III.18. They all have a strong mathematical background, and in particular they are proved to converge under our assumptions [Xu & Yin 2013] (*cf.* Appendix B for more details concerning this topic). It is however important to emphasize that due to the nonconvexity of the problem, the minimizers yielded by the different algorithms might be different.

C.3.1 BCD

The BCD algorithm [Tseng 2001] was one of the first algorithms to be able to tackle Eq. III.18. It uses the multi-convex structure of the problem by solving exactly and cyclicly convex subproblems of the form

$$\underset{\mathbf{U}_{(i)}\in\mathbb{R}^{n_i\times t_i}}{\operatorname{argmin}} \; h_{(i)}(\mathbf{U}_{(i)}) + \mathcal{J}_{(i)}(\mathbf{U}_{(i)}) \tag{III.19}$$

Each of these subproblems can then be handled for instance by the usual proximal algorithms presented in section B.3, depending on the difficulty to compute the proximal operator of $\mathcal{J}_{(i)}$ (in particular, if such a proximal operator is explicit, a FBS [Combettes & Wajs 2005] or one of its accelerated version is a straightforward choice).

BCD is summarized in Algorithm 1, where $h_{(i)}^{(l)}$ is a short-hand for $h_{(i)}^{(l)} : \mathbb{R}^{n_i\times t_i}, \mathbf{U} \to h(\mathbf{U}_{(1)}^{(l)}, ..., \mathbf{U}_{(i-1)}^{(l)}, \mathbf{U}, \mathbf{U}_{(i+1)}^{(l-1)} ..., \mathbf{U}_{(K)}^{(l-1)})$, that is $h_{(i)}$ with the fixed matrices chosen as the previously updated matrices.

While the convergence conditions are in practice verified in sparse BSS, more generally the minimum in each BCD step must be uniquely attained to prove the convergence [Zangwill 1969], otherwise the method may cycle indefinitely [Powell 1973]. Assuming a strict convexity of each subproblem III.19 then enables to prove that every limit point of the sequence of iterates is a critical point of III.18.

2. By heuristic, we here mean an approximate method enabling to obtain non-optimal but still decent results, while alleviating a difficulty of the initial problem (*e.g.* time computation issues, difficult hyper-parameter choice, need to perform relaunches with different initializations...). As such, the decreasing parameter choice of GMCA is an heuristic in that the parameters should rigorously stay fixed during the whole algorithm to minimize Eq. II.8 : we shall see in Chapter IV that the corresponding estimates $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$ are good while sub-optimal compared to what could be obtained with optimal fixed regularization parameters. This heuristic however alleviate the cumbersome search of regularization parameters.

Algorithm 1 BCD

function BCD($\hat{\mathbf{U}}_{(1)}^{(0)}, ..., \hat{\mathbf{U}}_{(K)}^{(0)}$) ▷ $\hat{\mathbf{U}}_{(1)}^{(0)}, ..., \hat{\mathbf{U}}_{(K)}^{(0)}$ is the initialization
  while not converged **do**
    for $i = 1...K$ **do**
      $\hat{\mathbf{U}}_{(i)}^{(l+1)} = \underset{\hat{\mathbf{U}}_{(i)}}{\text{argmin}}\ h_{(i)}^{(l)}(\hat{\mathbf{U}}_{(i)}) + \mathcal{J}_{(i)}(\hat{\mathbf{U}}_{(i)})$
    end for
    $l \leftarrow l + 1$
  end while
  return $\hat{\mathbf{U}}_{(1)}^{(l)}, ..., \hat{\mathbf{U}}_{(K)}^{(l)}$
end function

C.3.2 PBC

Removing this strict convexity assumption can be done by relaxing the minimization of each subproblem III.19 with a proximal term [Attouch *et al.* 2010], yielding new subproblems that each corresponds to a local minimization :

$$\underset{\mathbf{U}_{(i)} \in \mathbb{R}^{n_i \times t_i}}{\text{argmin}}\ h_{(i)}^{(l)}(\mathbf{U}_{(i)}) + \mathcal{J}_{(i)}(\mathbf{U}_{(i)}) + \frac{1}{2\mathcal{L}_{(i)}^{(l)}} \left\| \mathbf{U}_{(i)} - \mathbf{U}_{(i)}^{(l)} \right\|^2 \tag{III.20}$$

With the $\mathcal{L}_{(i)}^{(l)}$ positive numbers (that are finite). This is, by definition of proximal operators, equivalent to :

$$\underset{\mathcal{L}_{(i)} \times (h_{(i)}^{(l)} + \mathcal{J}_{(i)})}{\text{prox}} (\mathbf{U}_{(i)}^{(l)}) \tag{III.21}$$

Hence the name of the Proximal Block Coordinate – PBC – algorithm. This leads to the iterative scheme of Algorithm 2.

Algorithm 2 PBC

function PBC($\hat{\mathbf{U}}_{(1)}^{(0)}, ..., \hat{\mathbf{U}}_{(K)}^{(0)}$) ▷ $\hat{\mathbf{U}}_{(1)}^{(0)}, ..., \hat{\mathbf{U}}_{(K)}^{(0)}$ is the initialization
  while not converged **do**
    for $i = 1...K$ **do**
      $\hat{\mathbf{U}}_{(i)}^{(l+1)} = \underset{\mathcal{L}_{(i)}^{(l)} \times (h_{(i)}^{(l)} + \mathcal{J}_{(i)})}{\text{prox}} (\hat{\mathbf{U}}_{(i)}^{(l)})$
    end for
    $l \leftarrow l + 1$
  end while
  return $\hat{\mathbf{U}}_{(1)}^{(l)}, ..., \hat{\mathbf{U}}_{(K)}^{(l)}$
end function

While this algorithm has interesting properties (in particular, it is proved to converge under mild conditions and might be intuitively faster than BCD [Chenot 2017]), it has not been extensively used in sparse BSS. This might be linked to the fact that it requires the proximal operator of $h_{(i)}^{(l)} + \mathcal{J}_{(i)}$ to be explicit to yield computationally

cheap updates. If this is not the case, such a proximal operator must be computed using a subroutine (*e.g.* a FBS), which however leads to the non-trivial issue of computational error accumulations in each step. Thus, it has been claimed in [Bolte *et al.* 2014] that such a scheme is mainly "conceptual", calling instead for the PALM algorithm. Therefore, the PBC will not be studied in this work.

C.3.3 PALM

PALM has been introduced in [Bolte *et al.* 2014, Xu & Yin 2014] and has driven an intensive research with a large number of applications and extensions [3] in the scope of matrix factorization [Chouzenoux *et al.* 2014, Chouzenoux *et al.* 2016]. It can be seen as the merging of PBC with the FBS algorithm, or said differently, as an alternating minimization approach for the FBS.
Compared to BCD, instead of fully minimizing the subproblems III.19, PALM minimizes a local regularization of the Gauss-Seidel scheme. However, in contrast to PBC, it uses a proximal linearization of each subproblem. Therefore, it avoids the entirely implicit step required by PBC. More specifically, the minimization performed by PALM is :

$$\underset{\mathbf{U}_{(i)}\in\mathbb{R}^{n_i\times t_i}}{\operatorname{argmin}} \; h_{(i)}^{(l)}(\mathbf{U}_{(i)}^{(l)}) + \langle \nabla h_{(i)}^{(l)}(\mathbf{U}_{(i)}^{(l)}) | \mathbf{U}_{(i)} - \mathbf{U}_{(i)}^{(l)} \rangle + \mathcal{J}_{(i)}(\mathbf{U}_{(i)}) + \frac{\mathcal{L}_{(i)}^{(l)}}{2} \left\| \mathbf{U}_{(i)} - \mathbf{U}_{(i)}^{(l)} \right\|^2 \tag{III.22}$$

where $\mathcal{L}_{(i)}$ is a constant that we will take equal to the Lipschitz constant of $\nabla h_{(i)}^{(l)}$ in the following. Compared to BCD, the advantage is that due to the linearization, $\mathbf{U}_{(i)}$ do not appear anymore in the first term. Thus, the update can be rewritten using only the proximal operator of $\mathcal{J}_{(i)}$:

$$\underset{\frac{\mathcal{J}_{(i)}}{\mathcal{L}_{(i)}^{(l)}}}{\operatorname{prox}} \left(\mathbf{U}_{(i)}^{(l)} - \frac{1}{\mathcal{L}_{(i)}^{(l)}} \nabla h_{(i)}^{(l)} \left(\mathbf{U}_{(i)}^{(l)} \right) \right) \tag{III.23}$$

Therefore, for each block of coordinates, PALM performs one gradient step on the smooth part, while a proximal step is taken on the non-smooth part. Thus, provided that the proximal operators of the individual regularization terms $\mathcal{J}_{(i)}$ are explicit, PALM iterations have a low computational cost. In particular, it can prove in some settings to be faster than BCD, since it does not require to perform a full minimization of each subproblem at all iterations [Xu & Yin 2013]. The whole scheme is summarized in Algorithm 3.

3. Among these extensions, some accelerated versions of PALM have been proposed (*cf. e.g.* [Pock & Sabach 2016, Hien *et al.* 2019]). In this work and due to the difficulty of performing sparse BSS using PALM (*cf.* Chapter IV), we do not however consider such accelerations, which is left for future works.

Algorithm 3 PALM

function PALM($\hat{\mathbf{U}}_{(1)}^{(0)}, ..., \hat{\mathbf{U}}_{(K)}^{(0)}$) ▷ $\hat{\mathbf{U}}_{(1)}^{(0)}, ..., \hat{\mathbf{U}}_{(K)}^{(0)}$ is the initialization
  while not converged **do**
    for $i = 1...K$ **do**
      $\hat{\mathbf{U}}_{(i)}^{(l+1)} = \underset{\frac{\mathcal{J}_{(i)}}{L_{(i)}}}{\text{prox}} \left(\hat{\mathbf{U}}_{(i)}^{(l)} - \frac{1}{L_{(i)}} \nabla h_{(i)}^{(l)} \left(\hat{\mathbf{U}}_{(i)}^{(l)} \right) \right)$
    end for
    $l \leftarrow l + 1$
  end while
  return $\hat{\mathbf{U}}_{(1)}^{(l)}, ..., \hat{\mathbf{U}}_{(K)}^{(l)}$
end function

Example of application of PALM : sparse BSS Since PALM will be extensively used in this work, we here derive it for the specific case of sparse BSS corresponding to Eq. (II.8), which is done using the examples of proximal operators given in section B.2 :

PALM($\hat{\mathbf{A}}^{(0)}, \hat{\mathbf{S}}^{(0)}$)
Requires : $\mathbf{X}, \mathbf{R_S}$
While the stopping criterion $\Delta^{(l)}$ has not reached the desired value, iterate over (l) :

1 - Update of $\mathbf{S}$ using the current version of $\hat{\mathbf{A}}^{(l-1)}$:

$$\tilde{\mathbf{S}} = \hat{\mathbf{S}}^{(l-1)} - \frac{\gamma}{\left\| \hat{\mathbf{A}}^{(l-1)T} \hat{\mathbf{A}}^{(l-1)} \right\|_2} \hat{\mathbf{A}}^{(l-1)T} (\hat{\mathbf{A}}^{(l-1)} \hat{\mathbf{S}}^{(l-1)} - \mathbf{X}) \tag{III.24}$$

$$\hat{\mathbf{S}}^{(l)} = \mathcal{S}_{\frac{\gamma \mathbf{R_S}}{\left\| \hat{\mathbf{A}}^{(l-1)T} \hat{\mathbf{A}}^{(l-1)} \right\|_2}} (\tilde{\mathbf{S}} \mathbf{\Phi}_{\mathbf{S}}^{\mathbf{T}}) \mathbf{\Phi_S} \tag{III.25}$$

2 - Update of $\mathbf{A}$ using the current version of $\hat{\mathbf{S}}^{(l)}$:

$$\tilde{\mathbf{A}} = \hat{\mathbf{A}}^{(l-1)} - \frac{\delta}{\left\| \hat{\mathbf{S}}^{(l)} \hat{\mathbf{S}}^{(l)T} \right\|_2} (\hat{\mathbf{A}}^{(l-1)} \hat{\mathbf{S}}^{(l)} - \mathbf{X}) \hat{\mathbf{S}}^{(l)T} \tag{III.26}$$

$$\hat{\mathbf{A}}^{(l)} = \Pi_{\|.\|_2 = 1} (\tilde{\mathbf{A}}) \tag{III.27}$$

3 - Update stopping criterion $\Delta^{(l)} = \min_{j \in [1,n]} \langle \frac{\hat{\mathbf{A}}^{(l)j}}{\left\| \hat{\mathbf{A}}^{(l)j} \right\|_F}, \frac{\hat{\mathbf{A}}^{(l-1)j}}{\left\| \hat{\mathbf{A}}^{(l-1)j} \right\|_F} \rangle$

The adaptation of PALM to our cost function is detailed below :

— Update of $\mathbf{S}$: the term $\hat{\mathbf{A}}^{(l-1)T} (\hat{\mathbf{A}}^{(l-1)} \hat{\mathbf{S}}^{(l-1)} - \mathbf{X})$ is the gradient of the data fidelity term with respect to $\mathbf{S}$ and a Lipschitz modulus upper bound can be chosen as $\left\| \hat{\mathbf{A}}^{(l-1)T} \hat{\mathbf{A}}^{(l-1)} \right\|_2$, where $\|.\|_2$ is the spectral norm [4]. The parameter

4. More specifically, if $\mathbf{U}$ is a matrix, $\mathbf{x}$ a vector and $\|.\|_{\ell_2}$ is the ℓ_2 norm for vectors, the $\|.\|_2$

γ is chosen in the range $(0, 1)$, which is required to ensure the convergence [5].

— Update of $\mathbf{A}$: the term $(\hat{\mathbf{A}}^{(l-1)}\hat{\mathbf{S}}^{(l)} - \mathbf{X})\hat{\mathbf{S}}^{(l)T}$ is the gradient of the data fidelity term with respect to $\mathbf{S}$ and a Lipschitz modulus upper bound can be chosen as $\left\|\hat{\mathbf{S}}^{(l)}\hat{\mathbf{S}}^{(l)T}\right\|_2$. The parameter δ is chosen in the range $(0, 1)$, which is required to ensure the convergence.

— Stopping criterion : the stopping criterion $\Delta^{(l)}$ was chosen in this thesis as the cosine of the maximum angle between the columns of $\hat{\mathbf{A}}^{(l)}$ and that of the previous estimate of the mixing matrix $\hat{\mathbf{A}}^{(l-1)}$. The algorithm stops when $\Delta^{(l)}$ becomes higher than a threshold τ fixed by the user, that is when the changes in $\mathbf{A}$ become very small.

N.B. : while the blocks here were chosen as $\mathbf{U}_{(1)} = \mathbf{A}$ and $\mathbf{U}_{(2)} = \mathbf{S}$, other choices of coordinates are possible, which will be detailed in chapter V.

C.4 Algorithms minimizing an approximation of the cost function

We will now review some approximate algorithms. While such algorithms can both be cheaper and easier to use than exact ones and obtain good practical results, they however suffer from much less mathematical support.

C.4.1 pALS

The projected Alternating Least Square has first been introduced in the context of NMF [Berry *et al.* 2007] by [Paatero & Tapper 1994]. [6] As all the previous algorithms, it performs a cyclic update over all the coordinates. However, it does not perform a true minimization of each subproblem III.19 as in BCD, but rather uses a rough approximation : for a given $i \in [1, K]$, the subproblem is first minimized without taking into account the constraint $\mathcal{J}_{(i)}$; then the solution is projected through the proximal operator of $\mathcal{J}_{(i)}$. The whole procedure is described in Algorithm 4.

While this algorithm can be interesting in many practical setting, especially when the minimization of $h^{(l)}_{(i)}(\mathbf{U}_{(i)})$ is explicit as well as the proximal operator of $\mathcal{J}_{(i)}$, it suffers from a lack of mathematical grounding. In particular, it is neither proved to converge nor to minimize the cost function III.18 in general. Indeed, even in the historical case of NMF, this algorithm can increase the cost function when performing the minimization over one fixed variable, which precludes any convergence guarantees in the BCD framework [Kim *et al.* 2008] (*cf.* Appendix B).

induced matrix norm is defined as :

$$\|\mathbf{U}\|_2 = \sup_{x \neq 0} \frac{\|\mathbf{U}\mathbf{x}\|_{\ell_2}}{\|\mathbf{x}\|_{\ell_2}} \tag{III.28}$$

5. Note that such a parameter might influence the quality of the solution found by the algorithm, due to the non-convexity of the cost function. Such a study is however beyond the scope of this work.

6. While it historically started with the standard projected least-square approach, h being a quadratic loss, we will present it here in a more general setting for the sake of continuity with the previous subsection.

Algorithm 4 pALS

function pALS($\hat{\mathbf{U}}_{(1)}^{(0)}, ..., \hat{\mathbf{U}}_{(K)}^{(0)}$) ▷ $\hat{\mathbf{U}}_{(1)}^{(0)}, ..., \hat{\mathbf{U}}_{(K)}^{(0)}$ is the initialization
  while not stabilized or maximum number of iterations not reached **do**
    for $i = 1...K$ **do**
      $\tilde{\mathbf{U}}_{(i)}^{(l+1)} = \underset{\hat{\mathbf{U}}_{(i)}}{\text{argmin}}\ h_{(i)}^{(l)}(\hat{\mathbf{U}}_{(i)})$
      $\hat{\mathbf{U}}_{(i)}^{(l+1)} = \underset{\mathcal{J}_{(i)}}{\text{prox}}\left(\tilde{\mathbf{U}}_{(i)}^{(l+1)}\right)$
    end for
    $l \leftarrow l + 1$
  end while
  return $\hat{\mathbf{U}}_{(1)}^{(l)}, ..., \hat{\mathbf{U}}_{(K)}^{(l)}$
end function

Despite this lack of mathematical guarantees, its simplicity has however made pALS successful in the NMF community. Indeed, when the constraints $\mathcal{J}_{(i)}$ are fairly simple, it can give good practical results. Concerning sparse BSS, pALS has known a huge success in the context of the GMCA algorithm. However, when the constraints become more complicated than mere NMF or sparsity in the direct domain (such as for instance trying to combine both non negativity in the direct domain and sparsity in another domain using redundant transforms), its separation performances can deteriorate in comparison to a true BCD [Rapin *et al.* 2013].

pALS in sparse BSS Since the pALS is the basis of the GMCA algorithm [Bobin *et al.* 2007], which is the core of this PhD (with PALM), we here detail it in the context of sparse BSS. However and as explained above, contrary to the PALM algorithm of section C.3.3, pALS does not truly look for a minimizer of II.8 but rather for a minimizer of an *approximation* of the cost function. In that, the sparsity parameters of pALS do not fully correspond to $\mathbf{R_S}$, the ones of II.8, and we shall denote them as $\mathbf{M_S}$. The algorithm is then given by (for the sake of simplicity, $\mathbf{\Phi_S}$ is supposed to be the identity matrix) :

pALS($\hat{\mathbf{A}}^{(0)}, \hat{\mathbf{S}}^{(0)}$)
Requires $\mathbf{X}, \mathbf{M_S}$
While not stabilized or maximum number of iterations not reached, iterate over (l) :

1 - $\mathbf{S}$ is updated using the current $\hat{\mathbf{A}}^{(l-1)}$.

$$\hat{\mathbf{S}}^{(l)} = \mathcal{S}_{\mathbf{M_S}}\left(\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{X}\right) \tag{III.29}$$

2 - $\mathbf{A}$ is updated using the current $\hat{\mathbf{S}}^{(l)}$:

$$\hat{\mathbf{A}}^{(l)} = \Pi_{\|.\|_2=1}\left(\mathbf{X}\hat{\mathbf{S}}^{(l)\dagger}\right) \tag{III.30}$$

In this algorithm, the minimization of the quadratic data fidelity term of II.8 over one of the two matrices $\mathbf{A}, \mathbf{S}$ is explicit and is performed using the Moore-Penrose pseudo-inverse [Ben-Israel & Greville 2003] of the fixed matrix, denoted as †.
As a side remark, for this specific application of pALS to sparse BSS and in the case where at each iteration l, $\hat{\mathbf{A}}^{(l)}$ and $\hat{\mathbf{S}}^{(l)}$ are orthogonal, the pseudo-inverse of the matrices is equal to the transpose. Thus, the PALM algorithm trivially reduces to the pALS, and as such in this case the pALS both truly minimizes the cost function II.8 and is proved to converge (and $\mathbf{M_S} = \mathbf{R_S}$). [7]

C.4.2 GMCA as an optimized pALS for sparse BSS

GMCA is not a new optimization framework, but rather an enhancement of the pALS algorithm in the sparse BSS case of II.7. As explained in the previous subsection, while the pALS framework is appealing due to its simplicity and interpretability, it nevertheless suffers from several issues (both theoretical and practical, such as a lack of robustness when the thresholds are too low [Chenot 2017]). The GMCA algorithm [Bobin *et al.* 2007] enables to both assuage such issues and to yield good practical separation results [Bobin *et al.* 2008, Bobin *et al.* 2015], while alleviating the cumbersome hyperparameter choice.
More specifically, while GMCA is very similar to the algorithm of section C.4.1, its main strength is to propose an automatic adaptive parameter choice for $\mathbf{M_S}$, which is based on a fixed point argument and further enables to benefit from the morphological diversity assumption [Bobin *et al.* 2007]. In brief (the required qualities needed for a good hyperparameter choice will be more extensively discussed in chapter IV), such an automatic choice enables pALS to be more robust to the noise, more reliable (insensitive to the initialization) and more accurate.
In the following, we will first state the morphological diversity principle and then the GMCA automatic hyper-parameter choice.

Morphological diversity The morphological diversity dates back to mono-channel component separation, and has first been proposed in the context of morphological component analysis (MCA) [Starck *et al.* 2010]. In brief, in MCA the mixing $\mathbf{x} \in \mathbb{R}^t$ is assumed to be a linear combination of morphological components $\mathbf{x}_i^* \in \mathbb{R}^t$: $\mathbf{x} = \sum_{i=1}^n \mathbf{x}_i^*$. Each of this components $\mathbf{x}_i^*$ is assumed to be sparsest in its *own* dictionary $\mathbf{\Phi_i}$, all the $\mathbf{\Phi_i}$ being *different*, which corresponds to the assumption that all the $\mathbf{x}_i^*$ have different morphologies (said differently, different kinds of geometrical features). For instance, in the case $n = 2$, $\mathbf{x}_1^*$ could be broadly distributed while $\mathbf{x}_2^*$ could resemble punctual sources : thus, $\mathbf{x}_1^*$ and $\mathbf{x}_2^*$ would be sparsest in different dictionaries, encoding different kind of geometrical features. Using such an information

7. Note that instead of choosing the blocks as $\mathbf{A}$ and $\mathbf{S}$ but rather splitting the problem into $2n$ blocks corresponding to the columns of $\mathbf{A}$ and the rows of $\mathbf{S}$, the update is explicit and pALS also becomes exact (since $\forall i \in [0, n], \left\|\mathbf{A}^i\right\|_F = 1$ and $\mathbf{A}^{i^\dagger} = \mathbf{A}^{i^T}$).

then enables to recover $\mathbf{x}_1^*, ..., \mathbf{x}_n^*$ from $\mathbf{x}$ by looking for a minimizer of :

$$\underset{\mathbf{x}_1 \in \mathbb{R}^t, ..., \mathbf{x}_n \in \mathbb{R}^t}{\operatorname{argmin}} \left\| \mathbf{x} - \sum_{i=1}^{n} \mathbf{x}_i \right\|_{\ell_2}^2 + \sum_{i=1}^{n} \lambda_i \left\| \mathbf{x}_i \mathbf{\Phi}_\mathbf{i}^\mathbf{T} \right\|_0 \tag{III.31}$$

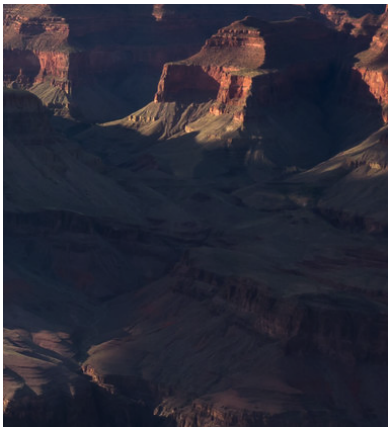
On the other hand, this principle has been extended for sparse BSS to the case of components having the *same* kind of geometrical features, that is, that are sparse in a *same* dictionary $\mathbf{\Phi_S}$. In that case, sparsity can still be used to discriminate the components, which are now the sources $\mathbf{S}_i^*$. As the $\mathbf{S}_i^*$ are sparse in $\mathbf{\Phi_S}$, the information is encoded in a small number of significant entries. Since the $\mathbf{S}_i^*$ are however different from each other, these significant entries are also likely to differ through the position at which they are active. Therefore, the discrimination between the sources can rely on their most significant coefficients in $\mathbf{\Phi_S}$ (*cf. e.g.* Fig. III.1, where the support of the largest wavelet samples are almost disjoint).

It is important to point out that many practical sources satisfies the morphological diversity hypothesis. In particular, i.i.d. sources drawn according to a Bernoulli-Gaussian, a Laplacian, a Generalized Gaussian with parameter $\alpha \leq 1$ or a strongly sub-gaussian distribution respect such an assumption.
In the remaining of this subsection, to explain the GMCA algorithm, we assume for the sake of clarity that $\mathbf{\Phi_S^T} = \mathbf{I_d}$, that is the sources are sparse in the direct domain. The generalization to sparsity in transformed domain is relatively straightforward.

Another point of view on sparse BSS : a geometrical interpretation Just before explaining how GMCA draws on the concept of morphological diversity to disentangle the sources, let us turn towards the geometrical interpretation of sparse BSS by starting back from the example of Fig. III.1. If we draw the scatter plot of the source $\mathbf{S}_1^*$ wavelet samples as a function of the ones of $\mathbf{S}_2^*$, we will get a star shape (*cf.* Fig. III.2) having its highest amplitude samples lying on the axes. This is typical of sources respecting the morphological diversity assumption, since the highest amplitude samples supports are (almost) disjoint. On the other hand, any mixing by a non-trivial $\mathbf{A}^*$ will break such a shape since the coefficients of the largest amplitude samples will be simultaneously large (*cf.* Fig. III.2). Such a loss of compressibility can be measured through the ℓ_1 norm : the samples of the sources are enclosed into a ℓ_1 norm ball of smaller radius than the ones of the mixing. Thus, a good unmixing matrix $\hat{\mathbf{A}}$ should ensure that the unmixing $\tilde{\mathbf{S}} = \hat{\mathbf{A}}^\dagger \mathbf{X} = \hat{\mathbf{A}}^\dagger \mathbf{A}^* \mathbf{X}$ lies into a ℓ_1 norm of smallest radius.

GMCA : a pALS scheme enhanced with an heuristic threshold choice We can now explain how the GMCA algorithm takes advantage from the morphological diversity to perform sparse BSS. We will then detail the algorithm through a second interpretation in terms of a fixed point condition and noise removal.

A decreasing threshold choice enabling to explicitly benefit from the morphological diversity assumption As stated just above, the sparse BSS



Source 1

Source 2

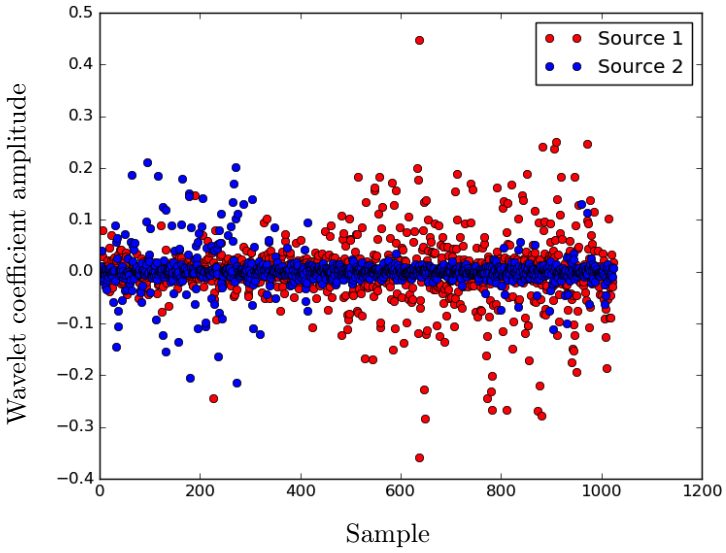


Figure III.1 – *Up :* Two natural images which are sparse in the same transformed domain. *Down :* Wavelet coefficients of the images (to obtain such coefficients, the images have been transformed into grayscale and resized). As hoped due to the morphological diversity hypothesis, since the two images are different the support of their most powerful wavelet samples are different (and here, they are almost disjoint).

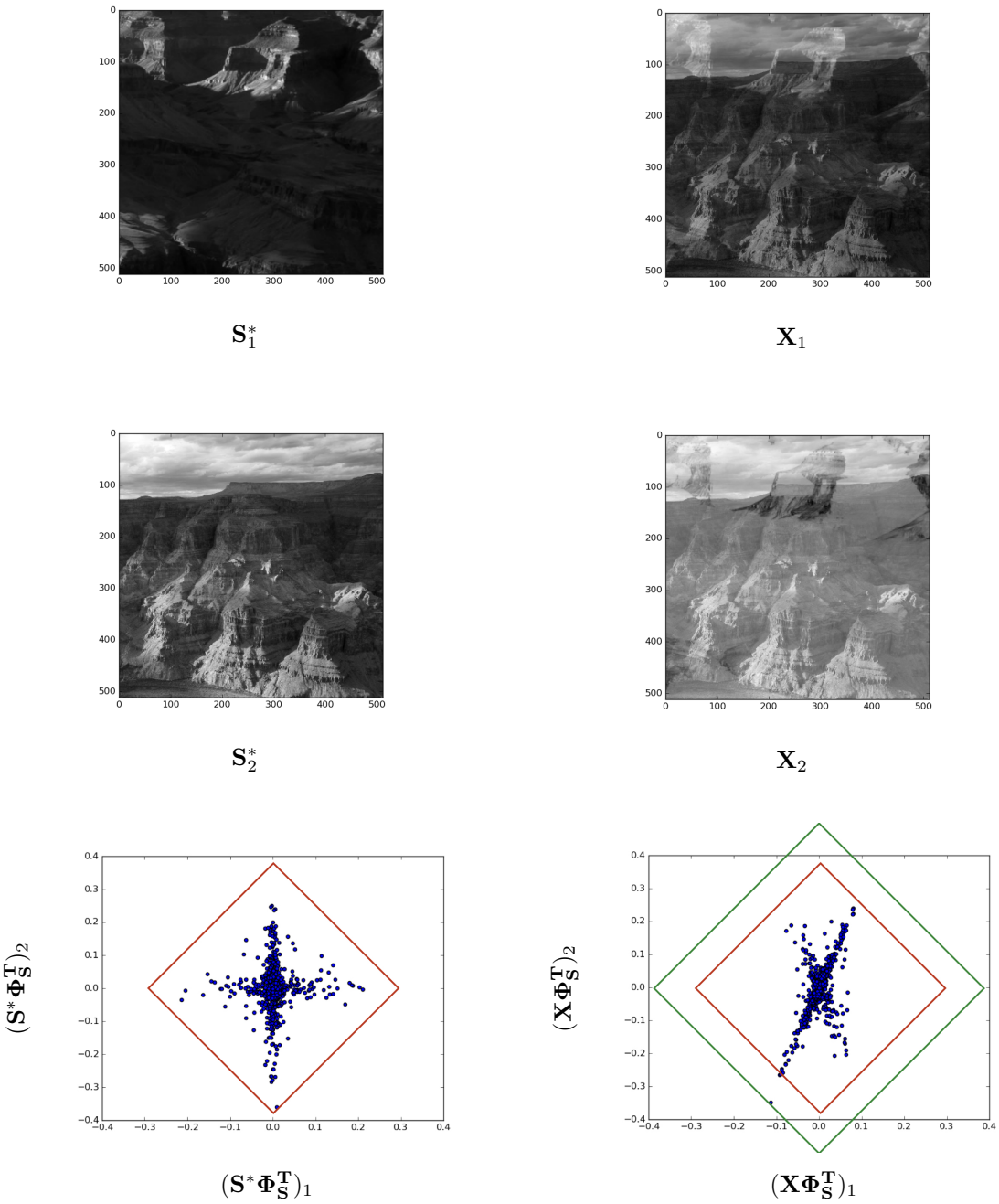


Figure III.2 – *Left, up and middle :* Two sources $\mathbf{S}_1^*$ and $\mathbf{S}_2^*$. *Left, down :* Scatter plot of the wavelet coefficients of the two sources. The red square corresponds to the ℓ_1 ball of minimum radius enclosing all the source wavelet coefficients. *Right, up and middle :* Two mixings $\mathbf{X}_1$ and $\mathbf{X}_2$ obtained from $\mathbf{S}_1^*$ and $\mathbf{S}_2^*$. *Right, down :* Scatter plot of the wavelet coefficients of the two mixings. The green square corresponds to the ℓ_1 ball of minimum radius enclosing all the mixing wavelet coefficients. As expected, its radius is larger than the one of the ℓ_1 ball found with the source wavelet coefficients, in red.

problem geometrically amounts to find $\hat{\mathbf{A}}$ corresponding to an unmixing lying within the smallest radius ℓ_1 ball. Due to the morphological diversity assumption, it is of uttermost importance to note that such a ball is mainly determined by the highest amplitude source samples. This is the core idea behind GMCA : emphasizing these few high amplitude samples enables an enhanced separation quality. Placing such an emphasis can be done by discarding all the lowest amplitude samples and setting them to zero, which corresponds to setting high hyperparameters $\mathbf{M_S}$ [8]. We would however suffer from a difficult trade-off :

— *If too many samples are kept within the estimation process* : taking into account the lowest amplitude samples can worsen the separation results since : i) due to the morphological diversity assumption, the lowest samples are the least discriminative (the support can be very joint between the lowest amplitude samples of the sources) ; ii) the lowest amplitude coefficients are highly likely to be mainly due to the noise $\mathbf{N}$.

— *If too few samples are kept* : taking only the very highest amplitude coefficients is treacherous since i) the morphological diversity assumption can be an approximation, and small partial correlations [9] can occur even for the highest amplitude samples ; ii) taking into account too few samples can create a lack of statistics, making the problem much more ill-posed.

To bypass this difficult trade-off, GMCA proposes an *adaptive* hyperparameter $\mathbf{M_S}$ choice, by making them decrease along the iterations. In the first iterations, only the most prominent samples of $\mathbf{A}^{(l-1)\dagger}\mathbf{X}$ are kept, and the lower amplitude samples are increasingly added within the estimation process to increase the statistics. Since the $\mathbf{M_S}$ hyperparameters are thus varying along the iterations l, we shall now denote them as $\mathbf{M_S}^{(l)}$ in the following.

A final threshold choice based on a fixed point interpretation We now have justified the principle of decreasing thresholds through the morphological diversity. Beyond this general decreasing principle, a natural remaining question is : which value to give to the $\mathbf{M_S}^{(l)}$ in practice ?
In GMCA, such values are chosen according to a fixed point argument. More specifically, since the algorithm is based on pALS, the thresholding applies to the least-square estimate of the sources (*cf.* Sec. C.4.1). If we assume that after many iterations the algorithm has ultimately stabilized on an estimate $\hat{\mathbf{A}}^{(\infty)}$ close to the true mixing matrix $\mathbf{A}^*$, the corresponding source update before thresholding is given

8. Since high $\mathbf{M_S}$ correspond to a high thresholding $\mathcal{S}_{\mathbf{M_S}}$ in the pALS algorithm.
9. That is, samples with multiple active coefficients.

by [10] :

$$\begin{aligned}\tilde{\mathbf{S}}^{(\infty)} &= \hat{\mathbf{A}}^{(\infty)\dagger}\mathbf{X} \\ &\simeq \mathbf{A}^{*\dagger}\mathbf{X} \\ &= \mathbf{A}^{*\dagger}(\mathbf{A}^*\mathbf{S}^* + \mathbf{N}) \\ &= \mathbf{S}^* + \mathbf{A}^{\dagger *}\mathbf{N}\end{aligned} \tag{III.32}$$

Where $\mathbf{A}^{\dagger *}\mathbf{N}$ is a Gaussian noise (since it was assumed that $\mathbf{N}$ was Gaussian). That is, the sources before thresholding $\tilde{\mathbf{S}}^{(\infty)}$ are equal to the true ones $\mathbf{S}^*$ up to an additive Gaussian noise. Thus, the ${\mathbf{M}_\mathbf{S}}^{(\infty)}$ hyper-parameter choice can be understood as choosing the hyper-parameters of a sparse signal Gaussian denoising problem, which has been well studied.

More specifically, let us assume that no reweighted ℓ_1 is used and that ${\mathbf{M}_\mathbf{S}}^{(\infty)} = \text{Diag}(\mu_1^{(\infty)}, \mu_2^{(\infty)}, ..., \mu_n^{(\infty)})\mathbf{1}_{n\times t}$. For each source $\tilde{\mathbf{S}}_i^{(\infty)}, i \in [1, n]$, the threshold μ_i is chosen such that it aims at setting to 0 the small coefficients of $\tilde{\mathbf{S}}^{(\infty)}$ that should mainly correspond to the noise $(\mathbf{A}^{\dagger *}\mathbf{N})_i$. Such a choice is usually performed through a detection procedure using the "$\kappa\sigma$ " rule, $\kappa \in \mathbb{R}^+$. For instance, if $\kappa = 3$, the probability that a coefficient of $\tilde{\mathbf{S}}_i^{(\infty)}$ with a larger amplitude than $3\sigma_i$, with σ_i the standard deviation of the Gaussian noise $(\mathbf{A}^{*\dagger}\mathbf{N})_i$, corresponds to noise only is roughly 0.4%.

A first practical difficulty is however that the standard deviation σ_i of the noise $(\mathbf{A}^{\dagger *}\mathbf{N})_i$ is unknown (if only because $\mathbf{A}^*$ itself is not known). It can fortunately be approximated using the Median Absolute Deviation, defined as :

$$\forall \mathbf{u} \in \mathbb{R}^t, \text{mad}(\mathbf{u}) = \underset{i\in[1,t]}{\text{median}} \, |\mathbf{u}_i - \underset{i\in[1,t]}{\text{median}}(\mathbf{u}_i)| \tag{III.33}$$

We further extend this definition to matrices, by taking their row-wise mad : MAD : $\mathbb{R}^{n\times t} \to \mathbb{R}^n$, such that $\forall i \in [1, n], \forall \mathbf{U} \in \mathbb{R}^{n\times t}, \text{MAD}(\mathbf{U})_i = \text{mad}(\mathbf{U}_i)$.

In our case, we have that $\sigma_i \simeq 1.48 \times \text{MAD}((\mathbf{A}^{\dagger *}\mathbf{N})_i)$. Furthermore, since $\mathbf{S}^*$ is assumed to be sparse, the MAD operator is quite insensitive to it, and it is thus possible to directly estimate $\kappa\sigma_i$ from $\tilde{\mathbf{S}}^{(\infty)}$, as $\kappa\,\text{MAD}(\tilde{\mathbf{S}}^{(\infty)}) \simeq \kappa\,\text{MAD}(\mathbf{S}^* + \mathbf{A}^{\dagger *}\mathbf{N}) \simeq \kappa\,\text{MAD}(\mathbf{A}^{\dagger *}\mathbf{N}) = (\mu_1^{(\infty)}, \mu_2^{(\infty)}, ..., \mu_n^{(\infty)})^T$.

A second, more challenging difficulty is that we have made the assumption that the hyperparameter ${\mathbf{M}_\mathbf{S}}^{(\infty)}$ were chosen according to the point where the algorithm stabilizes, which rises two issues :

— This requires to know the result of the algorithm before launching it ;

— This further requires that the result of the algorithm will be good in the sense that $\hat{\mathbf{A}}^{(\infty)} \simeq \mathbf{A}^*$, which is not trivial since $\mathbf{A}^*$ is precisely unknown.

To bypass such pitfalls, the hyperparameter choice in GMCA is done using the *current*, at iteration l, estimation of $\tilde{\mathbf{S}}^{(l)}$:

$$\left(\mu_1^{(l)}, \mu_2^{(l)}, ..., \mu_n^{(l)}\right)^T = \kappa \times \text{MAD}(\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{X}) \tag{III.34}$$

10. We point out that while notations such as $\mathbf{A}^{(\infty)}$ and $\mathbf{S}^{(\infty)}$ are slightly abusive in this context because GMCA is not proved to converge, the goal is to clarify the rational of hyper-parameter choice.

This is particularly appealing since :

— It alleviate the need to know the result of the algorithm ;

— It draws on the conclusions of the previous subsection, namely that an adaptive *decreasing* threshold choice enables to take into account the morphological diversity assumption. Indeed, we have that :

$$\left(\mu_1^{(l)}, \mu_2^{(l)}, ..., \mu_n^{(l)}\right)^T = \kappa \times \text{MAD}(\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{X}) = \kappa \times \text{MAD}(\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{A}^*\mathbf{S}^* + \hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{N}) \tag{III.35}$$

Thus, as during the first iterations $\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{A}^* \neq \mathbf{I_d}$, the term $\text{MAD}(\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{A}^*\mathbf{S}^*)$ is non zero (it is the MAD of a mixing of sparse signal, which is itself non-sparse), implying high $\mu_1^{(l)}, \mu_2^{(l)}, ..., \mu_n^{(l)}$. When the estimation improves, $\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{A}^*$ becomes closer to the identity and thus the term $\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{A}^*\mathbf{S}^*$ becomes sparser, and the corresponding $\mu_1^{(l)}, \mu_2^{(l)}, ..., \mu_n^{(l)}$ lower.

GMCA, summary GMCA [Bobin *et al.* 2007] can be written in the following way. The structure is similar to pALS, but the scheme is enhanced with an automatic hyperparameter choice :

GMCA$(\hat{\mathbf{A}}^{(0)}, \hat{\mathbf{S}}^{(0)})$
Requires *only* $\mathbf{X}$
While not stabilized or maximum number of iterations not reached, iterate over (l) :

1 - Automatic parameter choice :

$$\left(\mu_1^{(l)}, \mu_2^{(l)}, ..., \mu_n^{(l)}\right)^T = \kappa \times \text{MAD}(\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{X}) \tag{III.36}$$

$$\mathbf{M_S}^{(l)} = \text{Diag}(\mu_1^{(l)}, \mu_2^{(l)}, ..., \mu_n^{(l)})\mathbf{1}_{n \times t} \tag{III.37}$$

2 - $\mathbf{S}$ is updated using the current $\hat{\mathbf{A}}^{(l-1)}$.

$$\hat{\mathbf{S}}^{(l)} = \mathcal{S}_{\mathbf{M_S}^{(l)}}\left(\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{X}\right) \tag{III.38}$$

3 - $\mathbf{A}$ is updated using the current $\hat{\mathbf{S}}^{(l)}$:

$$\hat{\mathbf{A}}^{(l)} = \Pi_{\|.\|_2=1}\left(\mathbf{X}\hat{\mathbf{S}}^{(l)\dagger}\right) \tag{III.39}$$

Lastly, we would like to highlight that other, more advanced scheme for setting the hyperparameters $\mathbf{M_S}$ of GMCA have been proposed (see Appendix D for a presentation of some of these extensions). We do not however focus on this aspect, since the common rationale is the same : use a decreasing hyperparameter choice and set the final values according to the fixed point argument.

C.5 AMCA, an extension of GMCA

As evoked in the previous section and already relatively visible in Figure III.2, the morphological diversity hypothesis might be a too strong assumption in practical cases, for which some samples can have several high amplitude coefficients. In this situation GMCA, that grounds its hyperparameter estimation on such high amplitude samples, can have lowered performances.
The Adaptive Morphological Component Analysis (AMCA – [Bobin *et al.* 2015]) has been introduced to tackle such difficult cases. The main idea is the following : AMCA assigns a weight to each sample $\mathbf{X}^j, j \in [1, t]$ which will enable to discard in the estimation process samples suffering from partial correlations : that is, the ones that do not respect the morphological diversity. AMCA weights are grouped as the diagonal elements of a diagonal matrix $\mathbf{M} \in \mathbb{R}^{t \times t}$.
Let us write Ω^C the source samples for which the morphological diversity does not hold, and Ω^* the samples for which it does holds. If such sets are known beforehand, creating the $\mathbf{M}$ matrix is easy : for samples $\mathbf{X}^j \in \Omega^C, \mathbf{M}_{jj} = 0$ (thus discarding them) and for samples $\mathbf{X}^j \in \Omega^*, \mathbf{M}_{jj} = 1$ (thus keeping them unchanged). Instead of minimizing Eq. II.8, the new cost function [11] is then given by [12] :

$$\underset{\mathbf{A}\in\mathbb{R}^{m\times n},\mathbf{S}\in\mathbb{R}^{n\times t}}{\operatorname{argmin}} \frac{1}{2} \operatorname{Tr}\left[(\mathbf{X}-\mathbf{AS})\mathbf{\Phi}_\mathbf{S}^\mathbf{T}\mathbf{M}\mathbf{\Phi}_\mathbf{S}(\mathbf{X}-\mathbf{AS})\right] + \left\|\mathbf{R}_\mathbf{S} \odot (\mathbf{S}\mathbf{\Phi}_\mathbf{S}^\mathbf{T})\right\|_1 + \iota_{\{\forall i\in[1,n];\|\mathbf{A}^i\|_2^2=1\}}(\mathbf{A}) \tag{III.40}$$

Where, compared to Eq. II.8, the data fidelity term has been changed to take into account the weights $\mathbf{M}$. Since Ω^C and Ω^* are however unknown in practice, the authors proposed to relax this method using an adaptive reweighting procedure. They thus use a $\mathbf{M}$ matrix chosen based on the sparsity level of $\mathbf{S}\mathbf{\Phi}_\mathbf{S}^\mathbf{T}$ columns :

$$\mathbf{M} = \operatorname{Diag}_{j\in[1,t]}\left(\frac{1}{\frac{\left\|(\mathbf{S}\mathbf{\Phi}_\mathbf{S}^\mathbf{T})^j\right\|_{\ell_{p_{(l)}}}}{\left\|(\mathbf{S}\mathbf{\Phi}_\mathbf{S}^\mathbf{T})^j\right\|_{\ell_2}} + \varepsilon}\right) \tag{III.41}$$

where ε is a small constant that may be required when some columns have vanishing ℓ_p norms, and $p \in [0, 1]$ is a parameter determining the influence in the reweighting of the the source samples sparsity level. In brief, p changes during the iterations l : it is close to 1 at the beginning, to avoid to penalize too strongly imperfectly unmixed sources, and it decreases during the following iterations when the unmixing becomes better.

11. Note that only an *approximation* of such a cost function is minimized because similarly to GMCA the AMCA algorithm is based on pALS.
12. Assuming, without loss of generality, that $\mathbf{\Phi}_\mathbf{S}$ is orthogonal.

Chapitre IV

Using PALM in Sparse BSS

A Introduction

As mentioned in the previous chapter, the optimization strategy is of uttermost importance with regards to the performances of sparse BSS. Among the algorithms presented earlier, the GMCA algorithm [Bobin *et al.* 2007] (*cf.* Sec. III-C.4.2) has well established over the last decade its ability to handle various problems. This in particular due to its heuristic parameter choice – described in Sec. III-C.4.2 – making them decrease during the iterative process and benefitting from the morphological diversity assumption.
The goal of this chapter is however to depart from this approach and give a closer look to algorithms that have been introduced more recently and that are usually used with *fixed* parameters. More specifically, we will focus on PALM ([Bolte *et al.* 2014] – *cf.* Sec III-C.3.3), both exploring empirically its performances in the context of sparse BSS, and trying to draw the quintessence from it by re-using heuristics in the same spirit as in GMCA.

A.1 What is a good sparse BSS algorithm ?

But first, a questions is : what is a "good" sparse BSS algorithm ? Generally speaking, any sparse BSS algorithm yields an estimate $\hat{\boldsymbol{\Theta}}$ of the true mixture parameters $\boldsymbol{\Theta}^* = \{\mathbf{A}^*, \mathbf{S}^*\}$. This estimate depends on the data $\mathbf{X}$: of course, through $\boldsymbol{\Theta}^*$ but also through the noise $\mathbf{N}$. To find a good estimate $\hat{\boldsymbol{\Theta}}$, a practitioner has access to a given data set $\mathbf{X}$ but can choose a solver $\mathcal{A}$, an initial point $\hat{\boldsymbol{\Theta}}^{(0)}$ and a set of regularization parameters $\mathbf{R_S}$, which can be formulated as follows :

$$\hat{\boldsymbol{\Theta}} = \mathcal{A}\left(\hat{\boldsymbol{\Theta}}^{(0)}, \mathbf{R_S}; \mathbf{X}\right)$$

The goal of this chapter is to introduce a reliable, effective and versatile algorithmic framework to tackle sparse BSS problems, which is needed for real-world large-scale applications :

— **Reliability** means that the solution has a low dependence on the initial point, which is important when little is known about the solution. More specifically with given $\mathbf{R_S}$ and $\mathbf{X}$, the variance over $\hat{\boldsymbol{\Theta}}^{(0)}$ of the separation quality (measured by an estimator C_A – see Appendix B for the one used here) is as small as possible.

— **Efficiency** has to do with the algorithmic framework and the regularization parameter tuning strategy : for a given dataset $\mathbf{X}$, it should be possible to

obtain a solution close to the true $\mathbf{\Theta}^\star$ without having to perform a cumbersome regularization parameter choosing. In particular, this is achieved if the algorithmic framework is not too sensitive to the regularizing parameter choice (the variance of C_A over $\mathbf{R_S}$ is small for a given dataset $\mathbf{X}$ and initialization $\hat{\mathbf{\Theta}}^{(0)}$) or if some good automatic thresholding strategy is available.

— **Versatibility** in sparse BSS deals with the fact that the efficiency must hold for various data $\mathbf{X}$ settings : the thresholding strategy must generalize well to different data. In the case of a threshold choice performed through a mere grid search, versatility can be assessed by looking at the variance of C_A over different datasets $\mathbf{X}$ for given $\hat{\mathbf{\Theta}}^{(0)}$ and $\mathbf{R_S}$. If such a variance is low, it means that a threshold that has been determined for a specific dataset can be re-used easily for another one.

A.2 Questions arising from PALM use in sparse BSS

While the PALM algorithm has become one of the most attractive optimization frameworks for tackling generic matrix factorization [Repetti *et al.* 2015, Lanaras *et al.* 2015, Thouvenin *et al.* 2015, Bao *et al.* 2016, Pierre *et al.* 2015], its application to sparse BSS raises open questions :

— **What are the major limitations of minimizing Eq. (II.8) using the PALM algorithm to perform sparse BSS ?** The investigations presented in the following highlight the low *reliability* of PALM, while performing sparse BSS by minimizing Eq. (II.8) using it exhibits a dramatic lack of *efficiency* and *versatility*. The outputs of Eq. (II.8) are indeed highly sensitive to the regularization parameters in some scenarios. In these, PALM – and potentially any other algorithm – is therefore impractical if not associated with an automatic regularization parameter strategy. This is all the more problematic as the regularizing parameters choice is not always well discussed in the literature (although it is a difficult problem [Mensch *et al.* 2018], in many works it is either not discussed, or sums up to a grid search, or uses the true factorization [Repetti *et al.* 2015, Lanaras *et al.* 2015, Thouvenin *et al.* 2015, Bao *et al.* 2016]).
It is however also empirically shown here that potentially accurate results could be found if optimal regularization parameters for a given (good) initialization were known beforehand.

— **Is it possible to improve the estimation yielded by the minimization of Eq. (1) using PALM with heuristic techniques ?** The hope is then twofold : i) reach PALM potential accurate results when adapted regularizing parameters are used ; ii) benefit from PALM mathematical guarantees. It has to be emphasized that using heuristics in other matrix factorization algorithms has already known a wide success, both for sparse BSS [Bobin *et al.* 2007] and for NMF [Vandaele *et al.* 2016]. We show how combining PALM with

heuristic techniques can enhance the separation quality by providing adapted initializations and regularization parameters.

B Chapter outline

In section C, we investigate the practical limitations of performing sparse BSS by minimizing Eq. (II.8) using PALM. In section D, we show and explain the difficulty of straightforwardly extending existing heuristic techniques to PALM to make it easier to use. In section E, we re-use the results of section C and D to handle each element hindering PALM applicability in sparse BSS and, building on them, to rationalize a 2-step approach (with an initialization stage followed by a refinement procedure). In section F, the quality of the 2-step approach is demonstrated on realistic astrophysical data, while in section G, the limitations of the approach are studied. Several extensions of this work are furthermore shortly discussed in Appendix E (in particular, a quick study of BCD is performed, and other 2-step approaches are proposed).

C Limitations of minimizing Eq. (II.8) with PALM to perform sparse BSS, an empirical study

The objective of this section is to empirically shed light on the limitations of using the PALM algorithm along with Eq. (II.8) to handle the sparse BSS problem. For that purpose, we will evaluate the accuracy of the final point estimate $\hat{\boldsymbol{\Theta}} = \{\hat{\mathbf{A}}, \hat{\mathbf{S}}\}$ yielded by PALM in terms of an estimator (which can be computed here only due to the fact that the true matrices are simulated and therefore known). More specifically, we will highlight the limitations of using PALM to minimize problem (II.8) to perform sparse BSS in terms of efficiency, reliability and versatility.

At this point, it might be important to highlight two elements :

— We study the results in terms of *separation quality* measured by an estimator C_A, and not only in terms of the cost function (II.8). That is, we try to find a good critical point corresponding to a *true* physical factorization.

— The phenomena we study here could be inherent to the use of Eq. (II.8) for performing BSS, since the role of PALM is only to perform its minimization. However, in the light of the previous remark and since (II.8) is non-convex, the use of another minimization scheme could lead to a different (local) minimum with a different quality in terms of the estimator. For instance, another minimization algorithm could be more or less sensitive to the regularization parameter values [1]. In this work, we mainly study the *separation quality* when minimizing Eq. (II.8) *with PALM* (and therefore not problem (II.8) in general – however, preliminary results with BCD are also presented in Appendix E,

1. Due for example to an implicit regularization introduced by the minimization scheme.

tending to suggest that several conclusions might be quite inherent to the cost function).

C.1 Gist of the experiments : what is to be studied ?

The goal of this subsection is to determine which factors have a potential influence on the final separation quality and must therefore be taken into account in the study of PALM for sparse BSS. This can be inferred by detailing the first update step of the sources in the PALM algorithm :

$$\begin{aligned}\tilde{\mathbf{S}} &= \hat{\mathbf{S}}^{(0)} - \frac{\gamma}{\left\|\hat{\mathbf{A}}^{(0)T}\hat{\mathbf{A}}^{(0)}\right\|_2}\hat{\mathbf{A}}^{(0)T}(\hat{\mathbf{A}}^{(0)}\hat{\mathbf{S}}^{(0)} - \mathbf{X}) \\ &= \hat{\mathbf{S}}^{(0)} + \frac{\gamma}{\left\|\hat{\mathbf{A}}^{(0)T}\hat{\mathbf{A}}^{(0)}\right\|_2}(\hat{\mathbf{A}}^{(0)T}\mathbf{A}^*\mathbf{S}^* - \hat{\mathbf{A}}^{(0)T}\hat{\mathbf{A}}^{(0)}\hat{\mathbf{S}}^{(0)} + \hat{\mathbf{A}}^{(0)T}\mathbf{N})\end{aligned} \tag{IV.1}$$

The gradient descent step is followed by the application of the proximal operator of the penalization term :

$$\mathbf{S}^{(1)} = \mathcal{S}_{\frac{\gamma\mathbf{R_S}}{\left\|\hat{\mathbf{A}}^{(0)T}\hat{\mathbf{A}}^{(0)}\right\|_2}}\left(\tilde{\mathbf{S}}\right) \tag{IV.2}$$

Therefore, the best estimation of $\mathbf{S}^*$ will potentially depend on the starting point $\hat{\mathbf{\Theta}}^{(0)} = \{\hat{\mathbf{A}}^{(0)}, \hat{\mathbf{S}}^{(0)}\}$, the true mixture parameters $\mathbf{\Theta}^\star = \{\mathbf{A}^*, \mathbf{S}^*\}$, the noise $\mathbf{N}$ and the regularizing parameters $\mathbf{R_S}$.

More specifically, to relate this discussion to the desired properties of a sparse BSS algorithm and outline the experimental protocol :

- The *efficiency* will be studied by trying several values of $\mathbf{R_S}$ for a given experimental setting. This will give an insight of the estimate quality sensibility (in terms of C_A) yielded by the minimization of Eq. (II.8) using PALM.
- The *reliability* will be studied varying the initial points $\hat{\mathbf{\Theta}}^{(0)}$ and more specifically $\hat{\mathbf{A}}^{(0)}$.
- The *versatility* will be studied by changing the experimental setting : $\mathbf{\Theta}^\star = \{\mathbf{A}^*, \mathbf{S}^*\}$, and the noise $\mathbf{N}$. More specifically, 4 types/cases of experiments will be described in Sec. C.2 and we will analyse :
  - The inter-case versatility : for instance, how a change in the source distribution affects the estimate quality ?
  - The intra-case versatility : even focusing on a specific *case*, does using another random realization of $\mathbf{\Theta}^\star = \{\mathbf{A}^*, \mathbf{S}^*\}$, and $\mathbf{N}$ can affect the quality ?

C.2 Description of the data

To bring out the mechanisms at stake, we will perform experiments using 4 different datasets. In each case, there is $n = 2$ sources [2].

— *Case 1 :* The sources are assumed to be exactly sparse in the direct domain and to follow a Bernoulli-Gaussian distribution : a proportion $p = 0.1$ of the $t = 500$ samples is non-zero and drawn according to a standard normal distribution. The sources are equilibrated, *i.e.* they have equal variances. Their dynamic (maximum minus minimum value) is circa 0.6. This example therefore corresponds to very simple synthetic data. The mixing is performed through a matrix $\mathbf{A}^*$ drawn randomly following a standard normal distribution and modified to have unit columns. Its condition number is $C_d = 10$ and we focus on the exactly determined case : there is an equal number of observations and sources $m = n$. To complete the creation of the $\mathbf{X}$ data, a Gaussian noise is added to the mixing, such that the Signal to Noise Ratio is SNR = 60 dB.

— *Case 2 :* The sources are assumed to be approximately sparse in the direct domain and to follow a generalized Gaussian distribution of parameter $\alpha = 0.25$. There is again $t = 500$ samples and the sources are still equilibrated with a dynamic of circa 0.6. The mixing is performed with the same $\mathbf{A}^*$ matrix as in *case 1* and an additive noise $\mathbf{N}$ is added in a similar way. Compared to *case 1*, *case 2* corresponds to a more realistic setting, since the wavelet coefficients of natural images would have a similar distribution.

— *Case 3 :* The sources are constructed in a similar way as in *case 2*. However, the noise energy of the first observation is twice the one of the second observation. Furthermore, $\mathbf{A}^*$ is taken orthogonal (that is, $\mathbf{A}^{*T}\mathbf{A}^* = \mathbf{A}^*\mathbf{A}^{*T} = \mathbf{Id}$) to ensure that the noise projection on the source space has different amplitudes. While this setting might be simpler than the one of *cases 2*, it allows to study the impact of the noise on the parameter choice.

— *Case 4 :* The sources come from simulations obtained from real data of Cassiopeia A supernova remnant. These data originate from the Chandra [3] X-ray observatory. The sources in these wavelength values correspond to the thermal emission and the iron 2. As displayed in Fig. II.3, each of them consists in a 2D image of resolution $t = 128 \times 128$ pixels, supposed to be approximately sparse in the starlet domain (in which their dynamic range is circa 4×10^{-3} and 6×10^{-4} respectively). The mixing is performed with the same $\mathbf{A}^*$ matrix as in *case 1*. A Gaussian additive noise is added to the mixing, such that the SNR = 30 dB. Beyond being still more realistic than the other cases, *case 4* involves non-equilibrated sources.

2. While the case of $n = 2$ sources might seem too simplistic, the following experiments will already highlight the difficulty of performing sparse BSS using PALM in this setting. As such, it is not expected that the use of PALM would be made easier with more sources.

3. *http ://chandra.harvard.edu/*

C.3 Evaluation protocol

For each experimental scenario, a large number of values for the regularization parameters $\mathbf{R_S}$ are tested. More specifically, since no reweighting ℓ_1 in problem (II.8) is used in this part, the work will be done on $\mathbf{\Lambda_S} = \mathbf{R_S}$. As we will focus on the case of $n = 2$ sources, there will be 2 parameters to test, λ_1 and λ_2.
For each (λ_1, λ_2) value, a criterion measuring the separation quality is evaluated. Since a change in the regularization parameters directly impacts the source estimation, we will use the mixing matrix criterion described in [Bobin *et al.* 2015] (*cf.* Appendix B) :

$$C_A = \text{mean}(|\mathbf{P}\hat{\mathbf{A}}^{\dagger}\mathbf{A}^* - \mathbf{I_d}|) \tag{IV.3}$$

With $\mathbf{A}^*$ the true mixing matrix and $\mathbf{P}\hat{\mathbf{A}}^{\dagger}$ the pseudo-inverse of the estimate corrected through $\mathbf{P}$ for the scale and permutation indeterminacies. The mean is the average of all the elements inside the matrix. For the sake of clarity, the plots will display $-10\log_{10}(C_A)$. As such, the higher the values, the better the separation.

C.4 Results and interpretation

C.4.1 *Efficiency* : sensitivity to the regularization parameters

In this subsection, we study the *efficiency*, that is we want to get an idea of the sensitivity for a given $\mathbf{X}$ of PALM estimate to the regularizing parameters $\mathbf{\Lambda_S} = \mathbf{R_S}$. Therefore, we take one fixed experiment of each *case* and we try several (λ_1, λ_2) values. For each of them, the mixing matrix criterion C_A is computed from the estimate and reported in a 2D plot, displayed in Figure IV.1. To try to separate as much as possible the *efficiency* issue to the *reliability* issue, we launched the algorithm with 5 different random initializations $\hat{\mathbf{A}}^{(0)}$ and the median is displayed.
The first observation corresponds to the high variations induced by changes of regularizing parameters, with a dynamic range of more than 40 dB for *case 1*, 30 dB for *case 2* and 20 dB for *case 4*. The results change from good to catastrophic, making the choice of the regularizing parameters a crucial point, especially as these high variations are very fast on the plot.
In *case 1*, since the sources have identical distributions and the noise is supposed to be white, it would be expected from the best regularization parameters λ_1 and λ_2 to have similar values. However, while staying close to the diagonal could restrict the space of possible good parameters, it is unfortunately dangerous to fully restrict the search to the diagonal, as demonstrated in Fig. IV.2. In *case 1*, just using the parameters on the diagonal closest to the best ones of Fig. IV.2a yields a drop of more than 10 dB (which might be linked to a too small number of samples for the noise to be perfectly white).
In the non-equilibrated *case 4*, while not very visible in Fig. IV.1 due to the median over different initializations, very good parameter settings appear far from the diagonal for specific initializations as testified in Fig. IV.2b. Restricting the parameter search to the diagonal would hide these high quality results.

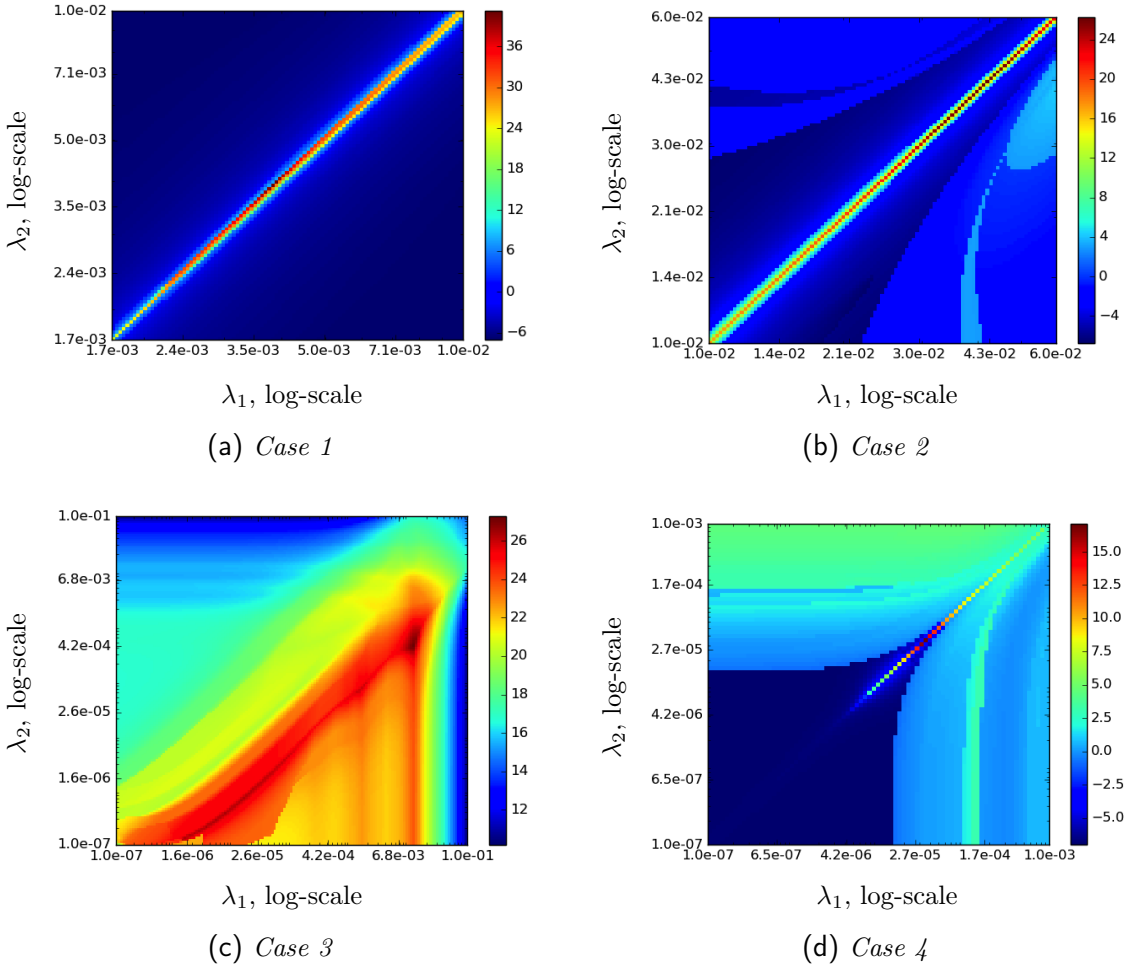
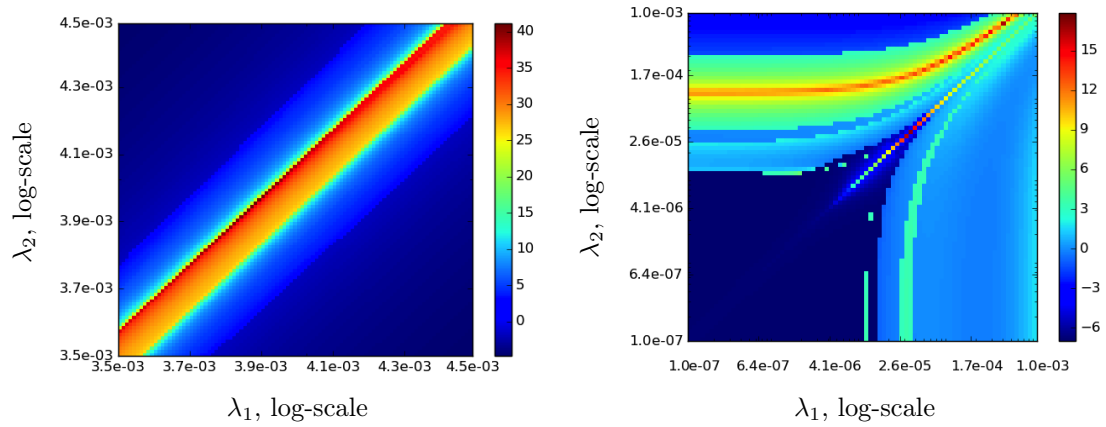


Figure IV.1 – Median of C_A (dB) for 5 initializations of PALM algorithm as a 2D function of the 2 thresholds corresponding to the $n = 2$ sources.



(a) *Case 1*, zoom around the maximum (median over 5 different initializations)

(b) *Case 4* for one specific experiment

Figure IV.2 – C_A (dB) of the output of PALM algorithm as a 2D function of the thresholds.

Even if we assume that it is possible to restrict the regularizing parameter search to the diagonal, the variations are still quick in comparison to the dynamic of the sources. In the simple *case 1*, a shift of 0.1% of the dynamic of the sources in the regularizing parameters in a parallel direction to the diagonal yields a 7 dB loss. In *case 2*, a shift of 5% yields a 8 dB drop and for *case 3*, a 1% change implies a 8 dB drop.
This highlights that, for a *single experiment* (*i.e.* for a given $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$), **the separation performances are highly sensitive to the choice of the regularization parameters**.

C.4.2 *Versatility* : impact on the regularization parameters of $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$

Inter-case *versatility*

We first look at the main potential source of lack of *versatility*, that is when the distributions of the matrices $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$ change. To do that and for computational reasons, we will merely compare the results displayed in Fig. IV.1 to highlight the lack of *versatility* implied by a change of the distribution of $\mathbf{S}^*$ or of the back-projected noise $\mathbf{A}^{*\mathbf{T}}\mathbf{N}$.

— *Impact of* $\mathbf{S}^*$
The shapes of the plots of the 4 different *cases* strongly differ. In particular, the observations we made in the previous subsection show that the best regularizing parameter values are highly dependent on the true source $\mathbf{S}^*$ distributions :

  - In addition to remove the noise, the regularizing parameters should limit the presence of remixing or interferences between the sources, which depend on the distribution of $\mathbf{S}^*$, the mixing matrix $\mathbf{A}^*$ as well as the initialization.
  - The resulting thresholding induces a bias, called artifacts, which also depend on the distribution of $\mathbf{S}^*$. These are also problematic as they can eventually convert into interferences between the sources in the iterative optimization process.

— *Impact of the back-projection of* $\mathbf{N}$

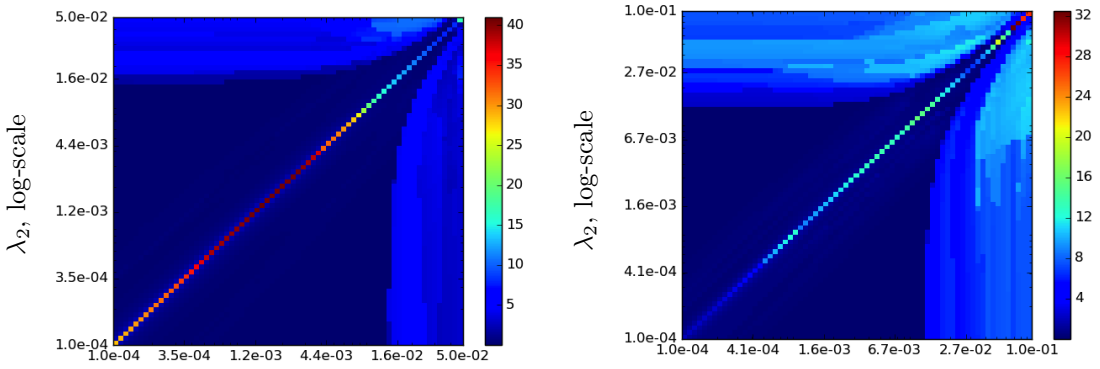
At the vicinity of the true mixture parameters $\mathbf{A}^*$ and $\mathbf{S}^*$, the only source of error that contributes to the estimate of $\mathbf{S}^*$ originates from propagated noise. The role of the regularization parameters is then merely to avoid a deterioration of the estimate by the noise, that is to threshold the update $\mathbf{A}^{*T}(\mathbf{X} - \mathbf{A}^*\mathbf{S}^*) = \mathbf{A}^{*T}\mathbf{N}$. Consequently, an intuitive choice for the regularization parameters for each sources i would be $\lambda_i = \left\|(\mathbf{A}^{*T}\mathbf{N})_i\right\|_\infty$, which implies a clear dependency between the thresholds and the back-projection of $\mathbf{N}$ through $\mathbf{A}^{*T}$. This can be studied through *case 3* : in this setting, $\mathbf{A}^*$ is orthogonal and the noise $\mathbf{N}$ energy is different over the observations. The thresholds enabling the convergence should therefore be different for the two sources : $\lambda_1 \neq \lambda_2$. This is confirmed by Figure IV.1c, which shows that contrary to the other *cases*, the highest separation quality lies far from the diagonal. Thus, unbalanced backprojected observation noise (that is, with different variances on each source) can make the threshold choice much more difficult [4] than in settings where the noise is equilibrated due to optimal parameters further from the diagonal.

Intra-case *versatility*

We now look at the intra-case *versatility*, that is we study the impact of changing the realization of the data for a fixed case – more specifically, we will focus on *case 1* and *2*. To do that, we draw for both cases 10 new random $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$ matrices and create new data. For each of these data and to try to separate the *versatility* issue from the *reliability* one, we try 10 initializations and take the median over them. Therefore, we get 10 plots similar to the ones of Fig. IV.1, each of them corresponding to a new random $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$. To study the *versatility*, we look at the diversity among this plots by looking at their dynamic : for each (λ_1, λ_2) value, we plot the mixing criterion of the best estimation minus the mixing criterion of the worst estimation. It gives us an idea of the intra-case variability for given parameters. The results are plotted in Fig. IV.3.

In the exactly sparse *case 1*, the dynamic of circa 40 dB is huge compared to the results yielded by the best value of some settings (which are in some realizations of

4. Introducing a weighted Frobenius norm - squared Mahalanobis distance - would avoid this situation but it requires a good estimate of the covariance matrix of the noise N.



(a) *Case 1* (b) *Case 2*

Figure IV.3 – Dynamic of C_A (dB) over different random realizations of $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$.

24 dB). For the approximately sparse *case 2*, the dynamic is relatively similar. It makes that even without changing the distribution of the random matrices (said differently, even if we know their distributions), choosing good (λ_1, λ_2) values is very difficult because this choice is extremely dependent on the specific data realization that must be handled.

Consequently, the best regularizing parameters are dramatically dependent on the data $\mathbf{X}$, which further highlights **the low versatility of performing sparse BSS by minimizing Eq. (II.8) using PALM without any automatic regularization parameter choice**. In particular, the extra-case *versatility* has pointed out that the regularization parameters should be chosen based on $\mathbf{S}^*$ as well as the backprojected noise.

C.4.3 *Reliability* : impact of the initialization $\hat{\mathbf{A}}^{(0)}$, $\hat{\mathbf{S}}^{(0)}$

The impact of the initialization has two theoretical groundings :

- As problem (II.8) is not convex but multi-convex, the algorithm performing its minimization can be trapped in spurious critical points depending on the initial matrices $\hat{\mathbf{A}}^{(0)}$ and $\hat{\mathbf{S}}^{(0)}$.
- The initialization directly impacts the quality of the estimate yielded by specific thresholds through Eq. (IV.2) and the threshold choice. Said differently, for a given initialization, it might be desirable to change the cost function via the choice of thresholds to avoid a spurious critical point that would not be an issue for another initialization.

To quantify this impact, the dynamic of C_A over 5 different initializations $\hat{\mathbf{A}}^{(0)}$ is plotted for *case 4* in Fig. IV.4. The high values, up to more than 13 dB, are to be compared with the best results of Fig. IV.2b, that is slightly less than 18 dB.

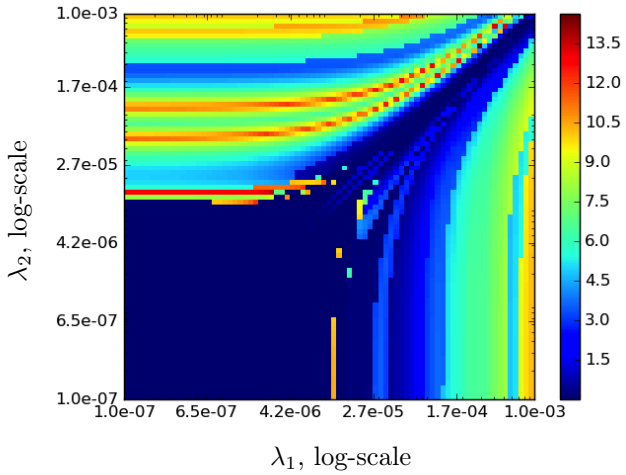


Figure IV.4 – Dynamic (maximum value minus minimum value) of C_A (dB) as a 2D function of the thresholds for 5 different initializations of PALM algorithm.

This is mainly due to regularizing parameters outside the diagonal, which unfortunately correspond to a high separation quality for some experiments. This means than choosing such parameters in these experiments can lead to almost the best performances, as well as very bad ones (3 dB), making the regularization parameter choice still more difficult.
These experiments emphasize that the quality of PALM results is highly connected to the initialization. More strikingly, the choice of good regularization parameters in terms of C_A is also connected to the initial point. In practice, to avoid relaunching the algorithm with different initializations for given regularization parameter values (which is especially important with *large-scale* data), **it is important to make PALM estimate, as well as the regularization parameter choice, more robust to the initial conditions**.

D Enhancing PALM with heuristic techniques

To bypass the cumbersome parameter choice described in the previous section, the goal of this section is to try and show the issues of a straight adaptation within PALM of the heuristic parameter choice used in GMCA. In the first subsection, GMCA heuristic is reminded, and some general remarks about the performances of GMCA made. In section D.2, the heuristic parameter choice is adapted to the PALM framework. In section D.3, the heuristic is tested and shown to work badly within PALM, while in D.4 these empirical results are explained.

D.1 GMCA heuristic and performances

Value displayed	*Case 1*	*Case 2*
Average	36.3	20.0
Dynamic over $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$	8.6	14.2
Minimum value over $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$	29.7	15.8
Variance over initializations	3.2×10^{-12}	5.6×10^{-13}

Tableau IV.1 – GMCA results in terms of C_A.

D.1.1 Reminder concerning GMCA heuristic

From Chapter III, we would like to recall that GMCA proposes an automatic threshold choice by using the following κ-MAD rule :

$$\mathbf{M_S} = \text{Diag}\left(\mu_1^{(l)}, \mu_2^{(l)}, ..., \mu_n^{(l)}\right) = \text{Diag}(\kappa \times \text{MAD}(\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{X})) \tag{IV.4}$$

D.1.2 Remark about the performances of GMCA

The practical success of the GMCA algorithm relies on its good *reliability* as well as the heuristic to automatically tune $\mathbf{M_S}$, enabling both *efficiency* and *versatility*. To give an idea of the performances of GMCA in the experimental setting of Sec. C, we launched it on *cases 1* and *2*, for 10 different initializations and 10 $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$ settings. The results are summarized in Table IV.1.

Compared to PALM, the dynamic over different realizations of $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$ is much lower than the one of PALM, and the worst values are still good, meaning a much higher *versatility*. The automatic parameter choice yields a high mean of C_A, showing GMCA *efficiency*. The *reliability* seems to be extremely high as shown by the very low standard deviation over the initializations. However, the results of GMCA are not always as good as the ones yielded by PALM *with the best parameters*. This might be due that the use of the pseudo inverse $\mathbf{A}^{(k)\dagger}$ in GMCA can strongly increase the noise back-projection when $\hat{\mathbf{A}}^{(l)}$ is badly conditioned. As an example of such a difference between GMCA and PALM, the results of GMCA are 36.3 dB in the experience of *case 1* displayed in Fig. IV.2, to be compared with a little more than 40 dB for PALM. This partially justifies the will to make PALM easy to use : to benefit from its potential high accuracy when it is well tuned.

D.2 Adaptation of GMCA heuristic parameter choice to PALM : a first idea

The goal is here to explain a straightforward adaptation to PALM of GMCA automatic parameter choice.
For the sake of simplicity, let us assume that there is no reweighting : $\mathbf{R_S} = \mathbf{\Lambda_S}$. The update of $\mathbf{S}$ by PALM in Eq. (III.25) is then :

$$\hat{\mathbf{S}}^{(l)} = \mathcal{S}_{\frac{\gamma\mathbf{\Lambda_S}}{\left\|\hat{\mathbf{A}}^{(l-1)T}\hat{\mathbf{A}}^{(l-1)}\right\|_2}}\left(\tilde{\mathbf{S}}_{\mathbf{PALM}}\right) \tag{IV.5}$$

Under the same assumptions, the update of GMCA is the following (with $\tilde{\mathbf{S}}_{\mathbf{GMCA}} = \hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{X}$) :

$$\hat{\mathbf{S}}^{(l)} = \mathcal{S}_{\mathbf{M_S}}\left(\tilde{\mathbf{S}}_{\mathbf{GMCA}}\right) \tag{IV.6}$$

Taking into account the differences between $\tilde{\mathbf{S}}_{\mathbf{PALM}}$ and $\tilde{\mathbf{S}}_{\mathbf{GMCA}}$, the parallel between Eq. (IV.5) and (IV.6) suggests that $\mathbf{\Lambda_S}$ could be choosen similarly as the parameters in GMCA (in which case the parameters change over the iterations, but are fixed at the end to ensure the convergence of the algorithm) :

$$\frac{\gamma}{\|\mathbf{A}^{*T}\mathbf{A}^{*}\|_2}\,(\lambda_1, \lambda_2, ..., \lambda_n)^T = \kappa \times \mathrm{MAD}\left(\tilde{\mathbf{S}}_{\mathbf{PALM}}\right) \tag{IV.7}$$

If so, and if PALM has converged to *both* the true matrices $\mathbf{A}^*$ and $\mathbf{S}^*$, then :

$$\begin{aligned}\frac{\gamma}{\|\mathbf{A}^{*T}\mathbf{A}^{*}\|_2}\,(\lambda_1, \lambda_2, ..., \lambda_n)^T &= \kappa \times \mathrm{MAD}\left(\mathbf{S}^* - \frac{\gamma}{\|\mathbf{A}^{*T}\mathbf{A}^{*}\|_2}\mathbf{A}^{*T}(\mathbf{A}^*\mathbf{S}^* - \mathbf{X})\right) \\ &\simeq \kappa \times \frac{\gamma}{\|\mathbf{A}^{*T}\mathbf{A}^{*}\|_2}\,\mathrm{MAD}(\mathbf{A}^{*T}\mathbf{N})\end{aligned} \tag{IV.8}$$

Where the last line is obtained because $\mathbf{S}^*$ is assumed sparse and the MAD is robust to outliers. Therefore, using the MAD enables a thresholding of a projection of the noise $\mathbf{N}$, which yields a similar interpretation as in GMCA (*cf.* Sec. III-C.4.2). This parallel must however be tempered :

— It only holds when and if PALM has converged towards good $\mathbf{A}$ *and* $\mathbf{S}$;

— The projection is not performed through the pseudo-inverse as in GMCA, but rather through $\mathbf{A}^{*T}$. Both projections however merge when $\mathbf{A}$ is orthogonal.

Remark : Choosing the regularization parameters in general inverse problems has been the subject of several studies, which propose various ways for setting them. Among them, one can highlight the Stein's Unbiased Risk Estimator – SURE – method (and its extended versions [Stein 1981, Eldar 2009, Giryes *et al.* 2011, Deledalle *et al.* 2014]), the generalized cross-validation [Golub *et al.* 1979, Lukas 2006], the L-curve [Hansen & O'Leary 1993], the discrepancy principle (or some variants [Almeida & Figueiredo 2013]) or some Bayesian methods [Pereyra *et al.* 2015, Vidal & Pereyra 2018]. However, most of these are not directly tractable in the case of sparse BSS for several reasons : i) we are dealing with the *blind* setting, in which the linear operator $\mathbf{A}^*$ must also be evaluated, making the use of these methods more complicated ; ii) some of these methods can be computationally expensive in the large-scale setting, since they require to try several regularization parameters and compute a criterion to decide which one to choose ; iii) some of them, such as SURE, use as a criterion for the parameter choice the estimated MSE : it is not clear if this criterion is the most relevant one in sparse BSS (*cf.* [Feng & Kowalski 2018]) ; iv) some of them have mostly been applied in the case of linear solutions (*e.g.* by using a ℓ_2 regularization instead of ℓ_1 – this has been emphasized in the context of deconvolution in [Almeida & Figueiredo 2013]).
Therefore, choosing the regularization parameters using the MAD seems one of the most interesting solutions since it has already lead to enhanced separation quality

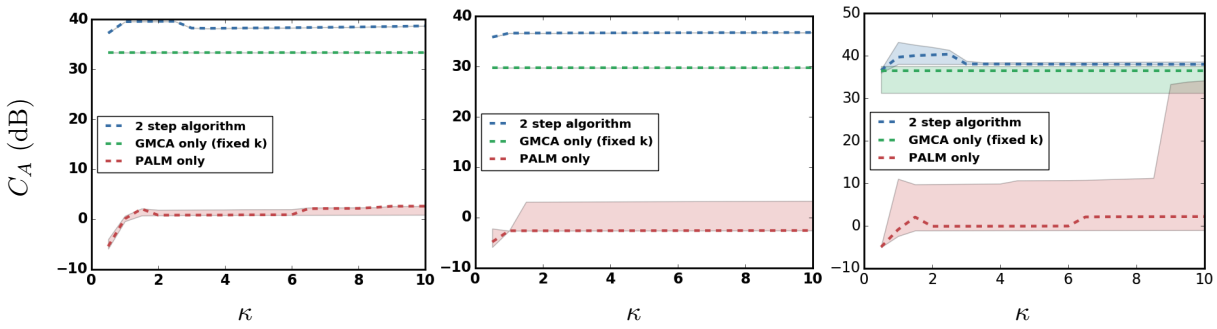


Figure IV.5 – Mixing matrix criterion for PALM and the 2-step strategy (*cf.* Sec. E) on *case 1*. The results of GMCA are also plotted as a baseline, with a fixed $\kappa = 3$, which is a classical value due to the corresponding hypothesis in terms of Gaussian noise removal. *Left* : the dashed line is the median of C_A over the different $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$, and the error bars correspond to the quartiles of the criterion over the initialization ; *center* : the dashed line corresponds to the values of C_A for one specific $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$, and the error bars to the quartiles of the criterion over the initialization ; *right* : the dashed line corresponds to the median of C_A over the initializations, and the error bars to the quartiles of the criterion over the realizations of $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$.

within the context of GMCA. In Appendix E, we however try one of the previous methods in sparse BSS to back this claim.

D.3 Illustration

The automatic parameter setting based on the MAD described in the previous subsection is tried inside of a PALM algorithm for different κ values. The parameter κ is the same for both sources, since the MAD is supposed to be adaptive enough to the noise. The metric on the separation quality C_A is the same as used in Sec. C and the data is the same as in *case 1* : it is obtained from two exactly sparse sources with a square mixing matrix having a condition number of 10. The experiments are conducted with 5 realizations of $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$ and 10 different initializations. The results are displayed in Fig IV.5 (red curve). The low *efficiency* is highlighted by two points : i) the low average values of C_A, reaching at most 1 dB in the leftmost plot of Fig. IV.5 ; ii) even in the experiments for which the strategy works, it is very difficult to choose a good κ value, as seen with the upper quartile of the rightmost plot.
Furthermore, for one specific experiment (see plot in the center of Fig. IV.5) the standard deviation for the different initializations is very high compared to the average value of C_A, which denotes a lack of *reliability*. Finally, the lack of *versatility* is shown by the high error bars in the rightmost plot.
The goal of the following subsection is to explain these empirical results.

D.4 Understanding the limitations of the heuristic in the scope of PALM

We previously pointed out that the thresholds are not identical in GMCA and PALM. A key difference is that the thresholding applies to different quantities :

— In GMCA, it is applied on a least-square estimate of the sources after a direct inversion of the mixing matrix ;

— In PALM, it applies on a single gradient descent step, that is during the inversion.

Therefore, the estimation errors originating from both an imperfect unmixing and the noise are different in the two algorithms. Since it is the role of the thresholds to filter out these estimation errors, they do not have the same impact and the optimal ones may differ from an algorithm to the other.
To better understand this role, let us assume that both GMCA and PALM algorithms are initialized with the true mixing matrix $\mathbf{A}^*$ *only*. Let us further assume on the contrary that the initialization of the sources is not perfect and can be written as $\mathbf{S} = \mathbf{S}^* + \mathbf{s}$ with $\mathbf{s}$ the error made on the sources. For GMCA, the thresholds given by the MAD heuristic then are :

$$(\mu_1, \mu_2, ..., \mu_n)^T = \kappa \, \mathrm{MAD}(\mathbf{A}^{*\dagger}\mathbf{X}) = \kappa \, \mathrm{MAD}(\mathbf{S}^* + \mathbf{A}^{*\dagger}\mathbf{N}) \simeq \kappa \, \mathrm{MAD}(\mathbf{A}^{*\dagger}\mathbf{N}) \tag{IV.9}$$

Where the last equality is obtained because the MAD operator is not sensible to sparse signals. In this case, the thresholds are thus set according to the back-projection of the noise only and an imperfect estimation of $\mathbf{S}^*$ does not affect the thresholding strategy as long as the mixing matrix is well estimated.
In contrast, when using PALM the thresholds are given by :

$$\begin{aligned} \frac{\gamma}{\|\mathbf{A}^{*T}\mathbf{A}^*\|_2} (\lambda_1, \lambda_2, ..., \lambda_n)^T &= \kappa \times \mathtt{MAD}\left(\mathbf{S} - \frac{\gamma}{\|\mathbf{A}^{*T}\mathbf{A}^*\|_2}\mathbf{A}^{*T}(\mathbf{A}^*\mathbf{S} - \mathbf{X})\right) \\ &= \kappa \times \mathtt{MAD}\left(\mathbf{S} - \frac{\gamma}{\|\mathbf{A}^{*T}\mathbf{A}^*\|_2}\mathbf{A}^{*T}(\mathbf{A}^*(\mathbf{S}^* + \mathbf{s}) - \mathbf{X})\right) \\ &\simeq \kappa \frac{\gamma}{\|\mathbf{A}^{*T}\mathbf{A}^*\|_2} \mathtt{MAD}\left(\mathbf{A}^{*T}\mathbf{N} - \mathbf{A}^{*T}\mathbf{A}^*\mathbf{s}\right) \end{aligned} \tag{IV.10}$$

The previous equation highlights the following (and unfortunately complementary) issues of using the MAD inside of PALM :

— *An inappropriate threshold choice :*
Compared to GMCA, the thresholds are calculated with the additional detrimental $\mathbf{A}^{*T}\mathbf{A}^*\mathbf{s}$ interference term. Indeed, they are computed after only one gradient step and therefore when the interferences are still high. Another interpretation is that the MAD is computed on an *approximation* of the minimization of the data fidelity term, namely a gradient step update, and its results are therefore not as accurate as in GMCA.

This hinders the interpretation of the use of the MAD in terms of noise removal, particularly when the remixing $\mathbf{A}^{*T}\mathbf{A}^{*}\mathbf{s}$ has more energy than the noise term $\mathbf{A}^{*T}\mathbf{N}$. More specifically, this is the case when the columns of $\mathbf{A}^{*}$ are strongly correlated, as testified by the product $\mathbf{A}^{*T}\mathbf{A}^{*}$, or when the error on the sources $\mathbf{s}$ are high (which is in particular the case if the initialization is random or if there are many artifacts). If $\mathbf{s}$ is further strongly non-Gaussian, the MAD use is still less relevant (at least, when the number of sources n is low).

— *Higher interferences in the estimation process :*
The gradient update on the sources creates at each iteration a remixing of the error $\mathbf{s}$ due to the $\mathbf{A}^{*T}\mathbf{A}^{*}\mathbf{s}$ term. While this could be a limited issue if $\mathbf{A}$ was fixed, since in PALM only one proximal gradient step is performed on $\mathbf{S}$ before the update of the mixing matrix, $\mathbf{A}$ will be computed from sources with high interferences.
In particular, this issue is very important with high thresholds that induce artefacts and therefore a large $\mathbf{s}$, creating more interferences (which in turn feeds the problem of the previous point, creating more artefacts and so on...). It makes that high thresholds are not that valuable in PALM. Therefore, contrary to GMCA in which the $\mathbf{A}^{*T}\mathbf{A}^{*}\mathbf{s}$ is not present, it is harder to exploit the Morphological Diversity assumption. Indeed, while the most significant coefficients are still assumed to be the most interesting for the separation, selecting only them will create interferences counterbalancing their positive effect.

To conclude, in PALM high thresholds are less interesting than in GMCA and **using the MAD heuristic does not results in good separations in practice if the level of the interferences is not negligible in comparison to the noise contribution**.

E Combining GMCA and PALM : a hybrid strategy

The previous part has demonstrated that the introduction of an heuristic based only on the back-projection of the noise does not yield satisfactory results within a single PALM. We therefore rationalize an approach tackling the various elements (*cf.* Sec. C) hindering the applicability of PALM in sparse BSS by combining the best of GMCA and PALM in a two-step approach. The first warm-up stage consists in a GMCA. It is followed by a refinement stage during which PALM is performed retaining as much information as possible coming from the warm-up stage. Inside the refinement stage, the MAD heuristic is used to choose PALM parameters.
The main idea of the two step approach is to reduce the interferences by a good initialization of the refinement stage and the artifacts using a reweighting information coming from the warm-up stage. It has to be emphasized that this 2-step approach has already been empirically used in two previous works [Chenot & Bobin 2018,Kervazo *et al.* 2018]. Therefore, the novelty of the following is to provide an in-depth justification of this approach while the previous results were only empirical.

More specifically, this justification describes in details all the required elements for this approach to work, which is done both with theoretical arguments along with experiments to confirm these on new datasets.

E.1 Motivation of the two step approach

Our 2-step approach has several motivations :

— Dealing with the factors deteriorating the results of sparse BSS (in terms of *versatility*, *efficiency* and *reliability*) when using PALM for minimizing Eq. (II.8) :
While the MAD heuristic enables the refinement stage to handle the noise back-projection, using some reweighting information from the warm-up stage enables to take into account the distribution of $\mathbf{S}^*$, which was the second factor of the lack of *versatility* of using PALM in the context of sparse BSS (*cf.* Sec. C). The automatic parameter choice then enables to circumvent in the refinement stage the difficult parameter choice and the lack of *efficiency*. Finally, using a *reliable* algorithm such as GMCA as a warm-up stage enables a *reliable* global 2-step algorithm. Furthermore, it enables a good initialization of PALM (which is reached using decreasing regularization parameters and benefiting from the morphological diversity, in contrast to what is usually done when using a single PALM with fixed regularization parameters).

— Enabling to use the MAD heuristic :
One of the main advantages of combining GMCA and PALM is to initialize PALM with the output $\hat{\mathbf{A}}_{\mathbf{GMCA}}$ and $\hat{\mathbf{S}}_{\mathbf{GMCA}}$ of GMCA. Since *both* $\hat{\mathbf{A}}_{\mathbf{GMCA}}$ and $\hat{\mathbf{S}}_{\mathbf{GMCA}}$ are close to $\mathbf{A}^*$ and $\mathbf{S}^*$, the level of the interferences is relatively low in comparison to the noise. Therefore, following the conclusion of Sec. D.4, the MAD is still relatively accurate to derive the thresholds once at the beginning of the refinement step and the interpretation in terms of noise removal is more accurate than with a poor initialization, due to smaller interferences. In addition, the interferences are further indirectly reduced by the use of the reweighting, which reduces the artefacts that would be transformed into interferences by the gradient step.

— Keeping mathematical guarantees and PALM high potential accuracy :
While GMCA is only a proxy, the 2-step algorithm will attempt to solve exactly Eq. (II.8) as PALM does. It will further benefit from PALM potential high accuracy (*cf.* Sec. D.1.2). Moreover, since PALM is proved to converge under mild assumptions [Bolte *et al.* 2014], so is the 2-step algorithm [5].

5. Note that here only a single PALM is used in the refinement step. It could be interesting to study a two-loop algorithm, in which several PALM would be launched until stabilization (if any), with each time new regularization parameters computed on the solution of the previous PALM. It seems however clear that such a process would preclude any convergence guarantee of the whole algorithm, as it is not straightforward that the regularization parameters would converge.

E.2 Use information on $\mathbf{S}^*$ from GMCA : reweighted ℓ_1

As evoked in the previous subsection, it is possible to take into account in the refinement stage the information from $\mathbf{S}^*$ coming from the warm-up stage. To that end, minimizing the artifacts can be carried out by improving the sparse regularizer. This is done by resorting to a reweighted ℓ_1 regularizer [Candes *et al.* 2008]. In this setting, one can benefit from the first guess estimate $\hat{\mathbf{S}}_{\mathbf{GMCA}}$ to compute the reweighting matrix $\mathbf{G}$ in problem (II.8). This generally leads to a significant decrease of the artifacts, which eventually reduces the importance of the interferences introduced by the $\mathbf{A}^{*T}\mathbf{A}^*\mathbf{s}$ term. This will improve the efficiency of the proposed hybrid heuristic, which interpretation is based on noise removal only. In this work, we use the following reweighting scheme :

$$\mathbf{G}_i^j = \frac{\varepsilon}{\varepsilon + \frac{|\hat{\mathbf{S}}^j_{\mathbf{GMCA}_i}|}{\|\hat{\mathbf{S}}_{\mathbf{GMCA}_i}\|_\infty}} \tag{IV.11}$$

where ε is a small constant (here 10^{-3}), $\mathbf{G}_i^j$ the coefficient of $\mathbf{G}$ corresponding to the i^{th} line and j^{th} column and $\hat{\mathbf{S}}_i$ the i^{th} line of $\hat{\mathbf{S}}$. In brief, the thresholds are lowered for the largest samples of the estimated sources, reducing the bias introduced by the soft-thresholding and therefore the error $\mathbf{s}$ and the interferences.

E.3 Final algorithm

The previous remarks lead to the following algorithm using a smart initialization and thresholding strategy :

Input : $\mathbf{X}$ (data matrix)

— *Warm-up stage* :
Random initialization $\hat{\mathbf{A}}^{(0)}$ and $\hat{\mathbf{S}}^{(0)}$
$\hat{\mathbf{A}}_{\mathbf{GMCA}}$, $\hat{\mathbf{S}}_{\mathbf{GMCA}}$ = GMCA($\mathbf{X}$,$\hat{\mathbf{A}}^{(0)}$,$\hat{\mathbf{S}}^{(0)}$)

— *Refinement stage* :
Update of the reweighting information in Eq. (II.8) :

$$\mathbf{G}_i^j = \frac{\varepsilon}{\varepsilon + \frac{|\hat{\mathbf{S}}^j_{\mathbf{GMCA_i}}|}{\|\hat{\mathbf{S}}_{\mathbf{GMCA}_i}\|_\infty}} \tag{IV.12}$$

Update of the parameters $\mathbf{\Lambda_S}$ in Eq. (II.8) :

$$\frac{\gamma}{\left\|\hat{\mathbf{A}}^T\hat{\mathbf{A}}\right\|_2}(\lambda_1, \lambda_2, ..., \lambda_n)^T$$

$$= \kappa \times \texttt{MAD}\left(\hat{\mathbf{S}}_{\mathbf{GMCA}} - \frac{\gamma}{\left\|\hat{\mathbf{A}_{GMCA}}^T\hat{\mathbf{A}}_{\mathbf{GMCA}}\right\|_2}\hat{\mathbf{A}}^T_{\mathbf{GMCA}}(\hat{\mathbf{A}}_{\mathbf{GMCA}}\hat{\mathbf{S}}_{\mathbf{GMCA}} - \mathbf{X})\right) \tag{IV.13}$$

PALM step, initialization coming from GMCA :
$\hat{\mathbf{A}}_{\mathbf{PALM}}$, $\hat{\mathbf{S}}_{\mathbf{PALM}}$ = PALM($\mathbf{X}$,$\hat{\mathbf{A}}_{\mathbf{GMCA}}$,$\hat{\mathbf{S}}_{\mathbf{GMCA}}$)

E.4 Complexity of the algorithm

We here derive the complexity of one iteration of each of the 2 steps of the algorithm.

E.4.1 Initialization stage

Each iteration of the warm-up stage can be decomposed into the following elementary steps : i) the pseudo-inverse is performed using the singular value decomposition of a $n \times n$ matrix, which yield an overall complexity of $\mathcal{O}(n^3 + m^2n + nmt)$; ii) the thresholding-strategy first requires the evaluation of the threshold values, which has a complexity of nt ; iii) the soft-thresholding step itself has a complexity of $\mathcal{O}(nt)$; and iv) updating $\mathbf{A}$ is finally performed using a conjugate gradient algorithm, whose complexity is known to depend on the number of non-zero entries in $\mathbf{S}$ and on the condition of this matrix $C_d(\mathbf{S})$. An upperbound for this complexity is $\mathcal{O}(nt\sqrt{C_d(\mathbf{S})})$. The final estimate of the complexity of a single iteration is thus given by :

$$m^2n + n^3 + nmt + nt\sqrt{C_d(\mathbf{S})} \tag{IV.14}$$

With $C_d(\mathbf{S})$ the condition number of $\mathbf{S}$.

E.4.2 Refinement stage

For the refinement stage, i) the update of $\mathbf{S}$ is dominated by the multiplication required for $\hat{\mathbf{A}}^T(\mathbf{X} - \hat{\mathbf{A}}\hat{\mathbf{S}})$, which has a $\mathcal{O}(nmt)$ complexity, and the computation of $\left\|\hat{\mathbf{A}}^T\hat{\mathbf{A}}\right\|_2$ having an overall complexity of $\mathcal{O}(n^2m)$ (for instance, using the power method). The thresholding and weight computation complexities are negligible ; ii) similarly, for $\mathbf{A}$ update the complexity is $\mathcal{O}(n^2t + nmt)$.
Therefore, the final estimate of the complexity of a single iteration of the refinement stage is given by :

$$nmt + n^2(m + t) \tag{IV.15}$$

Beyond this complexity, the overall number of required iterations needed to give "good" results for GMCA and convergence for PALM is also of uttermost importance. In practice, the number of PALM iterations tends to be much higher than the one of GMCA. Therefore, the refinement stage is in practice much more computationally expensive than the warm-up one. The whole 2-step algorithm is however much faster than a single isolated PALM. This is due to the warm-up stage that enables the refinement stage to start from a good initialization and therefore to converge in less iterations.

E.5 Illustration

The experimental protocol is the same as described in Sec. D.3 (two exactly sparse sources with a square mixing matrix having a condition number of 10 and a SNR of 60 dB) except that the 2-step algorithm is used instead of PALM. The results are plotted in Fig. IV.5. With values of C_A higher than 33 dB, the demixing is close to the best ones obtained with the exhaustive search in Sec. C. Compared to PALM only, the variance of results over different initializations is also largely improved as it is close to zero, which shows the increased *reliability* of the algorithm with regards to the initialization.

F Application to a realistic data separation problem in astrophysics

F.1 Description of the data

The following numerical experiments are carried out on simulated astrophysical data, which have been generated from real Chandra [6] observations of the Cassiopeia A supernova remnants (*cf.* Sec. II-A.1.3). The observations are made of a linear combination of three components : the synchrotron emission and 2 redshifted iron (Fe) emission lines [7]. Compared to *case 4*, there are now $n = 3$ sources (which creates

6. *http ://chandra.harvard.edu/*

7. More precisely, the emissions lines correspond to the telescope impulse response to Dirac-shaped emission lines, which are modeled as Gaussian-shaped spectra. These lines are centered about different energy values, which depend on the relative speed of propagation of each iron

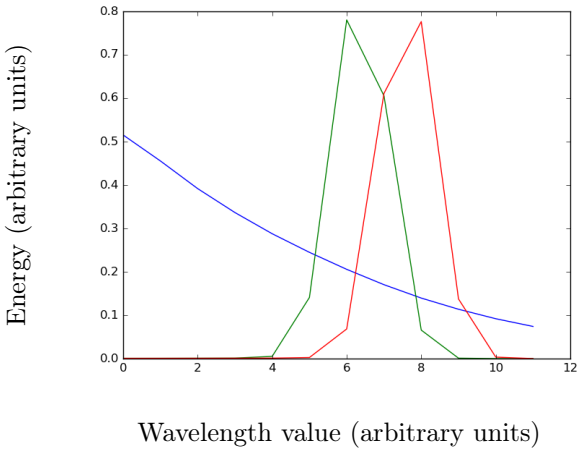


Figure IV.6 – Realistic Chandra $\mathbf{A}^*$ matrix with 12 channels.

more interferences and makes the unmixing more difficult). These are displayed in Fig. IV.7. Furthermore, contrary to *case 4* where $\mathbf{A}^*$ was taken equal to the one of *case 1* and *case 2* to enable simple comparisons, $\mathbf{A}^*$ is now obtained from realistic simulations (*cf.* Fig. IV.6). In particular, we are now in the over-determined setting since $m = 12$ while $n = 3$. Finally, comparisons are performed with 5 different levels of additional noise : SNR = 10, 15, 20, 30 and 60 dB. The practical SNR levels are between 10 and 35 dB.

F.2 Results

Results for different SNR values in terms of C_A are displayed in Table IV.2. The original sources and the ones estimated for a SNR of 30 dB, as well as the difference between both rows, are shown in Fig. IV.7. The "PALM" line corresponds to a PALM algorithm *equipped with the MAD heuristic* described in Section D (which is already a potential improvement compared to an exhaustive search on the regularizing parameters – in the sense that it makes possible to use PALM in the large-scale context – if the interferences are low compared to the noise level). To make the comparison fair with RNA [Zibulevsky 2003] and EFICA [Koldovsky *et al.* 2006], in this subsection the data were pre-whitened.

The separation quality of the 2-step algorithm is good, both for the estimation of $\mathbf{A}^*$ (*cf.* Table IV.2) and $\mathbf{S}^*$ (*cf.* Fig IV.7). In this experiment, the 2-step algorithm always obtains better results than both PALM and GMCA with a gain of about 2 dB for all tested SNR. This highlights the advantage provided by the 2-step approach compared to either GMCA or PALM alone.

Interestingly, the PALM algorithm alone provides rather reasonable separation re-

component due to the Doppler effect. The synchrotron component has a power emission law. These components are representative of typical supernovae remnants in the energy band $5000 - 6000$ eV (electron-volt).

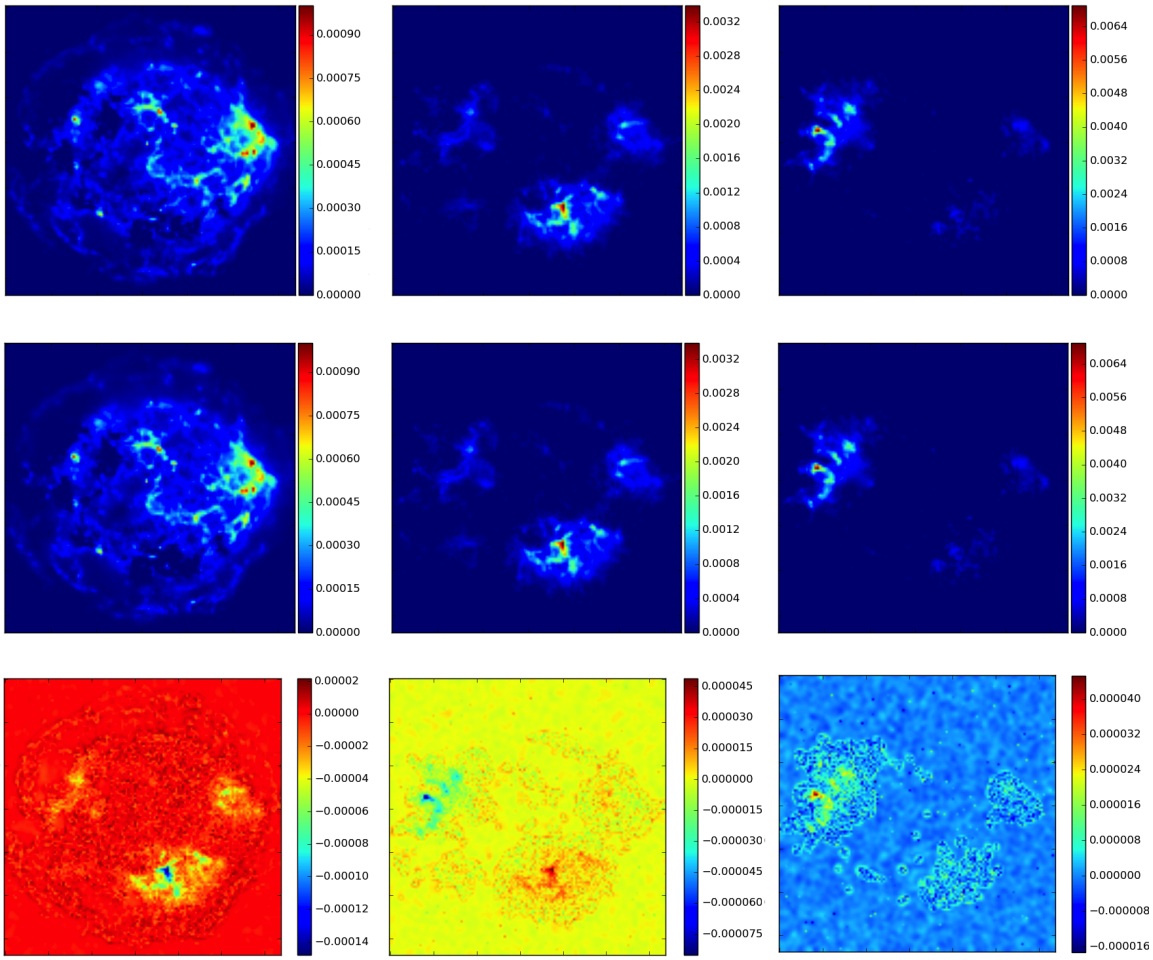


Figure IV.7 – *Up* : true sources $\mathbf{S}^*$; *Middle* : sources estimated by the 2-step algorithm (for a mixing with 30 dB of noise) ; *Down* : residual between upper and middle images.

sults in these experiments. It has to be noticed that in this setting the condition number of the mixing matrix is low, *e.g.* 1.8. Therefore, since $\mathbf{A}^*$ is close to the orthogonality, the $\mathbf{A}^{*T}\mathbf{A}^*\mathbf{s}$ error term of Eq. IV.10 is quite low and potentially sparse. This is precisely the regime where the MAD heuristic can perform correctly in the PALM algorithm (*cf.* Fig IV.9).

The standard sparse ICA-based methods, namely RNA and EFICA, perform rather poorly at low SNR, which highlights their higher sensitivity to noise. In contrast, they perform similarly (or even better) to GMCA in the low noise regime.

	10 dB	15 dB	20dB	30 dB	60 dB
2 step	**15.0**	**16.3**	**17.4**	**19.7**	**20.9**
PALM	11.9	13.3	13.5	14.2	14.5
GMCA	13.2	14.8	15.1	17.1	18.6
EFICA	8.8	10.3	14.0	18.9	19.4
RNA	9.8	12.6	15.6	18.3	18.4

Tableau IV.2 – C_A for 5 SNR values and 5 algorithms.

G Discussion on the 2-step approach

G.1 Limitations of the current 2-step strategy

While the 2-step approach has been shown to perform very well in a realistic astrophysical setting, its limitations and applicability are now discussed in more details.

G.1.1 Explanation of the limitations on the realistic setting

To further understand in which regimes the 2-step is valuable or not, let us have a closer look at Eq. (IV.10) where two important terms must be highlighted : i) the back-projected error on the first guess sources $\mathbf{A}^{*T}\mathbf{A}^{*}\mathbf{s}$ and ii) the noise back-projection $\mathbf{A}^{*T}\mathbf{N}$. For a fixed starting point (*i.e.* fixed error $\mathbf{s}$), both the condition number of the mixing matrix $\mathbf{A}^{*T}$ and the SNR define different regimes. As an illustration, experiments are performed with the astrophysical data described before but with random non-negative mixing matrices with a larger condition number equal to 10. We carry out 10 Monte-Carlo simulations over $\mathbf{A}^{*}$, with for each 10 different random initializations. The results are displayed in Fig. IV.8a, in which the average over the initializations is used. The error bars correspond to the quartiles of the results for different $\mathbf{A}^{*}$.
For larger condition numbers, the re-mixing effect in the PALM iterations plays a prominent role since it tends to concentrate the first guess errors in the subspace spanned by the eigenvectors that are associated with the largest eigenvalues of the Gram matrix of $\mathbf{A}^{*}$. Consequently, the MAD heuristic is more likely to yield badly estimated regularization parameters and the results of the 2-step approach compared to GMCA only are worse than in the previous subsection. This gain will likely depend on the relative levels of $\mathbf{A}^{*T}\mathbf{A}^{*}\mathbf{s}$ and $\mathbf{A}^{*T}\mathbf{N}$, which are displayed in Fig. IV.8b. When the noise level is large enough, the term $\mathbf{A}^{*T}\mathbf{N}$ dominates and the proposed MAD heuristic will perform correctly, which will be favorable for the 2-step approach. This can be observed when the SNR is below 25 dB but not too small : for still smaller SNR, GMCA does not work well, which makes that the 2-step approach has deteriorated performances due to a bad initialization (this is probably due to the pseudo-inverse in GMCA of a badly conditioned $\mathbf{A}$ in the presence of strong noise). For larger SNR (in the range $25 - 50$ dB in Fig. IV.8a), the term

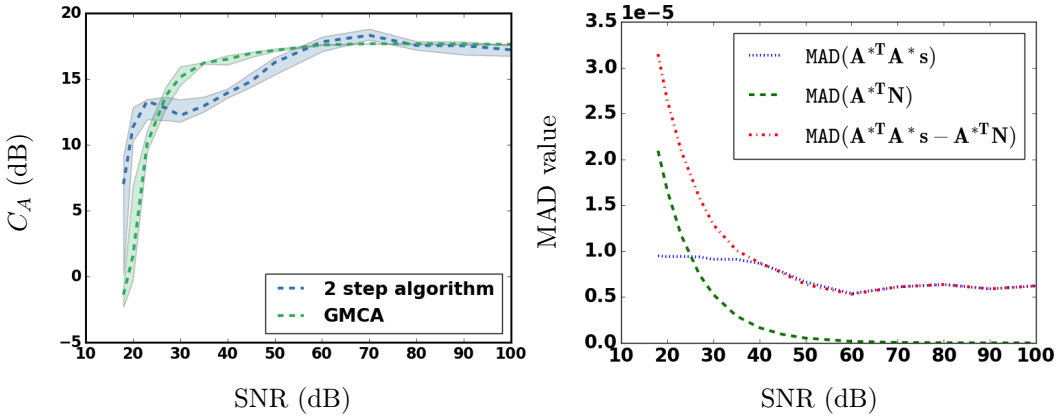


Figure IV.8 – *Left :* C_A as a function of the SNR (in dB), with a condition number of $\mathbf{A}^*$ of 10. To plot the figure, the mean over the realizations has been used. The dashed line corresponds to the mean over the different $\mathbf{A}^*$ and the error bars to the corresponding quartiles. *Right :* Practical influence of the two terms of Eq. (IV.10) in the experiment.

$\mathbf{A}^{*T}\mathbf{A}^{*}\mathbf{s}$ becomes dominant. In that case, the 2-step approach might degrade the GMCA solution. Finally for larger SNR, the contribution of the noise to the thresholds is almost zero. While the MAD therefore yields a bad estimation of optimal parameters, the thresholds are however low because of low interferences due to the relatively good initial point proposed by GMCA. It makes that the solution yielded by the refinement step does not deviate significantly from a simple minimization of the data fidelity term. Since in GMCA the data fidelity term is minimized at each iteration, the solution of the warm-up stage is close to one of its minimizers and therefore the solution of the refinement step does not change much from the one of GMCA. Said differently, in this regime even if the thresholds are bad, they are low enough for the refinement stage estimate to stay close from the one yielded by the warm-up stage, which is already quite good because GMCA performs well with low noise [8].

G.1.2 Further understanding of the limitations of the two step strategy

To further understand the limitations of the two step strategy and the mechanisms at stake, we propose to come back to simulated data. Doing so should in particular enable to understand more deeply the influence of the two terms $\mathbf{A}^{*T}\mathbf{A}^{*}\mathbf{s}$ and $\mathbf{A}^{*T}\mathbf{N}$ by playing on the sparsity level, the noise level and the condition number

8. Note however that while such a conclusion seems to hold true in this case, where the condition number of $\mathbf{A}^*$ is relatively low and where the sources are not too sparse, it might not be the case for all experiments, see the plots of next subsection.

of $\mathbf{A}^*$.

More specifically, the experiences are similar to the one of *case 2* : the sources follow a generalized Gaussian distribution of parameter α, which will vary in the range between $\alpha = 0.05$ (very sparse sources) and $\alpha = 1$ to study the impact of $\mathbf{s}$ in $\mathbf{A}^{*T}\mathbf{A}^*\mathbf{s}$. The noise is Gaussian, with a SNR that will vary from 10 dB to 120 dB, enabling to study $\mathbf{N}$ in the term $\mathbf{A}^{*T}\mathbf{N}$. Finally, we will look at the influence of $\mathbf{A}^*$ in both terms by trying six condition number C_d values, namely $1, 2, 5, 10, 20$ and 100. The matrix $\mathbf{A}^*$ is further drawn randomly from a standard normal distribution and modified to have unit columns.

For each parameter value, we launch the 2-step algorithm and compute the mixing matrix criterion C_A. Each value of C_d yields a 2-D image, displayed in Fig. IV.9. The 6 upper plots correspond the results of the GMCA warm-up stage, and the 6 lower plots to the improvement of the PALM refinement stage over the warm-up stage. The differences between lower and upper parts correspond to two phenomena : i) the different behavior between PALM and the pALS scheme ; ii) the relevance of MAD use within PALM.

Generally speaking, the differences between GMCA and the two step strategy are increased by higher condition numbers C_d, which was expected since in the case of orthogonal matrices ($C_d = 1$), the pseudo-inverse is equal to the transpose : $\mathbf{A}^\dagger = \mathbf{A}^T$.

Moreover, the plots of Fig. IV.9 confirm what was observed in the previous subsection with realistic sources : the 2-step strategy improves GMCA results when the noise level is high (left part of the plots), that is when GMCA does not work well. This is due to the fact that i) the pALS scheme of GMCA and in particular the use of the pseudo-inverse amplifies the noise, which is not the case in PALM ; ii) the MAD is relevant, since the Gaussian noise $\mathbf{A}^{*T}\mathbf{N}$ dominates over the interferences $\mathbf{A}^{*T}\mathbf{A}^*\mathbf{s}$, making the gradient step similar to Gaussian if the estimates are close to the true values.

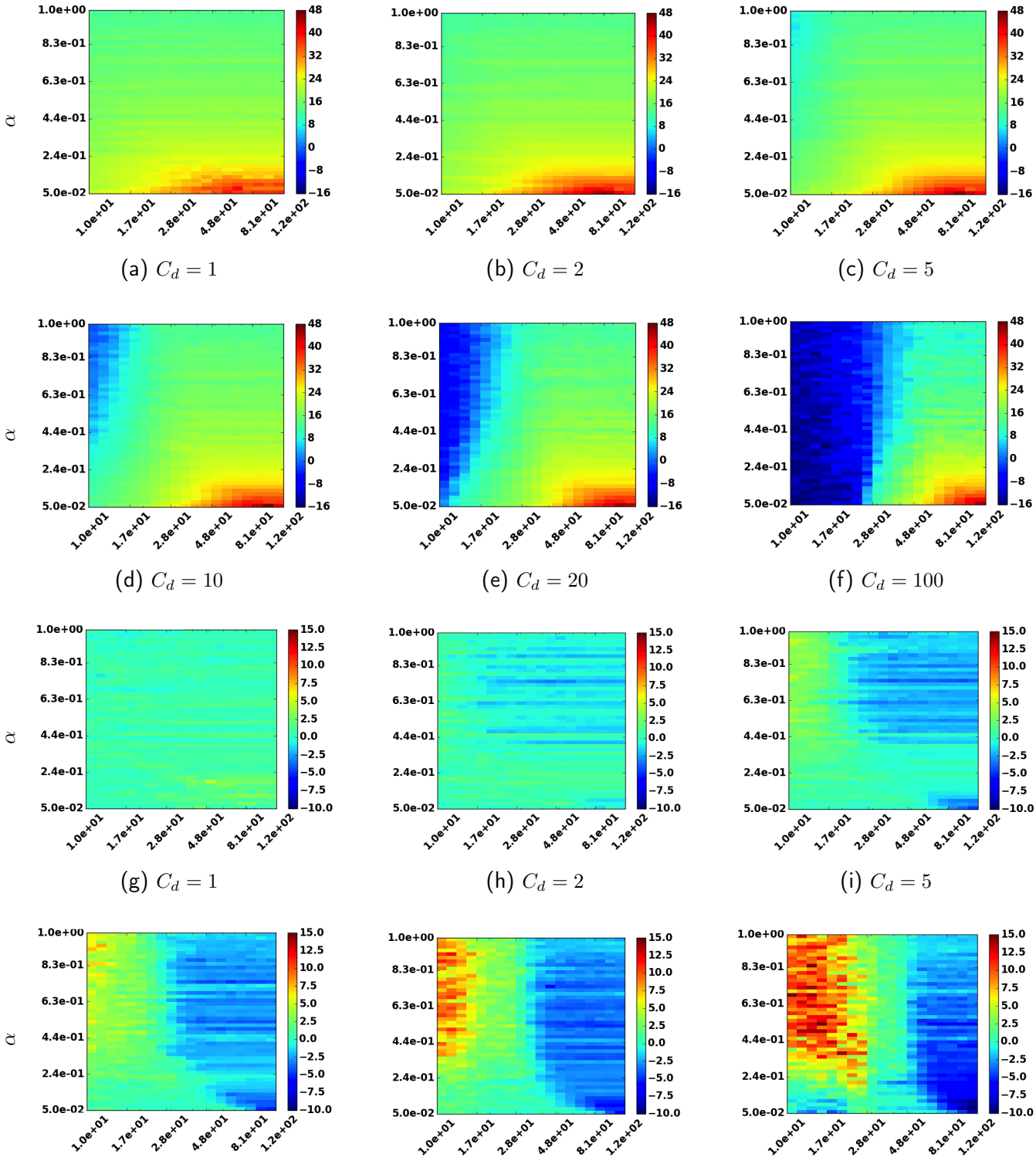
Point ii) can further be made more precise by looking at the shape of the first gradient step within the refinement step (that is, the shape of $\mathbf{A}^T(\mathbf{X} - \mathbf{AS})$ at the first iteration of PALM). Such a shape is displayed in Fig. IV.10 for extreme cases close to the corners of the plots of Fig. IV.9, for a condition number of $C_d = 20$. Fig. IV.10a displays a residual resembling closely to a Gaussian distribution, explaining the good [9] results yielded by the MAD within the PALM stage in the corresponding upper left corner of Fig. IV.9. On the contrary, Fig. IV.10d shows a fully non-Gaussian residual since the noise is very low and the $\mathbf{A}^{*T}\mathbf{A}^*\mathbf{s}$ term is far from Gaussianity (since α is far from 2), thus explaining the deterioration of the 2-step strategy over GMCA in the lower right corner of Fig. IV.9. On the diagonal, while the residual are not Gaussian they are not too different either, explaining acceptable results of the 2-step approach (although the residual seems to be further from a Gaussian in the lower left plots of Fig. IV.9, the better results are likely to come from the

9. The term "good" is to be understood in comparison to the output of the GMCA stage, and not necessarily in terms of the absolute performances of the whole 2-step algorithm – which can be quite bad in this difficult setting where the sources are not very sparse and the noise level high.

fact that the MAD is fairly insensitive to the sparse contamination coming from the interferences $\mathbf{A}^{*T}\mathbf{A}^{*}\mathbf{s}$ – such a contamination is however less sparse in the upper right corner).
Furthermore, we have already emphasized that the use of reweighted ℓ_1 enables to lower the artifacts, that might be transformed into interferences by the gradient step in PALM (*cf.* Sec. E.2). Looking in Fig. IV.11 at the (ideal) residual at convergence $\mathbf{A}^{*T}(\mathbf{N} - \mathbf{A}^{*}\mathbf{s})$ for highly sparse $\mathbf{S}^{*}$ and high SNR enables understanding another interest of reweighting : it transforms a highly non-Gaussian residual into a Gaussian one. Using reweighted ℓ_1, the MAD is therefore more likely to perform correctly and to enables a threshold choice cancelling the erroneous updates, which confirms that the reweighting is of uttermost importance in the 2-step strategy.

Conclusion

The applicability of sparse BSS methods to real-world data, especially for large-scale ones, largely depends on the design of *reliable*, *efficient* and *versatile* methods. Methods such as GMCA presenting these properties rely on heuristics that do not however guarantee any optimality of the estimate. On the contrary, the estimation using PALM algorithm to minimize a ℓ_1 regularized data fidelity term, which is now a standard approach to tackle generic matrix factorization problems due to mathematical guarantees, does not always exhibit such characteristics. To develop an optimization framework merging the best of the two worlds, we first investigated the behavior of PALM in the context of sparse BSS. These investigations show and explain the dramatic sensitivity of the estimate with respect to the regularization parameters (lack of *efficiency* and *versatility*) and also to the initialization (lack of *reliability*). To mitigate these limitations, we rationalize a hybrid approach that allows to circumvent the robustness issue of PALM with respect to the initialization and provides a proxy to automatically tune the regularization parameters for various data. Numerical experiments on both simulated and realistic data demonstrate the quality of the proposed approach with respect to the three desired characteristics. They also exhibits an improved separation accuracy with respect to state-of-the art methods. On the other hand, an extensive study of the limitations of the proposed strategy is performed, enlightening when to use it. As supplementary material, we also propose in Appendix E several exploratory alternative ways of how to perform a two-step strategy, which could pave the way for various enhancements.



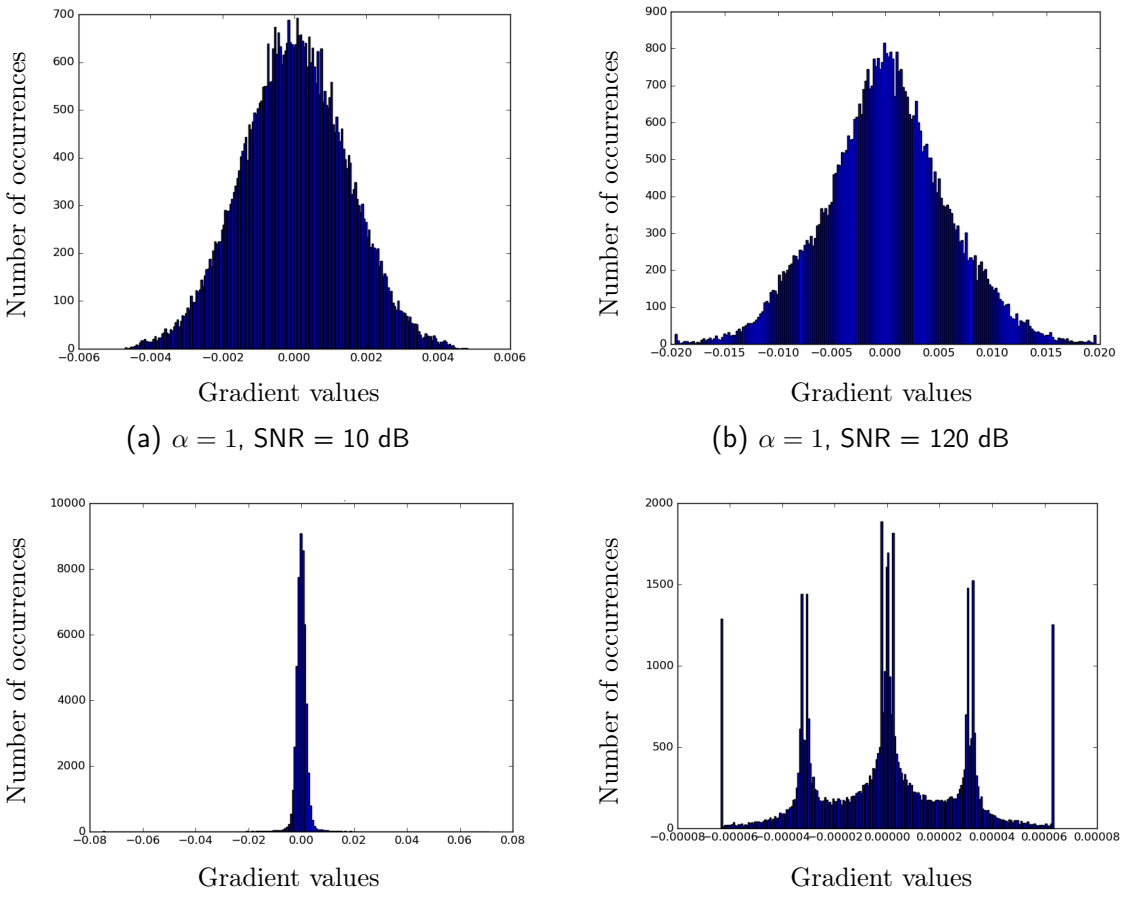
(a) $C_d = 1$ (b) $C_d = 2$ (c) $C_d = 5$

(d) $C_d = 10$ (e) $C_d = 20$ (f) $C_d = 100$

(g) $C_d = 1$ (h) $C_d = 2$ (i) $C_d = 5$

(j) $C_d = 10$ (k) $C_d = 20$ (l) $C_d = 100$

Figure IV.9 – Mixing matrix criterion C_A (dB) as a function of the SNR and the sparsity level, for 6 values of condition number C_d. *2 upper rows :* results of GMCA warm-up stage ; *2 bottom rows :* improvement over GMCA results yielded by the PALM refinement stage equipped with the MAD strategy.



(a) $\alpha = 1$, SNR = 10 dB

(b) $\alpha = 1$, SNR = 120 dB

(c) $\alpha = 0.1$, SNR = 10 dB

(d) $\alpha = 0.1$, SNR = 120 dB

Figure IV.10 – Histograms of the gradient in the first iteration of PALM : ${\mathbf{A}_{\mathbf{GMCA}}}^T(\mathbf{X} - \mathbf{A}_{\mathbf{GMCA}}\mathbf{S}_{\mathbf{GMCA}})$, for a condition number of $C_d = 20$. The sparsity level α and the SNR are specified for each plot.

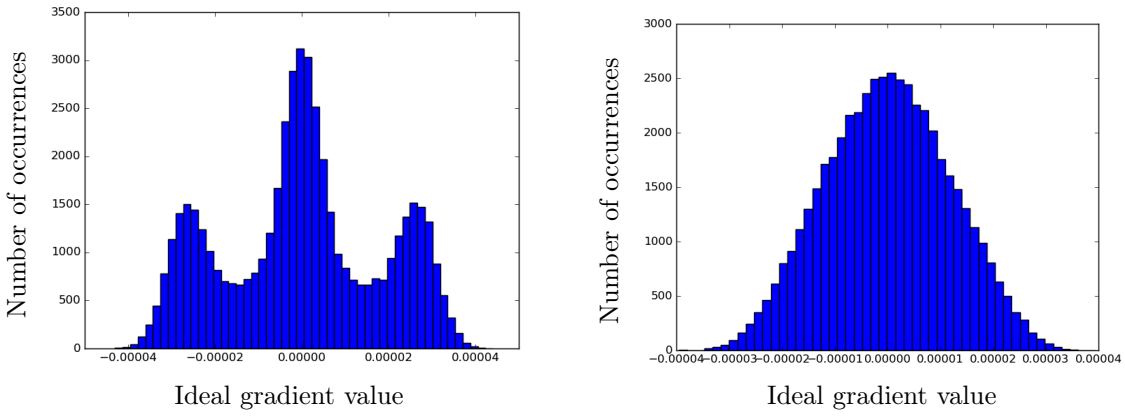


Figure IV.11 – Influence of reweighted ℓ_1 on the ideal gradient shape, that is the gradient computed with the true $\mathbf{A}^*$ and $\mathbf{S}^*$ according to Eq. IV.10. The error $\mathbf{s}$ then corresponds only to the bias introduced by the thresholding. In this experiment, $C_d = 20$, SNR $= 60$ and $\alpha = 0.2$. *Left :* $\mathbf{A}^{*T}(\mathbf{N} - \mathbf{A}^*\mathbf{s})$, with $\mathbf{s} = \mathcal{S}_{\mathbf{\Lambda}_{\mathbf{S}^*}}(\mathbf{S}^*) - \mathbf{S}^*$: no reweighted ℓ_1 is used for the threshold choice, *Right :* $\mathbf{A}^{*\mathbf{T}}(\mathbf{N} - \mathbf{A}^*\mathbf{s})$, with $\mathbf{s} = \mathcal{S}_{\mathbf{R}_{\mathbf{S}^*}}(\mathbf{S}^*) - \mathbf{S}^*$, $\mathbf{R}_{\mathbf{S}^*} = \mathbf{\Lambda}_{\mathbf{S}^*} \odot \mathbf{G}$, with $\mathbf{G}$ accounting for reweighted ℓ_1.

Chapitre V

Tackling a high number of sources : blockGMCA

The goal of the previous chapter was to find both a reasonable initialization for a sparse BSS algorithm and a way to set the regularization parameters, which was a first mandatory step to hope to handle large-scale problems. In this chapter, we now enter the core subject of this thesis with a first large-scale issue of BSS : handling a large number of sources n. This problem is both difficult, since most methods fail when the number of sources typically exceeds a few tens, and of paramount importance in various applications such as spectroscopy, astronomy [Bobin *et al.* 2008] or biomedical imaging [Biswal & Ulmer 1999].
The proposed approach focuses on the optimization strategy, which has already been deemed as crucial in the previous chapters. More specifically, the proposed block-Generalized Morphological Component Analysis (bGMCA) algorithm builds upon block-coordinate descent with intermediate size blocks. Numerical experiments are provided that show the quality of the approach when the sources are numerous.

A Problem and outline

A.1 Problem : decreased performances with large numbers of sources

We illustrate in Fig. V.1 the performance deterioration of most BSS methods when the number of sources n becomes large. The evolution of the mixing matrix criterion as a function of the number of sources shows that most methods do not perform correctly in such a large-scale regime. In this case, the main source of deterioration is very likely related to the non-convex nature of BSS problem and a regularization issue. Indeed, for a fixed number of samples t, an increasing number of sources n will make these algorithms more prone to be trapped in spurious local minima, which tends to hinder the applicability of BSS on practical issues with a large n. Consequently, the optimization strategy has a huge impact on the separation performances.

A.2 Outline

The goal of this chapter is to introduce a novel algorithm dubbed bGMCA to specifically tackle sparse BSS problems when many sources need to be estimated. In addition to sparse modelling, this algorithm builds upon an efficient minimization

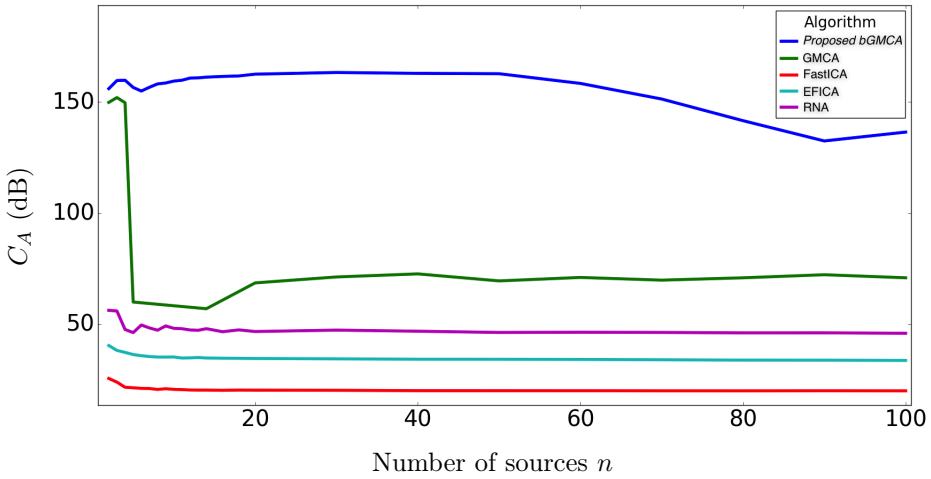


Figure V.1 – Evolution of the mixing matrix criterion C_A (*cf.* Appendix B) of four standard BSS algorithms for an increasing n. For comparison, the results of the proposed *bGMCA* algorithm are presented, showing that its use allows the good results of GMCA for low n (around 160 dB for $n = 3$) to persist for $n < 50$ and to stay much better than GMCA for $n > 50$. The experiment was conducted using exactly sparse sources $\mathbf{S}^*$, with 10% non-zero coefficients, the other coefficients having a Gaussian amplitude. The mixing matrix $\mathbf{A}^*$ was taken orthogonal. Both $\mathbf{A}^*$ and $\mathbf{S}^*$ were generated randomly, the experiments being done 25 times and the median used to draw the figure.

scheme based on block-coordinate descent, as explained in Section B. In contrast to state-of-the art methods [Zibulevsky 2003, Bobin *et al.* 2015, Rapin *et al.* 2014, Gillis & Glineur 2012], we show that block-based minimization with intermediate block sizes allows the bGMCA to dramatically enhance the separation performances for large n. This is demonstrated through comparisons with state-of-the art methods in Section C, which have been carried out on various simulation scenarios. The last part of the Chapter shows the flexibility of *bGMCA*, with an application to sparse and non-negative BSS in the context of spectroscopy.

B Proposed approach : use of intermediate block-sizes

B.1 State-of-art

As explained in Chapter III-C, to bypass both the coupling between $\mathbf{A}$ and $\mathbf{S}$ and the non-convexity of problem II.7, a common idea of several strategies (BCD [Tseng 2001], PALM [Bolte *et al.* 2014], ALS) is to benefit from the multi-convex structure of (II.7) by using blocks [Xu & Yin 2014] in which each sub-problem is convex. The minimization is then performed alternately with respect to one of the coordinate blocks while the other coordinates stay fixed, which entails solving a sequence of convex optimization problems. Most of the already existing methods

can then be categorized in one of two families, depending on the block sizes :

- *Hierarchical or deflation methods* : these algorithms use a block of size 1. For instance, Hierarchical ALS (HALS) ([Gillis & Glineur 2012] and references therein, [Comon & Jutten 2010]) updates only one specific column of $\hat{\mathbf{A}}$ and one specific row of $\hat{\mathbf{S}}$ at each iteration. The main advantage of this family is that each subproblem is often much simpler as their minimizer generally admits a closed-form expression. Moreover, the matrices involved being small, the computation time is much lower. The drawback is however that the errors on some sources/mixing matrix columns propagate from one iteration to the other since they are updated independently.
- *Full-size blocks* : these algorithms use as blocks the whole matrices $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$ (the block size is thus equal to n). For instance, GMCA [Bobin *et al.* 2008] is part of this family. One problem compared to hierarchical or deflation methods is that the problem is more complex due to the simultaneous estimation of a high number of sources. Moreover, the computational cost increases quickly with the number of sources, since large-size problems need to be handled (*e.g.* in GMCA, the pseudo-inverse of the whole matrices $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$ needs to be computed at each iteration).

The gist of the proposed *bGMCA* algorithm is to adopt an alternative approach that uses intermediate block sizes. The underlying intuition is that using blocks of intermediate size can be recast as relatively small-scale source separation problems, which are simpler to solve as testified by Fig. V.1. As a byproduct, these subproblems are also less costly to tackle. On the other hand, they are not small enough to incur dramatic error propagation.

B.2 Proposed approach

In the following, *bGMCA* minimizes the problem in eq. (II.7) with *blocks*, which are indexed by a set of indices I of size r, $1 \leqslant r \leqslant n$. In practice, the minimization is performed at each iteration on submatrices of $\hat{\mathbf{A}}$ (keeping only the columns indexed by I) and $\hat{\mathbf{S}}$ (keeping only the rows indexed by I).
We thereafter re-used the work described in the previous chapter and used a 2-step minimization strategy *using intermediate-size coordinate blocks*. As before, it comprehends a GMCA as warm-up stage and PALM as refining stage, enabling the various benefits described in Chapter III. In the following, we describe both steps.

B.2.1 Warm-up stage

In the framework of the proposed *bGMCA* algorithm, the GMCA-based warm-up stage uses blocks of size $1 \leqslant r \leqslant n$ and alternates between the update of some *submatrices* of $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$ (these submatrices will be noted $\hat{\mathbf{A}}_I$ and $\hat{\mathbf{S}}_I$). The whole stage is summarized below :

0 - Initialize the algorithm with random $\hat{\mathbf{A}}^{(0)}$.

For each iteration (l) :

1 - A submatrix $\hat{\mathbf{S}}_I$ is now updated instead of the whole $\hat{\mathbf{S}}$. This is performed using a projected least square solution :

$$\hat{\mathbf{S}}_I^{(l)} = \text{prox}_{\mathcal{G}(.)}(\hat{\mathbf{A}}^{I^{(l-1)\dagger}}\mathbf{R}_I) \tag{V.1}$$

where : $\mathbf{R}_I$ is the residual term defined by $\mathbf{R}_I = \mathbf{X} - \hat{\mathbf{A}}^{I^C{}^{(l-1)}}\hat{\mathbf{S}}_{I^C}^{(l-1)}$ (with I^C the indices of the sources outside the block), which is the part of $\mathbf{X}$ to be explained by the sources in the current block I.

2 - The mixing sub-matrix $\hat{\mathbf{A}}^I$ is similarly updated with a fixed $\mathbf{S}$:

$$\hat{\mathbf{A}}^{I^{(l)}} = \text{prox}_{\mathcal{J}(.)}(\mathbf{R}_I\mathbf{S}_I^{(l)\dagger}) \tag{V.2}$$

In this chapter, the penalizations $\mathcal{J}$ and $\mathcal{G}$ we will consider are the ones described in Section III-B.2.

B.2.2 Refinement stage

The PALM-based refinement stage using intermediate-size blocks reads as :

While the stopping criterion $\Delta^{(l)}$ has not reached the desired value, iterate over (l) :

1 - Update of a submatrix $\hat{\mathbf{S}}_I$ instead of the whole $\hat{\mathbf{S}}$:

$$\hat{\mathbf{S}}_I^{(l)} = \text{prox}_{\frac{\gamma\mathcal{G}(.)}{\left\|\hat{\mathbf{A}}^{I^{(l-1)T}}\hat{\mathbf{A}}^{I^{(l-1)}}\right\|_2}}\left(\hat{\mathbf{S}}_I^{(l-1)} - \frac{\gamma}{\left\|\hat{\mathbf{A}}^{I^{(l-1)T}}\hat{\mathbf{A}}^{I^{(l-1)}}\right\|_2}\hat{\mathbf{A}}^{I^{(l-1)T}}(\hat{\mathbf{A}}^{(l-1)}\hat{\mathbf{S}}^{(l-1)} - \mathbf{X})\right) \tag{V.3}$$

2 - Update of a submatrix $\hat{\mathbf{A}}^I$ instead of the whole $\hat{\mathbf{A}}$:

$$\hat{\mathbf{A}}_I^{(l)} = \text{prox}_{\frac{\delta\mathcal{J}(.)}{\left\|\hat{\mathbf{S}}_I^{(l)}\hat{\mathbf{S}}_I^{(l)T}\right\|_2}}\left(\hat{\mathbf{A}}^{I^{(l-1)}} - \frac{\delta}{\left\|\hat{\mathbf{S}}_I^{(l)}\hat{\mathbf{S}}_I^{(l)T}\right\|_2}(\hat{\mathbf{A}}^{(l-1)}\hat{\mathbf{S}}^{(l)} - \mathbf{X})\hat{\mathbf{S}}_I^{(l)T}\right) \tag{V.4}$$

3 - Update stopping criterion : $\Delta^{(l)} = \frac{\sum_{j\in[1,n]}\left\|\hat{\mathbf{A}}^{j^{(l)}} \odot \hat{\mathbf{A}}^{j^{(l-1)}}\right\|_1}{n}$

Where the notations and the different constants are the same as in Chapter III.

B.2.3 Block choice

Several strategies for selecting at each iteration new block indices I have been investigated :

— *Sequential* : at each iteration, r sources are selected sequentially in a cyclic way ;

— *Random* : at each iteration, r indices in $[1, n]$ are randomly chosen following a uniform distribution and the corresponding sources updated ;

— *Random sequential* : this strategy combines the sequential and the random choices to ensure that all sources are updated an equal number of times.

In the experiments, random strategies tended to provide better results. Indeed, compared to a sequential choice, randomness is likely to make the algorithm more robust with respect to spurious local minima. Since the results between the random strategy and the random sequential one are similar, the first was eventually selected.

B.2.4 Convergence

The use of intermediate-size blocks do not alter the convergence guarantees of PALM algorithm with fixed thresholds. As such, the refinement stage converges to a stationary point of eq. (II.7), as long as the blocks are updated following an essentially cyclic rule [Chouzenoux *et al.* 2016] or even if they are chosen randomly and updated one by one [Patrascu & Necoara 2015].

B.2.5 Complexity

In this part, we focus only on the warm-up stage, which iterations are the most computationally expensive. Each iteration can then be decomposed into the following elementary steps : i) a residual term is computed with a complexity of $\mathcal{O}(mtr)$, where m is the number of observations, t the number of samples and r the block size ; ii) the pseudo-inverse is performed with the singular value decomposition of a $r \times r$ matrix, which yield an overall complexity of $\mathcal{O}(r^3 + r^2 m + m^2 r)$; iii) the thresholding-strategy first requires the evaluation of the threshold values, which has a complexity of rt ; iv) then the soft-thresholding step which has complexity $\mathcal{O}(rt)$; and v) updating $\mathbf{A}$ is finally performed using a conjugate gradient algorithm, whose complexity is known to depend on the number of non-zero entries in $\mathbf{S}$ and on the condition of this matrix $C_d(\mathbf{S})$. An upperbound for this complexity is thus $\mathcal{O}(rt\sqrt{C_d(\mathbf{S})})$. The final estimate of the complexity of a single iteration is finally given by :

$$r[mt + rm + m^2 + r^2 + t\sqrt{C_d(\mathbf{S})}] \tag{V.5}$$

With $C_d(\mathbf{S})$ the conditioning number of $\mathbf{S}$. Thus, both the r factor and the behavior in r^3 show that small r values will lower the computational budget of each iteration. Since the algorithm is iterative, the final running time will however depend on both the complexity of each iteration and of the number of iterations. Intuitively, the required number of iterations should be inversely proportional to r, since only r

sources are updated at each iteration, requiring $\lceil n/r \rceil$ times the number of iterations needed by an algorithm using the full matrices. As will be emphasized later on, the number of required iterations will be smaller than expected, which on overall makes that the *bGMCA* algorithm enables a reduction of the computation time.

C Explaining the behavior of bGMCA : numerical experiments on simulated data

In this part, we present our results on simulated data. The goal is to show and to explain on simple data how *bGMCA* works.

C.1 Experimental protocol

The simulated data were generated in the following way :

1 - Source matrix $\mathbf{S}^*$: the sources are exactly sparse in the sample domain (that is, $\mathbf{\Phi_S} = \mathbf{I_d}$ – the results would however be identical for any source sparse in an orthogonal representation). Their coefficients are drawn randomly according to a Bernoulli-Gaussian distribution : among the t samples ($t = 1\ 000$), a proportion p (unless specified, $p = 0.1$) of the samples is non-zero, with an amplitude drawn according to a standard normal distribution.

2 - Mixing matrix $\mathbf{A}^*$: the mixing matrix is drawn randomly according to a standard normal distribution and modified to have unit columns and a given condition number C_d (unless specified, $C_d = 1$).

The number of observations m is taken equal to the number of sources : $m = n$. In this first simulation, no noise is added. The number of iterations for the warm-up stage is 10 000. Here, the following penalizations $\mathcal{G}$ and $\mathcal{J}$ were used (for more mathematical details and the corresponding proximal operators, see Sec. III-B.2) :

- ℓ_1 *sparsity constraint in some transformed domain* : The constraint $\mathcal{G}$ on $\mathbf{S}$ is a ℓ_1-norm penalization.
- *Oblique constraint* : to avoid degenerated $\mathbf{A}$ and $\mathbf{S}$ matrices, the columns of $\mathbf{A}$ are constrained through $\mathcal{J}$ to lie onto the ℓ_2 hyper-sphere.

To measure the accuracy of the separation, we again followed the definition in [Bobin *et al.* 2015] to use the global criterion C_A on $\mathbf{A}$ (*cf.* Appendix B). The data matrices being drawn randomly, each experiment was performed several times (typically 25 times) and the median of C_A over the experiments will be displayed.

C.2 Modeling block minimization

In this section, a simple model is introduced to describe the behavior of the warm-up stage of the *bGMCA* algorithm. As described in section B.2, updating a given block is performed at each iteration from the residual $\mathbf{R}_I = \mathbf{X} - \hat{\mathbf{A}}^{I^C}\hat{\mathbf{S}}_{I^C}$ (to

lighten the notations, we dropped the iteration number $l-1$). If the estimation were perfect, the residual would be equal to the part of the data explained by the true sources in the current block indexed by I, which would read : $\mathbf{R}_I = \mathbf{A}^{I*}\mathbf{S}_I^*$.
It is nevertheless mandatory to take into account the noise $\mathbf{N}$, as well as a variety of flaws in the estimation by adding a term $\mathcal{E}$ to model the estimation error. This entails :

$$\mathbf{R}_I = \mathbf{X} - \hat{\mathbf{A}}^{I^C}\hat{\mathbf{S}}_{I^C} = \mathbf{A}^{I*}\mathbf{S}_I^* + \mathcal{E} + \mathbf{N} \tag{V.6}$$

A way to further describe the structure of $\mathcal{E}$ is to decompose the estimated $\hat{\mathbf{S}}_I$ matrix in the true matrix plus an error : $\hat{\mathbf{S}}_I = \mathbf{S}_I^* + \mathbf{s}_I$ and $\hat{\mathbf{S}}_{I^C} = \mathbf{S}_{I^C}^* + \mathbf{s}_{I^C}$, where $\mathbf{s}$ is the error on $\mathbf{S}^*$. Assuming that the errors are small and neglecting the second-order terms, the residual $\mathbf{R}_I$ can now be written as :

$$\mathbf{R}_I = \mathbf{X} - \hat{\mathbf{A}}^{I^C}\hat{\mathbf{S}}_{I^C} = \mathbf{A}^{I*}\mathbf{S}_I^* + \mathbf{A}^{I^{C*}}\mathbf{S}_{I^C}^* - \hat{\mathbf{A}}^{I^C}\hat{\mathbf{S}}_{I^C}^* - \hat{\mathbf{A}}^{I^C}\mathbf{s}_{I^C} + \mathbf{N} \tag{V.7}$$

This implies that :

$$\mathcal{E} = (\mathbf{A}^{I^{C*}} - \hat{\mathbf{A}}^{I^C})\mathbf{S}_{I^C}^* - \hat{\mathbf{A}}^{I^C}\mathbf{s}_{I^C} \tag{V.8}$$

Equation (V.8) highlights two terms. The first term can be qualified as interferences in that it comes from a leakage of the *true* sources that are *outside* the currently updated block. This term vanishes when $\hat{\mathbf{A}}^{I^C}$ is perfectly estimated. The second term corresponds to interferences as well as artefacts. It originates indeed from the *error* on the sources *outside* the block I. The artefacts comprehend in particular the errors on the sources induced by the soft thresholding corresponding to the ℓ_1-norm. Equation (V.8) also allows us to understand how the choice of a given block size $r \leqslant n$ will impact the separation process :

- Updating small-size blocks can be recast as a small-size source separation problem where the actual number of sources is equal to r. As testified by Fig. V.1, updating small-size block problems should be easier to tackle.
- Small-size blocks should also yield larger errors $\mathcal{E}$. It is intuitively due to the fact that many potentially badly estimated sources in I^C are used for the estimation of $\mathbf{A}^{I*}$ and $\mathbf{S}_I^*$ through the residual, deteriorating this estimation. It can be explained in more details using equation (V.8) : with more sources in I^C, the energy of $\mathbf{A}^{I^C}$, $\hat{\mathbf{A}}^{I^{C*}}$, $\mathbf{S}_{I^C}^*$ and ε_{I^C} increases, yielding bigger error terms $(\mathbf{A}^{I^{C*}} - \hat{\mathbf{A}}^{I^C})\mathbf{S}_{I^C}^*$ and $-\hat{\mathbf{A}}^{I^C}\mathbf{s}_{I^C}$. Therefore the errors $\mathcal{E}$ become higher, deteriorating the results.

C.3 Experiment

In this section, we investigate the behavior of the proposed block-based GMCA algorithm with respect to various parameters such as the block size r, the number of sources n, the conditioning of the mixing matrix C_d and the sparsity level of the sources p.

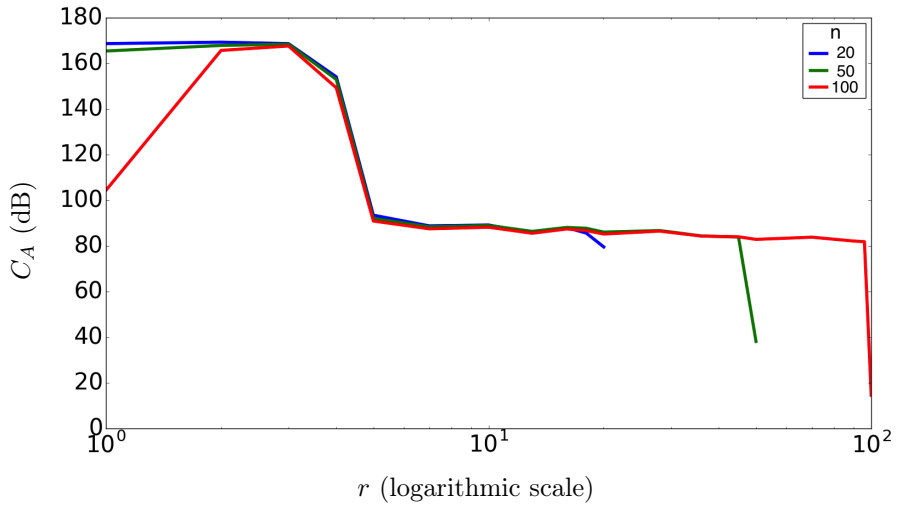
C.3.1 Study of the impact of r and n

In this subsection, *bGMCA* is evaluated for different numbers of sources $n = 20, 50, 100$. Each time the block sizes vary in the range $1 \leq r \leq n$. In this experiment and to complete the description of section C.1, the parameters for the matrix generation were : $p = 0.1$, $t = 1\ 000$, $C_d = 1$, $m = n$, with a Bernoulli-Gaussian distribution for the sources. These results are displayed in Fig. V.2a. Interestingly, three different regimes characterize the behavior of the *bGMCA* algorithm :

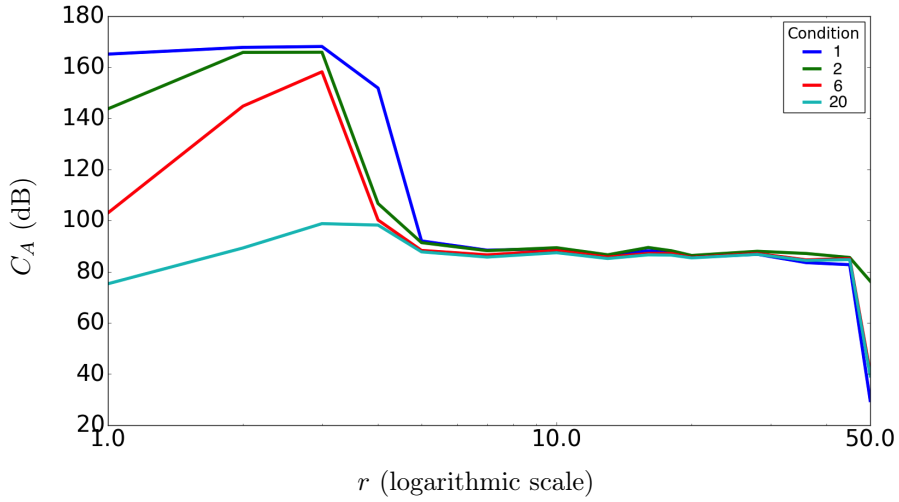
- For intermediate and relatively large block sizes (*typically* $r > 5$ and $r < n-5$) : we first observe that after an initial deterioration around $r = 5$, the separation quality does not vary significantly for increasing block sizes. A degradation of several dB can then be observed for r close to n. In all this part of the curve, the error term $\mathcal{E}$ is composed of residuals of sparse sources, and thus $\mathcal{E}$ will be rather sparse when the block size is large. Based on the MAD, the thresholds are set according to dense and not to sparse noise. Consequently the automatic thresholding strategy of the *bGMCA* algorithm will not be sensitive to the estimation errors.
- A very prominent peak can be observed when the block size is of the order of 3. Interestingly, the maximum yields a mixing matrix criterion of about 10^{-16}, which means that perfect separation is reached up to numerical errors. This value of 160 dB is at least 80 dB larger than in the standard case $r = n$, for which the values for the different n are all below 80 dB. In this regime, error propagation is composed of the mixture of a larger number of sparse sources, which eventually entails a densely distributed contribution that can be measured by the MAD-based thresholding procedure. Therefore, the threshold used to estimate the sources is able to filter out both the noise and the estimation errors. Moreover, $r = 5$ is quite small compared to n. Following the modeling introduced in section C.2, small block sizes can be recast as a sequence of low-dimensional blind source separation problems, which are simpler to solve.
- For small block sizes (*typically* $r < 4$), the separation quality is deteriorated when the block size decreases, especially for large n values. In this regime, the level of estimation error $\mathcal{E}$ becomes large, which entails large values for the thresholds $\mathbf{\Lambda}$. Consequently, the bias induced by the soft-thresholding operator increases, which eventually hampers the performance quality. Furthermore, for a fixed block size r, $\mathcal{E}$ increases with the number of sources n, making this phenomenon more pronounced for higher n values.

C.3.2 Condition number of the mixing matrix

In this section, we investigate the role played by the conditioning of the mixing matrix on the performances of the *bGMCA* algorithm. Fig. V.2b displays the empirical results for several condition numbers C_d of the $\mathbf{A}^*$ matrix. There are $n = 50$ sources generated in the same way as in the previous experiment : with a Bernoulli-



(a) Number of sources.



(b) Condition number.

Figure V.2 – *Up :* mixing matrix criterion as a function of r for different n. *Right :* mixing matrix criterion as a function of r for different C_d.

Gaussian distribution and $p = 0.1$, $t = 1000$. One can observe that when C_d increases, the peak present for r close to 5 tends to be flattened, which is probably due to higher projection errors. At some iteration l, the sources are estimated by projecting $\mathbf{X} - \hat{\mathbf{A}}^{I^C}\hat{\mathbf{S}}_{I^C}$ onto the subspace spanned by $\hat{\mathbf{A}}_I$. In the orthogonal case, the projection error is low since $\hat{\mathbf{A}}^{I^C}$ and $\hat{\mathbf{A}}_I$ are close to orthogonality at the solution. However, this error increases with the condition number C_d.

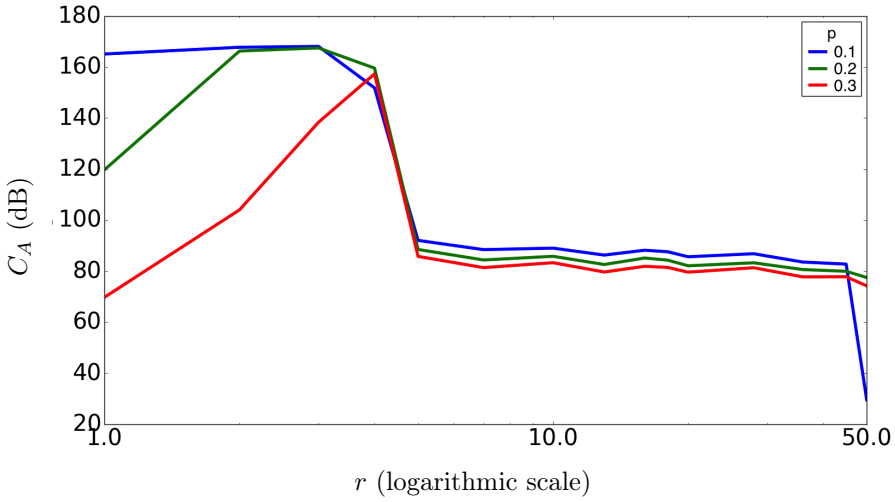


Figure V.3 – Mixing matrix criterion as a function of r for different sparsity degrees.

C.3.3 Sparsity level p

In this section, the impact of the sparsity level of the sources is investigated. The sources are still following a Bernoulli-Gaussian distribution. The parameters are : $n = 50$, $t = 1\ 000$, $C_d = 1$. As featured in Figure V.3, the separation performances at the maximum value decrease slightly with larger p, while a slow shift of the transition between the small/large block size regimes towards larger block sizes operates. Furthermore, the results tend to deteriorate quickly for small block sizes ($r < 4$). Indeed, owing to the model of subsection C.2, the contribution of $\mathbf{S}^*_{IC}$ and $\mathbf{s}_{IC}$ in the error term (V.8) increases with p, this effect being even more important for small r (which could also explain the shift of the peak for $p = 0.3$, by a deterioration of the results at its beginning, $r = 3$). When p increases, the sources in $\hat{\mathbf{S}}_I$ become denser. Instead of being mainly sensitive to the noise and $\mathcal{E}$, the MAD-based thresholding tends to be perturbated by $\hat{\mathbf{S}}_I$, resulting in more artefacts, which eventually hampers the separation performances. This effect increases when the sparsity level of the sources decreases.

C.3.4 Number of iterations and computation time

We have already seen in Section B.2.5 that the *bGMCA* algorithm enables a gain in terms of computational complexity of each iteration (roughly speaking, this gain is almost linear if t is much larger than m and r). We here further empirically assess the actual *number of iterations* required by the warm-up stage to yield a good initialization. To this end, the following experiment has been conducted :

1. First, the algorithm is launched with a large number of iterations (*e.g.* 10 000) to give good $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$ matrices. The corresponding value of C_A is saved and called C^*_A.

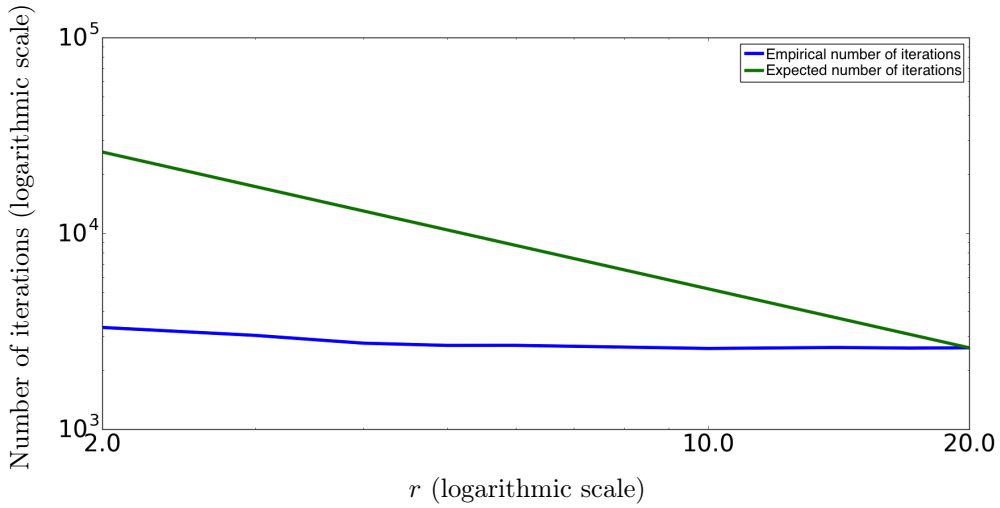


Figure V.4 – Right : number of iterations in logarithmic scale as a function of r.

2. Using the same initial conditions, the warm-up stage is re-launched and stops when the mixing matrix criterion reaches $1.05 \times C_A^*$ (*i.e.* 5% of the "optimal" solution for a given setting).

The number of iterations needed to reach the 5% accuracy is reported in Fig. V.4. Intuitively, one would expect that when the block size decreases, the required number of iterations should increase by about n/r to keep the number of updates per source constant. This trend is displayed with the straight green curve of Fig. V.4. Interestingly, Fig. V.4 shows that the actual number of iterations to reach the 5% accuracy criterion almost does not vary with r. Consequently, on top of leading to computationally cheaper iterations, using small block sizes almost requires the same number of iterations for the warm-up stage to give a good initialization. Therefore, the use of blocks allows a huge decrease of the computational cost of the warm-up stage and thus of sparse BSS.

D Validation of the approach on realistic sources

D.1 Context

The goal of this part is to evaluate the behavior of *bGMCA* and show its efficiency in a more realistic setting. Our data come from a simulated LC - ^{1}H NMR (Liquid Chromatography - ^{1}H Nuclear Magnetic Resonance) experiment. The objective of such a experiment is to identify each of the chemicals compounds present in a fluid, as well as their concentrations. As explained in Chapter II-A.1.2 (the principles of LC - ^{1}H NMR and LC - MS experiments are the same), the LC - ^{1}H NMR experiment enables a first physical imperfect separation during which the fluid goes through a chromatography column and its chemicals are separated according to their speeds (which themselves depend on their physical properties). Then, the spectrum of the

output of the column is measured at a given time frequency. These measurements of the spectra at different times can be used to feed a *bGMCA* algorithm to refine the imperfect physical separation.

The fluids on which we worked could for instance correspond to drinks. The goal of *bGMCA* is then to identify the spectra of each compound (*e.g.* caffeine, sucrose, menthone...) and the mixing coefficients (which are proportional to their concentrations) from the LC - ^{1}H NMR data. BSS has already been successfully applied [Toumi *et al.* 2013] to similar problems but generally with lower number of sources n.

The sources ($n = 40$ sources with each $t = 10\,000$ samples) are composed of elementary sparse non-negative theoretical spectra of chemical compounds taken from the SDBS database [1], which are further convolved with a Laplacian having a width of 3 samples to simulate a given spectral resolution. Therefore, each convolved source becomes an approximately sparse non-negative row of $\mathbf{S}^*$ (*cf.* Fig. V.6). The mixing matrix $\mathbf{A}^*$ of size (m,n) = (320, 40) is composed of Gaussians (see Fig. V.5), the objective being to have a matrix that could be consistent with the first imperfect physical separation. It is designed in two parts : the first columns have relatively spaced Gaussian means while the others have a larger overlap to simulate compounds for which the physical separation is less discriminative. More precisely, an index $\bar{m} \in [1, m]$ is chosen, with $\bar{m} > m/2$ (typically, $\bar{m} = \lceil 0.75m \rceil$). A set of $\lfloor n/2 \rfloor$ indices $(m_i)_{i=1\ldots\lfloor n/2 \rfloor}$ is then uniformly chosen in $[0, \bar{m}]$ and another set of $\lceil n/2 \rceil$ indices $(m_i)_{i=\lceil n/2 \rceil \ldots n}$ is chosen in $[\bar{m}+1, m]$. Each column of $\mathbf{A}^*$ is then created as a Gaussian whose mean is m_i. Monte-Carlo simulations have been carried out by

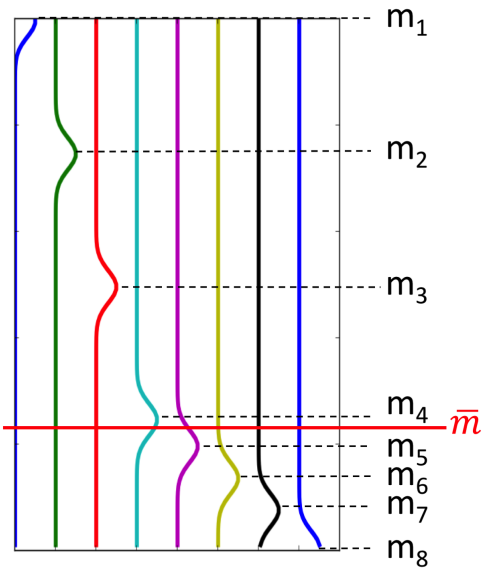


Figure V.5 – Exemple of $\mathbf{A}^*$ matrix with 8 columns : the four first columns have spaced means, while the last ones are more correlated

1. National Institute of Advanced Industrial Science and Technology (AIST), Spectral database for organic compounds : http://sdbs.db.aist.go.jp

randomly assigning the sources and the mixing matrix columns. The median over the results of the different experiments will be displayed.

D.2 Experiments

There are two main differences with the previous experiments of section C : i) the sources are sparse in the undecimated wavelet domain $\mathbf{\Phi_S}$, which is chosen as the starlet transform [Starck *et al.* 2007] in the following, and ii) the non-negativity of $\mathbf{S}$ and $\mathbf{A}$ is enforced. Fig. V.6 (left) displays the evolution of the mixing matrix criterion with varying block sizes with and without the non-negativity constraints. The algorithm was launched with 2 000 iterations.

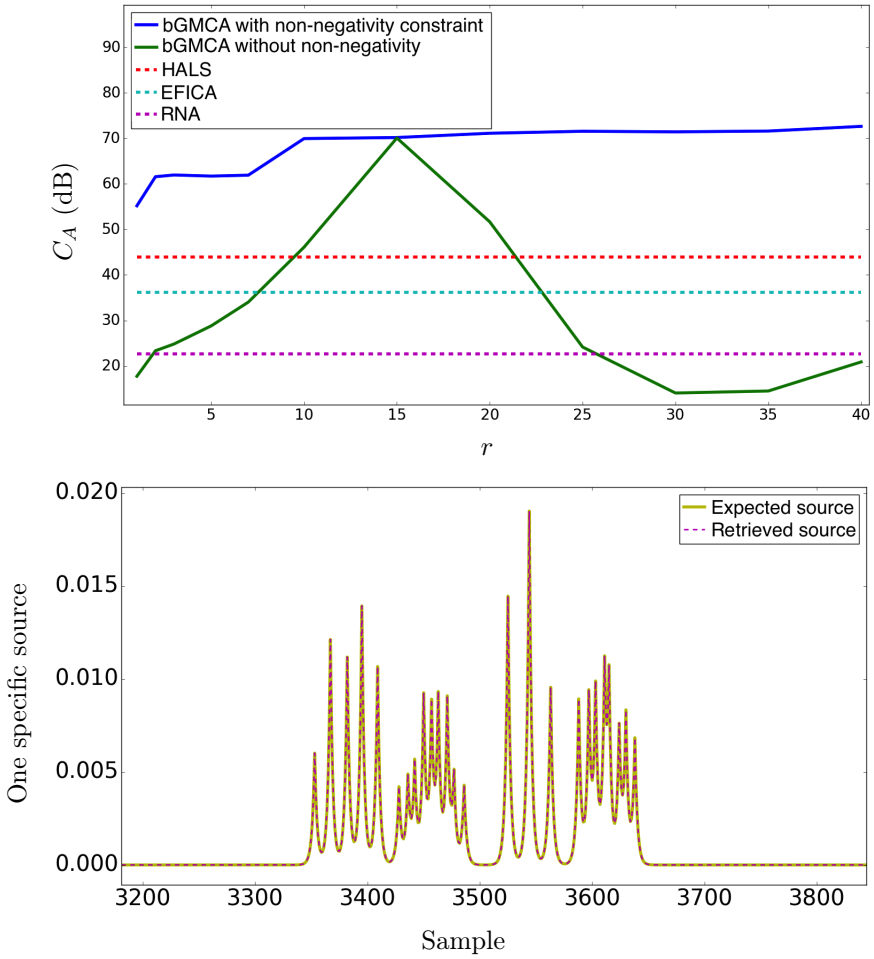


Figure V.6 – *Up :* mixing criterion on realistic sources, with and without a non-negativity constraint. *Down :* example of a retrieved source, which is almost perfectly superimposed on the true source, therefore showing the quality of the results.

These results show that non-negativity yields a huge improvement for all block sizes r, which is expected since the problem is more constrained. This is probably due to

the fact that all the small negative coefficients are set to 0, thus artificially allowing lower thresholds and therefore less artefacts. This is especially advantageous in the present context with very low noise [2] (the Signal to Noise Ratio - SNR - has a value of 120 dB) where the thresholds do not need to be high to remove noise.

Furthermore, the separation quality tends to be constant for $r \geq 10$. In this particular setting, non-negativity helps curing the failure of sparse BSS when large blocks are used. However, using smaller block sizes still allows reducing the computation cost while preserving the separation quality. The *bGMCA* with non-negativity also compares favorably with respect to other tested standard BSS methods, yielding better results for all values of r. A single original source is displayed in the right panel of Fig. V.6 after its convolution with a Laplacian. Its estimation using *bGMCA* with a non-negativity constraint is plotted in dashed line on the same graph, showing the high separation quality because of the nearly perfect overlap between the two curves. Both sources are drawn in the direct domain.

The robustness of the *bGMCA* algorithm with respect to additive Gaussian noise has further been tested. Fig. V.7 reports the evolution of the mixing matrix criterion for varying values of the signal-to-noise ratio. It can be observed that *bGMCA* yields the best performances for all values of SNR. Although it seems to particularly benefit from high SNR compared to HALS and EFICA, it still yields better results than the other algorithms for low SNR despite the small block size used ($r = 10$), which could have been particularly prone to error propagations.

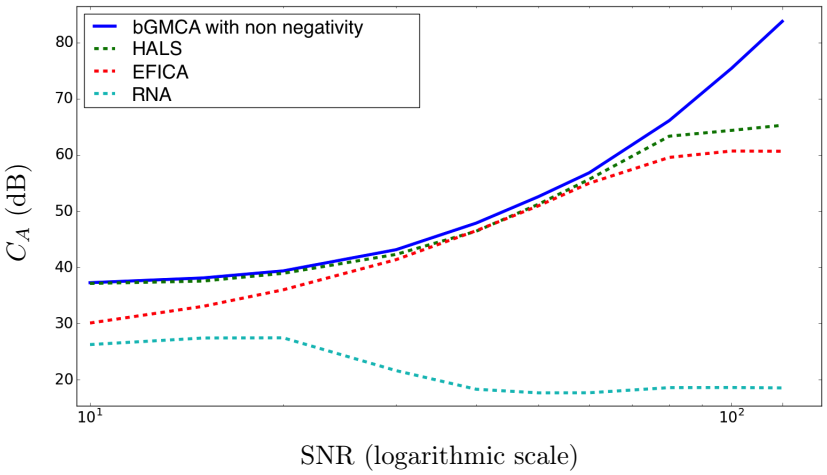


Figure V.7 – Mixing criterion on realistic sources, using a non-negative constraint with r = 10.

2. Depending on the instrumentation, high SNR values can be reached in such an experiment.

Conclusion

While being central in numerous applications, tackling sparse BSS problems when the number of sources is large is highly challenging. In this chapter, we described the block-GMCA algorithm, which is specifically tailored to solve sparse BSS in such a *large-scale* regime. In contrast to other state-of-the-art algorithms, *bGMCA* builds upon block-coordinate optimization with intermediate-size blocks. The mechanisms enabling *bGMCA* to have improved performances compared to its full-size block counterparts are explained both using exactly sparse simulated data and a mathematical modeling. While on such exactly sparse data *bGMCA* can lead to numerically perfect separations, comparisons have also been carried on simulated spectroscopic data, which demonstrates the reliability of the proposed algorithm in a realistic setting and its superior performances for high SNR. All the numerical comparisons conducted show that *bGMCA* performs at least as well as standard sparse BSS on mixtures of a high number of sources and most of the experiments even show dramatically enhanced separation performances. As a byproduct, the proposed block-based strategy yields a significant decrease of the computational cost of the separation process.

Chapitre VI

Tackling large-scale datasets : mini-batch optimization with aggregation on Riemannian manifold

---

The above chapter proposed a way to handle problems involving a high number of sources n, that is matrix factorization problems with a large inner dimension. In this chapter, we introduce a scalable sparse BSS algorithm enabling to *handle large-scale datasets* $\mathbf{X}$, which is of uttermost importance due to current ever growing data-sizes.
The proposed distributedGMCA (dGMCA) combines a robust projected alternating least-squares method with mini-batches, which enables to both handle large-scale datasets and to benefit from the *efficiency*, *reliability* and *versatility* of pALS (see Chapter IV). The originality lies in the use of a manifold-based aggregation of the different estimates of the mixing matrices. This approach is showed to maintain high performances compared to full-batch methods, which are in contrast not distributed. Remarkably, dGMCA can further outperform such algorithms when the sources have highly sparse distributions. Numerical experiments are carried out on synthetic data as well as realistic simulations of spectroscopic data.

This work has been started by the internship of Tobias Liaudat, which was proposed during the present PhD.

A Introduction

As explained in Chapter II, the challenges implied by huge datasets $\mathbf{X}$ are twofold : *time computational issues* (the dataset sizes make that it is currently impossible to perform BSS in a decent amount of time) and *memory limitation issues* (datasets are even too large to fit into memory). To give an order of the magnitude at stake, one could expect from astronomical devices such as the SKA data of up to $m \simeq 10^4$ observations and $t \simeq 10^9$ samples.
In this context, the GMCA algorithm [Bobin *et al.* 2007, Bobin *et al.* 2015], which was shown in Chapter IV to perform well for handling small-scale to middle [Bobin

et al. 2013] size datasets $\mathbf{X}$, is not anymore usable. Indeed, it needs at each iteration both an inversion of the factors $\mathbf{A}$ and $\mathbf{S}$ and multiplications with $\mathbf{X}$, which is particularly costly. This is furthermore an issue for the two-step approach of Chapter IV, that at least required a few iterations of GMCA as a warm-up stage for PALM.
In this chapter, we will focus on the largest dimension of $\mathbf{X}$: **we will propose a new sparse BSS method enabling to cope with datasets X comprehending a large number of samples** t. Before detailing the method, we will present the context by reviewing other related works. Then, we explain the challenges and our contribution.

A.1 Sparse matrix factorization for large-scale datasets X

The sparse BSS problem of Eq. (II.8) can be seen as a generic sparse matrix factorization problem, for which some works have been dedicated to large-scale datasets. A classical idea to tackle such an issue is to use only a submatrix (a set of columns) of $\mathbf{X}$ at each iteration, that is to use mini-batches. In contrast to successful small scale sparse BSS algorithms, most of the corresponding works do not use pALS.
Among them, one can distinguish the ones that :

— builds on stochastic gradient descent (SGD – [Bottou 2010]) : for instance, [Davis *et al.* 2016] extended the use of PALM to mini-batches, making it possible to tackle large datasets. Such a (potentially asynchronous) approach has also been extended to the case of hyperspectral imaging [Thouvenin *et al.* 2018]. In this work, the authors use the framework of [Cannelli *et al.* 2016], arguing a higher flexibility than in [Davis *et al.* 2016] in which the asynchronicity has a high impact on the allowable step sizes, counter-balancing its positive effects ;

— builds upon stochastic approximations. In the context of dictionary learning, [Mairal *et al.* 2009] proposed an online algorithm in which the dictionary is computed only minimizing an *upperbound* of the empirical cost. While the online setting is a special case with mini-batches size of $t_b = 1$, the algorithm is generalized to arbitrary t_b values. More recently, this algorithm has been extended in [Mensch *et al.* 2018] to tackle datasets that, in our context, would be huge both in the number of samples t and number of observations m. An extension has also been envisioned in the context of hyperspectral imaging with spectral variability [Thouvenin *et al.* 2016].

A.2 Challenges and contributions

A.2.1 Challenges

While the approaches of the previous subsection using mini-batches might sound appealing for large-scale sparse BSS, it must be highlighted that they have been elaborated for different kind of problems. In these, they generally do not aim at retrieving *physical* factors $\mathbf{A}^*$ and $\mathbf{S}^*$, which might not even exist. As such, algorithms

based on stochastic approximations as in [Mairal *et al.* 2009,Mensch *et al.* 2018] were initially targeting the dictionary learning problem and thus focus only on finding factors yielding good results for a given application (*e.g.* denoising...). Therefore, they generally do not provide accurate results in the context of sparse BSS (this claim will be further backed by the comparisons performed throughout this whole work). On the other hand, it has been shown in Chapter IV that gradient descent (GD) methods usually suffer from a low reliability in the sparse BSS context. Indeed, and in contrast to the pALS scheme of [Bobin *et al.* 2007], the use of GD methods makes automatic hyper-parameter choice much more difficult. Therefore, the mini-batches SGD methods are not expected to work well either (at least currently without requiring a warm-up stage based on pALS).
This leaves sparse BSS with very few satisfying options, as :

— Large-scale optimization algorithms using mini-batches yield bad physical factors $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$ in practice ;

— Usual sparse BSS algorithms working well on small-scale problems are not scalable.

Consequently, the solution we propose for large-scale BSS is to introduce mini-batches in the GMCA algorithm of [Bobin *et al.* 2007], which would enable to benefit from the best of the two worlds : re-using the mini-batch approach enabling scalable algorithms ; benefiting from the automatic parameter choice yielded by the pALS scheme and its reliability.

A.2.2 Other related works

The ALS algorithm has already been studied in the large-scale setting, but in the different application of recommendation systems. In [Zhou *et al.* 2008], the authors proposed a parallel ALS with weighted-λ-regularization to solve a low-rank matrix factorization problem. To do that, they split the $\mathbf{X}$ matrix, the factors $\mathbf{A}$ and $\mathbf{S}$ accordingly, and each node processes its own submatrices, in an approach reminiscent to mini-batches. The method was extended by [Teflioudi *et al.* 2012] to a distributed (shared nothing) setting. The works of [Hastie *et al.* 2015, Kampffmeyer 2015] also tackle a similar issue. However, several differences between such a problem and ours must be highlighted as : i) The application is different from ours : in particular, as $\mathbf{X}$ is sparse (due to the high number of missing entries), it is much easier to work on smaller submatrices and the mini-batches are much more naturally chosen ; ii) As a consequence, no aggregation (see below) is used since during each iteration each matrix $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$ (or more specifically, a submatrix of $\hat{\mathbf{A}}$ and $\hat{\mathbf{S}}$) is updated once ; iii) The cost function is different from ours, as there is no ℓ_1 sparsity promoting term.
As such, while the main idea of using mini-batches within an ALS scheme will be re-used in this Chapter, the similarities with these works are limited.

A.2.3 Contributions

In this Chapter, we propose to extend sparse BSS to the large-scale setting, enabling to tackle datasets $\mathbf{X}$ with a huge number of columns t. Our approach consists in introducing a mini-batch (stochastic) version of [Bobin *et al.* 2007] coined distributedGMCA (dGMCA). The algorithm enables to tackle datasets that would not even fit into the memory of one computer. The approach is furthermore robustified through the introduction of an *aggregation* step based on a robust weighted mean *on a Riemannian manifold*, enabling to aggregate several estimators of the mixing matrix $\mathbf{A}$. Beyond enabling scalability, the introduction of such a step is empirically shown to improve in some experiments the results of dGMCA over the full batch version of the algorithm. Lastly, realistic experiments are carried out to demonstrate the quality of the approach, and relationship to SGD methods in machine learning are highlighted to explain the results.

B Distributed sparse Alternating Least-Squares

B.1 Distributing the GMCA algorithm

B.1.1 Naive approach

A simple approach to deal with large-scale data would be to split the large dataset $\mathbf{X}$ into B submatrices. As such, the t columns would be split into B disjoint submatrices, each of them having $t_b = t/B$ columns (for the moment, t is assumed to be a multiple of B). Each of the corresponding submatrix is denoted as $\mathbf{X}^{J_b}, b \in [1, B]$ and the corresponding indices of the columns as $J_b, b \in [0, B]$ ($\#J_b = t_b$).
A naive approach would then be to work independently on each submatrix $\mathbf{X}^{J_b}, b \in [1, B]$. As such, instead of looking for an (approximate) minimizer of Eq. (II.8) using GMCA, we would rather tackle small-scale subproblems :

$$\underset{\mathbf{A}[J_b]\in\mathbb{R}^{m\times n},\mathbf{S}^{J_b}\in\mathbb{R}^{n\times t_b}}{\operatorname{argmin}} \frac{1}{2}\left\|\mathbf{X}^{J_b} - \mathbf{A}[J_b]\mathbf{S}^{J_b}\right\|_F^2 + \left\|\mathbf{R}_{\mathbf{S}^{J_b}} \odot \mathbf{S}^{J_b}\right\|_1 + \iota_{\{\forall i\in[1,n], \|\mathbf{A}[J_b]^i\|_{\ell_2}^2=1\}}(\mathbf{A}[J_b]), \tag{VI.1}$$

where, for the sake of simplicity, the $\mathbf{\Phi}_\mathbf{S}$ has been omitted and taken equal to identity. The final estimate of $\mathbf{S}^*$ is straightforwardly obtained through the concatenation of the columns of the different final estimates $\hat{\mathbf{S}}^{J_b}, b \in [1, B]$.
The question is however more intricate concerning the mixing matrix, as each submatrix $\mathbf{X}^{J_b}$ yields a different estimate $\mathbf{A}[J_b]$ of the *same* full matrix $\mathbf{A}^*$. At this point, we should make clear that the $\mathbf{A}[J_b]$ are different despite the assumed stationarity of the signals, as they rely on a specific *small-case* realization $\mathbf{X}^{J_b}$. As such, some will be better estimates of the true underlying $\mathbf{A}^*$ than others. To get the final estimate $\hat{\mathbf{A}}$, we thus need to resort to an *aggregation* step that will merge the information yielded by the various $\mathbf{A}[J_b]$ to yield a final $\hat{\mathbf{A}}$:

$$\hat{\mathbf{A}} = \texttt{AGGREGATE}(\mathbf{A}[J_1], \mathbf{A}[J_2], ..., \mathbf{A}[J_B]), \tag{VI.2}$$

where the `AGGREGATE` function is used as a generic term for the aggregation. As a simple example of such a function, one could for instance use the coefficient-wise Euclidean mean of the different $\mathbf{A}[J_b], b \in [1, B]$. This option is however simplistic and can be enhanced (*cf.* Sec. B.2). The naive version of dGMCA is summarized in Algorithm 5.

Algorithm 5 Naive dGMCA

procedure NAIVE DGMCA($\mathbf{X}$,$\hat{\mathbf{A}}^{(0)}$)
  Choose $J_1, J_2, \cdots, J_B$ as a partition of $[1, t]$
  for $b = 1, \cdots, B$ **do**
    $\mathbf{A}[J_b], \hat{\mathbf{S}}^{J_b} = \text{GMCA}(\mathbf{X}^{J_b}, \hat{\mathbf{A}}^{(0)})$
  end for
  $\hat{\mathbf{A}} = \texttt{AGGREGATE}(\mathbf{A}[J_1], \mathbf{A}[J_2], ..., \mathbf{A}[J_B])$
  return $\hat{\mathbf{A}}, \hat{\mathbf{S}}$
end procedure

Nevertheless, one of the main flaws of the naive dGMCA is that the information yielded by a given submatrix $\mathbf{X}^{J_{b_1}}$ is not shared at all to enhance the mixing matrix $\mathbf{A}[J_b]$ found from another submatrix $\mathbf{X}^{J_b}, b \neq b_1$. Said differently, processing each submatrix $\mathbf{X}^{J_b}$ fully independently makes that much less information than using the whole matrix $\mathbf{X}$ is used to estimate each of the $\mathbf{A}[J_b]$. It could be argued that this is not a real issue, as we are not directly interested by each of the $\mathbf{A}[J_b], b \in [1, B]$, but rather by the final $\hat{\mathbf{A}}$ obtained after the aggregation step, which goal is precisely to merge the information between the subproblems. Unfortunately, this is not fully accurate and practical experiments further tend to hinder such arguments. Indeed, performing *independent* GMCA makes that none of the estimated $\mathbf{A}[J_b]$ is descent (as only too small submatrices of the original data $\mathbf{X}$ are used). As such, the aggregation step – said differently, information sharing among subproblems – is applied "too late" on only badly estimated $\mathbf{A}[J_b]$: since $\hat{\mathbf{A}}$ is estimated from mainly bad $\mathbf{A}[J_b]$, it is also bad.

B.1.2 dGMCA algorithm

To alleviate the previous difficulty induced by working independently on subproblems of the form Eq. (VI.1), we propose to enable the estimation of each $\mathbf{A}[J_{b_1}]$ to share information with the other $\mathbf{A}[J_b], b \neq b_1$ *during the iterations of GMCA*. To do that, we propose to apply the aggregation step during each iteration of GMCA (following the machine learning terminology, we can now speak of epochs of GMCA), in contrast to the naive approach in which it was performed only after the end of GMCA.

Using such an approach, the indices J_b may now change during GMCA iterations, as well as the corresponding submatrices $\mathbf{X}^{J_b}$. As such, we will rather write for each epoch $J_b(l)$. By analogy with the works of [Mairal *et al.* 2009, Davis *et al.* 2016,

Mensch *et al.* 2018], we will further call $\mathbf{X}^{J_b(l)}$ a mini-batch of $\mathbf{X}$ [1].
A last remaining question before detailing the algorithm is how to choose the regularization parameters $\mathbf{R_S}$ within dGMCA, to keep the advantage of the adaptive parameter tuning of GMCA [2]. Indeed, we will here aim at extending the most recent parameter choice of GMCA [3], based on using an increasing percentile of the whole distribution of the currently estimated $\hat{\mathbf{S}}^{(l)}$, which is difficult to distribute for large t values. As such, we propose to rather use an exponential decay of the parameters of the form :

$$\mathbf{R}_{i_{\mathbf{S}_i}}^{i(l)} = \kappa\sigma_i + \left(\left\|\hat{\mathbf{S}}_i^{(l)}\right\|_\infty - \kappa\sigma_i\right)\exp(-l\alpha_i), \tag{VI.3}$$

where $\mathbf{R}_{i_{\mathbf{S}_i}}^{i(l)}$ is the estimated regularization parameter at epoch l for source $\mathbf{S}_i$, σ_i is an estimation of the noise back-projected on source $\mathbf{S}_i$, κ is chosen according to the fixed point Gaussian noise removal argument (here $\kappa = 3$), $\left\|\hat{\mathbf{S}}_i^{(l)}\right\|_\infty$ is the maximum absolute value of $\hat{\mathbf{S}}_i^{(l)}$ and α_i is a parameter controlling the exponential decay decrease [4]. An interesting property of the parameter choice of (VI.3) is that it is highly distributable : $\left\|\hat{\mathbf{S}}_i^{(l)}\right\|_\infty$ is the maximum over each mini-batch ($\left\|\hat{\mathbf{S}}_i^{(l)}\right\|_\infty = \max_{b\in[1,B]}\left\|\hat{\mathbf{S}}_i^{J_b(l)}\right\|_\infty$), and σ_i can be chosen as the median of its estimations over the mini-batches : $\sigma_i = \text{median}_{b\in[1,b]}\,\sigma_i[J_b(l)]$. Each estimation $\sigma_i[J_b(l)]$ is in turn performed according to the usual method used in GMCA, based on the MAD operator (*cf.* Chapter II [Bobin *et al.* 2007]).

B.1.3 Summary of the algorithm

B.2 Manifold-based mixing matrix aggregation

We now detail the aggregation step used to build the estimate $\hat{\mathbf{A}}^{(l)}$ from the various $\mathbf{A}^{(l)}[J_b], b \in [1, B]$. To lighten the notations, we will drop in this subsection the l index, but the aggregation is of course performed at each iteration. We propose to define $\hat{\mathbf{A}}$ as the barycenter of the different estimates according to some ϕ as follows :

1. Note that in these works, the aggregation step enabling to share the information between mini-batches is however implicit and made simpler due to the fact that the algorithms use only *local* updates such as gradient steps instead of least-square solutions.

2. While in the naive dGMCA approach the answer to such a question was straightforward, it is important to emphasize that choosing the regularization parameters using small $\mathbf{X}^{J_b}$ could lead to estimate them badly.

3. The reader might wonder why we are not implementing here the regularization parameter choice used in Chapter IV, which was fully based on the MAD. This is due to the fact that while this would is easier to distribute, the use of the percentile decrease generally provides better results in GMCA. In contrast, the experiments we perform in the Appendix E show that the percentile does not seem to work well within PALM, thus our choice to use the MAD.

4. The α_i parameter intrinsically depends on the sparsity level of $\mathbf{S}_i$. As such, it can be estimated by fitting a generalized Gaussian distribution to the current estimation $\hat{\mathbf{S}}_i^{(l)}$ and using a maximum likelihood estimator. However, we experimentally found out that the final result of dGMCA is quite robust to the estimation of α_i, enabling to rather use a value fixed beforehand.

Algorithm 6 dGMCA

procedure DGMCA($\mathbf{X}$,$\hat{\mathbf{A}}^{(0)}$)
  for $l = 1, \cdots, L$ **do**
    Choose $J_1, J_2, \cdots, J_B$ as a partition of $[1, t]$
    for $b = 1, \cdots, B$ **do**
      $\hat{\mathbf{S}}^{J_b(l)} = \mathcal{S}_{\mathbf{R_S}^{(l)}}(\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{X}^{J_b(l)})$ ▷ Use Eq. (VI.3) for $\mathbf{R_S}$ choice
      $\mathbf{A}[J_b(l)] = \Pi_{\|.\|_2=1}(\mathbf{X}^{J_b(l)}\hat{\mathbf{S}}^{J_b(l)\dagger})$
    end for
    $\hat{\mathbf{A}}^{(l)} = \texttt{AGGREGATE}(\mathbf{A}[J_1(l)], \mathbf{A}[J_2(l)], ..., \mathbf{A}[J_B(l)])$
  end for
  return $\hat{\mathbf{A}}^{(L)}, \hat{\mathbf{S}}^{(L)}$
end procedure

$$\hat{\mathbf{A}} = \underset{\mathbf{A}\in\mathbb{R}^{m\times n}}{\operatorname{argmin}} \sum_{b=1}^{B} \omega_b \phi\left(\mathbf{A}, \mathbf{A}[J_b]\right), \tag{VI.4}$$

where the barycentric weights are positive and sum to one : $\forall b \in [1, b]; \omega_b \geq 0$ and $\sum_{b=1}^{B} \omega_b = 1$.
A straightforward example amounts to choose ϕ as the standard Euclidean distance applied on each column : $\phi\left(\mathbf{A}, \mathbf{A}[J_b]\right) = \sum_{j=1}^{n} \|\mathbf{A}^j - \mathbf{A}^j[J_b]\|_{\ell_2}^2$. This choice will eventually define the aggregated estimator as a weighted sum of the different mini-batch estimators :

$$\hat{\mathbf{A}} = \sum_{b=1}^{B} \omega_b \mathbf{A}[J_b]. \tag{VI.5}$$

However, in the context of BSS, the mixing matrix is assumed to belong to the Oblique ensemble. In the limit of small angular distances between the different estimators, the Euclidean metric is likely an aggregated estimate that fulfills the Oblique constraint, which can be further constrained by projecting $\hat{\mathbf{A}}$ onto $\mathcal{O}_b$. In the general case, this is unlikely to hold true, especially when considering small size mini-batches, that can lead to more larger angular deviations. This implies that the Oblique constraint has to be preserved in the aggregation step.

Fréchet mean on the hypersphere

Recall that the Oblique constraint implies that each column of the mixing matrix $\mathbf{A}$ belongs to the m-dimensional hypersphere $\mathcal{S}_m$ or m-sphere, which is a Riemannian manifold. We assume the reader to be familiar with some basic notions about optimization on Riemannian manifolds. If it is not the case, we refer to [Absil *et al.* 2009] or to the Appendix F for a summary of useful elements for this work.

A natural approach to take into account the Oblique constraint consists in building an aggregated estimator by defining each of the estimate $\hat{\mathbf{A}}$ column as being

the Fréchet mean of the corresponding columns in the estimators $\mathbf{A}[J_b]$:

$$\forall j \in [1,n], \quad \hat{\mathbf{A}}^j = \underset{\mathbf{a}\in\mathbb{R}^m}{\operatorname{argmin}} \sum_{b\in[1,B]} \omega_b \, d^\beta \left(\mathbf{a}, \mathbf{A}[J_b]^j\right), \tag{VI.6}$$

where $1 \leq \beta < +\infty$. The case $\beta = 2$ corresponds to the ℓ_2 norm along the geodesics of the manifold $\mathcal{S}_m$.
Following [Afsari 2011], it is possible to find a local critical point of this problem using an iterative gradient descent algorithm on the m-sphere presented in Algorithm 7. The parameter ρ is the step size[5]. The term $\nabla\mathcal{J}^\beta(\hat{\mathbf{A}}^{j^{(l_f)}})$, where l_f is the iteration number in the Fréchet mean algorithm, is the gradient of the mean cost function $\mathcal{J}^\beta(\hat{\mathbf{A}}^{j^{(l_f)}}) = \sum_{b=1}^B \omega_b \, d^\beta \left(\hat{\mathbf{A}}^{j^{(l_f)}}, \mathbf{A}^j[J_b]\right)$, which takes the following expression for $\beta \geq 1$:

$$\nabla\mathcal{J}^\beta(\hat{\mathbf{A}}^{j^{(l_f)}}) = -\sum_{b=1}^{B} \omega_b d^{\beta-2} \left(\hat{\mathbf{A}}^{j^{(l_f)}}, \mathbf{A}^j[J_b]\right) \log_{\hat{\mathbf{A}}^{j^{(l_f)}}} \left(\mathbf{A}^j[J_b]\right). \tag{VI.7}$$

Algorithm 7 Fréchet mean

procedure Fréchet mean on the Oblique ensemble $\mathcal{O}_b(m)$
  Correct for permutations
  for $j = 1, \cdots, n$ **do** ▷ Loop over all the sources
    while Convergence is not reached **do**
      $\nabla\mathcal{J}^\beta(\hat{\mathbf{A}}^{j^{(l_f)}}) = -\sum_{b=1}^B \omega_b \log_{\hat{\mathbf{A}}^{j^{(l_f)}}} \left(\mathbf{A}^j[J_b]\right)$ ▷ Gradient
      $\hat{\mathbf{A}}^{j^{(l_f+1)}} = \exp_{\hat{\mathbf{A}}^{j^{(l_f)}}} \left(-\rho\nabla\mathcal{J}^\beta(\hat{\mathbf{A}}^{j^{(l_f)}})\right)$
      $l_f \leftarrow l_f + 1$
    end while
  end for
  Return $\hat{\mathbf{A}}^{(L_f)}$
end procedure

Robust Fréchet mean on the hypersphere

The use of small mini-batches makes the separation process more sensitive to several factors. In practice, this will tend to generate outliers in the estimated $\mathbf{A}[J_b]$ for some mini-batches, which is discussed in more details in Section B.3 and supported by several numerical experiments in Section C. Unfortunately, the Fréchet mean is not robust to such outliers [Fletcher *et al.* 2008, Arnaudon *et al.* 2013]. In this case, a natural choice would be to choose $\beta = 1$, which corresponds to the usual ℓ_1 norm. However, the ℓ_1 norm is not differentiable about 0 and the gradient of the cost function depends on the inverse of the distance d^1 as highlighted in Equation (VI.7), which makes it quite unstable in practice. To alleviate this issue,

5. It can possibly vary during the optimization process. It will be kept fixed in this Chapter.

we propose to build a differentiable approximation of d^1 based on Nesterov's smoothing technique [Nesterov 2005]. Such a smooth approximation of d^1 can be built as follows [Becker *et al.* 2011] for $\mathbf{a}, \mathbf{b} \in \mathbb{R}^m$:

$$d^1_\nu(\mathbf{a}, \mathbf{b}) = \underset{\|\mathbf{u}\|_\infty \leq 1}{\operatorname{argmax}} \; < \mathbf{a} - \mathbf{b}, \mathbf{u} > - \frac{\nu}{2} \|\mathbf{u}\|^2_{\ell_2}. \tag{VI.8}$$

This approximated distance is differentiable and its gradient is ν-Lipschitz. This entails that the gradient of the cost function takes the form :

$$\nabla \mathcal{J}^\beta(\hat{\mathbf{A}}^{j^{(l_f)}}) = - \sum_{b=1}^{B} \omega_p \nabla d^1_\nu \left(\hat{\mathbf{A}}^{j^{(l_f)}}, \mathbf{A}^j[J_b] \right) \log_{\hat{\mathbf{A}}^{j^{(l_f)}}} \left(\mathbf{A}^j[J_b] \right), \tag{VI.9}$$

where the gradient of d^1_ν is given by [Becker *et al.* 2011] for all $i \in [1, m]$:

$$\nabla d^1_\nu(\hat{\mathbf{A}}^{j^{(l_f)}})_i = \begin{cases} \nu^{-1} \, \hat{\mathbf{A}}_i^{j^{(l_f)}}, & \text{if } |\hat{\mathbf{A}}_i^{j^{(l_f)}}| < \nu, \\ \operatorname{sign}(\hat{\mathbf{A}}_i^{j^{(l_f)}}), & \text{otherwise.} \end{cases} \tag{VI.10}$$

Algorithm 8 robust Fréchet mean

procedure FRÉCHET MEAN ON THE OBLIQUE ENSEMBLE $\mathcal{O}_b(m)$
  Correct for permutations
  for $j = 1, \cdots, n$ **do** ▷ Loop over all the sources
    while Convergence is not reached **do**
      $\nabla \mathcal{J}^\beta(\hat{\mathbf{A}}^{j^{(l_f)}}) = - \sum_{b \in [1,B]} \omega_b^j \nabla d^1_\nu \left(\hat{\mathbf{A}}^{j^{(l_f)}}, \mathbf{A}^j[J_b] \right) \log_{\hat{\mathbf{A}}^{j^{(l_f)}}} \left(\mathbf{A}^j[J_b] \right)$
      $\hat{\mathbf{A}}^{j^{(l_f+1)}} = \exp_{\hat{\mathbf{A}}^{j^{(l_f)}}} \left(-\rho \nabla \mathcal{J}^\beta(\hat{\mathbf{A}}^{j^{(l_f)}}) \right)$
      $l_f \leftarrow l_f + 1$
    end while
  end for
  Return $\hat{\mathbf{A}}^{(L_f)}$
end procedure

From Fréchet mean to barycenter

In practical settings, the data are assumed to be contaminated with Gaussian noise. As pointed out in the previous paragraph, the use of small size mini-batches might lead to very diverse estimates of the mixing matrix. These estimates could yield sources with largely different signal-to-noise ratio. A first solution could be to choose weights $\{\omega_b\}_{b \in [1,B]}$ that penalize mini-batches with smaller SNR. However this would not take into account the actual noise that would contaminate the estimated sources. Therefore, we propose to choose these weights as being a function of the SNR of the estimated sources. Assuming that the data noise covariance matrix

is denoted $\mathbf{\Sigma_N}$, the weights are defined as :

$$\forall b \in [1, B]; \quad \omega_b^j = \frac{\left(\mathbf{W}[J_b]^j \mathbf{\Sigma_N} \mathbf{W}[J_b]^{jT}\right)^{-1}}{\sum_{p=1}^{B} \left(\mathbf{W}[J_b]^j \mathbf{\Sigma_N} \mathbf{W}[J_b]^{jT}\right)^{-1}} \tag{VI.11}$$

where $\mathbf{W}[J_b]^j = [\mathbf{A}[J_b]^+]^j$. In the numerical experiments, this weighting strategy will be applied to both the Fréchet mean and its robust version.

B.3 Another point of view about dGMCA : connections with stochastic gradient descent

To give a better insight of dGMCA principle, we aim in this subsection to highlight some connections with works on SGD.

From GD to ALS

Beyond matrix factorization problems, GD is a very popular approach in machine learning. In the scope of BSS and forgetting the oblique constraint, estimating $\mathbf{A}^*$ using GD would yield the following update rule at epoch l for estimate $\hat{\mathbf{S}}$ of the sources :

$$\begin{aligned} \hat{\mathbf{A}}^{(l+1)} &= \hat{\mathbf{A}}^{(l)} + \eta^{(l)} \mathbf{\Delta} && \text{(VI.12)} \\ &= \hat{\mathbf{A}}^{(l)} + \eta^{(l)} \left(\mathbf{X} - \hat{\mathbf{A}}^{(l)} \hat{\mathbf{S}}\right) \hat{\mathbf{S}}^T && \text{(VI.13)} \end{aligned}$$

where $\mathbf{\Delta}$ is the gradient of the data-fidelity term of Eq. (II.8) with respect to $\hat{\mathbf{S}}$ and $\eta^{(l)}$ the learning rate (*i.e.* the gradient step size when using the machine learning terminology).
On the other hand, it is possible to incorporate second order information, which amounts to write a Newton update as follows :

$$\begin{aligned} \hat{\mathbf{A}}^{(l+1)} &= \hat{\mathbf{A}}^{(l)} + \eta^{(l)} \left(\mathbf{X} - \hat{\mathbf{A}}^{(l)} \hat{\mathbf{S}}\right) \hat{\mathbf{S}}^T \mathbf{H}^{-1} && \text{(VI.14)} \\ &= \hat{\mathbf{A}}^{(l)} + \eta^{(l)} \left(\mathbf{X} - \hat{\mathbf{A}}^{(l)} \hat{\mathbf{S}}\right) \hat{\mathbf{S}}^T \left(\hat{\mathbf{S}} \hat{\mathbf{S}}^T\right)^{-1} && \text{(VI.15)} \end{aligned}$$

where $\mathbf{H} = \left(\hat{\mathbf{S}} \hat{\mathbf{S}}^T\right)^{-1}$ is the Hessian of the data fidelity term with respect to $\mathbf{A}$. Assuming the sources to be decorrelated yields a diagonal Hessian, which is equivalent to a single iteration of the standard GD algorithm up to a scaling factor and with $\eta^{(l)} = 1$. More generally, fixing $\eta^{(l)} = 1$ entails : $\hat{\mathbf{A}}^{(l+1)} = \mathbf{X} \hat{\mathbf{S}}^T \left(\hat{\mathbf{S}} \hat{\mathbf{S}}^T\right)^{-1}$. In this setting, ALS can be regarded as a special type of GD when curvature or second-order information is used.

From SGD to mini-batch ALS

However, when it turns to mini-batch optimization the connections become slightly delicate : whether in Newton descent algorithm or in ALS, the Hessian matrix $\hat{\mathbf{S}}^{J_b}\hat{\mathbf{S}}^{J_bT}$ becomes mini-batch dependent. Small size mini-batches entail extra errors on the estimated Hessian, which eventually leads to more stochasticity on the estimated mixing matrices. Therefore, *ALS leads to more stochasticity on the estimates than standard SGD*, which can be detrimental to the optimization procedure when the mini-batch size becomes very small [6].

From averaged SGD to aggregated stochastic ALS

The parallel between mini-batch ALS and SGD can be prolonged to give an insight concerning the aggregation step. More precisely, it is well known that the stochasticity of SGD can lead to an optimization path with more fluctuations and decreased rates of convergence in comparison to full-batch GD updates. To bypass this issue, a classical modification of SGD is to *average* the iterates of the estimates (Polyak-Rupper averaging [Ruppert 1988, Polyak & Juditsky 1992]), improving the convergence rates and reducing the impact of noise. In such an averaging process, the estimate of a variable $\hat{\mathbf{A}}^{(l)}$ at iteration l is chosen as the mean over the previous iterations : $\hat{\mathbf{A}}^{(l)} = \frac{1}{l}\sum_{i=1}^{l}\hat{\mathbf{A}}^{(i)}$, which resembles the *Euclidean* aggregation step discussed in Section B.2. Recent works [Tripuraneni *et al.* 2018] have extended the Polyak-Rupper averaging to a Riemannian setting, creating links with the aggregation step we use in dGMCA : instead of using a mere average of the previous iterates, the averaging is performed on a manifold. This parallel must however be tempered as : i) we apply a *weighted* Polyak-Rupper averaging : in particular, the weights are zero for the estimates found before the current epoch l [7] ; ii) we do not aggregate after each mini-batch, but rather once per epoch, which helps distributing the algorithm ; iii) [Tripuraneni *et al.* 2018] explores SGD, and we deal with pALS, implying the differences evoked above. However, we highlight that as pALS might increase the discrepancy between the different estimations compared to GD, such an aggregation might be particularly relevant to smooth the optimization path.
As a side remark, while more different from the dGMCA aggregation, we would like to point out the works of [Sato *et al.* 2017, Zhang *et al.* 2016], which are also variance reduced Riemannian optimization methods.

C Numerical experiments

In the following experiments, the performances of the dGMCA algorithm are evaluated in various experimental settings, on both simulated and realistic sources.

6. On the other hand, we shall see later that such stochasticity can also have benefits to some extent.

7. Note that using specific weights has been proved in some different settings to lead to interesting results through creating an implicit regularization [Neu & Rosasco 2018].

C.1 Experiments on simulated data

Comparison set-up

In this subsection, we first make use of synthetic random data, which allows to perform Monte-Carlo simulations to assess the robustness of the different methods and to study the performances of dGMCA when varying the experimental parameters. More precisely, we will look at the influence of different numbers of sources n and observations m, as well as the one of mixing matrices $\mathbf{A}^*$ with different condition numbers and sources with various sparsity levels p. To that end, the data are synthesized as follows :

— The sources $\mathbf{S}_j^*, j \in [1, n]$ have entries which are distributed independently and identically according to a Generalized Gaussian distribution with parameter $0 < \alpha \leq 1$.

— The mixing matrix $\mathbf{A}^*$ is picked at random from a Gaussian distribution, and further processed to have columns with unit ℓ_2 norm and a pre-defined condition number.

Unless stated differently, each single experimental result will be given as the mean over 10 Monte-Carlo simulations with different mixing matrices, sources and noise realizations.

Beyond the proposed two versions of the dGMCA algorithm, namely the one using as aggregation the Fréchet mean and the one using the robust Fréchet mean, comparisons will be carried out with :

— **GMCA** : This algorithm is used as a baseline to compare its parallelized dGMCA counterpart.

— **Online dictionary learning** : see Section A.1 and [Mairal *et al.* 2009]. This algorithm is a classical one for solving large-scale sparse matrix factorization problems. [8]

To assess the separation quality, we will use the mixing matrix criterion C_A.

Studying the impact of the number of observations m

In this paragraph, we evaluate the performances of the dGMCA with respect to the number of observations m. The role played by this parameter is twofold. First, for a fixed signal-to-noise ratio, the source SNR roughly evolves as the ratio m/n and therefore improves with m. Second, the manifold-based aggregation may behave quite differently when the dimensionality of the ambient space changes : for some fixed entry-wise error on the estimated mixing matrices, the angular error (*i.e.* that is proportional to the distance used in the tangent space of the hypersphere) decreases when the number of observations increases.

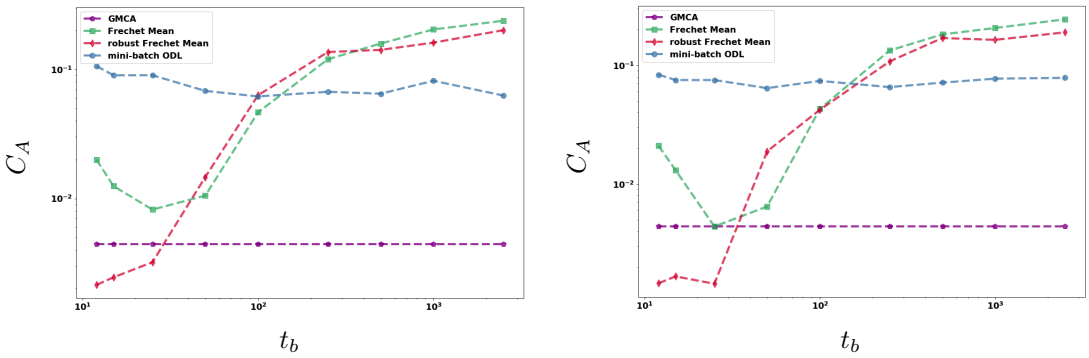
8. When using ODL, the regularization parameter choice has to be performed by the user. In the simulations we propose, we tried several values and kept only the best results.

Figure VI.1 shows the evolution of the mixing matrix criterion as a function of the mini-batch size for $m = 5$ (which corresponds to the determined case, as there are $n = 5$ sources) and $m = 20$. The sparsity level is fixed to $p = 0.1$ and the condition number of the mixing matrix $\mathbf{A}^*$ is equal to 3. The SNR is fixed to 40 dB.
At first glance, it is interesting to observe that the dGMCA - Fréchet mean algorithm provides in both experiments decent results when the mini-batch size t_b is lower than 80 : here, using small mini-batches enables to alleviate the computational burden while not deteriorating too much the results over GMCA. Being more specific, dGMCA slightly benefits from a larger number of observations m when the mini-batch size is small. This is very likely the consequence of lower errors as measured by the angular distance. Furthermore, for small t_b some mini-batches are likely to have a too small number of statistics to be well estimated when m is small. To that regards, the results of the robust Fréchet mean algorithm are particularly informative : while for the smallest mini-batches and small m the results yielded by the mere Fréchet mean were deteriorated, it is not anymore the case. This probably means that the deterioration previously observed with small t_b was likely to stem from a *small number* of badly estimated mini-batches, that is outliers. The use of a robust aggregation lowers their impact, enabling an improvement of about 1 order of magnitude for very small mini-batch sizes ($t_b < 20$). Unexpectedly, equipped with this aggregation procedure, the dGMCA algorithm further allows to improve the separation process with a gain of up to about a factor 2 with respect to the standard GMCA algorithm. We will propose an explanation of such a phenomenon in Section C.3.
Lastly, the reader might wonder why dGMCA does not seem to reach the accuracy of GMCA for the largest mini-batch sizes. To explain this, let us recall that the thresholding strategy differs from GMCA to dGMCA : in the case of GMCA, it adapts to the actual distribution of the estimated sources using a percentile. Since such an automatic regularization parameter choice was not applicable in dGMCA, we rather chose a fixed deterministic strategy which is likely to be less effective. When the mini-batch size t_b becomes a relatively large fraction of the full number of samples t, the strategy used on GMCA should be also implemented in dGMCA (note that in this setting, it is assumed that the computational burden is not a main issue).

Condition number

The ill-conditioning of the mixing matrix $\mathbf{A}^*$, measured by its condition number, plays an important role in the difficulty of a given BSS problem. Mixing matrices with large condition numbers will lead to two major challenges : i) an increased noise level in the source domain [9], and ii) mixtures that are more colinear.
In these experiments, the noise level is fixed to 40 dB and the sparsity level is $p = 0.1$. Figure VI.2 shows the evolution of C_A as a function of the mini-batch

9. To that regards, it is expected to see similarities between an increased condition number and a small number of observations.



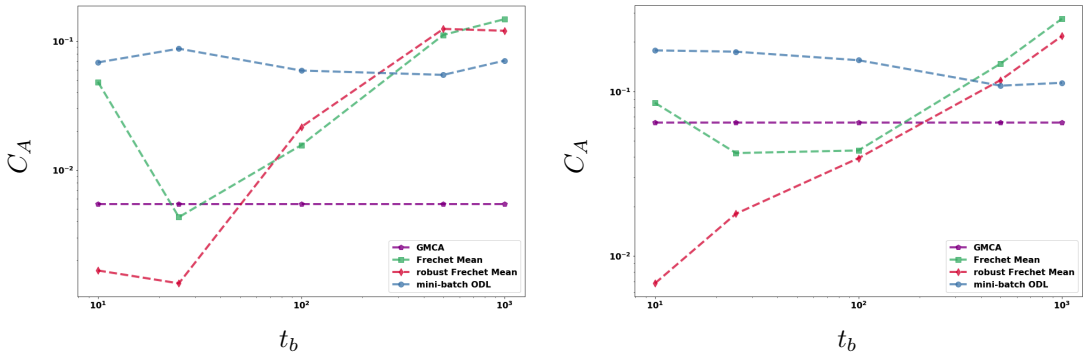
(a) $m = 5$ (b) $m = 20$

Figure VI.1 – Evolution of the mixing matrix criterion with respect to the mini-batch size for *left* : $m = 5$ and *right* : $m = 20$. The number of sources is fixed to $n = 5$.

size t_b for two values of the condition numbers : left panel 2.5 and right panel 7. Alternatively, Figure VI.3 shows the same kind of results but as a function of the condition number for $t_b = 10$ and $t_b = 100$.
A general comment is that as expected the separation quality of all the methods decrease when the condition number increases. Similarly to the tests performed in the previous section, the dGMCA algorithm has better results for relatively small mini-batch sizes (but when the Fréchet mean is used, it eventually deteriorates for $t_b < 25$, *cf.* Fig. VI.2). The use of small-batches along with the robust Fréchet mean leads to an improvement for $t_b < 25$, which becomes more significant when the condition number increases and leads to a gain of about one order of magnitude. This was to be expected, as higher condition number lead to more diversity and less stability among the mini-batches. Similarly, when the mini-batch size decreases, the discrepancy between the two methods increases.
Furthermore, while the GMCA results are the best ones when $\mathbf{A}^*$ is close to orthogonality, the robust dGMCA tends to deteriorate slower as a function of the condition number (*cf.* Fig VI.3) : that is, here again the aggregation is able to reject the outliers that impede GMCA use (such outliers probably comes from a specific bad noise realization in some mini-batches, which is amplified by high condition numbers). As such, the gain of robust dGMCA over GMCA is very significant for conditions number of about 5. Beyond 10, the curves tends to merge to give bad results, as most of the mini-batches are badly estimated.

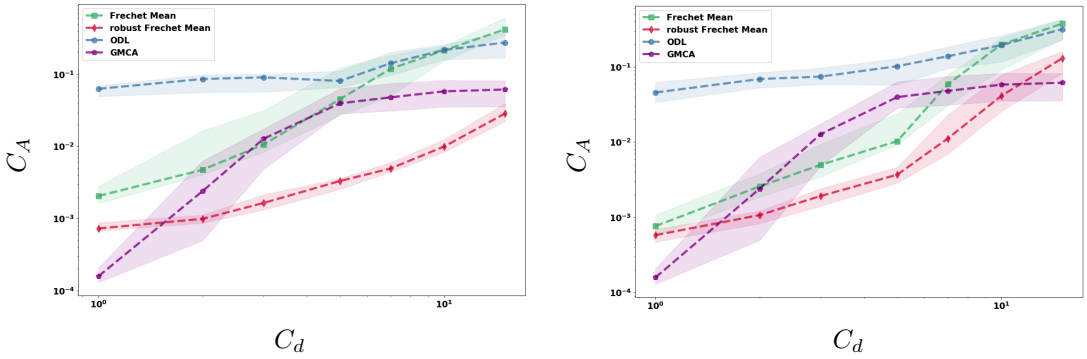
Number of sources n

As seen in Chapter V, the number of sources is one of the elements that drive the complexity of blind source separation, as higher n values imply more difficult unmixing problems.
In these experiments, the sparsity level is fixed to 0.1, the number of observations



(a) Cd = 2.4 (b) Cd = 7

Figure VI.2 – Evolution of the mixing matrix criterion as a function of the mini-batch size for two distinct values of the mixing matrix condition number.



(a) $t_b = 10$ (b) $t_b = 100$

Figure VI.3 – Evolution of mixing matrix criterion as a function of the condition number for mini-batch sizes $t_b = 10$ and $t_b = 25$.

to 20 and the SNR to 40 dB. The number of sources evolves from 5 (right panel of Fig. VI.1) to 15 sources (Fig. VI.4).

For a large number of sources, a general comment is that GMCA obtains degraded performances, consistently with Chapter V. Then, a striking element is that equipped with the Fréchet mean the dGMCA algorithm no more exhibits the peak of improved performances for t_b around $20 - 50$, and eventually does not reach the quality of GMCA in the setting. In contrast, making use of the robust aggregation allows preserving very good separation results for $t_b < 100$, with a gain of almost one order of magnitude with respect to the other methods. These results were to be expected : with a fixed number of observations, finding an increasing number of sources n is more and more difficult, which makes that some mini-batches are strongly badly unmixed. Furthermore, an increasing number of sources also implies an increasing number of partial correlations (*i.e.* coefficients for which multiple sources

are active), which are notoriously difficult to handle and make most sparse BSS algorithm deteriorate [Bobin *et al.* 2015].
From an alternative point of view, a larger number of sources generally increases the number of potential spurious local critical points of the cost function, making robust dGMCA more successful. A more detailed discussion concerning this aspect will be developed in Section C.3.

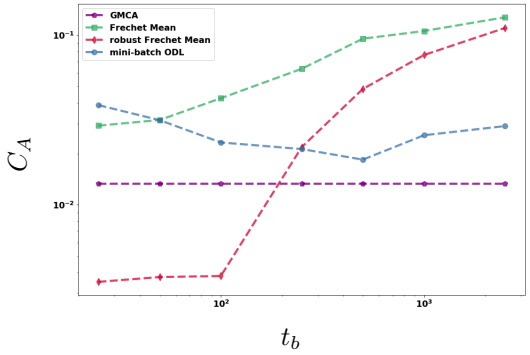


Figure VI.4 – Evolution of the mixing matrix criterion as a function of the mini-batch size for 15 sources.

Sparsity level p

In these experiments, the impact of the sparsity level of the sources p is assessed. Such an impact can be twofold :

— The sources are statistically stationary. However, when they are very sparse (*e.g.* $p = 0.1$), the values taken by a single small mini-batch may largely change between two realizations, which can lead to outliers that will impact the aggregation process. In contrast, mildly sparse sources should lead to a more stable aggregation procedure ;

— Very sparse sources tend to lead to hard-to-escape local minima, which are further smoothed out when sparsity level decreases.

Figure VI.5 displays the evolution of C_A with respect to the mini-batch size for two values of the sparsity level of the mixing matrix : $p = 0.25$ in the left panel and $p = 0.5$ in the right panel. The case $p = 0.1$ is featured in the left panel of Fig. VI.1. These results seem to reveal three distinct regimes :

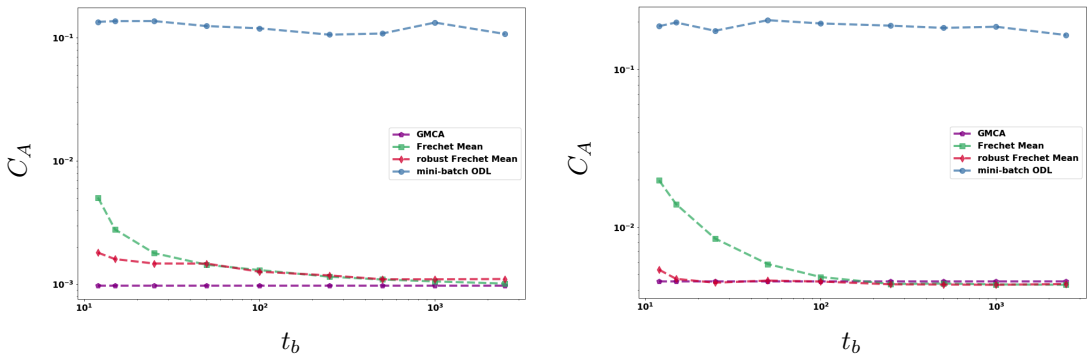
i) For very sparse sources (left panel of Fig. VI.1), the GMCA algorithm yields decent results, but the robust dGMCA algorithm performs significantly better when small mini-batch sizes are used. In this regime, the realizations of the sources are very different for different mini-batches, explaining the discrepancy between a) robust dGMCA and b) GMCA and Fréchet mean dGMCA ;

ii) For mildly sparse sources (left panel of Fig. VI.5), the GMCA algorithm yields better results and there is far less discrepancy between the two dGMCA me-

thods. This highlights that equipped with the (robust) Fréchet mean, combining mini-batch optimization and aggregation leads no performance loss : parallelization allows to go faster and large-scale without diminishing the separation quality. This can be explained by the fact that in this regime the realization of the sources is smoother among mini-batches.

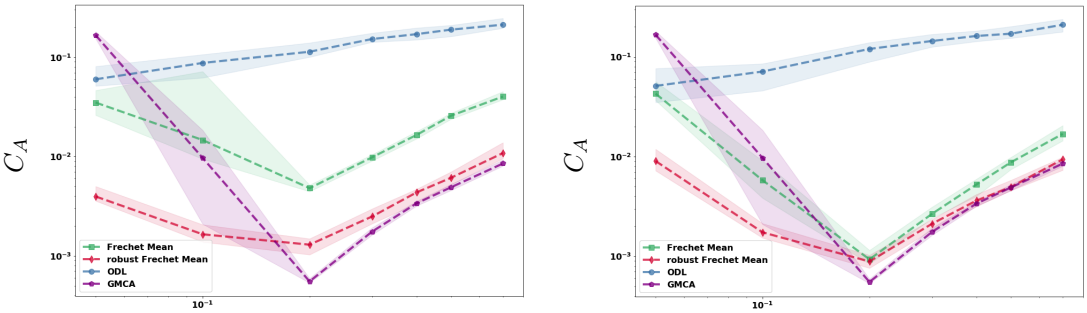
iii) For still less sparse sources (*cf.* right panel of Fig. VI.5), the results of robust dGMCA are consistent with the previous regime : since the mini-batches are on average still more similar, robust dGMCA obtains a separation quality closer to GMCA for still smaller mini-batches. One of the differences with regime ii) is that GMCA obtains slightly worse results. This is expected, as more partial correlations occur. Furthermore, the discrepancy between Fréchet mean and its robust counterpart increases again strongly. This might also be due to an increased number of partial correlations : a few small mini-batches containing such samples might be much more difficult to unmix.

However, such arguments do not fully answer several observations for the *very sparse* regime. In particular, why are the results of GMCA enhanced when going from $p = 0.1$ to $p = 0.25$? Furthermore, why is the robust dGMCA better working than GMCA when $p = 0.1$? Indeed, it should be the opposite, as very large mini-batches should offer in a very sparse regime more stationarity of the precise realization of the sources. To answer to these questions, we will first confirm in the next subsection such observations with a realistic experiment in which the sources are very sparse. In Section C.3, we will then propose an explanation. In brief, the hypothesis we propose is that highly sparse sources induce a less smooth optimization landscape (which is further backed by Fig. VI.6 : the error bars of GMCA are larger for lower p values – at least when the results are still acceptable), with potentially more spurious critical points. Using mini-batches enables a deeper exploration of such a landscape, and thus to find a better solution.



(a) $p = 0.25$ (b) $p = 0.5$

Figure VI.5 – Evolution of the mixing matrix criterion as a function of the sparsity level with $p = 0.25$ (left panel) and $p = 0.5$.



(a) $t_b = 10$ (b) $t_b = 100$

Figure VI.6 – Evolution of mixing matrix criterion as a function of the sparsity level for mini-batch sizes $t_b = 10$ and $t_b = 25$.

C.2 Application to γ-ray spectroscopy realistic simulations : sparse case

In this section, the behavior of the dGMCA algorithm is evaluated in the context of γ-ray spectroscopy. This is one of the main methods used for measuring the activity concentrations of radionuclides in environmental samples. It particularly plays a key role to monitor the radiological environment or perform radioecology studies and nuclear incident preparedness. A γ-ray spectrum is the histogram of the number of detected γ-ray photons in the sensors. In this context, an observation is formed by the linear combination of the contributions from various radionuclides. Each one is described by a signature in energy which is composed of one or several emission lines to which a Compton continuum is associated, as displayed in Figure VI.7. The goal of this experiment is to jointly estimate the activity of each radionuclide (*a.k.a.* the mixing matrix) as well as their signature (*a.k.a.* the sources) from several observations. These simulations are composed of 5 radionuclides : ^{7}Be, ^{22}Na, ^{40}K, ^{137}Cs, ^{210}Pb, which are representative of aerosol samples [Xu *et al.* 2019] and featured in Figure VI.7. The number of observations is fixed to $m = 20$ and the number of samples per source is equal to 16240.

The goal of this section is to evaluate the performances of dGMCA in a setting where the samples are non-stationary, highly sparse and with a large dynamic range ; the information content of a single signature basically spans 2 to 3 orders of magnitude. The sources are modeled in the wavelet domain : γ-ray observations are first decomposed into an undecimated unidimensional wavelet frame [Starck *et al.* 2010] before applying any BSS method. The number of scales is fixed to 5, which yields a number of wavelet coefficients equal to 81200 ; these are obviously not large-scale data but it already allows to highlight some remarkable results.

Figure VI.8 shows the reconstructed solution with GMCA, ODL and dGMCA equipped with the robust Fréchet mean with $t_b = 10$; it also displays the estimation error in transparent solid line. This figure first shows that dGMCA provides a very good

reconstruction of $^{2}_{2}$Na signature, while both GMCA and ODL exhibit clear leakage from other sources. The estimation error of the dGMCA solution does not present any structure and is mainly dominated by noise.
Figure VI.9 features the evolution of the mixing matrix criterion as a function of the mini-batch size for two different levels of the signal-to-noise ratio : 40 and 80 dB. These values might seem very large but it has to be recalled that the dynamic range is very large ; a small amount of noise might already erase a significant part of the Compton continuum while leaving only the photon peaks. This experiment first shows that GMCA, ODL and dGMCA with the standard Fréchet mean performs rather poorly. In agreement with the results of the previous subsection studying the impact of the sparsity level p, the use of the robust aggregation makes the dGMCA algorithm largely outperforms these methods, especially when the mini-batch size is smaller than $t_b = 50$. Further randomizing the mini-batches entails an extra improvement, especially for middle-size mini-batches for $100 < t_b < 1000$. The gain is particularly large when the noise level is small.
To explain the results of such a setting, we advocate the fact that the sources are by a large extent dominated by few photon peaks, which is likely to create spurious hard to escape critical points. This might explain why neither the GMCA algorithm nor the dGMCA algorithm without the robust aggregation are able to perform correctly. The use of the robust Fréchet mean along with ALS clearly yields empirical robustness to the algorithm as testified by the very small scatter of the results (the shaded areas). As such, it is likely that the proposed minimization scheme generate some implicit regularization that is beneficial to efficiently tackle sparse BSS problems by making the optimization easier. More concerning this topic is said in the following.

C.3 Discussion - robustness and implicit regularization

During the last few years, understanding the impact of optimization in matrix factorization problems [Gunasekar *et al.* 2017] or learning (deep) neural networks [Neyshabur 2017] has attracted a lot of interest. More specifically, it has been emphasized in many works that a specific optimization method might enable *implicit regularization*. In this subsection, we first detail the notion of implicit regularization and then highlight links to such works in order to better understand the behavior of dGMCA. Our hypothesis is backed through further experiments.

Optimization landscape and implicit regularization

Most of the investigations studying the optimization landscape of learning problems focus on the underdetermined or over-parameterized case [Gunasekar *et al.* 2017, Neyshabur *et al.* 2017]. In this regime, it has been showed that under some conditions local minimizers are likely to be global. However, such a claim does not help concerning the quality of these critical points [Neyshabur *et al.* 2017]. In sparse BSS, the factorization problem at play is furthermore determined or over-determined, which

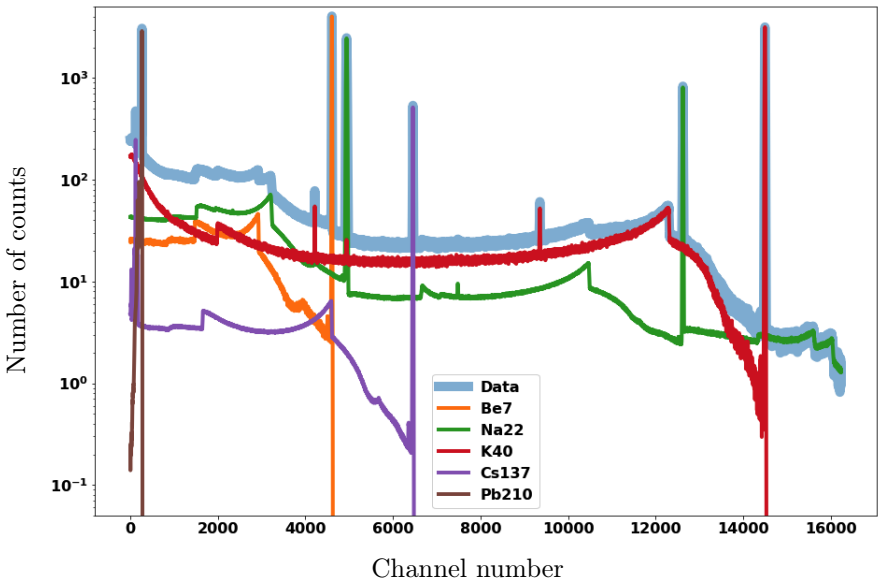


Figure VI.7 – Experiment in γ-ray spectroscopy : example of a single observation and the contribution of each of the radionuclide sources.

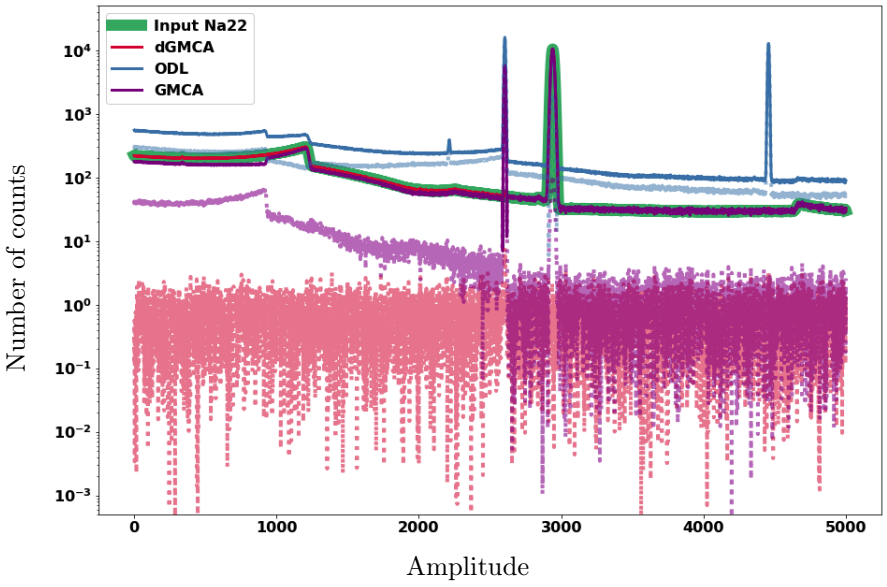
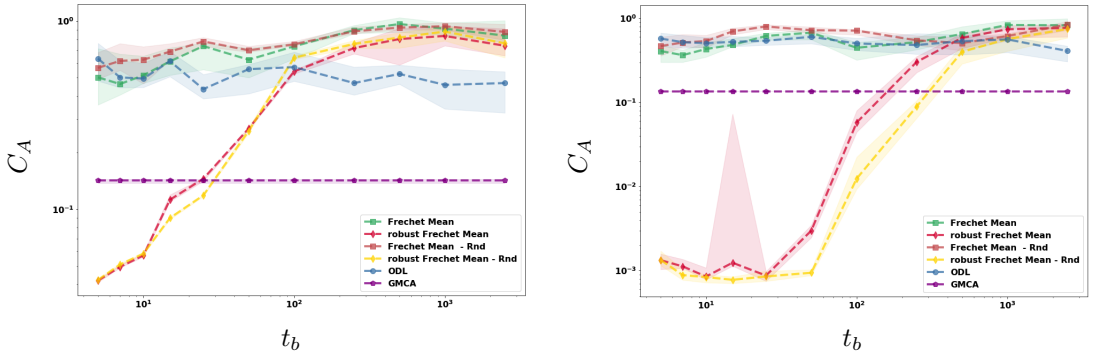


Figure VI.8 – Experiment in γ-ray spectroscopy : $^{2}_{2}$Na radionuclide estimated with GMCA, ODL and dGMCA equipped with the robust Fréchet mean. Errors with respect to the input spectrum are displayed in transparent solid lines.

means that the optimization landscape is likely to be largely different and probably less smooth due to the presence of spurious local minimizers. In this setting, regu-



(a) SNR = 40 (b) SNR = 80

Figure VI.9 – Experiment in γ-ray spectroscopy : evolution of the mixing matrix criterion C_A as a function of the mini-batch size for SNR 40 dB and 80 dB.

larization, either being it explicit or implicit, is of paramount importance.
By *explicit* regularization, we mean a regularization appearing in the cost function (*e.g.* the sparsity promoting term of Eq. (II.8)). By *implicit*, we denote an existing regularization only due to the optimization scheme, that is which does not appear explicitly. To give an example of implicit regularization in general matrix factorization problems, one can cite the role induced by gradient descent in many algorithms [Gunasekar *et al.* 2017]. To tell a long story short, using GD for underdetermined problems implies a *regularization penalizing the complexity of the learnt model* (*e.g.* with low nuclear norm) [10].
In our case, while such regularization might appear because of the analogy made between ALS and GD in Section B.3, we rather study more specifically the implicit regularization implied by the use of mini-batches.

Regularization induced by mini-batches in machine learning

To that respect, the case of highly sparse sources and more particularly the γ-spectroscopy example is particularly illustrative : dGMCA performs very well as the batch-size decreases, while GMCA does not. Such a phenomenon is well known in the context of machine learning, where it has been noticed for long that using small stochastic mini-batches can indeed improve the results over full batch methods [LeCun *et al.* 2012, Keskar *et al.* 2016, Hardt *et al.* 2015]. A common interpretation is that using small size mini-batches is important as it injects noise in the optimization process, which is essential to escape certain types of minimizers.
Very closely related to dGMCA, several articles have recently emphasized the interest of using non-vanishing learning rates [Smith *et al.* 2017, Xing *et al.* 2018].

10. Note that beyond matrix factorization, similar implicit regularizations have been studied in machine learning and it has been emphasized that in contrast to standard belief GD algorithm leads to minima [Lee *et al.* 2016] that surprisingly generalize well [Gunasekar *et al.* 2017, Neyshabur 2017, Gidel *et al.* 2019, Azizan *et al.* 2019]

This is appealing, as following our analogy with ALS we are specifically in a regime where such a rate is constant over the iterations (set to $\eta^{(l)} = 1$). More specifically, and beyond a potential computation gain in terms of the number of parameter updates [Smith *et al.* 2017], it has been empirically shown by [Xing *et al.* 2018] that using stochastic mini-batches (respectively high learning rate) enables to explore broader areas (respectively to jump optimization landscape borders and escape bad areas), thus finding results further from the initialization. The authors argue that the stochasticity introduced by small size mini-batches through a structured noise strongly favors flat minimizers that are akin to generalize better [Hochreiter & Schmidhuber 1997], since the introduced noise would be roughly aligned with the sharpest loss directions [11].

Links to dGMCA

A similar phenomenon is very likely to be at play within dGMCA, for which the generalization notion would translate into minimizers that are less sensitive to a given realization of the sources. Indeed, the algorithm performs much better in the highly sparse case, where spurious minima tend to be sharper and more difficult to escape. The stochasticity induced by mini-batches then helps exploring the optimization landscape.
To further investigate such an exploratory ability within the experimental setting of Section C.2, Figure VI.11 displays the histograms of the mixing matrix criterion after 250 iterations of dGMCA (close to "convergence") when the Fréchet mean (left panel) or its robust version (right panel) is used. In the first case, it is interesting to notice that using large mini-batches tends to provide more stable solutions, with a smaller scatter of the criterion across mini-batches. In contrast, using smaller mini-batches leads to a broader exploration of the parameter space as testified by more widespread values of the mixing matrix criterion. As most of the values are bad (close to 10 dB), the aggregated estimate is rather poor.
Switching to robust aggregation (right panel Figure VI.11), the results might seem to go against the previous hypothesis of exploratory power : while in accordance to the previous experiments the separation with small mini-batches is good, the scatter is however smaller than when using large mini-batches. Nevertheless, we are here looking at the results *after 250 iterations*. That is, it seems that at this point dGMCA already found a good minimum of the optimization landscape. As such :

— The fact that the scatter with small t_b values is smaller that with high t_b values indicates that most of the mini-batches are contained within a basin of attraction of a good minimum. This is relieving, as it means that the noise introduced by small mini-batches (and also the high learning rate $\eta^{(l)}$) is not high enough to make most of the mini-batches "jump out" of the good mini-

11. Note that for similar reasons, some authors have been injecting noise in their algorithm to benefit from such a better exploration of the optimization landscape [Neelakantan *et al.* 2015] – see [Rapin 2014] in sparse BSS.

mum. As such, the few outliers present in the histogram are easily handled by the robust aggregation (there is much less than 50 % of them) ;

— The fact that on the other hand the scatter is non-zero, nor even negligible might further highlight that the local minimum we are looking for is rather flat.

Figure VI.10 gives a last interesting insight concerning the optimization landscape of Eq. (II.8). It shows the evolution of the mixing matrix criterion of dGMCA with the robust Fréchet mean as a function of the number of mini-batches B (*i.e.* this is somehow unrealistic in applications since the limiting factor is the total number of samples). The best quartile of the large mini-batches obtains in general almost as good performances as the small ones. Therefore, the lack of exploratory power with large B seems to impede the separation because *most* of the large mini-batches are unable to move from bad minima. However, some of them are still able to find good solutions, but they are too few for the aggregation to highlight them.

To sum up, an optimization landscape exploratory phenomenon related to the stochasticity of mini-batch optimization occurs when small mini-batches are considered (typically a few times the number of sources n). In this regime, the separation quality will improve when the number of mini-batches increases. This phenomenon will vanish when the mini-batch size is too large.

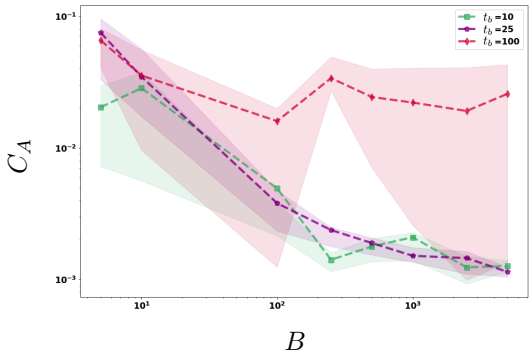


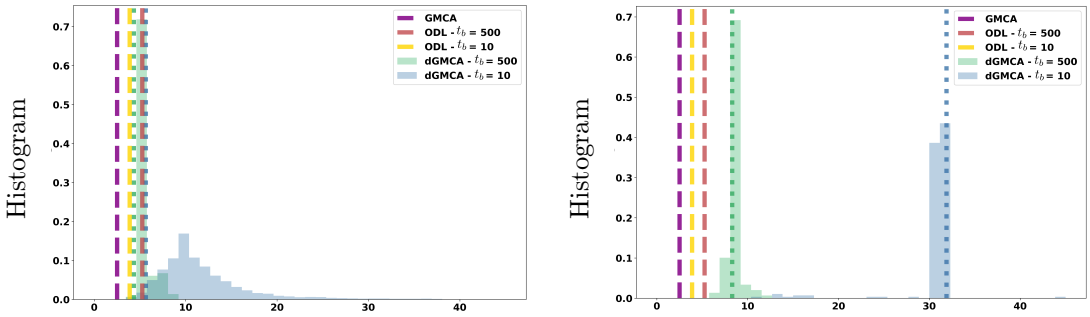
Figure VI.10 – Evolution of the mixing matrix criterion as a function of the number of mini-batches for $B = 10$, $B = 25$ and $B = 100$.

C.4 Computation time

We now conclude this experimental section with the computation time of dGMCA, which depends both on the complexity of one epoch and the number of required epochs.

Complexity of one epoch

Consistently with Chapter V, each iteration of the GMCA algorithm has a complexity of $\mathcal{O}(t(mn + n^2 + m))$. The complexity of one epoch of dGMCA is similar,



(a) Fréchet mean (b) robust Fréchet mean

Figure VI.11 – Histogram of the mixing matrix criterion across mini-batches with *left* the Fréchet mean and *right* the robust Fréchet mean and $B = 500$ and $B = 10$.

once the cost of the Frechet mean as been taken into account :

$$\mathcal{O}\left(b(mn + n^2 + m) + \frac{t}{b} nmL_f\right), \tag{VI.16}$$

where the last term correspond to the Fréchet mean and L_f corresponds to the number of iterations required for its computation. As such, except for very small mini-batches, the linear gain of using dGMCA over GMCA dominates. This is experimentally confirmed (see Fig. VI.12) : in practice the computation time for a given number of iterations does not deviate much from linearity (in particular, the transfer costs between the nodes are negligible in comparison to the computation time).

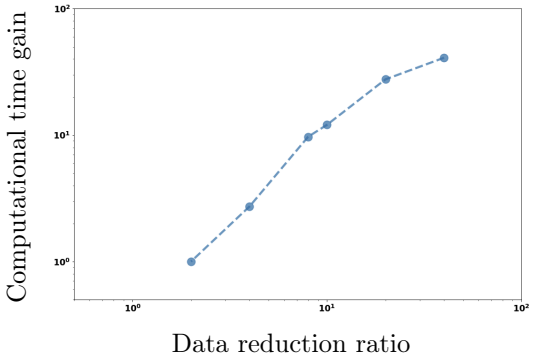


Figure VI.12 – Computational time gain of the dGMCA algorithm with respect to the GMCA algorithm as a function of the reduction gain t/t_b. The dGMCA algorithm has been run on a PC cluster equipped with 8 Amd CPUs, each one has 6 cores Istanbul Opteron 8431 at 2, 4Ghz.

Number of epochs

Concerning the number of epochs, an interesting observation can be drawn from Figure VI.13, which shows the evolution of the mixing matrix as a function of the epoch number for the γ-ray spectroscopy experiment. It highlights that the smaller mini-batches are, the faster the algorithm is. It is very striking to remark that dGMCA almost reaches its stationary regime (up to noise related to stochastic mini-batch optimization) in about 25 iterations, which further makes the algorithm very interesting to provide a fast estimate from large-scale data.

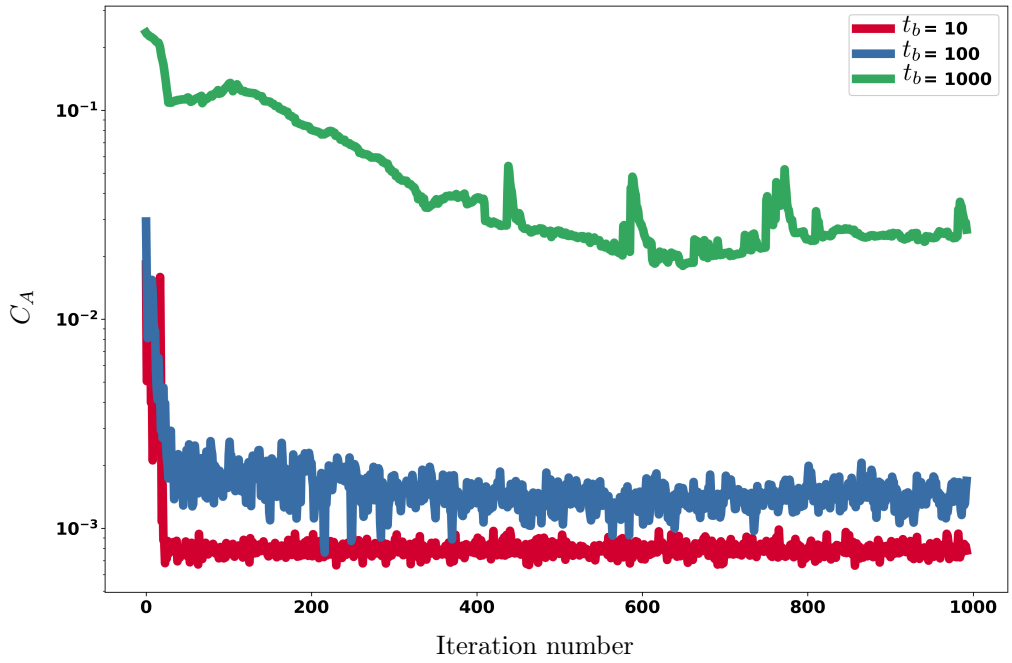


Figure VI.13 – Evolution of the mixing matrix criterion along the separation process for $t_b = 10$, 100 and 1 000. The dGMCA algorithm is used with the robust Fréchet mean

C.5 Experimental results : summing-up

The previous results concerning the computation time have validated the fact that the dGMCA enables to work with large-scale datasets, as there is a linear gain in the complexity of each iteration, and the number of iterations can further be highly reduced in some experimental settings.
It was however expected that switching to a distributed version of the GMCA algorithm would not have come at no cost : using mini-batches should have led to larger estimation errors. Seen from the distance, the experiments tend to highlight two distinct regimes :

— **Mildly sparse sources :** in this setting, a key result is that the robust dGMCA and GMCA algorithms perform similarly ; going distributed comes at almost no cost as soon as the mini-batch size is large enough. This shows that the proposed dGMCA is an useful approach for the separation of large-scale mildly sparse signals, which describe numerous natural signals represented in multiscale representations.

— **Sparse to very sparse sources :** mini-batch optimization with very sparse sources cause a series of problems. In particular, small-size mini-batches tend to be less homogeneous, which is likely to lead to more outliers and degrade the aggregation procedure. In this case, the robust Fréchet mean is an interesting aggregation procedure. Unexpectedly, it can further continue to improve for very small mini-batch sizes, the best solutions being reached for sizes of the order of the number of sources. Strikingly, the results are then more accurate than the ones of the GMCA algorithm. The explanation we proposed is that dGMCA enables a better exploration of the optimization landscape : if the randomness introduced by the mini-batches is well handled by the aggregation, it makes that the final unmixings are better.

Conclusion

To tackle the large-scale sparse BSS problem, we introduced in this work the dGMCA algorithm, which uses mini-batches in a projected Alternating Least-Square framework. The different estimations of the mixing matrix are aggregated taking into account the Riemannian manifold structure of the problem constraints. To perform such an aggregation, we proposed to use a robust Fréchet mean, which further improve the separation quality. While for mildly sparse sources the approach is experimentally demonstrated to yield a huge gain in computation time at almost no cost in terms of separation accuracy, dGMCA can even improve the separation results over its full batch counterpart when used for very sparse sources. The explanation we propose is that using stochastic mini-batches enables a better exploration of the optimization landscape. Many numerical experiments are proposed to show the relevance of the approach.

Chapitre VII

Sparse BSS : from Linear to Non-Linear Mixtures

In this last chapter, we extend the previous work by departing from the usual linear setting and tackling the case in which the sources are mixed by an unknown *non-linear* function. We propose a stacked sparse BSS method enabling a sequential decomposition of the data through a linear-by-part approximation. In this context, non-linear BSS can be seen has solving a potentially large number of linear BSS sub-problems. Therefore, an automatic hyper-parameter choice for each sub-problem is mandatory due to the computational burden (*cf.* Chapter IV [1]), and optimization strategies such as block-coordinate methods (see blockGMCA in Chapter V) or mini-batches (see dGMCA in Chapter VI)) could yield improved performances.
Beyond separating non-linearly mixed sources and despite increased indeterminacies compared to linear BSS (*cf.* Chapter II-A.2.2), the introduced StackedAMCA can under discussed conditions further learn an approximation of the inverse of the unknown non-linear mixing, enabling to reconstruct the sources. The quality of the method is experimentally demonstrated, and a comparison is performed with state-of-the art non-linear BSS algorithms. We also propose an in-depth discussion of StackedAMCA required hypotheses.

A Non-Linear BSS

A.1 Mixing model

In this chapter, the BSS model will change into the *non-linear* one presented in chapter II-A.2.2 :

$$\mathbf{X} = \mathbf{f}^*(\mathbf{S}^*) + \mathbf{N} \tag{VII.1}$$

Where $\mathbf{f}^*$ is an unknown *non-linear* function from $\mathbb{R}^{n\times t}$ to $\mathbb{R}^{m\times t}$ (here, $n \leq m$). We will here consider general functions $\mathbf{f}^*$, by mostly assuming that $\mathbf{f}^*$ is invertible and symmetrical around the origin, as well as regular enough. Regular means that $\mathbf{f}^*$ is $\mathcal{L}$-Lipschitz with $\mathcal{L}$ small enough and that $\mathbf{f}^*$ does not deviate from a linear mixing too fast as a function of the input amplitude (which can for instance be the case with sensor saturations, or chemical sensors etc.) ; see Fig. VII.1 for examples of mixings StackedAMCA can or cannot handle. The other required hypotheses, both

1. Note that we however only use only the warm-up stage ; the introduction of a refinement step is left for future work.

on the mixing and the sources, are discussed in Sec. E.

A.2 Previous work

In this subsection, we present some previous works concerning non-linear BSS. A summary of the different algorithm families is displayed in Figure VII.2.

A.3 Independent component analysis

While most of the previous work has been devoted to ICA, the independence prior is not sufficient to ensure source separation in the general non-linear setting [Comon & Jutten 2010]. A first possibility to bypass this separability issue is to explicitly focus on a special kind of mixing $\mathbf{f}^*$, for which separability can be shown (under conditions that are not discussed here for the sake of compactness). Among the most well-known kind of mixings, one can cite [Deville & Duarte 2015] :

— Post Non-Linear – PNL. In these, each observed mixture $\mathbf{X}_i, i \in [1, m]$ is an univariate nonlinear function of a linear mixture of the sources :

$$\mathbf{X}_i = \mathbf{f}_i^* \left(\sum_{j=1}^{n} \mathbf{A}_i^{*j} \mathbf{S}_j^* \right) \tag{VII.2}$$

— Linear Quadractic mixtures – LQ. In these, each observed mixture $\mathbf{X}_i, i \in [1, m]$ is a polynomial function of second degree of the sources :

$$\mathbf{X}_i = \sum_{j=1}^{n} \mathbf{A}_i^{*j} \mathbf{S}_j^* + \sum_{j=1}^{n} \sum_{k=j}^{n} \mathbf{B}_i^{*j} \mathbf{S}_j^* \mathbf{S}_k^* \tag{VII.3}$$

Where $\mathbf{B}^*$ is, similarly to $\mathbf{A}^*$, a mixing matrix. Another possibility is to use an explicit or implicit regularization making the problem better posed. For explicit regularization, one can cite additional priors on the sources such as temporal dependencies [Hyvarinen & Morioka 2017, Ehsandoust *et al.* 2017]. However, in this work we will not assume any such additional explicit prior. In such situations, several algorithms use an implicit regularization. Among the most well-known, one can cite [Almeida 2003, Honkela *et al.* 2007, Brakel & Bengio 2017]. The work of [Almeida 2003] is an extension of INFOMAX to the case of non-linear mixtures, in which a neural network is learnt by minimizing the mutual information. The difference with the linear case is that the functions enabling to approximate the cumulative distribution functions of the sources have to be learnt accurately and not chosen a priori, since the non-linear problem is much less constrained. [Honkela *et al.* 2007] summarizes several works using a variational Bayesian approach bringing the required additional regularization. The derived cost function is used to train a non-linear model of the mixing $\mathbf{f}^*$ (*e.g.* a Multi-Layer Perceptron). In [Brakel & Bengio 2017], the authors use a Generative Adversarial Network : the idea is to maximize independency measures implicitly thanks to adversarial objectives. The goal of the generator is

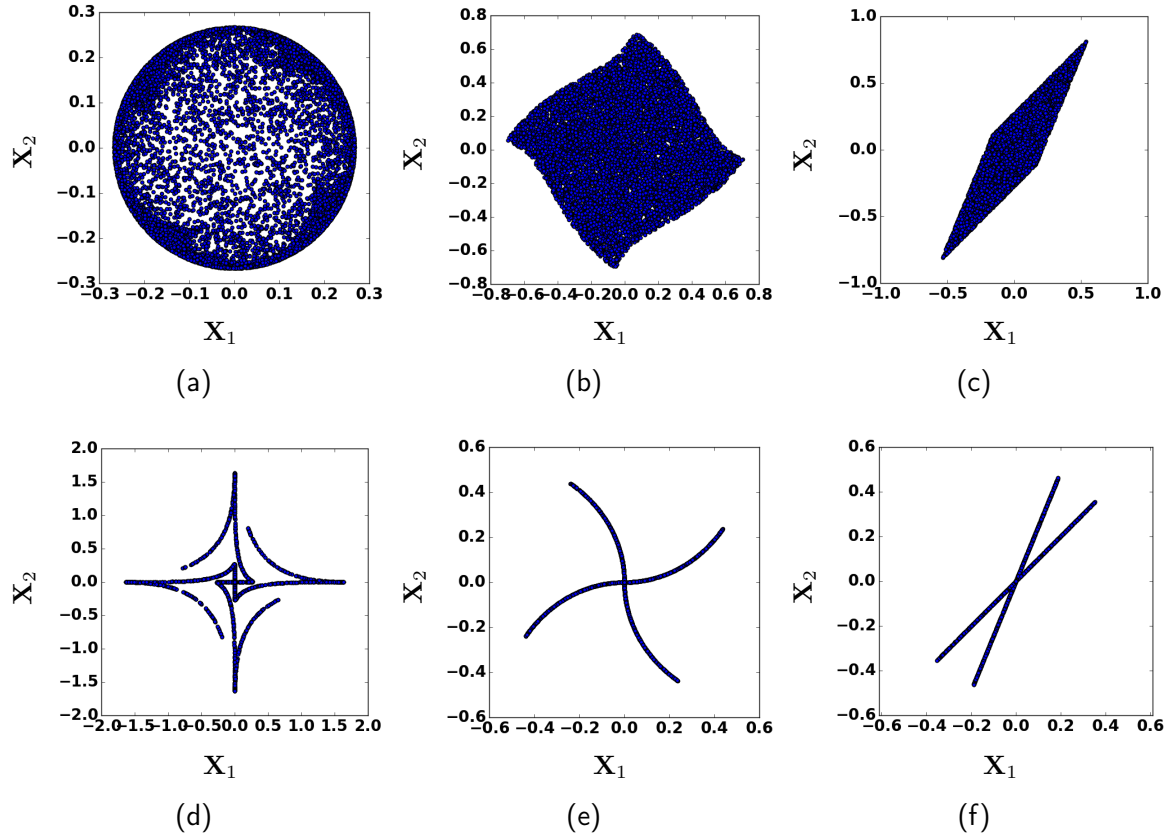


Figure VII.1 – Examples of three different mixings on *Up* : non-sparse (uniformly distributed) sources ; *down :* Sparse (Bernoulli-Gaussian) sources. *Left :* Mixing highly deviating from non-linearity, for which StackedAMCA would not work : $\mathbf{X}_0^i = \cos(\alpha(i))^5\mathbf{S}_0^{*i} - \sin(\alpha(i))^5\mathbf{S}_1^{*i}$ and $\mathbf{X}_1^i = \sin(\alpha(i))^5\mathbf{S}_0^{*i} + \cos(\alpha(i))^5\mathbf{S}_1^{*i}$; *Middle :* Mixing deviating from linearity, but regular enough for StackedAMCA to work well : $\mathbf{X}_0^i = \cos(\alpha(i))\mathbf{S}_0^{*i} - \sin(\alpha(i))\mathbf{S}_1^{*i}$ and $\mathbf{X}_1^i = \sin(\alpha(i))\mathbf{S}_0^{*i} + \cos(\alpha(i))\mathbf{S}_1^{*i}$, with $\alpha(i) = \frac{\pi}{2}(1 - \sqrt{\mathbf{S}_0^{*i^2} + \mathbf{S}_1^{*i^2}})$; *Right :* Linear mixing : $\mathbf{X} = \mathbf{A}^*\mathbf{S}^*$.

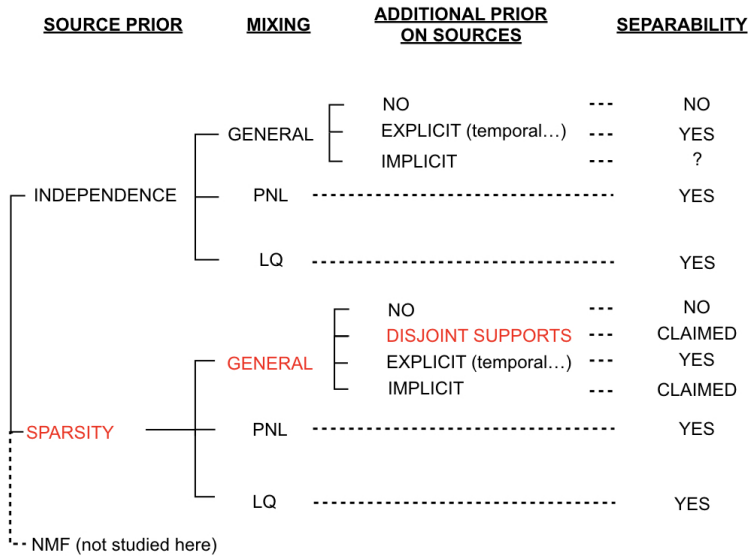


Figure VII.2 – Summary of the different families of algorithms. In red, settings we study in this chapter. The "YES" mention for separability might be subject to other assumptions. The "CLAIMED" refers to the work of [Ehsandoust *et al.* 2016].

then to produce samples reconstructing the data well (therefore, a reconstruction is performed through a decoder) and which are undistinguishable by the discriminator of independent samples. The implicit additional prior thus comes here from the network structure itself.

A.4 Sparsity

Concerning works based on the sparsity of the sources, [Theis & Amari 2004, Van Vaerenbergh & Santamaría 2006, Duarte *et al.* 2015] attempted the problem of PNL mixtures, and [Duarte *et al.* 2012] the problem of LQ mixtures. General settings similar to the framework of the current article have mainly been studied in [Ehsandoust *et al.* 2016, Puigt *et al.* 2012] and focus on the geometric interpretation. In [Ehsandoust *et al.* 2016] the authors claimed the *separability* of the sources when using a sparsity prior, if only one source is active for each sample. Note that the mere sparsity prior is however in general not sufficient for source *reconstruction*. The algorithm they propose uses a clustering approach, followed by a manifold unfolding (which, in particular aims at tackling the partially correlated samples if additional information is known – *e.g.* source regularity). However, they do not propose a specific method to perform the clustering in the case of more than $n > 2$ sources. In [Puigt *et al.* 2012], the authors first propose to find zones in the observation domain in which only a single source is active. This gives a scattered representation of the mixing non-linearities. As several zones may correspond to the same source, these are then aggregated to get more statistics for estimating the mixing. Unfortunately,

the experiments also only focus on the case $n = 2$.

A.5 Contribution

We propose to tackle the general problem of *non-linear* BSS presented in Eq. (II.3) by using a sparsity prior on the sources, without assuming any additional explicit priors on them. To the best of our knowledge, our method is the first attempting to find a linear-by-part approximation of the underlying non-linearities using a stacked sparse BSS approach. This departs from usual methods as :

— In contrast to neural network approaches [Almeida 2003, Brakel & Bengio 2017], we explicitly use the geometric interpretation in terms of 1-dimensional (1D) manifolds (*cf.* Sec. B) existing in the case of sparse sources ;

— In contrast to clustering approaches [Ehsandoust *et al.* 2016, Puigt *et al.* 2012], the clustering we use is only *local* and *linear-by-part*. As such, it is easier in settings with a high number of sources : in particular, we will apply it in the case of more than $n > 2$ sources contrary to [Ehsandoust *et al.* 2016, Puigt *et al.* 2012]. Furthermore, it is based on usual sparse *linear* BSS algorithms, enabling to re-use much of the corresponding work in terms of optimization and automatic hyper-parameter choice, which has lead to enhanced separation quality [Bobin *et al.* 2015] (*cf.* previous Chapters).

Beyond separating sources, the algorithm yields a possible source reconstruction by inverting the estimated linear-by-part model. Despite the usual non-linear BSS indeterminacies, this reconstruction is empirically shown to estimate well the true sources under some discussed hypotheses. In Sec. B, the method is further described. In Sec. D, some experiments are conducted on three different mixings, two of them with a high number of sources. In Sec. E, the required hypotheses for the proposed approach are studied.

B Proposed Approach

B.1 A Geometrical Perspective on Sparse Non-Linear BSS

The proposed method is first described by adopting a geometric point of view in the case of $n = 2$ sources. The principle can however be generalized to higher values, and the relevance of the method will be empirically testified on such difficult settings in Section D.
When plotting $\mathbf{S}_1^*$ as a function of $\mathbf{S}_2^*$, most of the source samples lie on the axes due to the morphological diversity hypothesis ([Bobin *et al.* 2007] – *cf.* Chap. III-C.4.2). In this chapter, we will even assume that *all* the source samples lie on the axes (*cf.* Fig VII.3, left) – this disjoint support hypothesis is further discussed in Sec. E. Once mixed through the *non-linear* $\mathbf{f}^*$ function, the source samples on the axes are transformed into n non-linear 1D manifolds [Ehsandoust *et al.* 2016, Puigt *et al.* 2012], each corresponding to one source (see Fig VII.3, right).

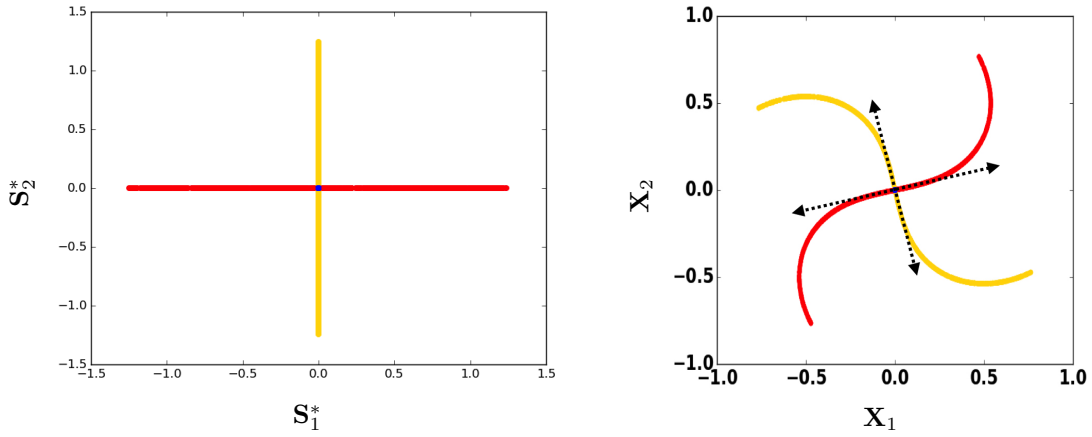


Figure VII.3 – *Left :* Scatter plot of $n = 2$ sources. The red color is associated with samples where only source 1 is active. Yellow is associated with source 2 ; *Right :* A non-linear mixing of $n = 2$ sparse sources. The dashed arrows correspond to the mixing directions found by a *linear* model. The colors, corresponding to each source, are displayed for explaining the distortion introduced by the mixing $\mathbf{f}^*$ of the source samples but are unknown in a blind setting.

To perform source *separation*, BSS then geometrically aims at back-projecting each manifold on one of the axes. As evoked in Chapter II-A.2.2, we then obtain separated sources which are an approximation of the true ones $\mathbf{S}^*$ up to the non-linear indeterminacy function $\mathbf{h}$. Source *reconstruction*, on the other hand, amounts at finding a *specific* non-linear back-projection, which corresponds to a $\mathbf{h}$ being only a scale factor.
In the context of StackedAMCA, we propose to perform the back-projection on the axes by approximating the 1D-manifolds by a *linear-by-part* function (see the example of Fig. VII.4 corresponding to the data of Fig VII.3), which is inverted.

B.2 Overview of The Proposed Approach

More specifically, the lowest amplitude samples of the data $\mathbf{X}$ can be well approximated by a classical linear model (*cf.* Fig. VII.3, right) because of the regularity assumption on $\mathbf{f}^*$ stating that the mixing must not deviate from linearity too fast as a function of the amplitude (*cf.* Sec. E). Thus, it is possible to find a first linear model for the low-amplitude samples using a **sparse linear BSS algorithm**, provided that this one is robust to the higher amplitude non-linearities.
However, this approximation is too rough for higher amplitude samples, for which the mixing deviates strongly from linearity. It is nevertheless possible to improve it through the introduction of another linear model fitted to higher amplitude samples, thus creating a linear-by-part approximation (with initially two segments). This is

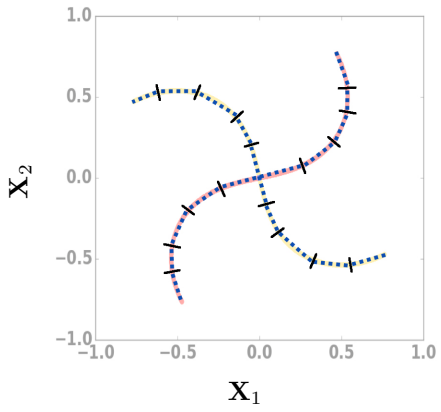


Figure VII.4 – Linear-by-part approximation of the non-linearities.

done by introducing a **non-linear step** that first *selects the (highly non-linear) samples* that are not currently well linearly separated and then *creates a new dataset comprehending only them*. A new linear model can next be fitted to the lowest amplitude samples of this new dataset. A graphical example of a naive version of StackedAMCA showing the main ideas and the corresponding challenges is shown in Fig. VII.5.

B.3 StackedAMCA, detailed description and notations

The whole algorithm iterates the process described in the previous subsection by alternating at each iteration l a linear BSS step and a non-linear step. The first step computes a linear model $\hat{\mathbf{A}}^{(l)}$ on the current residual $\mathbf{R}^{(l)}$. The second one paves the way for the next iterations by computing a new residual $\mathbf{R}^{(l+1)}$. This is done by finding for each source i a maximum amplitude value $\tau_i^{(l)}$ above which the non-linearities are too high to be considered as currently well estimated, and then shrinking the current data using the $\tau_{1..n}^{(l)}$ (*cf.* Fig. VII.6 for the first residual computation). This shrinkage enables to sequentially reduce the amplitudes of the originally higher non-linearities, and therefore to compute linear models describing them. Thus, at each iteration a new linear model is fitted to increasingly higher amplitude samples of the original data $\mathbf{X}$.

More details concerning the two main steps are given below. The algorithm as well as the notations are summarized in Fig. VII.7 and an illustration on a concrete example is given in Fig VII.8.

B.4 Linear Sparse BSS Step : AMCA

The main requirement for the linear sparse BSS algorithm is its abillity to find a linear model representing well the lowest amplitude samples of the residual $\mathbf{R}^{(l)}$,

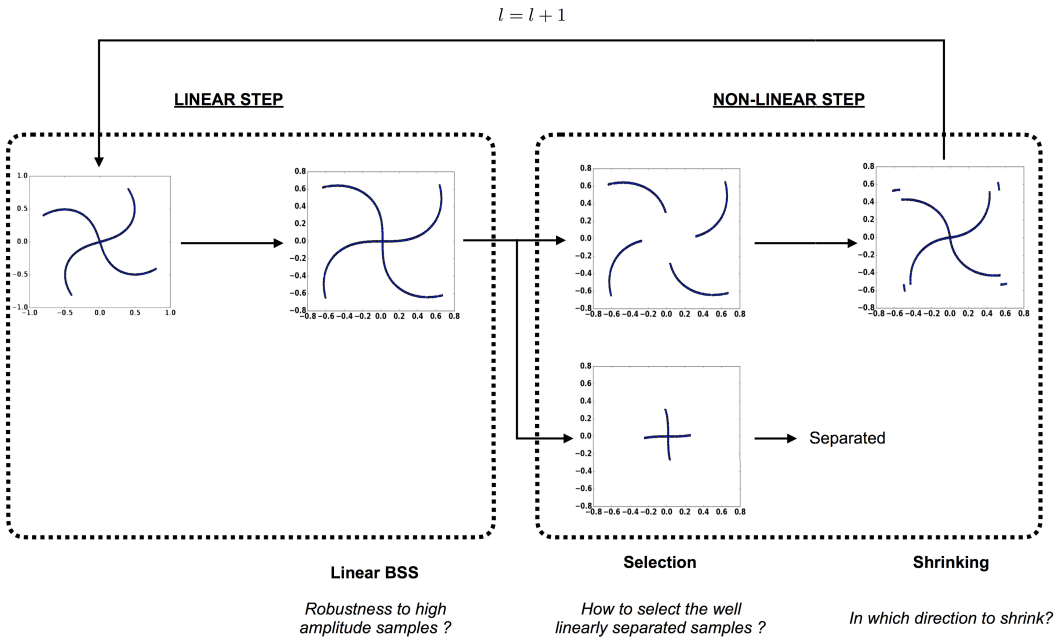
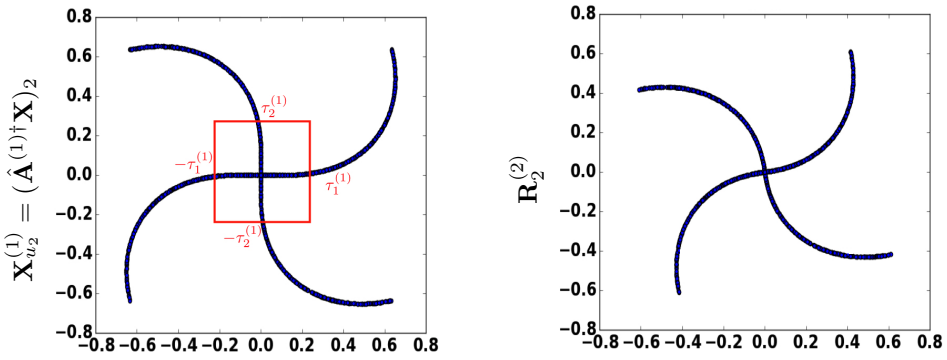


Figure VII.5 – Summary of the challenges faced by StackedAMCA on Fig. VII.3 data. Note that here, only the naive approach described in Sec. B.5.1 is shown, and thus it corresponds to the final version of StackedAMCA for the first iteration only.



(a) (b)

Figure VII.6 – *Left :* In blue, output of the manifold unfolding at the first iteration (for which it coincides to an inversion of the linear model of Fig. VII.3 to align the found dashed arrows with the axes). In addition, the red square delimits the low amplitude sample areas where the linear model is a good approximation – the corresponding maximum amplitudes are denoted as $\tau_{1..n}^{(1)}$ – which means the areas where the samples almost lie on the axes. *Right :* Residual $\mathbf{R}^{(2)}$. The residual $\mathbf{R}^{(l+1)}$ is computed from the left figure data $\mathbf{X}_u^{(l)}$ by shrinking the amplitudes of the samples by $\tau_{1..n}^{(1)}$. .

StackedAMCA$(\mathbf{X})$
$\mathbf{R}^{(1)} = \mathbf{X}$
for l in $1...L$:

— **Linear step** : estimates $\hat{\mathbf{A}}^{(l)}$ with AMCA through the minimization of (*cf.* Sec. III-C.5 for more details) :

$$\min_{\hat{\mathbf{A}}^{(l)},\hat{\mathbf{S}}} \frac{1}{2}\mathtt{Tr}[(\mathbf{R}^{(l)} - \hat{\mathbf{A}}^{(l)}\hat{\mathbf{S}})\mathbf{M}(\mathbf{R}^{(l)} - \hat{\mathbf{A}}^{(l)}\hat{\mathbf{S}})^T] + \sum_{i=1}^{n} \left\| \lambda_i \odot \hat{\mathbf{S}}_i \right\|_1 + \iota_{\|\hat{\mathbf{A}}^i\|_{\ell_2}=1}(\hat{\mathbf{A}}) \tag{VII.4}$$

— **Non-linear selection step** : compute $\mathbf{R}^{(l+1)}$
  — Unroll manifolds in $\mathbf{X}$ using $\hat{\mathbf{A}}^{(1)}...\hat{\mathbf{A}}^{(l)}$: result denoted $\mathbf{X}_u^{(l)}$
  — From $\mathbf{X}_u^{(l)}$, select highly non-linear samples : find $\tau_{1..n}^{(l)}$
  — Compute $\mathbf{R}^{(l+1)}$ through soft-thresholding $\mathbf{R}^{(l+1)} = \mathcal{S}_{\tau_{1..n}^{(l)}}(\mathbf{X}_u^{(l)})$

return $\hat{\mathbf{A}}^{(1)}... \hat{\mathbf{A}}^{(L)}$

Figure VII.7 – StackedAMCA algorithm summary.

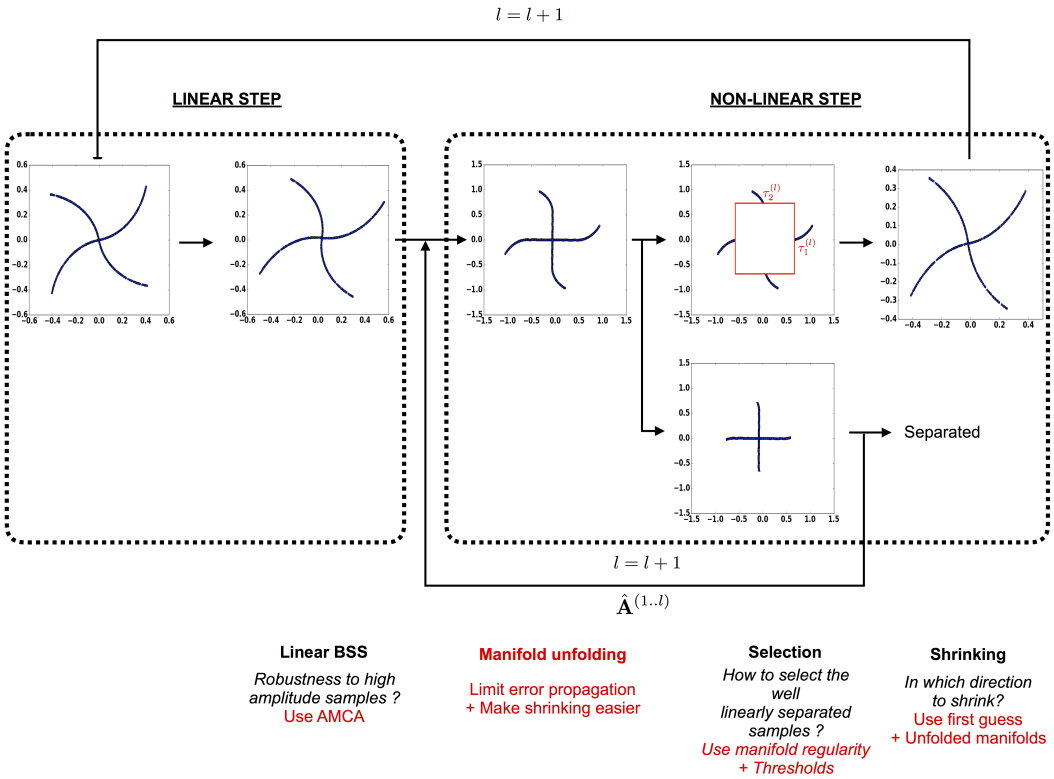


Figure VII.8 – Illustration of the iteration $l = 4$ of the final version of StackedAMCA. In red : proposed solutions to the challenges of Fig. VII.5.

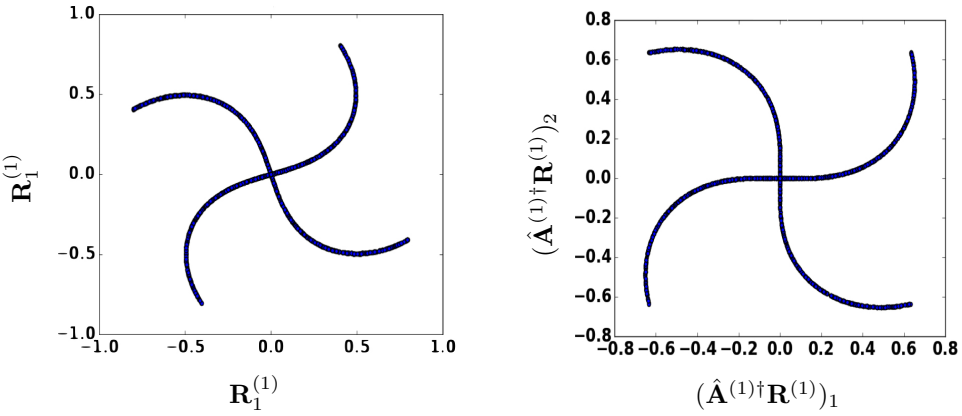


Figure VII.9 – Illustration of the AMCA step at the first iteration $l = 1$. *Left :* Residual $\mathbf{R}^{(1)}$ (note that at the first iteration, $\mathbf{R}^{(1)} = \mathbf{X}$) ; *Right :* Inversion of the linear model found by AMCA : the lowest amplitude samples are well aligned with the axes.

while being insensitive to the higher amplitude samples that are more affected by the non-linearities. Indeed, such highly non-linear samples are detrimental to most usual linear sparse BSS algorithms, as i) they behave as partially correlated samples (i.e. samples for which multiple sources are simultaneously active), while many linear BSS algorithms assume disjoint supports of the sources ; ii) they correspond to high amplitude samples, which hinder the use of the Morphological Diversity hypothesis ([Starck *et al.* 2010], *cf.* Chapter III-C.4.2), stating that the highest amplitude samples are the most discriminative for BSS.
Due to these two issues, we propose to use the Adaptive Morphological Component Analysis (AMCA - [Bobin *et al.* 2015], *cf.* Chap. III-C.5) algorithm, which enables a linear separation of sources having samples with both partial correlations and large amplitudes. In the case of non-linear mixings, the weights $\mathbf{M}$ in AMCA enable to discard the samples with high amplitudes which are the most affected by the non-linearities, because these are considered as partial correlations. At iteration l, the algorithm is thus able to find a good linear model $\hat{\mathbf{A}}^{(l)}$ of the lowest amplitude samples of $\mathbf{R}^{(l)}$.

B.5 Non-linear step : computing $\mathbf{R}^{(l+1)}$

B.5.1 Issues for $\mathbf{R}^{(l+1)}$ computation

The goal of the selection function is to create a new dataset $\mathbf{R}^{(l+1)}$ containing only the samples that are not well explained by the current linear-by-part $\hat{\mathbf{A}}^{(1..l)}$ model, thus paving the way for the next iteration.

For selecting only such contributions, there are two issues :

i) *Determine which samples are well separated by the current linear-by-part model*;

ii) *Actually compute a new residual* $\mathbf{R}^{(l+1)}$ *containing the currently badly separated samples*, which is done through shrinking their amplitude towards zero.

Solving these two issues is however non-trivial. We could have a fully-sequential naive approach, in which we would re-use at iteration l the previous residual $\mathbf{R}^{(l)}$ to compute the new one $\mathbf{R}^{(l+1)}$ (this naive approach is illustrated in Fig. VII.5). Issue i) would then be solved by inverting the current linear model found by AMCA, which would align the lowest amplitude samples of $\mathbf{R}^{(l)}$ with the axes (*cf.* Fig. VII.9, where the inversion is performed in the right plot). The badly estimated samples would then be the ones far from the axes, making their selection quite straightforward (*cf.* Fig. VII.6 left : the badly estimated samples are the ones outside the red square).
The difficulty with such a naive approach however arises when attempting to solve issue ii). Indeed, for each sample the amplitude shrinkage must be performed in the direction of the axis *corresponding to the source associated to the sample manifold*. For low and middle amplitude samples, this is not a major difficulty : since these are supposed to be well or at least decently unmixed respectively, we can quite reliably assume to which source they belong, and thus in the direction of which axis to shrink. For high amplitude samples, things are however completely different as we have almost no clue concerning the unmixing. Consequently, these can be shrunk in parallel to wrong axes. While this is not a direct issue in the following iteration $l+1$, during which the linear BSS step will aim only at unmixing the lowest amplitude samples of $\mathbf{R}^{(l+1)}$, this will definitely impact the algorithm through error propagation after several iterations, as $\mathbf{R}^{(l+1)}$ will be used to compute $\mathbf{R}^{(l+2)}$, $\mathbf{R}^{(l+3)}$...
In Figure VII.10, we illustrate such a shrinking issue with an example. To determine in which direction each sample must be shrinked, we use an angular criterion ($\theta = \pi/4$) to associate the samples with the closest axis. In the left plot, the samples in red are associated with the horizontal axis and the samples in yellow with the vertical axis. The shrinking is then operated in the corresponding direction. As can be seen in Figure VII.10 right, the subsequent residual is good for low amplitude samples but bad for higher amplitudes, for which the shrinkage has been performed in the direction of the wrong axis. This will create error propagation in a few iterations, when the algorithm will try to fit a linear model on these high amplitude samples.

B.5.2 Selection Function : proposed solution for computing $\mathbf{R}^{(l+1)}$

To bypass the error propagation issue that would appear in the previous naive fully sequential approach, we introduce at each iteration a preliminary step, that is first presented. The whole solution to issue i) and ii) is then detailed.

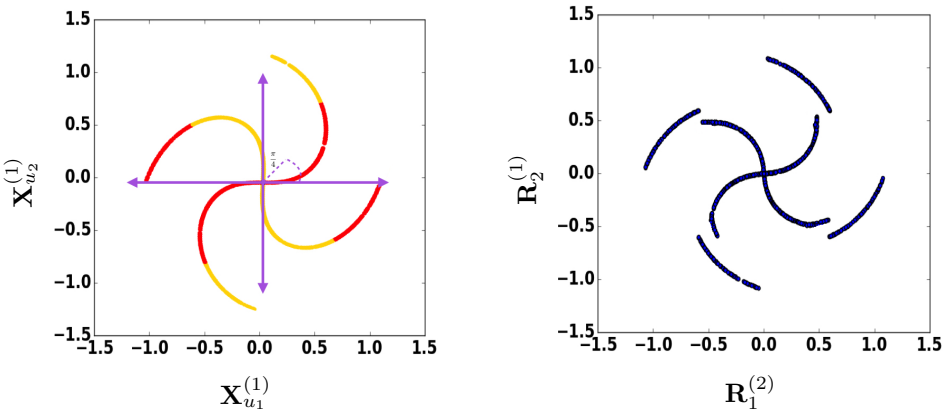


Figure VII.10 – Example of the residual computation difficulty, which is illustrated when working on more difficult examples than the one of Fig. VII.3. *Left :* Data obtained after the inversion of AMCA linear model. To know in which direction to shrink each sample, we associate each sample to the closest axis. Red corresponds to the samples associated with the horizontal axis and yellow to the vertical axis. This method works well for low amplitude samples, that are well unmixed by the linear step, but not for high amplitude ones. *Right :* Residual resulting from a shrinking of the left plot data with the directions corresponding to the colors. While the residual is relevant for low amplitudes, enabling to work on it at iteration l+1, it is bad for the extremities of the manifolds, potentially creating errors in the following iterations.

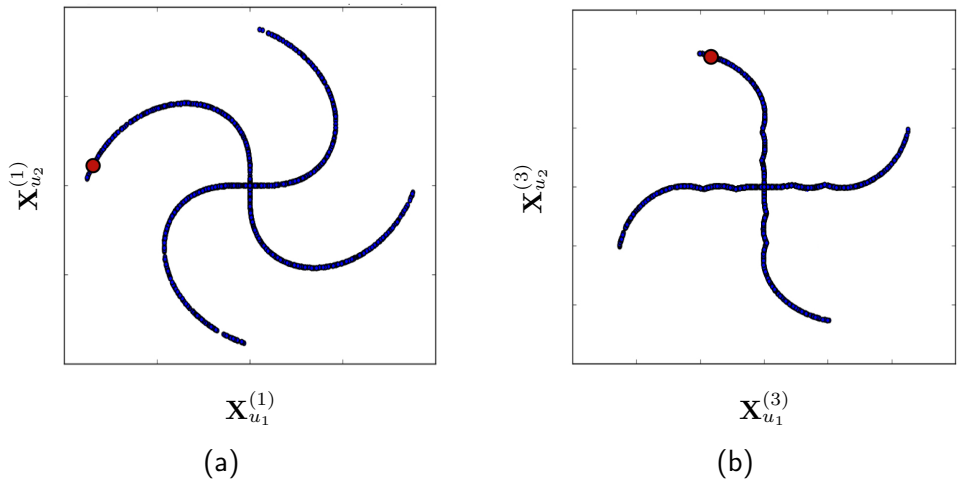


Figure VII.11 – Unfolded manifolds at the iteration $l = 1$ *(left)* and $l = 3$ *(right)*, showing the interest of such an unfolding to enhance the first guesses about the unmixing. The sample in red can be seen to lie much closer to the good axis after the $l = 3$ iterations, justifying the simple use of an angular distance to the different axes to make a decent first guess in the residual $\mathbf{R}^{(l+1)}$ computation.

Preliminary step : unfolding the manifolds. Instead of working directly on the previous residual, at each iteration l the algorithm starts back from the raw data $\mathbf{X}$ and *unfolds the manifolds* using all the previously computed linear models $\hat{\mathbf{A}}^{(1..l)}$ and $\tau^{(1..l-1)}$ (*cf.* Fig. VII.11 for an illustration of the unfolding at iteration $l = 1$ and $l = 3$). The unrolled data is denoted $\mathbf{X}_u^{(l)}$. This approach has two advantages [2] :

— Less error propagation occurs in the iterative process, as at each iteration we start back from the *raw* data $\mathbf{X}$, instead of $\mathbf{R}^{(l)}$;

— The contrast between the sources for high amplitude samples is increased, which makes that the separation is easier. For instance, in Fig. VII.11, the red dot is naturally made closer to the good axis after $l = 3$ iterations through the re-use of all the previously computed linear models.

Solution to problem i) StackedAMCA uses for each sample of $\mathbf{X}_u^{(l)}$ the distance (or more specifically the angle) to the axes. If such a distance is small enough, it is assumed that the sample is well separated by the current linear-by-part model. For each source i, we denote $\tau_i^{(l)}$ the maximum amplitude of the samples close [3] enough to the axis of i. In practice, the choice of the $\tau_i^{(l)}$ is made much more robust by using a clustering method based on the amplitude of the samples and enabling to discard

2. In addition to these two advantages, the unrolled $\mathbf{X}_u^{(l)}$ is also used to correct possible permutations between the different linear models $\hat{\mathbf{A}}^{(1..l)}$, *cf.* Section B.6

3. Such a closeness criterion depends on the manifold regularity, and is one of the few algorithm hyper-parameters to be tuned by the user.

potential outliers or samples that would be too much affected by noise .

Solution to problem ii) $\mathbf{X}_u^{(l)}$ can then be shrunk using the values $\tau_{1..n}^{(l)}$ to obtain the residual $\mathbf{R}^{(l+1)}$. For high amplitude samples, as evoked above the current unmixing is bad and we thus do not know in which direction to threshold and which threshold $\tau_i^{(l)}, i \in [1..n]$ to use. Therefore, for these we need to resort to a first guess based on their angular distance to each axis (similarly to Fig. VII.10). While this potentially introduces temporary errors, these are in practice strongly limited due to the two advantages of using the unfolded data $\mathbf{X}_u^{(l)}$ instead of merely working sequentially and re-using the previous residual $\mathbf{R}^{(l)}$.

B.6 Enhancements

In this subsection, we shortly detail several major enhancements of StackedAMCA that makes it much more robust in practice and enabling to work on more difficult datasets $\mathbf{X}$.

— *Correction of the permutations between the linear models* $\hat{\mathbf{A}}^{(1..l)}$ *:*
The issue is here due to the permutation indeterminacy in usual *linear* BSS. In our approach, multiple permutations between the linear models $\hat{\mathbf{A}}^{(1..l)}$ could create a remixing between the sources if they were not properly corrected.
In StackedAMCA, the linear model $\hat{\mathbf{A}}^{(l-1)}$ found in the previous iteration is used to initialize AMCA at iteration l. Thus, if the mixing $\mathbf{f}^*$ is regular enough, one can hope that the permutations will not be an issue as the output of AMCA will resemble the input. However, we can introduce a much better permutation correction through re-using the unrolled $\mathbf{X}_u^{(l)}$. To do so, we compare after the AMCA step the angle with the axes of the samples that are well linearly separated (i.e. the ones in the red square of Fig. VII.6) to the angle of the corresponding samples with the axes in the unrolled manifold of the previous iteration $\mathbf{X}_u^{(l-1)}$. The permutation correction is then done by minimizing the discrepancy. Note that such a correction is only made possible due to the regularity hypotheses detailed in Section E. It has furthermore been extensively experimentally tested, showing good practical results as confirmed in Section D.

— *Determining the sign of the samples on the manifold :*
In Sec. B.5, we explained how the *direction* in which to threshold was determined. A simple approach to determine the *way* (*e.g.* from right to left for the samples in the upper right quadrant of Fig. VII.10 but from left to right for the ones in the lower left quadrant) in which to shrink would be to use the sample signs. However, this approach would lead to errors for mixing highly deviating from linearity : in these, the manifolds could cross the axes, making the source sample signs in $\mathbb{R}^n$ to differ from the ones *on the manifold*. Therefore, in practice we compute the sign on the manifold of each sample : the process is similar to the one for attributing a source to each sample (*cf.*

solution to problem ii) in the residual computation), but in addition we also attribute a sign.

— *Handling non-negative data :*
Several extensions of NMF to non-linear mixtures have been proposed. In these, the sources are assumed to have non-negative coefficients. Examples of applications are hyperspectral imaging [Meganem *et al.* 2011,Eches & Guillaume 2013, Meganem *et al.* 2014] or show-through removal [Liu & Wang 2013]. A possible extension of StackedAMCA is therefore to handle properly the case of non-linear sparse non-negative sources (for realistic examples of such sources, see e.g. [Rapin 2014]). Due to the linear-by-part structure of the algorithm, such an enhancement is quite straightforward as it mostly amounts to transform the sparse *linear* BSS step into a NMF sparse *linear* one, making it possible to re-use all the corresponding literature (see *e.g.* [Rapin *et al.* 2013, Rapin *et al.* 2014]). Therefore, instead of looking for a minimizer of Eq. (VII.4), AMCA will here aim at minimizing :

$$\min_{\hat{\mathbf{A}}^{(l)},\hat{\mathbf{S}}} \frac{1}{2}\mathtt{Tr}[(\mathbf{R}^{(l)} - \hat{\mathbf{A}}^{(l)}\hat{\mathbf{S}})\mathbf{M}(\mathbf{R}^{(l)} - \hat{\mathbf{A}}^{(l)}\hat{\mathbf{S}})^T] + \sum_{i=1}^{n} \left\| \lambda_i \odot \hat{\mathbf{S}}_i \right\|_1 + \iota_{\|\hat{\mathbf{A}}^i\|_{\ell_2}=1}(\hat{\mathbf{A}}) + \iota_{\cdot\geqslant 0}(\hat{\mathbf{S}}) \tag{VII.5}$$

Where the last term enforces an elementwise non-negativity of the coefficients of $\hat{\mathbf{S}}$. The rest of the algorithm is the same, except that we only focus on the positive branches of the manifolds when determining the sign of the samples on the manifold [4].

— *Enabling different regularities for each manifold :*
In the current form of the algorithm, all the manifolds must have the same regularity. That is, their curvature radius must be similar. Otherwise, the most regular manifolds would need less linear models to be well estimated than the other ones. Thus, in the last iterations less sources would need to be separated as the whole most regular manifolds would already have been handled (*cf.* Fig. VII.12 for a concrete example). We propose to bypass this issue by limiting at each iteration l the amplitude of the thresholds $\tau_{1..n}^{(l)}$ and rather use $\tau_{1..n}^{\prime(l)} = \min_{i\in[1,n]}(\tau_i^{(l)})$ (*cf.* Fig. VII.12 - right for the resulting threshold choice). As such, the condition of equal regularity is transformed into an equal length condition (*cf.* more about this limitation in Sec. B.6).

C Neural Network Interpretation

StackedAMCA can be interpreted as the multilayer neural network of Fig. VII.13. The different network layers correspond to the iterations l and each layer computes a linear approximation of some part of the data.

4. Note that in this case, the symmetry of the mixing is not anymore required.

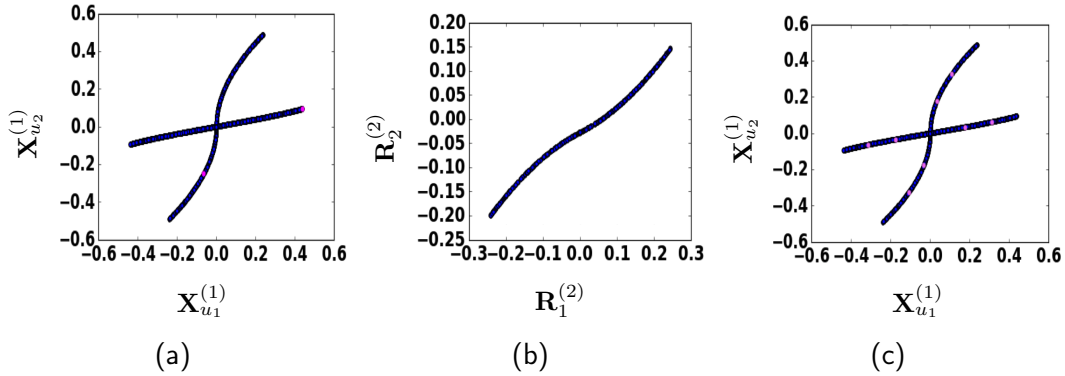


Figure VII.12 – Regularity issue. *Left :* The two manifolds do not have the same regularity, as the horizontal one is almost linear. The thresholds found by StackedAMCA without any enhancement after the first iteration are displayed in magenta. Therefore, the horizontal manifold is fully handled during the first iteration, while the vertical manifold would need at least two linear models to be well estimated. *Middle :* Residual after the first iteration : only one manifold is left, making the search for two sources irrelevant ; *Right :* Proposed solution for the threshold choice, to be compared with the left plot.

Slightly altering the notations and writing $\mathbf{W}^{(l)} = \hat{\mathbf{A}}^{(l)\dagger}$, each neuron layer corresponds to the estimate $\hat{\mathbf{A}}^{(l)}$ yielded by the linear BSS step. In Fig. VII.13, the non-linear step corresponds to the residual computation. Due to the thresholding, this step has similarities with classical non-linearities in neural networks such as the Rectified Linear Unit (ReLU – [Maas *et al.* 2013]). This similarity is in particular stronger for the case of sparse non-negative sources described in Sec. B.6, for which a ReLU is truly used to compute the residual. When no non-negativity is enforced, the soft-thresholding operator used resembles a symmetrical ReLU. The network thus roughly possesses the classical alternating between neuron layers and non-linearities. We further need skip connections to complete the transcription of the algorithm. In particular, these enable to re-use $\mathbf{X}$ directly, reducing the error propagations and improving the results similarly as usual skip connections [Huang *et al.* 2017]).

However, contrary to many learning processes the layers are trained one-by-one, each of them minimizing the cost function of AMCA. This would correspond to a greedy training. A global refinement step could however be added. Furthermore, while the non-linear step encompasses a classical non-linearity, it first requires an unfolding of the manifolds, which is different from usual neural networks. Another difference is also that the thresholds $\tau_{1..n}^{(1..l)}$ are not learnt by minimizing a global cost function through backpropagation but roughly speaking directly from the data itself.

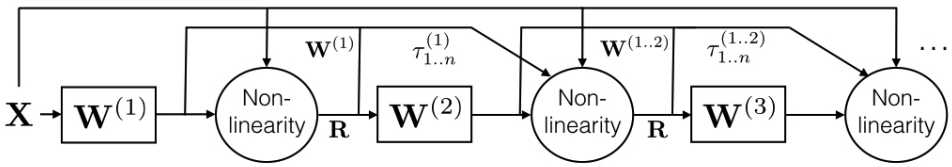


Figure VII.13 – StackedAMCA as neural network

D Experiments

D.1 Metrics

Before demonstrating the relevance of the method with three experiments, we present the metrics used to assess the result quality, as these are slightly less usual than in the linear case. Due to the indeterminacy by the non-linear function $\mathbf{h}$ in non-linear BSS, it is important to differentiate measures about the reconstruction of the sources and their separation. The reconstruction metrics are the usual *linear ones* : the Mean Error (ME), or the more accurate SDR (capturing both the separation and reconstruction quality) [Vincent *et al.* 2006], or SAR (focusing only on the reconstruction).
Concerning the separation quality, a classical approach is to estimate $\mathbf{h}$ [Ehsandoust *et al.* 2017] by fitting a non-linear curve $\mathcal{P}$ to the 1D-manifold of the scatter plot of each estimated source $\hat{\mathbf{S}}_i$ as a function of the true one $\mathbf{S}_i^*$, and to look at the thickness of the manifold around $\mathcal{P}$. Here, we use for $\mathcal{P}$ a polynomial function of degree 20 and the thickness is measured by the :

— logarithmic median absolute distance to $\mathcal{P}$:

$$C_{med} = -10\log(\sum_{i=1}^{n} \underset{j}{\text{median}}(|\hat{\mathbf{S}}_i^j - \mathcal{P}(\mathbf{S}_i^{*j}))|) \tag{VII.6}$$

— logarithmic mean absolute distance to $\mathcal{P}$:

$$C_{mean} = -10\log(\sum_{i=1}^{n} \frac{1}{t}\sum_{j=1}^{t} |\hat{\mathbf{S}}_i^j - \mathcal{P}(\mathbf{S}_i^{*j})|) \tag{VII.7}$$

However, the results of these metrics are sensitive to the choice of $\mathcal{P}$. We thus propose to introduce for exactly sparse sources a new metric based on the *angular distance to the axes* :

$$C_{ang} = -10\log\left(\frac{1}{n(n-1)}\sum_{i=1}^{n}\left(\sum_{\substack{i'=1\\ i'\neq i}}^{n} 1 - \frac{1}{\#Z}\sum_{j\in Z}\frac{\mathbf{S}_{i'}^{*j}}{\sqrt{\hat{\mathbf{S}}_i^{j^2} + \mathbf{S}_{i'}^{*j^2}}}\right)\right) \tag{VII.8}$$

where $Z = \{j \in [1,t] | \mathbf{S}_{i'}^j \neq 0\}$.
Lastly, for the first linear-by-part experiment, it is also possible to use the mixing criterion C_A [Bobin *et al.* 2008], as true underlying linear models exist.

D.2 Linear-By-Part Mixing

The objective of the first experiment is to study StackedAMCA main mechanisms and test it in a ideal setting where it should be able to separate and reconstruct the sources well. The mixing is indeed linear-by-part, thus perfectly matching the unmixing process of StackedAMCA. More specifically, they have $t = 10000$ samples, with disjoints support and a sparsity level of $p = 10\%$. There is $m = n = 2$ observations, obtained with a linear-by-part $\mathbf{f}^*$ (no noise $\mathbf{N}$ is added). Each part is an orthogonal $\mathbf{A}^{*(l)}$ matrix. The data $\mathbf{X}$ is shown in Fig. VII.14 (while the mixing might seem simplistic, it however deviates much from the linearity). Since both the true mixing matrices $\mathbf{A}^{*(1..l)}$ and the optimal thresholds are known, it is possible to assess the quality of their estimation by StackedAMCA.
Qualitatively and as shown in Figure VII.15 - left, the data reconstruction is almost perfect (except for 2 outliers – likely to come from a thresholding in the wrong direction). This however does not guarantee the separation of the sources. Fig. VII.15-right therefore also displays the scatter plot of $\mathbf{X}$ with colors corresponding to the estimated different sources : each manifold is correctly labeled with only one source. Furthermore, the thresholds $\tau_{1..n}^{(1..l)}$ in violet seem to be well estimated (the cumulated error is 4.9×10^{-3}).
Quantitively, Fig. VII.15 displays the evolution of C_A and the SDR as a function of the iterations. The high values of C_A confirms that the separation is good, while the decent SDR shows that despite the non-linear setting, the source reconstruction is correct. The high quality results for the first iteration indicates as expected that AMCA is robust enough to discard the highly non-linear high amplitude samples. The decrease of both C_A and the SDR with the iteration number is not strictly monotonic, probably due to the fact that some errors done at a given layer l can be compensated at the following layer (*e.g.* by still finding the good thresholds $\tau_{1..n}^{(l+1)}$).

D.3 Star Mixing and Comparison to Other Methods

We now compare the results of our algorithm to other existing ones on a much more complicated experiment. Only a few algorithms for non-linear BSS are open source, and we mostly found three of them : MISEP [Almeida 2003], NFA [Honkela *et al.* 2007] and ANICA [Brakel & Bengio 2017]. The experiment itself is inspired from [Ehsandoust *et al.* 2016] but is made much more difficult due to a) the presence of noise $\mathbf{N}$ with a SNR = 30 dB ; b) a high number of sources (the original experiment being restricted to $n = 2$).
The sources follow a Bernoulli-Uniform distribution, $p = 10\%$ of the $t = 9500$ samples being non-zeros and drawn according to a uniform distribution in [-0.5, 0.5]. We further ensure that the supports of the $n = 6$ sources are disjoint and there are $m = 6$ observations. These are computed recursively through the application 15 times on each element $j \in [1, t]$ of a mixing of the form $\mathbf{U}_d^j = \cos(\alpha(j))\mathbf{U}_d^j - \sin(\alpha(j))\mathbf{U}_f^j$ and $\mathbf{U}_f^j = \sin(\alpha(j))\mathbf{U}_d^j + \cos(\alpha(j))\mathbf{U}_f^j$, with $\alpha(j) = \frac{\pi}{2}(1 - \sqrt{\mathbf{U}_d^{j^2} + \mathbf{U}_f^{j^2}})$, for random $d, f \in [1, 6]$ and with the initial $\mathbf{U}_d = \mathbf{S}_d^*$ and $\mathbf{U}_f = \mathbf{S}_f^*$. The data $\mathbf{X}$ is

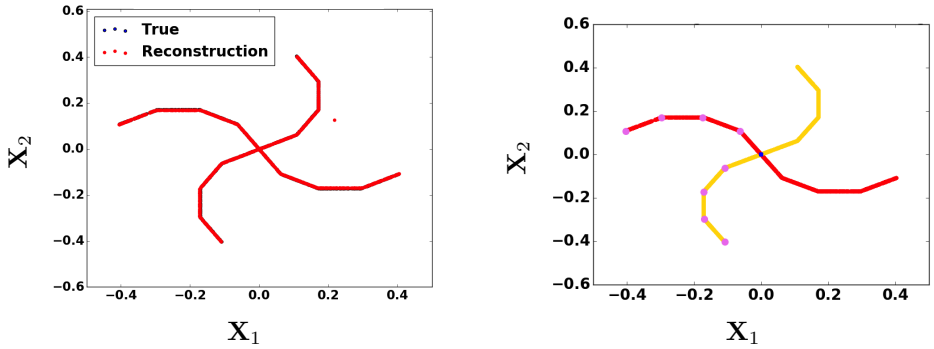


Figure VII.14 – *Left :* Reconstruction of the data from the model estimated by StackedAMCA, superimposed on the true data $\mathbf{X}$; *Right :* True data, with the colors coming from the demixing : red corresponds to source one and yellow to source two. Points in violet correspond to the samples used to compute the thresholds $\tau_{1..n}^{(1..l)}$.

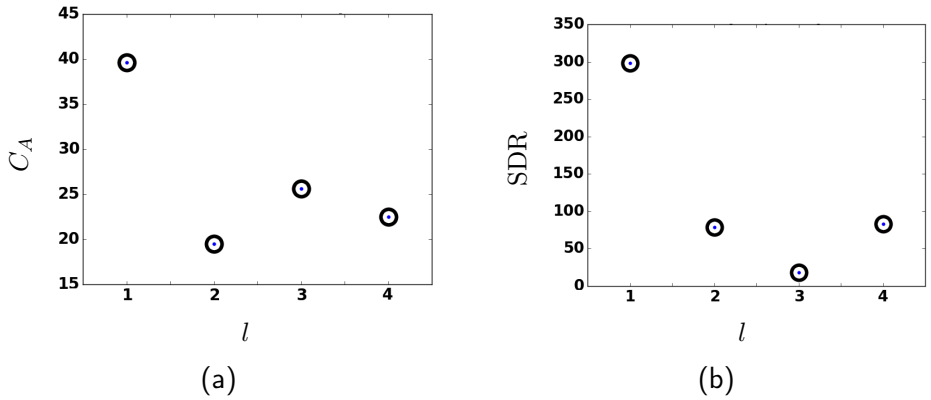


Figure VII.15 – *Left :* C_A as a function of the iteration l ; *Right :* SDR as a function of l.

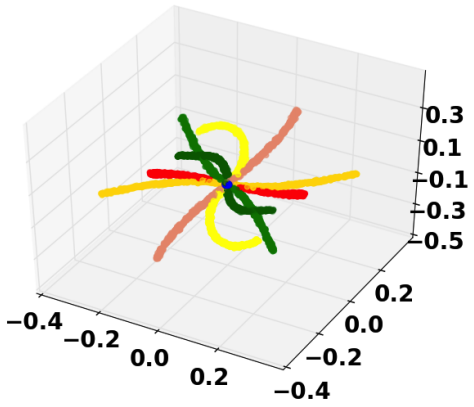


Figure VII.16 – A 3-dimensional projection of a non-linear mixing with $n = m = 6$ sparse sources.

taken equal to $\mathbf{U}$ after the 15 iterations plus the noise $\mathbf{N}$. A 3-dimensional projection of the mixing is displayed in Fig. VII.16.

The different separation and reconstruction metrics are displayed in Table VII.1 and VII.2. The corresponding results are shown in Fig. VII.17, where the scatter plot of one estimated source is drawn as a function of the true one for each method. First, it seems that neither ANICA nor NFA truly separate the sources (concerning ANICA, the results seem however to improve when no noise is added). This bad separation is translated into plots that do not resemble a 1D-manifold. It is in particular visible by looking at the estimated values for $\mathbf{S}_1^* = 0$: many samples are such that $\hat{\mathbf{S}}_1 \neq 0$, which corresponds to high interferences and a leakage of the other sources in $\hat{\mathbf{S}}_1$. Although such bad unmixings could come from our lack of familiarity with the parameter tuning of these methods, it is possible that the regularization introduced by the network structure for ANICA and the Bayesian setting for NFA is not sufficient to enable the separation of the sources (since the independence is not either, *cf.* Sec A). On the contrary, MISEP separates the sources well, probably due to a good implicit regularization. StackedAMCA displays a very small number of outliers (a single one in Fig. VII.17). While these probably come from small remaining error propagations due to the temporary first guess in the residual computation (*cf.* Sec. B), their small numbers shows the interest of the manifold unfolding from the raw data $\mathbf{X}$ at each iteration. Therefore, looking at the various metrics the sources are still in general much better separated by StackedAMCA : C_{mean} improves by almost 7 dB compared to MISEP. The good separation of StackedAMCA is confirmed by the best C_{ang}.
Second, MISEP does not reconstruct well the sources as StackedAMCA does and Fig. VII.17 clearly indicates that it did not invert the non-linearity $\mathbf{h}$. On the contrary, the scatter plot yielded by stackedAMCA resembles the identity and the good ME (and best SAR, despite the outliers) indicates that the algorithm structure was sufficient to regularize well the reconstruction problem. Some non-linearities $\mathbf{f}^*$

Tableau VII.1 – Separation quality of 4 methods : StackedAMCA, MISEP, NFA and ANICA. The curve $\mathcal{P}$ fitted to the scatter plots displayed in Fig. VII.17 is chosen as a polynomial function of degree 20.

Method	C_{med}	C_{mean}	C_{ang}
StackedAMCA	**41.4**	**28.1**	**41.2**
MISEP	22.6	21.8	20.6
NFA	15.8	10.8	11.1
ANICA	17.9	11.7	4.0

Tableau VII.2 – Reconstruction quality of StackedAMCA, MISEP, NFA and ANICA.

Method	-10log(ME)	SAR
StackedAMCA	**26.4**	**19.4**
MISEP	21.7	17.2
NFA	12.9	8.1
ANICA	0.52	−18.8

for which StackedAMCA is able to perform such a good reconstruction are characterized in Sec. E.

D.4 Experiment without source reconstruction

The objective is here to try stackedAMCA on a mixing for which it should not be able to reconstruct the sources well.
The experimental setting is relatively similar to the one of the previous subsection : the sources follow a Bernoulli-Uniform distribution, $p = 10\%$ of the $t = 9\ 500$ samples being non-zeros and drawn according to a uniform distribution in [-0.65, 0.65]. We further ensure that the supports of the $n = 6$ sources are disjoint and there is $m = 6$ observations. The mixing is computed recursively through the application 9 times on each element $j \in [1, t]$ of $\mathbf{U}_d^j = \cos(\alpha(j)^2)\exp(\alpha(j))\mathbf{U}_d^j/2 - \sin(\alpha(j)^2)\mathbf{U}_f^j$ and $\mathbf{U}_f^j = \sin(\alpha(j)^2)\exp(\alpha(j))\mathbf{U}_d^j/2 + \cos(\alpha(j)^2)\mathbf{U}_f^j$, where all the variables where defined in the previous subsection. The most important difference is due to the presence of the $\exp(\alpha(j))/2$ factor, making that the first column of the corresponding mixing matrix is not anymore unitary, and thus stackedAMCA cannot be expected to perform source reconstruction (*cf.* Sec. E). Furthermore, the presence of this term on the first column *only* makes that the different manifolds do not have anymore the same regularity, enabling also to test StackedAMCA in this setting. A 3D-projection of the mixing is displayed in Fig. VII.18.
Concerning the *separation* quality presented in Table VII.3, all the methods obtain

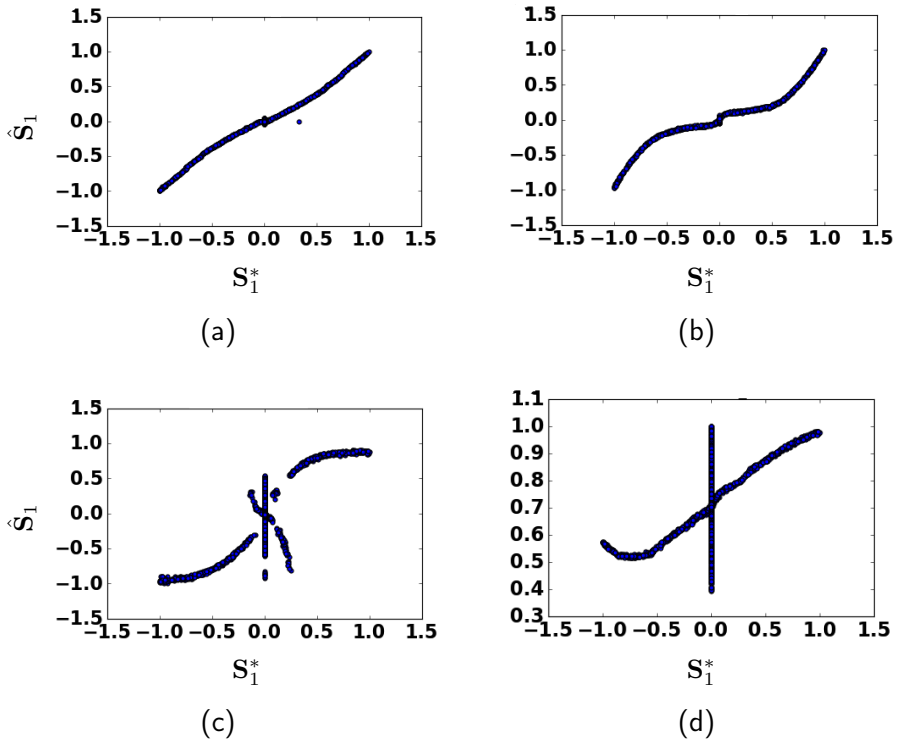


Figure VII.17 – Scatter plot of one estimated source as a function of the true source : *a)* StackedAMCA ; *b)* MISEP ; *c)* NFA ; *d)* ANICA.

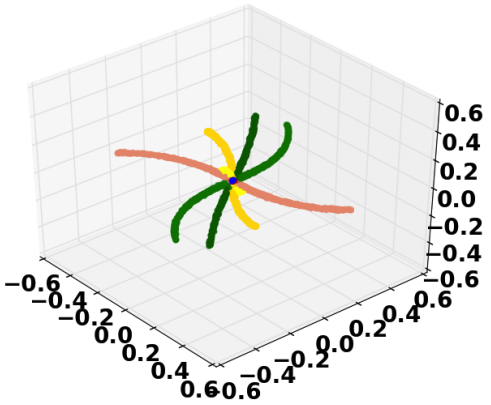


Figure VII.18 – A 3-dimensional projection of a non-linear mixing with $n = m = 6$ sparse sources.

Tableau VII.3 – Separation quality of 4 methods : StackedAMCA, MISEP, NFA and ANICA. The curve $\mathcal{P}$ fitted to the scatter plots displayed in Fig. VII.19 is chosen as a polynomial function of degree 20.

Method	C_{med}	C_{mean}	C_{ang}
StackedAMCA	**37.4**	**27.3**	**38.1**
MISEP	23.0	22.2	21.2
NFA	18.4	11.4	12.2
ANICA	16.7	12.0	4.14

slightly worst results compared to the previous subsection, as the problem is more complicated due to the different manifold regularities. StackedAMCA however still obtains the best ones, which are good as testified by Figure VII.19. Therefore, as expected the algorithm is still able to perform the separation.
Concerning the *reconstruction*, it can already be seen in Figure VII.19 that the non-linearity is not well inverted. This is confirmed by a decrease of 8.4 dB of the SAR and a loss of 3.1 dB for the ME (to be compared with a loss of only 0.8 dB for C_{mean}).
In conclusion, even in this experimental setting where the source reconstruction cannot be expected from StackedAMCA and is indeed bad in practice, the algorithm is still able to separate the sources well.

E Discussion : required Hypotheses for StackedAMCA and possible enhancements

In this section, we discuss some required hypotheses for StackedAMCA to work. While such hypotheses might seem restrictive, we however emphasize again the

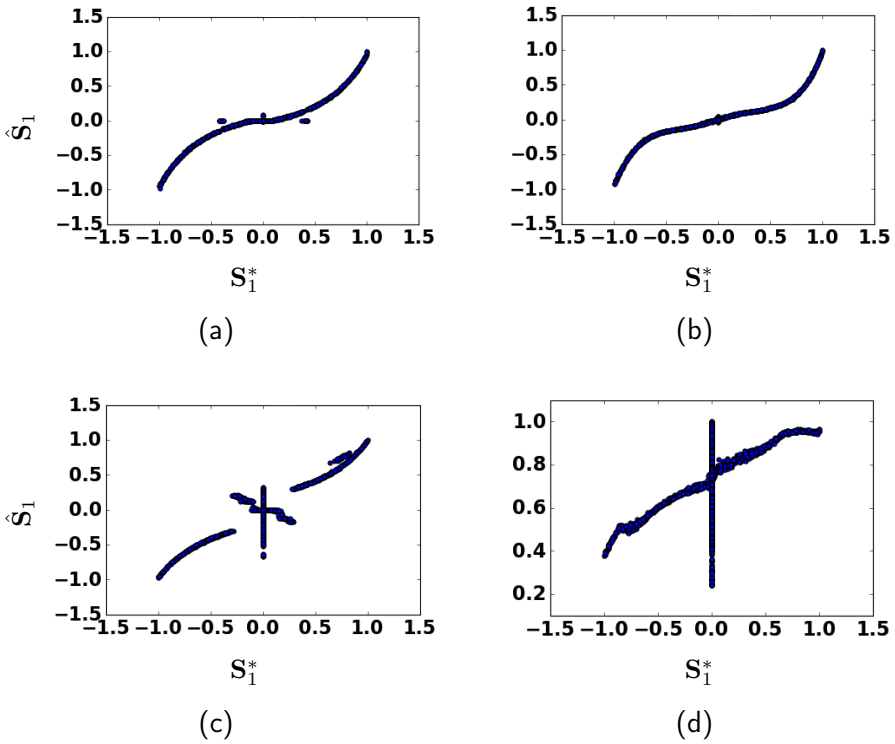


Figure VII.19 – Scatter plot of one estimated source as a function of the true source. In contrast to Fig. VII.17, the scatter plot of the source obtained by Stacked AMCA does not resemble identity : *a)* StackedAMCA ; *b)* MISEP ; *c)* NFA ; *d)* ANICA.

Tableau VII.4 – Reconstruction quality of StackedAMCA, MISEP, NFA and ANICA.

Method	-10log(ME)	SAR
StackedAMCA	**23.3**	**11.0**
MISEP	20.9	7.46
NFA	13.7	5.51
ANICA	2.09	−16.9

difficulty of the problem at hand and the fact that most of the previous works based on ICA and general mixings did not provide required conditions under which the source separation can be hoped [Almeida 2003, Brakel & Bengio 2017]. In contrast, the geometric interpretation of StackedAMCA enables to perform such a discussion. Concerning the other works based on sparsity, the works making the hypotheses explicit also used strong ones (*cf. e.g.* [Puigt *et al.* 2012]).
Furthermore, most of the discussed conditions are either intrinsic to the problem or can be mitigated. Hypotheses E.1 and E.2 are intrinsic to sparse non-linear BSS based on clustering (at least, without any additional explicit priors), as well as E.5 (with the regularity potentially depending on the specific clustering algorithm used). While conditions E.3, E.4 and E.6 are specifically required by our algorithm, we propose exploratory paths for alleviating E.3 and E.6. Assumptions E.7 and E.8 must be seen as advantages of StackedAMCA over most other existing algorithms. Concerning E.7, most algorithms do not incorporate any noise in the mixing process. Thus, being able to deal with noise is a progress, even if this one should be limited to some extent. Concerning E.8, to the best of our knowledge StackedAMCA is the first non-linear BSS algorithm for which the source reconstruction of the sources can be hoped for a characterized class of non-trivial mixings.

E.1 Sparsity of the sources and disjoint supports

The sparsity of the sources $\mathbf{S}^*$ in the direct domain is assumed for regularizing problem (II.3). Such an assumption can for instance be verified in realistic experiments such as spectrometry.

We have furthermore assumed the supports of the sources to be disjoint. While this is not very realistic in practical cases, it seems difficult to bypass this condition as we only explore the span of $\mathbf{f}^*$ that the 1D-manifolds created by the sparse sources uncover. By the morphological diversity assumption, the points outside these manifolds are too rare to enable a proper estimation of $\mathbf{f}^*$ without either further conditions on the mixing (*e.g.* the separability over the different sources) or additional priors on the sources (*cf.* Fig. VII.20). Similarly to the results of [Hyvarinen & Morioka 2017], a promising prior could be to use potential temporal dependencies, which is left for future work.

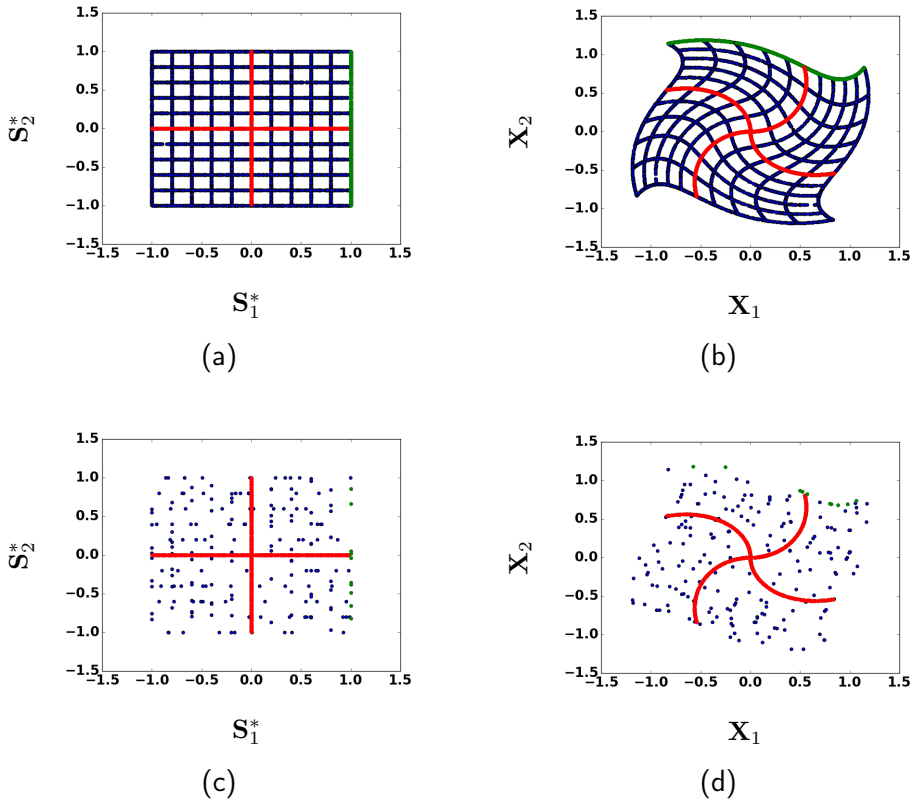


Figure VII.20 – Illustration of the difficulty to estimate the partially correlated samples in non-linear BSS. Note that with more than $n = 2$ sources, it is not even possible to determine which sources are active. *a)* Non-sparse sources : the samples in red are the only ones to correspond to disjoint supports of the sources. All the other ones are partially correlated ; *b)* A dataset corresponding to the mixing of the sources in a). Since the mixing is not separable, the manifolds in red are not sufficient to estimate the non-linear mixing for the samples in green (the green and red manifolds are not parallel). The non-linearity thus would need to be estimated using the information yielded by the green samples ; *c)* Sparse sources : the partially correlated samples are in much smaller number ; *d)* The number of green samples is now too small to estimate the mixing far from the disjoint support samples.

We however did some tests *without* disjoint supports. The samples with multiple active sources were badly separated but the estimation of the 1D-manifolds by StackedAMCA was not much perturbated, which is due to AMCA robustness to multiple active sources. For instance, re-doing the experiment of Section D.3 *without enforcing disjoint supports in the data creation*, we obtain similar results : $C_{med} = 42$ dB, $C_{mean} = 26$ dB, $C_{ang} = 37$ dB and -10log(ME) = 27 dB. Note that these metrics were computed on the disjoint samples only since we do not claim to handle well the joint ones, but only that these do not perturbate much the unmixing.

E.2 Density of the 1D-manifolds

For StackedAMCA to work, the manifolds must be continuous, which is to be related with the clustering nature of the algorithm. In particular, this entails that the sources must be dense enough within their support, to avoid any "gap" in the manifolds. The size of affordable gaps depends on the manifold regularity. Furthermore, for each linear BSS step a sufficient number of samples must be non-zero in order to have enough statistics for solving each linear sub-problem (*cf.* [Gribonval & Schnass 2010, Gribonval *et al.* 2015] for more detailed conditions concerning sparse linear BSS identifiability).

E.3 Symmetry of f* around the origin

StackedAMCA currently needs the symmetry of the mixing around the origin. Indeed, after the non-linear shrinkage step, the residual must be almost linear around zero. This might not be verified for non-symmetrical mixings (*cf.* Fig. VII.21 – the residual is not interpretable as a mixture with $n = 2$ sources). Such an assumption could in principle be leveraged. First, the data can be symmetrical around a different point as long as a preprocessing step is introduced to center it. Then, tackling non-symmetrical data could probably be dealt with by introducing non symmetrical non-linear steps (*i.e.* using a soft-thresholding function with two different thresholds for the positive and negative parts) and using the non-negativity constraint in the linear BSS step. As such, the positive and negative parts of each manifold could be treated separately, and aggregated at the end.

E.4 Well-conditioned linear sub-problems

Each linear sub-problem to be solved by StackedAMCA must be well-conditioned. That is, the corresponding underlying linear model $\mathbf{A}^{*(l)}$ must have a high enough condition number so that AMCA can find a good unmixing, which is to be linked with the classical performances of the algorithm in the linear case [Bobin *et al.* 2015]. Note that however the re-use of the linear model of the previous iteration $l-1$ as an initialization of AMCA can help it though re-using past knowledge to find a good linear model at the current iteration.
An extreme case of badly conditioned (and even colinear) sub-problem at the second

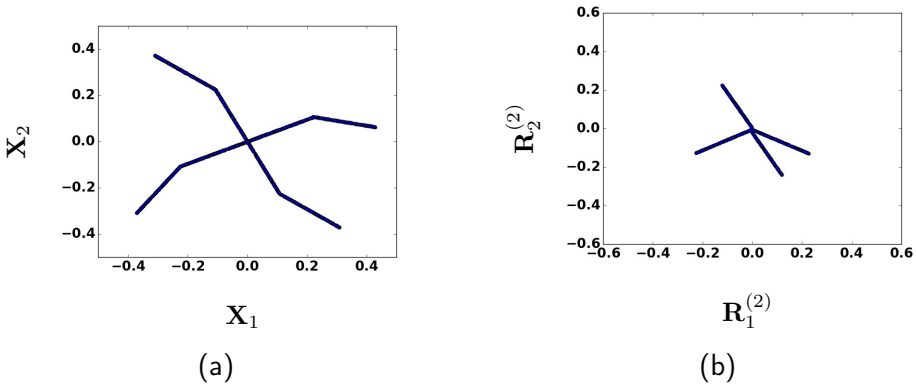


Figure VII.21 – Example of non-symmetric case. *Left :* Data $\mathbf{X}$. *Right :* corresponding residual $\mathbf{R}^{(2)}$. The non-symmetrical manifold cannot be tackled as a single source, as it is not linear in zero.

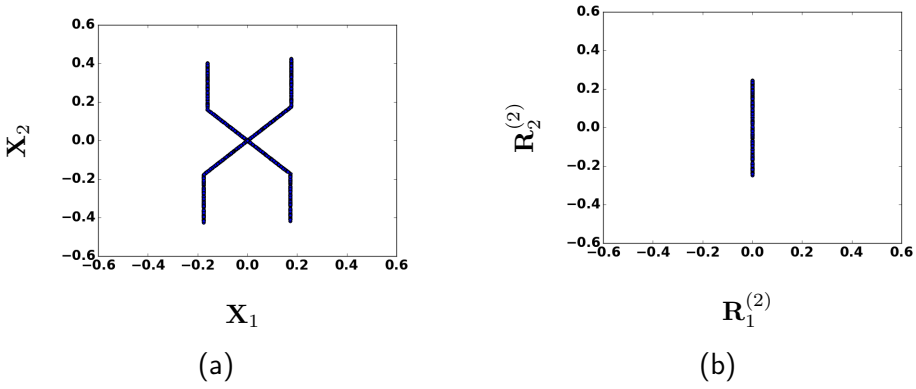


Figure VII.22 – Example of badly conditioned case. *Left* : Data $\mathbf{X}$; *Right :* Residual at iteration $l = 2$. The two manifolds are not anymore distinguishable.

iteration is given in Fig. VII.22. In this one, StackedAMCA cannot be expected to yield good results, as at the second iteration the residual is colinear.

E.5 Regularity of the Mixing $\mathbf{f}^*$

The mixing function $\mathbf{f}^*$ is assumed to be instantaneous, injective, and to depend on the sample amplitude (which is for instance the case with LQ mixtures [Hosseini & Deville 2003, Duarte *et al.* 2015, Deville & Duarte 2015]). We have furthermore assumed that $\mathbf{f}^*$ does not deviate too fast from linearity as a function of the amplitude. For differentiable curves, it mathematically means that at every point of the 1D-manifolds described by the mixing $\mathbf{X}$, the local curvature radius must be large enough. This condition is of primary importance to enable StackedAMCA to separate the sources and alleviate the issue of potential permutations between

layers (the permutation correction during the iterations is furthermore based on this assumption). For a similar reason, $\mathbf{f}^*$ must also be $\mathcal{L}$-Lipschitz with $\mathcal{L}$ small enough.

E.6 Same length for all the manifolds

All the manifolds must currently have the same length. Otherwise, one could be fully estimated in less iterations than the others. Thus, the residual would comprehend less sources in the last iterations than in the first ones.
A possible solution would be to re-estimate at each iteration the number of sources. While the estimation of such a number is a difficult question in BSS, the problem would here be made easier as we would only need to test a smaller number at each iteration. More specifically, at each iteration StackedAMCA could try reduce the number of sources to be estimated by checking if there are enough remaining samples for each manifold in the residual $\mathbf{R}^{(l)}$ to perform the separation.

E.7 Low noise

It is important to emphasize that in general, most non-linear BSS algorithm do not assume the mixing to be contaminated by noise (to the notable exception of [Honkela *et al.* 2007]). While in our approach the noise must be relatively low (*cf.* [Gribonval *et al.* 2015] for conditions on the noise concerning the separability of each linear sub-problems), our algorithm seems to be empirically quite robust as demonstrated in Section D. This is mainly due the use of AMCA, which is itself fairly robust [Bobin *et al.* 2015].

E.8 What Sources can StackedAMCA Reconstruct Well ?

In contrast to other methods, it is further possible to *characterize* at least one non-trivial type of mixings for which StackedAMCA approximately reconstructs the true sources up to a simple scaling and permutation indeterminacy. A sufficient condition (in addition to the ones required for separability) is that for each sample of the mixing indexed by j, the mixing $\mathbf{f}^*$ can be written as a product of an unitary matrix and the sources :

$$\mathcal{G} = \{\mathbf{f}^* : \mathbb{R}^{n\times t} \to \mathbb{R}^{m\times t} \| \forall j \in [1,t], \mathbf{X}^j = \mathbf{A}^*(\mathbf{S}^j)\ \mathbf{S}^{*j}, \mathbf{A}^*(\mathbf{S}^j) \in \mathcal{O}\} \tag{VII.9}$$

where $\mathcal{O}$ is the oblique set. The function $\mathbf{A}^*(.) : \mathbb{R}^n \to \mathbb{R}^{m\times n}$ is potentially non-linearly depending on $\mathbf{S}^*$. Since in AMCA the scale of the matrices $\hat{\mathbf{A}}^{(l)}$ is fixed to 1, if $\mathbf{f}^* \in \mathcal{G}$ there is no ambiguity left for the scale of each layer. Due to the regularity assumption, it will then be possible to backproject linearly for each layer the manifold on the axes with small errors and get an *approximate* reconstruction.

F Conclusion

We have introduced in this Chapter StackedAMCA, a new algorithm tackling the sparse non-linear BSS problem. Based on a new stacked sparse BSS approach,

this method enables to sequentially compute a linear-by-part approximation of the underlying non-linearities. Each linear part is estimated by a robust linear BSS algorithm step, which is followed by a non-linear step . The non-linear step enables to work on increasingly higher non-linearities and is itself composed of an unfolding of the source 1D-manifolds and then a thresholding. We show the relevance of StackedAMCA compared to other state-of-art methods. Beyond separating the sources, in some experiments the algorithm is also able to reconstruct them well despite a severely ill-posed problem. A discussion of the required hypotheses for StackedAMCA to work is furthermore proposed, as well as a characterization of some datasets for which it should be able to reconstruct the sources well.

Chapitre VIII

Conclusion and perspectives

Although sparse Blind Source Separation has well established over the last two decades its capacity to extract meaningful information from multivalued data, most algorithms do not handle properly *large-scale* problems. The objective of this PhD was to tackle such an issue.

In Chapter IV, we aimed at introducing a robust sparse BSS method using modern-art optimization frameworks such as PALM. While the difficulties to be tackled by such an approach were first discussed, the proposed method enables an automatic choice of the regularization parameters and an increased reliability with regards to the initialization, potentially obviating any relaunch of PALM that would be precluded in the large-scale setting.
Chapter V handled mixings comprehending a high number of sources. This problem is especially challenging, as many usual algorithms suffer from decreased separation qualities in such a setting. The introduced bGMCA, based on intermediate-size block coordinate optimization, enables to maintain high quality results while reducing the computational cost.
The case of large-scale datasets is dealt with in Chapter VI, in which we introduce mini-batches in GMCA, improving its scalability both in terms of computation time and memory burden. Strikingly, such a method can also in some settings improve the separation quality over full batch ones.
Lastly, Chapter VII is an extension of the previous work to the non-linear BSS problem. The proposed StackedAMCA method constructs a linear-by-part approximation of the underlying non-linearities. Numerical experiments highlight its capacity to separate the sources, as well as to reconstruct them in some settings.

The methods have been extensively tested, both on simulated and realistic experiments. As such, they are shown to work well in a wide range of settings. To foster reproducible research, the codes will be made available online at `http://www.cosmostat.org/software/gmcalab`.

Perspectives

The work presented in this thesis have raised several questions, some of which are still open and might be the subject of future research.

Algorithmic framework and regularization parameter choice

The work presented in Chapter. IV could be prolongated. Although the proposed 2-step approach was shown to work well in many settings, several pathways could be explored :

— Study other optimization schemes. While the use of BCD is shortly discussed in the Appendix E, more work could be dedicated to determine potential differences with PALM, which might appear in more difficult settings than the studied ones (*e.g.* with Poisson noise, more sources...).

— Use alternative regularization parameter choices than the MAD. In particular, a proper extension of the Stein Unbiased Risk Estimator (SURE) to the sparse BSS problem could be envisioned, as it has led to good results in other inverse problems [Eldar 2009, Giryes *et al.* 2011]. Preliminary results (in which an already existing SURE method is applied without any further adaptation to the sparse BSS problem) are shown in Appendix E ;

— Beyond the regularization parameter choice, use machine learning tools to enhance the separation.

Use of intermediate block-sizes in sparse BSS

The introduction of bGMCA leads to several questions :

— For the moment, the block choice at each iteration is still relatively unexplored. While we proposed three strategies in Chapter V, they would deserve more work. Finding a good (or even optimal) bloc size is still an open question ;

— Giving more mathematical grounding to the approach. In particular, links with recent works on dropout in matrix factorization [Cavazza *et al.* 2017, Mianjy *et al.* 2018] can be highlighted. In these, it is shown that using dropout introduces an implicit regularization promoting low-rank solutions. It has however to be emphasized that bGMCA is quite different from the aforementioned works (in particular, due to the fact that it works on a residual and it is not based on gradient descent but rather on least-squares) ;

— Estimation of the number of sources : such an issue might be difficult in BSS. However, re-using bGMCA and the above remark, it might be possible to perform such an estimation (first results have been obtained in this direction, but are still yet too preliminary to be discussed and must be confirmed) ;

Use of mini-batches in sparse BSS

The use of mini-batches in dGMCA has shown surprising results, making this approach particularly appealing and calling for further extensions :

— Handle datasets with a large number of observations, that is a high number of lines. This is reminiscent in the context of dictionary learning of the recent ODL extension [Mensch *et al.* 2018]. The main question would then be how to

aggregate the different estimations of the sources. The performances of using mini-batches both for the lines and columns would also need to be studied ;

— Combine mini-batches and blocks and study the performances of the resulting algorithm. In particular, this might call for a good block choice, as all the sources might not be present in a given mini-batch ;

— Introduce a refinement stage, akin to what is done the 2-step approach. This should be relatively straightforward as an asynchronous version of PALM has already been proposed in [Davies 2004].

Non-linear sparse BSS

In Chapter VII, we already highlighted some limitations of the current approach and some corresponding possible enhancements. To summarize and add a few elements :

— Building on the geometrical interpretation, we derived required hypotheses for StackedAMCA to work. It would be attractive to mathematically demonstrate these conditions to be sufficient for the separation and the reconstruction (or to find new ones that are) ;

— While similarly to most other works numerical experiments were here conducted on simulations, it would be interesting to apply StackedAMCA on concrete real-life problems ;

— A major issue of the current approach is its inability to handle partially correlated samples properly. To alleviate such an issue, we could use, if they are verified, additional explicit priors on the sources such as temporal dependencies.

Annexe A

Proximal operators

Definition of proximal operators

The proximal operator of an extended-valued proper and lower semi-continuous convex function $f : \mathbb{R}^t \to (-\infty, \infty]$ is defined as :

$$\text{prox}_f(\mathbf{u}) = \underset{\mathbf{y}\in\mathbb{R}^t}{\text{argmin}}\ f(\mathbf{y}) + \frac{1}{2}\|\mathbf{u}-\mathbf{y}\|_{\ell_2}^2 \tag{A.1}$$

Where $\mathbf{u} \in \mathbb{R}^t$.

Definition of the soft thresholding operator

For two scalar λ and b, the soft thresholding operator $\mathcal{S}_\lambda(.)$ is defined as :

$$\forall b \in \mathbb{R}, \forall \lambda \in \mathbb{R}^+, \mathcal{S}_\lambda(b) = \begin{cases} b - \lambda \times \text{sign}(b) \texttt{ if } |b| \geqslant \lambda \\ 0 \texttt{ otherwise} \end{cases} \tag{A.2}$$

For two matrices $\mathbf{\Lambda}$ and $\mathbf{U}$, we extend our notation to :

$$\forall \mathbf{U} \in \mathbb{R}^{n\times t}, \forall \mathbf{\Lambda} \in \mathbb{R}^{+^{n\times t}}, \forall i \in [1,n], \forall j \in [1,t], \mathcal{S}_{\mathbf{\Lambda}}(\mathbf{U})_{ij} = \mathcal{S}_{\mathbf{\Lambda}_{ij}}(\mathbf{U}_{ij}) \tag{A.3}$$

Definition of the projection of the columns of a matrix on the ℓ_2 unit hypersphere

We define the projection of a column vector $\mathbf{u}$ on the ℓ_2 unit hypersphere as :

$$\forall \mathbf{u} \in \mathbb{R}^m, \pi_{\|.\|_{\ell_2}=1}(\mathbf{u}) = \begin{cases} \mathbf{u}/\|\mathbf{u}\|_{\ell_2} \texttt{ if } \|\mathbf{u}\|_{\ell_2} \neq 0 \\ \texttt{undefined otherwise} \end{cases} \tag{A.4}$$

We extend the notation to a matrix $\mathbf{U}$ by projecting all its columns on the ℓ_2 unit hypersphere :

$$\forall \mathbf{U} \in \mathbb{R}^{m\times n}, \forall j \in [1,n], \Pi_{\|.\|_{\ell_2}=1}(\mathbf{U})^j = \pi_{\|.\|_{\ell_2}=1}(\mathbf{U}^j) \tag{A.5}$$

Definition of the projection of a matrix on the positive orthant K^+

$$\forall \mathbf{U} \in \mathbb{R}^{m\times t}, \forall i \in [1,m], \forall j \in [1,t], \Pi_{K^+}(\mathbf{U}_{ij}) = \begin{cases} \mathbf{U}_{ij} \texttt{ if } \mathbf{U}_{ij} \geqslant 0 \\ 0 \texttt{ otherwise} \end{cases} \tag{A.6}$$

Annexe B

Performance metrics for Blind Source Separation

The choice of a performance metric is key to quantitively assess the quality of a given separation. In this Appendix, we summarize the criteria used in this thesis. First, we highlight that finding such a criterion might not be trivial, and has lead to several propositions [Bobin *et al.* 2015, Vincent *et al.* 2006, Ehsandoust *et al.* 2017]. For instance, a mere recovery of the data (*e.g.* $\mathbf{X} \simeq \hat{\mathbf{A}}\hat{\mathbf{S}}$) does not mean that $\mathbf{A}^*$ and $\mathbf{S}^*$ are well estimated. Second, we will categorize the metrics into two families : the ones used for linear BSS, and the one used in the non-linear setting, for which the separation and reconstruction quality are distinct.

Linear BSS

The most largely used metric in this work is the mixing matrix criterion C_A [Bobin *et al.* 2015] :

$$C_A = \text{mean}(|\mathbf{P}\hat{\mathbf{A}}^{\dagger}\mathbf{A}^*| - \mathbf{I_d}) \tag{B.1}$$

With $\mathbf{A}^*$ the true mixing matrix and $\mathbf{P}\hat{\mathbf{A}}^{\dagger}$ the pseudo-inverse of an estimate corrected through $\mathbf{P}$ for the scale and permutation indeterminacies. The mean is the average of all the elements inside the matrix. The smaller C_A, the better the separation. However, in most of this work we used for the sake of clarity a logarithmic criterion : $-10\log_{10}(C_A)$.
Note that due to the use of the pseudo-inverse of $\hat{\mathbf{A}}$, this metric might be sensitive. Furthermore, C_A does not allow to distinguish between a fair but not fully accurate estimate of $\mathbf{A}^*$ and a good estimate in which for instance only one column is badly estimated. As such, an angular criterion between the columns of $\hat{\mathbf{A}}$ and $\mathbf{A}^*$ might be sometimes preferred [Chenot 2017].

Originally introduced in the context of audio BSS, the SDR, SAR, SIR, SNR [Vincent *et al.* 2006] are also of interest. In brief, the authors decompose each of the estimated source $\hat{\mathbf{S}}_i, i \in [1, n]$ in :

$$\hat{\mathbf{S}}_i = \hat{\mathbf{S}}_{i_{target}} + \hat{\mathbf{s}}_{i_{interferences}} + \hat{\mathbf{s}}_{i_{noise}} + \hat{\mathbf{s}}_{i_{artifacts}} \tag{B.2}$$

Where the four terms should be seen as respectively the part of $\hat{\mathbf{S}}_i$ coming from the wanted source $\hat{\mathbf{S}}_i^*$, the one coming from other sources, from noise and from other

causes (*e.g.* distortions). More specifically :

$$\hat{\mathbf{S}}_{i_{target}} = \Pi_{\mathbf{S}_i^*}(\hat{\mathbf{S}}_i)$$

$$\hat{\mathbf{s}}_{i_{interferences}} = \Pi_{\mathbf{S}^*}(\hat{\mathbf{S}}_i) - \Pi_{\mathbf{S}_i^*}(\hat{\mathbf{S}}_i)$$

$$\hat{\mathbf{s}}_{i_{noise}} = \Pi_{\mathbf{S}^*,\mathbf{N}}(\hat{\mathbf{S}}_i) - \Pi_{\mathbf{S}^*}(\hat{\mathbf{S}}_i)$$

$$\hat{\mathbf{s}}_{i_{artifacts}} = \hat{\mathbf{S}}_i - \Pi_{\mathbf{S}^*,\mathbf{N}}(\hat{\mathbf{S}}_i)$$

This decomposition is then used to derive the following metrics :

— Signal-to-Distortion-Ratio : $\mathrm{SDR}(\hat{\mathbf{S}}_i) = 10\log_{10}\frac{\left\|\hat{\mathbf{S}}_{i_{target}}\right\|_{\ell_2}^2}{\left\|\hat{\mathbf{s}}_{i_{interferences}}+\hat{\mathbf{s}}_{i_{noise}}+\hat{\mathbf{s}}_{i_{artifacts}}\right\|_{\ell_2}^2}$

— Signal-to-Interferences-Ratio : $\mathrm{SIR}(\hat{\mathbf{S}}_i) = 10\log_{10}\frac{\left\|\hat{\mathbf{S}}_{i_{target}}\right\|_{\ell_2}^2}{\left\|\hat{\mathbf{s}}_{i_{interferences}}\right\|_{\ell_2}^2}$

— Signal-to-Noise-Ratio : $\mathrm{SNR}(\hat{\mathbf{S}}_i) = 10\log_{10}\frac{\left\|\hat{\mathbf{S}}_{i_{target}}+\hat{\mathbf{s}}_{i_{interferences}}\right\|_{\ell_2}^2}{\left\|\hat{\mathbf{s}}_{i_{noise}}\right\|_{\ell_2}^2}$

— Signal-to-Artifacts-Ratio : $\mathrm{SAR}(\hat{\mathbf{S}}_i) = 10\log_{10}\frac{\left\|\hat{\mathbf{S}}_{i_{target}}+\hat{\mathbf{s}}_{i_{interferences}}+\hat{\mathbf{s}}_{i_{noise}}\right\|_{\ell_2}^2}{\left\|\hat{\mathbf{s}}_{i_{artifacts}}\right\|_{\ell_2}^2}$

The median over all the sources $\hat{\mathbf{S}}_i, i \in [1,n]$ can then be taken to obtain a global criterion. As a side remark, we would like to highlight that such metrics can be highly sensitive to a few badly estimated samples due to the use of a squared ℓ_2-norm.

Non-linear BSS

Metrics for source separation

A classical approach is to estimate the indeterminacy function $\mathbf{h}$ [Ehsandoust *et al.* 2017] by fitting a non-linear curve $\mathcal{P}$ to the 1D-manifold of the scatter plot of each estimated source $\hat{\mathbf{S}}_i$ as a function of the true one $\mathbf{S}_i^*$, and to look at the thickness of the manifold around $\mathcal{P}$. The thickness can then be measured by the :

— logarithmic median absolute distance to $\mathcal{P}$:

$$C_{med} = -10\log(\sum_{i=1}^{n}\underset{j\in[1,t]}{\mathrm{median}}(|\hat{\mathbf{S}}_i^j - \mathcal{P}(\mathbf{S}_i^{*j}))|) \tag{B.3}$$

— logarithmic mean absolute distance to $\mathcal{P}$:

$$C_{mean} = -10\log(\sum_{i=1}^{n}\frac{1}{t}\sum_{j=1}^{t}|\hat{\mathbf{S}}_i^j - \mathcal{P}(\mathbf{S}_i^{*j})|) \tag{B.4}$$

However, the results of these metrics are sensitive to the choice of $\mathcal{P}$. We thus used for exactly sparse sources a new metric based on the *angular distance to the axes* :

$$C_{ang} = -10 \log \left(\frac{1}{n(n-1)} \sum_{i=1}^{n} \left(\sum_{\substack{i'=1 \\ i' \neq i}}^{n} 1 - \frac{1}{\#Z} \sum_{j \in Z} \frac{\mathbf{S}_{i'}^{*j}}{\sqrt{\hat{\mathbf{S}}_i^{j^2} + \mathbf{S}_{i'}^{*j^2}}} \right) \right) \tag{B.5}$$

where $Z = \{j \in [1,t] | \mathbf{S}_{i'}^{j} \neq 0\}$.

Metrics for source reconstruction

The reconstruction metrics are the usual *linear ones*, once the scale indeterminacy corrected : the SDR, SAR... In addition, we also used the Median absolute Error (ME) :

$$\mathrm{ME}(\hat{\mathbf{S}}) = \sum_{i=1}^{n} \sum_{j=1}^{t} |\mathbf{S}_i^{*j} - \hat{\mathbf{S}}_i^{j}| \tag{B.6}$$

and the Mean Square Error (MSE) :

$$\mathrm{MSE}(\hat{\mathbf{S}}) = \sum_{i=1}^{n} \sum_{j=1}^{t} \left\| \mathbf{S}_i^{*j} - \hat{\mathbf{S}}_i^{j} \right\|_{\ell_2} \tag{B.7}$$

Annexe C

Convergence conditions

We here detail the conditions ensuring that the algorithms we use converge, if it is the case. Since this work is not new and is not the focus of this thesis, such details were omitted into the main text.

A BCD

Problems of the form of Eq. II.8 can be shown to (almost [1]) follow the conditions of Lemma 3.1 a) and Theorem 4.1 b) of [Tseng 2001] (*cf.* example 6.4 therein) :

- The domain of $h(\mathbf{A},\mathbf{S}) = \frac{1}{2}\|\mathbf{X}-\mathbf{AS}\|_F^2$ is open, and h is differentiable on it.
- Writing $f(\mathbf{A},\mathbf{S}) = \frac{1}{2}\|\mathbf{X}-\mathbf{AS}\|_F^2 + \left\|\mathbf{R_S}\odot(\mathbf{S}\mathbf{\Phi}_\mathbf{S}^\mathbf{T})\right\|_1 + \iota_{\{\forall i\in[1,n];\|\mathbf{A}^i\|_{\ell_2}^2=1\}}(\mathbf{A})$ and taking feasible points [2] for the initialization $(\hat{\mathbf{A}}^{(0)},\hat{\mathbf{S}}^{(0)})$, the level set $\mathcal{U} = \{(\mathbf{A},\mathbf{S}); h(\mathbf{A},\mathbf{S}) \leq h(\hat{\mathbf{A}}^{(0)},\hat{\mathbf{S}}^{(0)})\}$ is compact, and h is continuous on it.
- $f(\mathbf{A},\mathbf{S}) = f(\mathbf{A},\mathbf{S}_1^1,\mathbf{S}_2^1,...,\mathbf{S}_n^t)$ is convex in $(\mathbf{S}_1^1,\mathbf{S}_2^1,...,\mathbf{S}_n^t)$ and convex in $\mathbf{A}$. The function f is also regular on $\mathcal{U}$.
- The updates are essentially cyclic, ensuring that each variable is updated enough time. More specifically, in our case the updates are cyclic.

B PALM

It can be proved that our cost function follows the convergence condition of [Bolte *et al.* 2014].

- The function f defined above follows the Kurdyka-Lojasiewicz property [Bolte *et al.* 2014]. In brief, such property mainly implies that f is sharp (in particular, close to the critical points) up to a reparameterization of its values (there exists a function ϕ such that the sub-gradients of $u \to \phi\circ(f(u)-f(\bar{u}))$ have a norm greater than 1, no matter how u is close to the critical point $\bar{u}$). More specifically, all the terms of Eq. II.8 are semi-algebraic functions – the data

1. To be more precise, the use of $\iota_{\{\forall i\in[1,n];\|\mathbf{A}^i\|_2^2=1\}}(\mathbf{A})$ makes that the required conditions do not hold, since in particular f is no more convex in $\mathbf{A}$. This is however not an important issue, since the problem only stems for the specific point where $\mathbf{A}$ is a matrix having one (or more) columns filled with 0, which is not likely to occur in our context. For more mathematical accuracy, the hyper-sphere constraint $\iota_{\{\forall i\in[1,n];\|\mathbf{A}^i\|_{\ell_2}^2=1\}}(\mathbf{A})$ can be replaced by the ball constraint $\iota_{\{\forall i\in[1,n];\|\mathbf{A}^i\|_{\ell_2}^2\leq 1\}}(\mathbf{A})$.

2. That is, the initialization respects the conditions enforced by the indicator function.

fidelity term is a polynomial function, $\|.\|_p$ is semi-algebraic whenever p is a rational number and the indicator functions are semi-algebraic, making that the sum is semi-algebraic and thus follows the Kurdyka-Lojasiewicz property.

— Both the sparsity and the oblique constraints are proximable.

— The data fidelity term h is $\mathcal{C}^2$. As such, its gradients ∇h with respect to $\mathbf{A}$ and $\mathbf{S}$ are Lipschitz continuous on bounded subsets. Furthermore, the corresponding Lipschitz constants are bounded (and non-zeros).

— We use a cyclic update of the variables.

C pALS

The pALS algorithm cannot be proved to converge and can even diverge. In this section, we aim at giving an understanding of such an issue with a simple (historical) example related to NMF. Let us assume that we want to solve the problem, with $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{x} \in \mathbb{R}^m$:

$$\underset{\mathbf{s} \in \mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{x} - \mathbf{A}\mathbf{s}\|_F^2 + \iota_{\mathbf{s} \in K^+}(\mathbf{s}) \tag{C.1}$$

The pALS solution is :

$$\hat{\mathbf{s}} = \Pi_{K^+}[\mathbf{A}^\dagger \mathbf{x}] = \Pi_{K^+}[(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{x}], \tag{C.2}$$

which can be, as explained in [Kim *et al.* 2008], rewritten as :

$$\hat{\mathbf{s}} = \Pi_{K^+}[\hat{\mathbf{s}} - (\mathbf{A}^T\mathbf{A})^{-1}(\mathbf{A}^T\mathbf{A}\hat{\mathbf{s}} - \mathbf{A}^T\mathbf{x})] = \Pi_{K^+}[\hat{\mathbf{s}} - 1 \times \mathbf{H}^{-1}(\mathbf{A}^T\mathbf{A}\hat{\mathbf{s}} - \mathbf{A}^T\mathbf{x})] \tag{C.3}$$

with $\mathbf{H}$ the Hessian of the data fidelity term. Thus, instead of performing a gradient step as it would be the case in PALM, the pALS algorithm performs a quasi-Newton step with projection, but with an arbitrary step 1. Such an update can however increase the cost function as displayed in Fig. C.1, which precludes any convergence guarantee. As a side remark, it is however intuitively visible that the weaker the enforced constraints, the easier for pALS to be able to converge. For instance, when confronted to a sparse minimization problem of the following form :

$$\underset{\mathbf{s} \in \mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{x} - \mathbf{A}\mathbf{s}\|_F^2 + \|\mathbf{R}_\mathbf{S} \odot \mathbf{S}\|_1 , \tag{C.4}$$

pALS is more likely to converge when $\mathbf{R}_\mathbf{S}$ coefficients are small (and in the limit, if $\|\mathbf{R}_\mathbf{S}\| = 0$, it converges). This explains why in practice GMCA tends to stabilize in most cases thanks to its decreasing hyper-parameter strategy, since the hyper-parameters are low during the last iterations.

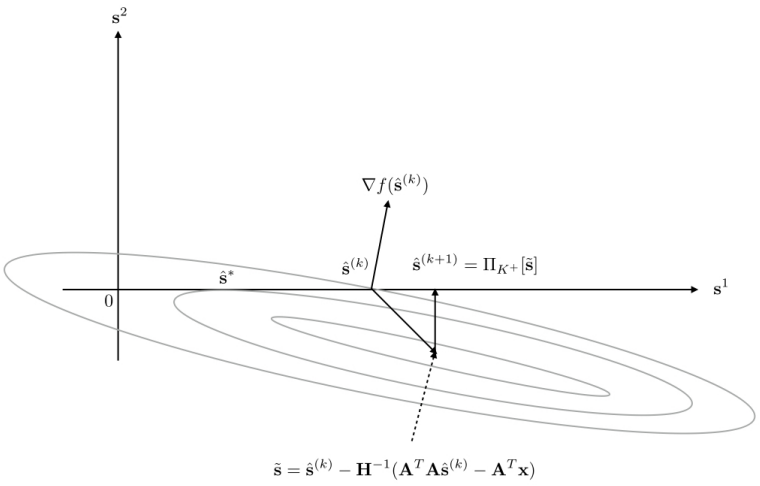


Figure C.1 – A graphical example of an iteration of pALS, during which the cost function does not decrease. The gray ellipses represent the level sets of the data-fidelity term. The algorithm starts at the current iterate $\hat{\mathbf{s}}^{(k)}$. The update before thresholding $\tilde{\mathbf{s}}$ reaches the minimum of the data fidelity term. However, the projection of $\tilde{\mathbf{s}}$ on the non-negativity constraint increases the whole cost function (which is confounded with the data-fidelity term for feasible points) since the new estimation $\tilde{\mathbf{s}}^{(k+1)}$ goes from an inner ellipse towards an outer one. Figure inspired from [Kim *et al.* 2008].

Annexe D

Variants of GMCA thresholding strategy

We here present a few variants of the originally-proposed GMCA automatic parameter choice based on the use of the κ-MAD. The main idea is the same, and all these strategies aim at accentuating the decrease of the parameters to still further benefit from the morphological diversity assumption.

A κ-MAD with varying κ

In this strategy, the hyper-parameters are chosen as described in chapter III on the κ-MAD rule :

$$\left(\mu_1^{(l)}, \mu_2^{(l)}, ..., \mu_n^{(l)}\right)^T = \kappa \times \text{MAD}(\hat{\mathbf{A}}^{(l-1)\dagger}\mathbf{X}) \tag{D.1}$$

However and contrary to what was described, in this strategy κ is not chosen fixed with a value of $\kappa = 3$ but decreases linearly during the iterations, starting for instance from a value of 7 and finishing to a value of 3. This enables to better take into account the morphological diversity, since the hyper-parameters now start from higher values.

B Increasing percentile

The value of $\kappa = 7$ of the previous strategy might seem somewhat arbitrary. Therefore, this method is fully based on the estimated $\tilde{\mathbf{S}}_i^{(l)}$ coefficients :

- First the support of $\tilde{\mathbf{S}}_i^{(l)}$ is determined, keeping only the samples of $\tilde{\mathbf{S}}_i^{(l)}$ larger than $\kappa\,\text{MAD}(\tilde{\mathbf{S}}_i^{(l)})$ (with $\kappa = 3$) ;
- Denoting L the maximum number of iterations, the threshold value $\mu_i^{(l)}$ at iteration l is set as the $100 \times \frac{L-l}{L}$ percentile of $\tilde{\mathbf{S}}_i^{(l)}$ entries in the support : $|\tilde{\mathbf{S}}_i^{(l)}|_{|\tilde{\mathbf{S}}_i^{(l)} \geq |\kappa\,\text{MAD}(\tilde{\mathbf{S}}_i^{(l)})|}$.

Compared to the previous one, this strategy has two advantages : i) it avoids the need for setting κ to an arbitrary value ; ii) it enables to take into account the distribution of the sources.
However, it has to be highlighted that since the thresholds values are dependent on the final number of iterations L, the results are in general much more dependent to L than with the previous strategy.

Annexe E

Other exploratory ways of performing a 2 steps strategy

In this appendix, we shortly discuss some alternatives for the refinement step of the 2-step strategy of Chapter IV. This must not be understood as a fully accomplished work, but rather as preliminary results that could pave the way for future research.

A Discussion about BCD

In sparse NMF, the use of a two-step strategy using as refinement a BCD algorithm instead of PALM was proposed in [Rapin 2014]. While using BCD can lead to a computational overload [Chenot 2017,Xu & Yin 2013] which would not be suitable in our large-scale setting, we tried it on relatively small problems to determine whether this approach could lead to a potential improvement of the separation quality. This section is structured similarly as Chapter IV : we first study the results of an isolated BCD with fixed parameters, and then its behavior as a refinement stage equipped with the MAD heuristic within a two-step approach.

A.1 Fixed parameters

In this section, we use a similar approach as in Chapter IV-C to assess the separation quality of a single BCD. Such a task is again performed using a grid search on the regularization parameters for the data of *case 1* and *case 2*. The protocol is exactly the same as the one we used for PALM. Figure E.1, which is to be compared with Fig. IV.1, indicates very similar results between BCD and PALM and does not seem to suggest that BCD has a better *efficiency* than PALM when coupled with a grid-search regularization parameter choice : that is, the hyper-parameter choice is not less sensitive when using BCD. Fig. E.2 and Fig. E.3, which are compared with Fig. IV.3 in the right plots, tend to show that the *versatility* of BCD is further similar as the one of PALM, at least close to the diagonal where the best results are achieved.
As a conclusion, such results seem to clearly indicate that a BCD suffers from the same difficult regularization parameter choice than a PALM. This was fully expected on such simple experiments, where the local minima seem to be rare (the standard deviation over the initialization of both PALM and BCD is low). Indeed, since both algorithms are minimizing the same cost function, the only source of discrepancy

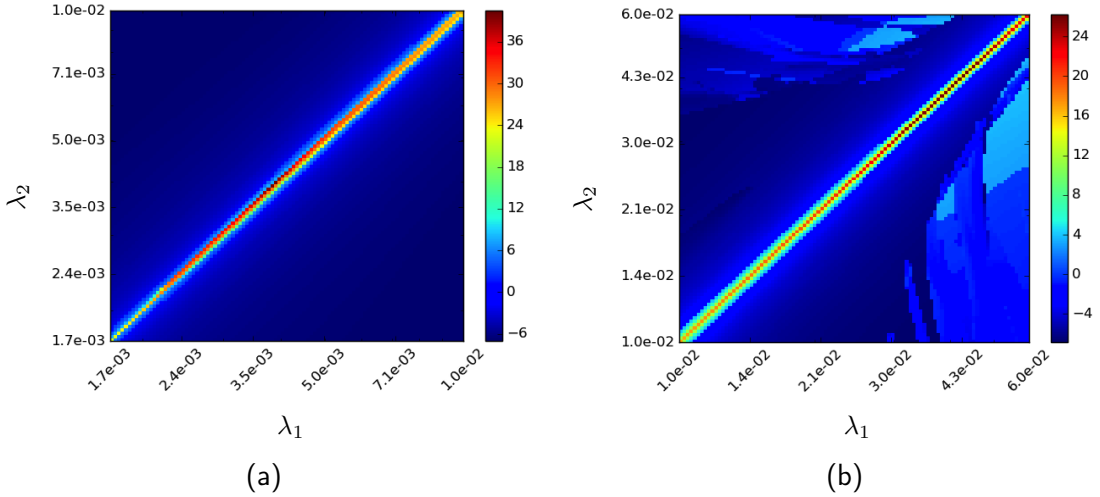


Figure E.1 – Median of C_A (dB) for 5 initializations of BCD algorithm as a 2D function of the 2 thresholds corresponding to the n = 2 sources. To be compared with Fig. IV.1. *Left : Case 1, Right : Case 2.*

between both must come from such potential spurious minima.
Such preliminary conclusions must however be mitigated, and should be confirmed in more complicated settings with realistic sources, for which such spurious minima could appear, as well as with a higher number of sources n. [1]

A.2 BCD as a refinement stage

In this section, we investigate the behavior of BCD within a 2-step strategy as a refinement stage. Similarly to PALM, we use both an initialization by a warm-up GMCA stage and the MAD as an automatic regularization parameter choice within the BCD refinement. The results we obtain in Fig. E.4 for the 2-step BCD do not vary much [2] from the ones of the 2-step PALM of Fig. IV.5. As such, using a BCD does not yield an improvement that would justify the higher computational cost.
As a side remark, while for some rare experiments and initializations using the MAD directly within a single PALM could yield decent results as testified by the right plot of Figure IV.5, it does not seem to be the case for BCD. Indeed, in BCD the thresholds computed from a bad initialization are re-used for the whole update of the $\mathbf{S}$ matrix, making them more important than in PALM, where they are used only for one proximal gradient step and then updated. This seems to highlight that using a good initialization for a BCD based on a MAD strategy for the regularization

1. As a side remark, changing the structure of the noise and using a Poisson noise instead of a Gaussian one seems also to create discrepancies between BCD and PALM results.

2. Note that in the previous subsection the only source of differences between BCD and PALM was spurious minima. In contrast, this is not anymore the case here because using the MAD within BCD leads to different regularization parameter choices than using it in PALM.

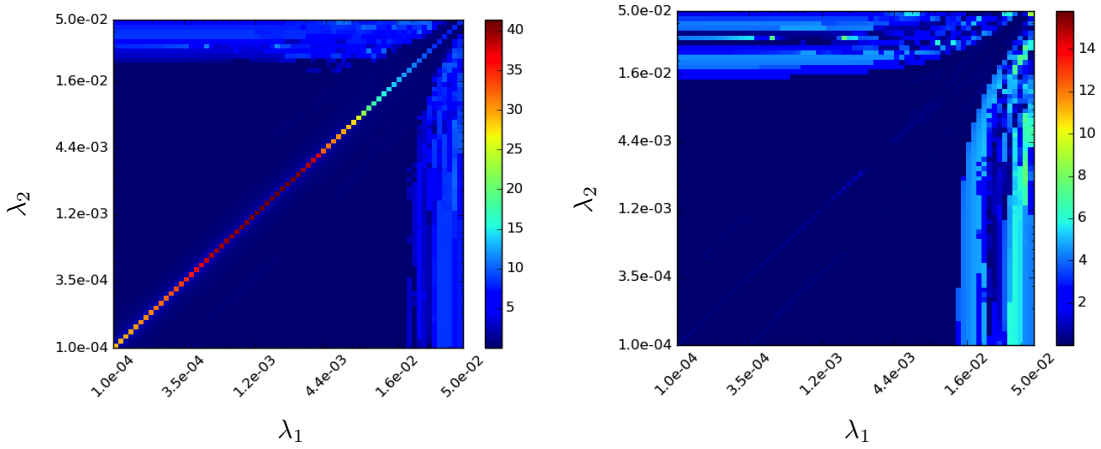


Figure E.2 – Dynamic of C_A for BCD over different random realizations of $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$ in *case 1*. To be compared with Fig. IV.3 left plot, in the case of PALM. *Left :* BCD results ; *Right :* Absolute difference between left plot and the corresponding one for PALM of Fig. IV.3.

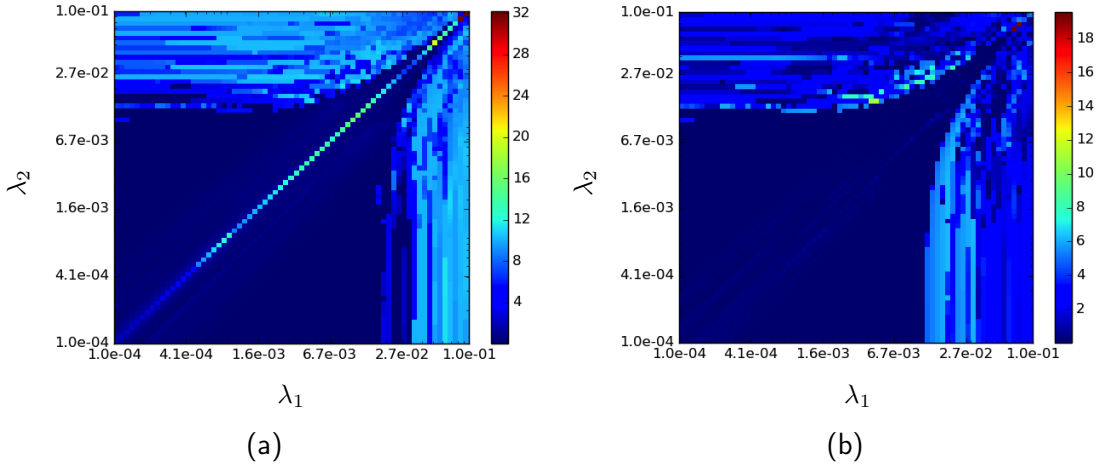


Figure E.3 – Dynamic of C_A for BCD over different random realizations of $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$ in *case 2*. To be compared with Fig. IV.3 right plot, in the case of PALM. *Left :* BCD results ; *Right :* Absolute difference between left plot and the corresponding one for PALM of Fig. IV.3.

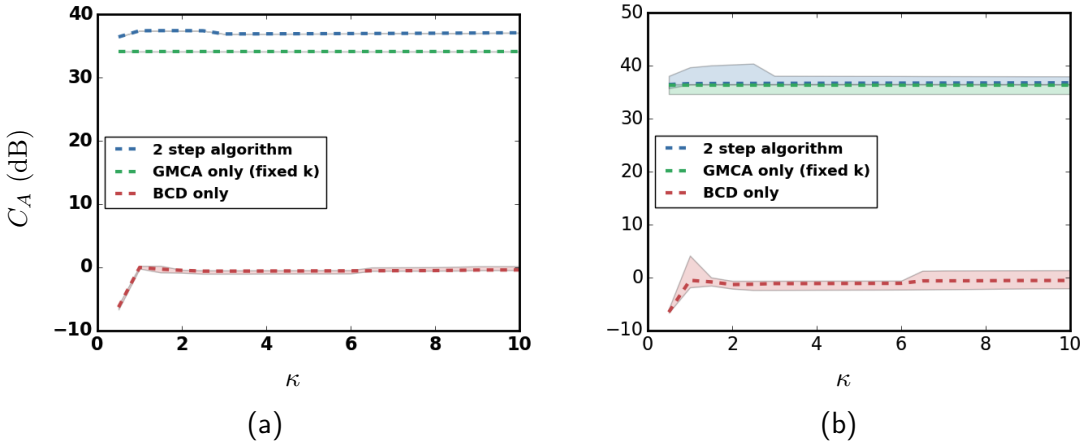


Figure E.4 – Results of a two-step approach with a BCD refinement stage on *Case 1*. For the sake of completness, the results of an isolated BCD equipped with a MAD strategy is also displayed. This figure is to be compared with the left and right plots of Fig. IV.5. *Left :* the dashed line is the median of C_A over the different $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$, and the error bars corresponds to the quartiles of the criterion over the initialization ; *Right :* the dashed line corresponds to the median of C_A over the initializations, and the error bars to the quartiles of the criterion over the realizations of $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$.

parameter choice is even more important than in the case of PALM.

B Decreasing thresholds based on the source distribution

Recent improvements of GMCA enforce more strongly decreasing thresholds by basing their choice on the estimated source distribution (for more details, see previous Appendix). A natural extension of the 2-step strategy is then to replace the MAD by such an enhanced threshold choice based on an increasing percentile of the estimated sources.

Preliminary to using it inside of a 2-step approach, a test on a single PALM yields decent results (at least, for a large number of maximal iterations, implying very slowly decreasing regularization parameters – *cf.* Fig. E.5). The robustness with regards to the initialization is however low compared to the one of the two step approach [3]. Furthermore, a single PALM always gives worse results than the two step algorithm (probably in part due to the fact that the lowering of the high thresholds implies high interferences – this seems to be confirmed by the bad results of

3. While such decent results could suggest to use a two-step PALM - PALM with the first stage using a threshold choice based on the decreasing strategy, and the second one using a MAD strategy in addition to reweighted ℓ_1, the results would likely be unstable with regards to the initialization similarly to the first step.

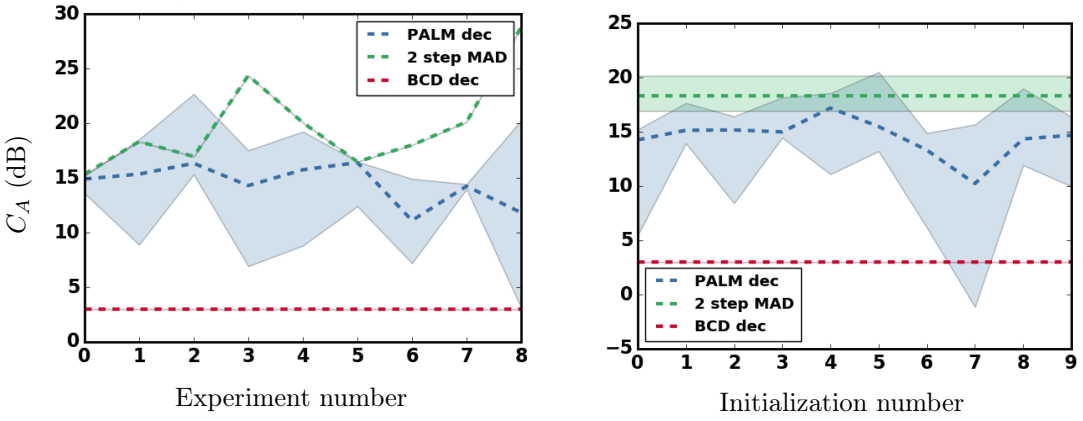


Figure E.5 – Results of a isolated PALM and BCD with a percentile threshold choice on *case 2*. For comparison, the results of the usual two-step approach are displayed. *Left :* the dashed line is the median of C_A over the different $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$, and the error bars corresponds to the quartiles of the criterion over the initialization ; *Right :* the dashed line correspond to the median of C_A over the initializations, and the error bars to the quartiles of the criterion over the realizations of $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$.

a decreasing threshold strategy within PALM with exactly sparse $\mathbf{S}^*$ sources, for which the decrease is faster).
Such a strategy is however made more robust to the initialization when used within a 2-step approach. Indeed, the choice of the initial regularization parameters is then made on a decently estimated $\hat{\mathbf{S}}$ matrix, making it much better. Since GMCA is robust to the initialization, so do the threshold choice. The final results are however slightly worse (2.5 dB on average) than the ones of the 2-step κ-MAD approach presented in chapter IV.
An interesting result is furthermore given when using a 2-step strategy with a refinement stage based on BCD instead of PALM and a threshold choice based on the percentile instead of the MAD. In this setting, BCD yields better results than a refinement stage based on the PALM. This can be explained by the fact that in BCD $\mathbf{S}$ is updated until convergence with fixed thresholds before updating $\mathbf{A}$. Thus, the interferences implied by the gradient step are partially removed before updating $\mathbf{A}$, implying better results. While we do not aim at performing a full study of BCD, this result suggests to use within PALM a decreasing by steps threshold choice, aiming at partially mimicking BCD.

C Decreasing by steps threshold

This new strategy is based on the previous remark that a decreasing threshold choice based on an increasing percentile of the estimated sources worked better

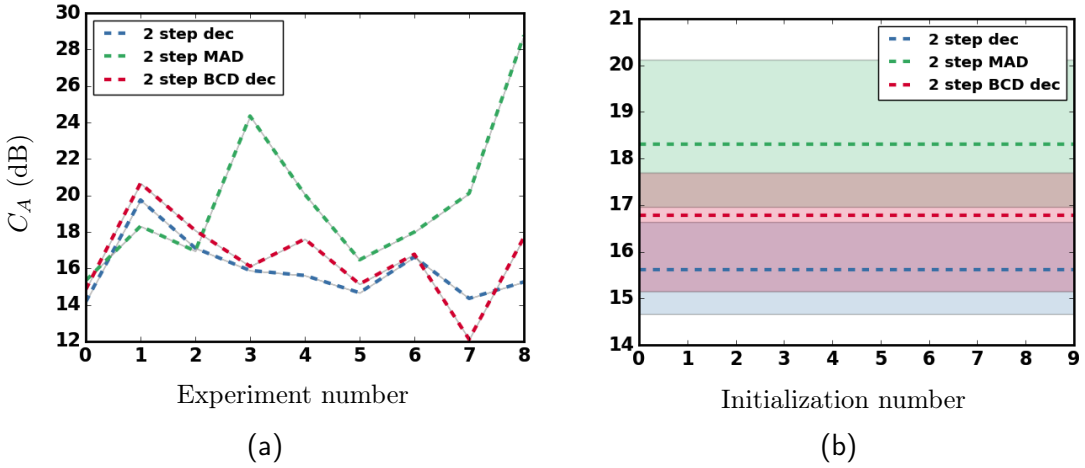


Figure E.6 – Results on *case 2* of PALM and BCD used as refinement stage within a two-step approach with a percentile threshold choice. For comparison, the results of the usual two-step approach are displayed. *Left :* the dashed line is the median of C_A over the different $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$, and the error bars corresponds to the quartiles of the criterion over the initialization ; *Right :* the dashed line corresponds to the median of C_A over the initializations, and the error bars to the quartiles of the criterion over the realizations of $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$.

within a 2-step BCD than a 2-step PALM, and that it might be linked to lower interferences in BCD. More specifically, high thresholds creates high artifacts due to the use of the ℓ_1 norm. If the thresholds decrease quickly, such artifacts will be directly partially transformed into interferences due to the PALM gradient step on $\mathbf{S}$. In turns, these interferences will impact the estimation of $\mathbf{A}$, deteriorating it, which is limited by the use of BCD.

While this seems to be relatively intrinsic to the PALM scheme, we can at least try to limit the deterioration by improving the threshold choice and not choosing it on a solution containing a high level of interferences. More specifically, instead of decreasing the threshold at each iteration, we will compute a new threshold only after several iterations, when the interferences become low. That is, we will decrease the thresholds by step, by choosing when to change the thresholds using a convergence criterion on $\hat{\mathbf{S}}$. The results of this method are displayed in Fig. E.7.

Such a strategy yields deteriorated results with a single PALM compared to the previous decreasing strategy. This is understandable, since at the beginning of the algorithm the bad initialization implies bad thresholds, which are kept during more iterations in a decreasing by steps scheme.

On the other hand, this strategy gives similar results as the usual κ-MAD rule (while they are in fact slightly better, the difference is not significative with respect to the

error bars). Compared to a continuous decrease, the gain is not as high as expected [4], which is to be explained by the fact that the reweighting already cancels out most of the interferences appearing due to the gradient step. As such, the strategy by step only brings a limited gain (it however seems to robustify the process with respect to the realization of $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$, which was expected since the good thresholds found through GMCA solution are kept during more iterations).

D SURE

The Stein Unbiased Risk Estimator (SURE) is an automatic regularization parameter tuning method based on an unbiased estimate of the MSE of a candidate solution $\hat{\mathbf{S}}$ [Giryes *et al.* 2011]. The hyper-parameter is chosen as minimizing such an estimation of the MSE. While originally restricted to additional white Gaussian noise [Stein 1981], it has been generalized to models having the form of an exponential family distribution [Eldar 2009], extending its use to much more various inverse problems.

We propose in this subsection to test this approach on the sparse BSS problem. Note that SURE is usually used on *non-blind* problems, that is problems where $\hat{\mathbf{A}} = \mathbf{A}^*$ is known in advance, which is not our case here. A full extension of SURE methods to the sparse BSS problem, while interesting, is out of the scope of this work. We rather propose to use the SURE method within the 2-step approach. Following [Giryes *et al.* 2011], the computation of the threshold [5] could then be based on the currently estimated – at iteration l – $\hat{\mathbf{A}}$, which would be supposed to be a decent guess of $\mathbf{A}^*$. The SURE estimated parameter should then correspond to an approximation of the best corresponding $\hat{\mathbf{S}}^{(l+1)}$ in terms of MSE. The hope is that such a $\hat{\mathbf{S}}^{(l+1)}$ in turns corresponds to a good $\hat{\mathbf{A}}^{(l+1)}$ and that the refinement step will enter a virtuous circle.

Unfortunately this strategy, although itself a cheaper iterative approximation [Giryes *et al.* 2011] of what a classical SURE method would do, is already too expensive for the large-scale setting since it requires to test many regularization parameters at each iteration. Therefore, we approximated it in our experiments by computing the thresholds only once at the beginning of the refinement stage, based on the first update of GMCA solution [6]. The results of such a method are given in Fig. E.8. While this methods gives relatively decent solutions, the results are deteriorated compared to the κ-MAD two-step approach. This can be understood by the fact that the method does not take explicitly into account the estimation of the unmixing matrix $\hat{\mathbf{A}}$ and thus the threshold choice is only based on noise removal. As such, the

4. Note that the discrepancy increases when the decrease of the thresholds is quicker in PALM – that is, when the number of maximal iterations is smaller.

5. Here, the regularization parameters are the same for all the sources.

6. This strategy thus corresponds to the first update of the thresholds in the iterations of [Giryes *et al.* 2011]. However, instead of using an initialization based on the *golden rule*, we here use a pseudo-inverse $\tilde{\mathbf{S}}^{(0)} = \hat{\mathbf{A}}^{\dagger}_{\mathbf{GMCA}}\mathbf{X}$ instead of $\tilde{\mathbf{S}}^{(0)} = \hat{\mathbf{A}}^{T}_{\mathbf{GMCA}}\mathbf{X}$, consistently to GMCA solution and giving much better results.

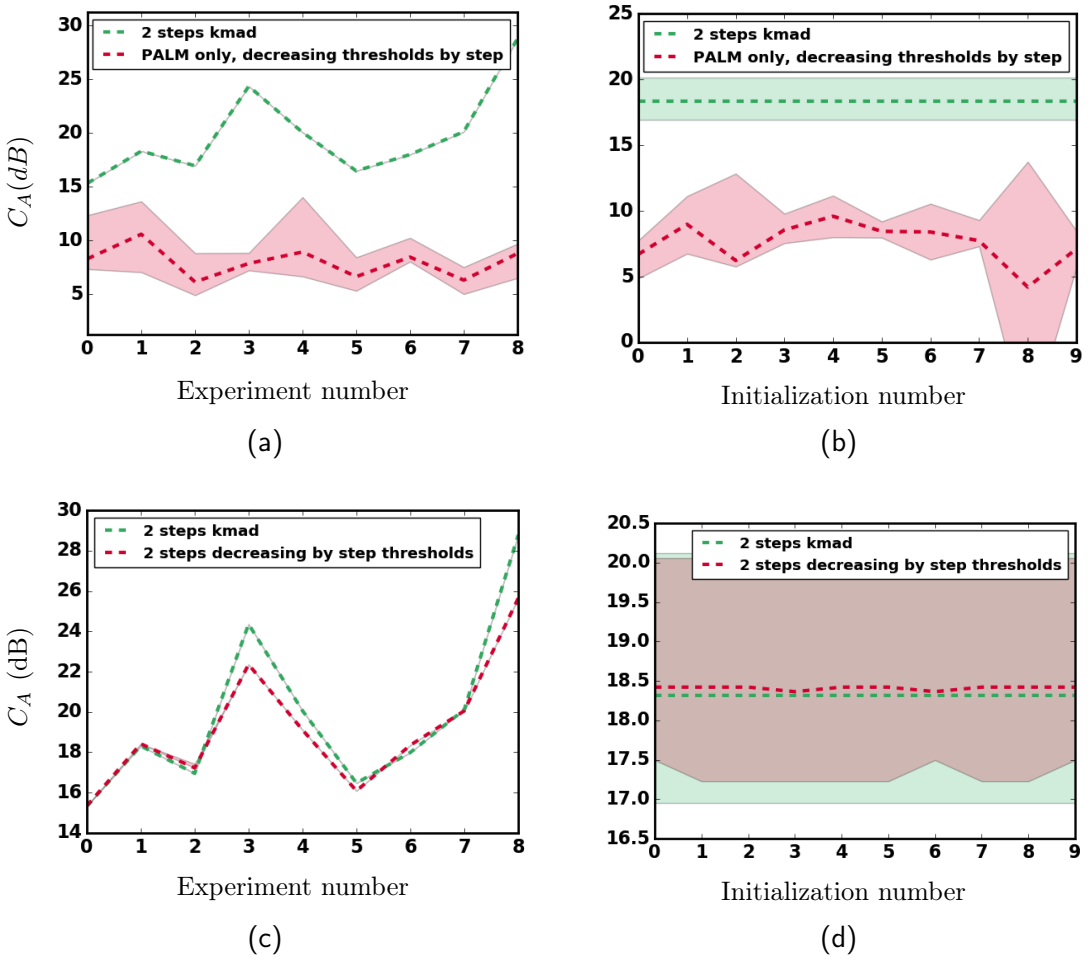


Figure E.7 – Results on *case 2* of PALM with a percentile-by-step threshold choice. For comparison, the results of PALM with a continuous percentile decrease of the thresholds are displayed, as well as the usual 2-step κ-MAD algorithm. The upper plots correspond to isolated PALMs and the lower ones to PALMs included in a two-step algorithm. *Left :* the dashed line is the median of C_A over the different $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$, and the error bars corresponds to the quartiles of the criterion over the initialization ; *Right :* the dashed line corresponds to the median of C_A over the initializations, and the error bars to the quartiles of the criterion over the realizations of $\mathbf{A}^*$,$\mathbf{S}^*$ and $\mathbf{N}$.

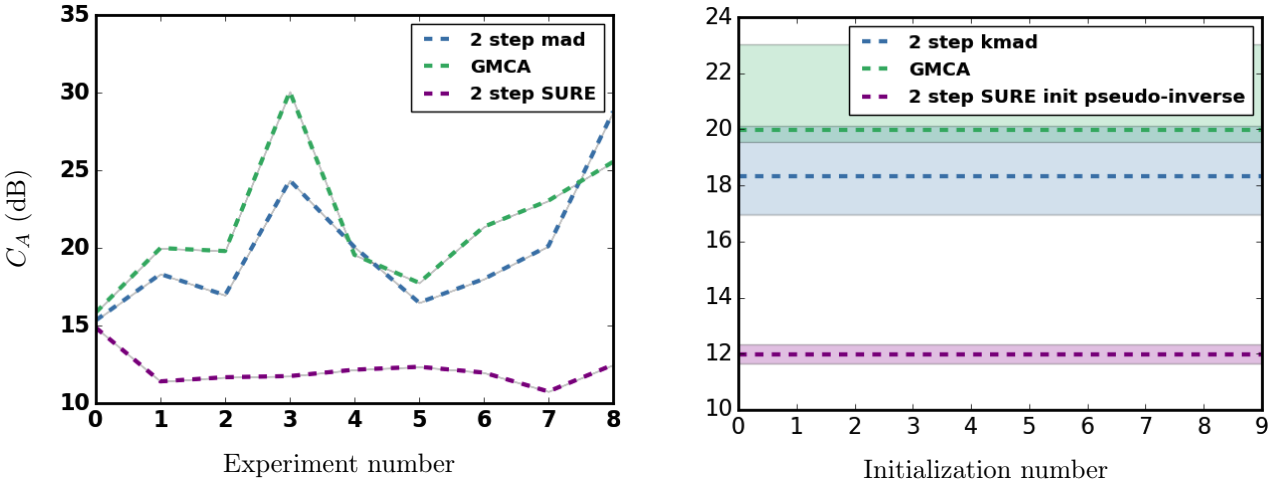


Figure E.8 – Results on *Case 2* of PALM used as refinement stage within a 2-step approach with a SURE threshold choice. For comparison, the results of the usual two-step approach are displayed. *Left :* the dashed line is the median of C_A over the different $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$, and the error bars corresponds to the quartiles of the criterion over the initialization ; *Right :* the dashed line corresponds to the median of C_A over the initializations, and the error bars to the quartiles of the criterion over the realizations of $\mathbf{A}^*$, $\mathbf{S}^*$ and $\mathbf{N}$.

thresholds are much too low for a high quality unmixing. It therefore confirms that in high SNR experiments, the interferences plays a non-negligible role compared to the noise and thus the threshold choice fully based on noise removal would not be efficient [7].

7. Which confirms that the use of MAD, while having a fixed-point noise removal interpretation, rather draws much of its interest from the morphological diversity assumption.

Annexe F

Elements of Riemannian geometry

Let define $\mathcal{M}$ some Riemannian manifold equipped with the metric $<,>$. One can define for every point x_0 its tangent space $\mathcal{T}_{\mathcal{M}}(x_0)$ as displayed Fig. F.1. The geodesic between two points x_0 and x_1 is defined as its shortest path on the manifold, which then turns to generalize the concept of straight lines on manifolds. In this thesis, the manifold $\mathcal{M}$ is assumed to be geodesically complete, which means that there always exists a minimal length geodesic between any two points x_0 and x_1. The length of the geodesic is then the distance $d(x_0, x_1)$ between these two points. This also entails that one can define for any point $x_0 \in \mathcal{M}$ the exponential map $\exp_{x_0}$. This function maps any point of the tangent space at x_0 to $\mathcal{M}$. Under some conditions, the exponential map is bijective and invertible ; its inverse is defined as the logarithm map $\log_{x_0}(x_1) = \exp_{x_0}(x_1)^{-1}$.
For some smooth function $\mathcal{J} : \mathcal{M} \to \mathbb{R}$ (*e.g.* the square geodesic distance to only name one), one can define uniquely its gradient at any point $x_0 : \nabla\mathcal{J}(x_0) \in \mathcal{T}_{\mathcal{M}}(x_0)$.

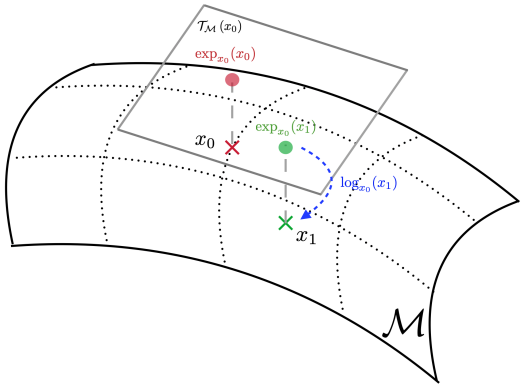


Figure F.1 – Sketch of some Riemannian manifold $\mathcal{M}$ and its tangent space at the point x_0.

Bibliography

[Absil *et al.* 2009] P-A Absil, Robert Mahony et Rodolphe Sepulchre. Optimization algorithms on matrix manifolds. Princeton University Press, 2009. (cited in page 109)

[Afsari 2011] Bijan Afsari. Riemannian L^p center of mass : existence, uniqueness, and convexity. Proceedings of the American Mathematical Society, vol. 139, no. 2, pages 655–673, 2011. (cited in page 110)

[Almeida & Figueiredo 2013] Mariana SC Almeida et Mário AT Figueiredo. Parameter estimation for blind and non-blind deblurring using residual whiteness measures. IEEE Transactions on Image Processing, vol. 22, no. 7, pages 2751–2763, 2013. (cited in page 69)

[Almeida 2003] Luís B Almeida. MISEP–Linear and Nonlinear ICA Based on Mutual Information. Journal of Machine Learning Research, vol. 4, no. Dec, pages 1297–1318, 2003. (cited in pages 5, 130, 133, 147 et 154)

[Arnaudon *et al.* 2013] Marc Arnaudon, Frédéric Barbaresco et Le Yang. Medians and means in Riemannian geometry : existence, uniqueness and computation. In Matrix Information Geometry, pages 169–197. Springer, 2013. (cited in page 110)

[Attouch *et al.* 2010] Hédy Attouch, Jérôme Bolte, Patrick Redont et Antoine Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems : An approach based on the Kurdyka-Łojasiewicz inequality. Mathematics of Operations Research, vol. 35, no. 2, pages 438–457, 2010. (cited in pages 41 et 43)

[Azizan *et al.* 2019] N. Azizan, S. Lale et B. Hassibi. Stochastic mirror descent on overparameterized nonlinear models : convergence, implicit regularization and generalization. In ArXiv :1906.03830v1, 2019. (cited in page 123)

[Bao *et al.* 2016] Chenglong Bao, Hui Ji, Yuhui Quan et Zuowei Shen. Dictionary learning for sparse coding : Algorithms and convergence analysis. IEEE transactions on pattern analysis and machine intelligence, vol. 38, no. 7, pages 1356–1369, 2016. (cited in page 58)

[Beck & Teboulle 2009] Amir Beck et Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM journal on imaging sciences, vol. 2, no. 1, pages 183–202, 2009. (cited in page 40)

[Becker *et al.* 2011] S. Becker, J. Bobin et E. Candes. NESTA : a fast and accurate first-order method for sparse recovery. SIAM Journal on Imaging Science, vol. 4, no. 11, 2011. (cited in page 111)

[Ben-Israel & Greville 2003] Adi Ben-Israel et Thomas NE Greville. Generalized inverses : theory and applications, volume 15. Springer Science &amp ; Business Media, 2003. (cited in page 48)

[Berry *et al.* 2007] Michael W Berry, Murray Browne, Amy N Langville, V Paul Pauca et Robert J Plemmons. Algorithms and applications for approximate nonnegative matrix factorization. Computational statistics & data analysis, vol. 52, no. 1, pages 155–173, 2007. (cited in page 46)

[Biswal & Ulmer 1999] Bharat B Biswal et John L Ulmer. Blind source separation of multiple signal sources of fMRI data sets using independent component analysis. Journal of Computer Assisted Tomography, vol. 23, no. 2, pages 265–271, 1999. (cited in page 87)

[Blake *et al.* 2004] Chris A Blake, Filipe B Abdalla, Sarah L Bridle et Steve Rawlings. Cosmology with the SKA. New Astronomy Reviews, vol. 48, no. 11-12, pages 1063–1077, 2004. (cited in page 15)

[Bobin *et al.* 2007] Jérôme Bobin, Jean-Luc Starck, Jalal M Fadili et Yassir Moudden. Sparsity and morphological diversity in blind source separation. IEEE Transactions on Image Processing, vol. 16, no. 11, pages 2662–2674, 2007. (cited in pages 9, 10, 24, 30, 31, 32, 41, 47, 48, 54, 57, 58, 103, 105, 106, 108 et 133)

[Bobin *et al.* 2008] Jérôme Bobin, Jean-Luc Starck, Yassir Moudden et Mohamed Jalal Fadili. Blind Source Separation : The Sparsity Revolution. Advances in Imaging and Electron Physics, vol. 152, no. 1, pages 221–302, 2008. (cited in pages 22, 48, 87, 89 et 146)

[Bobin *et al.* 2013] J Bobin, J-L Starck, F Sureau et S Basak. Sparse component separation for accurate cosmic microwave background estimation. Astronomy & Astrophysics, vol. 550, page A73, 2013. (cited in page 103)

[Bobin *et al.* 2014] J Bobin, F Sureau, J-L Starck, A Rassat et P Paykari. Joint Planck and WMAP CMB map reconstruction. Astronomy & Astrophysics, vol. 563, page A105, 2014. (cited in pages 7, 15 et 23)

[Bobin *et al.* 2015] Jerome Bobin, Jeremy Rapin, Anthony Larue et Jean-Luc Starck. Sparsity and Adaptivity for the Blind Separation of Partially Correlated Sources. IEEE Transanctions on Signal Processing, vol. 63, no. 5, pages 1199–1213, 2015. (cited in pages 9, 14, 48, 55, 62, 88, 92, 103, 118, 133, 139, 156, 158 et 167)

[Bobin *et al.* 2019] Jerome Bobin, I El Hamzaoui, A Picquenot et F Acero. Sparse BSS from Poisson Measurements. 2019. (cited in page 29)

[Bolte *et al.* 2010] Jérôme Bolte, Patrick L Combettes et J-C Pesquet. Alternating proximal algorithm for blind image recovery. In 2010 IEEE International Conference on Image Processing, pages 1673–1676. IEEE, 2010. (cited in page 35)

[Bolte *et al.* 2014] Jérôme Bolte, Shoham Sabach et Marc Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. Mathematical Programming, vol. 146, no. 1-2, pages 459–494, 2014. (cited in pages 9, 10, 31, 35, 41, 44, 57, 73, 88 et 171)

[Bottou 2010] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In Proceedings of COMPSTAT'2010, pages 177–186. Springer, 2010. (cited in page 104)

[Boyd *et al.* 2011] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine learning, vol. 3, no. 1, pages 1–122, 2011. (cited in pages 35 et 38)

[Bracewell & Bracewell 1986] Ronald Newbold Bracewell et Ronald N Bracewell. The Fourier transform and its applications, volume 31999. McGraw-Hill New York, 1986. (cited in page 26)

[Brakel & Bengio 2017] Philemon Brakel et Yoshua Bengio. Learning Independent Features with Adversarial Nets for Non-linear ICA. arXiv preprint arXiv :1710.05050, 2017. (cited in pages 130, 133, 147 et 154)

[Bronstein *et al.* 2005] Alexander M Bronstein, Michael M Bronstein, Michael Zibulevsky et Yehoshua Y Zeevi. Sparse ICA for blind separation of transmitted and reflected images. International Journal of Imaging Systems and Technology, vol. 15, no. 1, pages 84–91, 2005. (cited in page 24)

[Candes & Donoho 2002] Emmanuel J Candes et David L Donoho. Recovering edges in ill-posed inverse problems : Optimality of curvelet frames. Annals of statistics, pages 784–842, 2002. (cited in page 26)

[Candes & Tao 2004] Emmanuel Candes et Terence Tao. Near optimal signal recovery from random projections : Universal encoding strategies ? arXiv preprint math/0410542, 2004. (cited in page 24)

[Candes *et al.* 2008] Emmanuel J Candes, Michael B Wakin et Stephen P Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. Journal of Fourier analysis and applications, vol. 14, no. 5-6, pages 877–905, 2008. (cited in pages 29 et 74)

[Cannelli *et al.* 2016] Loris Cannelli, Francisco Facchinei, Vyacheslav Kungurtsev et Gesualdo Scutari. Asynchronous parallel algorithms for nonconvex optimization. Mathematical Programming, pages 1–34, 2016. (cited in page 104)

[Cavazza *et al.* 2017] Jacopo Cavazza, Pietro Morerio, Benjamin Haeffele, Connor Lane, Vittorio Murino et René Vidal. Dropout as a low-rank regularizer for matrix factorization. arXiv preprint arXiv :1710.05092, 2017. (cited in page 162)

[Chambolle & Pock 2011] Antonin Chambolle et Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of mathematical imaging and vision, vol. 40, no. 1, pages 120–145, 2011. (cited in page 38)

[Chen *et al.* 2001] Scott Shaobing Chen, David L Donoho et Michael A Saunders. Atomic decomposition by basis pursuit. SIAM review, vol. 43, no. 1, pages 129–159, 2001. (cited in page 26)

[Chenot & Bobin 2018] Cécile Chenot et Jérôme Bobin. Blind Source Separation with Outliers in Transformed Domains. SIAM Journal on Imaging Sciences, vol. 11, no. 2, pages 1524–1559, 2018. (cited in page 72)

[Chenot 2017] Cécile Chenot. Parcimonie, diversité morphologique et séparation robuste de sources. PhD thesis, Université Paris-Saclay, 2017. (cited in pages 24, 43, 48, 167 et 177)

[Chouzenoux *et al.* 2014] Emilie Chouzenoux, Jean-Christophe Pesquet et Audrey Repetti. Variable metric forward–backward algorithm for minimizing the sum of a differentiable function and a convex function. Journal of Optimization Theory and Applications, vol. 162, no. 1, pages 107–132, 2014. (cited in page 44)

[Chouzenoux *et al.* 2016] Emilie Chouzenoux, Jean-Christophe Pesquet et Audrey Repetti. A block coordinate variable metric forward–backward algorithm. Journal of Global Optimization, vol. 66, no. 3, pages 457–485, 2016. (cited in pages 35, 44 et 91)

[Combettes & Pesquet 2011] Patrick L Combettes et Jean-Christophe Pesquet. Proximal splitting methods in signal processing. In Fixed-point algorithms for inverse problems in science and engineering, pages 185–212. Springer, 2011. (cited in pages 35 et 38)

[Combettes & Wajs 2005] Patrick L Combettes et Valérie R Wajs. Signal recovery by proximal forward-backward splitting. Multiscale Modeling & Simulation, vol. 4, no. 4, pages 1168–1200, 2005. (cited in pages 35, 39 et 42)

[Comon & Jutten 2010] Pierre Comon et Christian Jutten. Handbook of Blind Source Separation : Independent component analysis and applications. Academic Press, 2010. (cited in pages 7, 20, 21, 22, 89 et 130)

[Comon 1994] Pierre Comon. Independent component analysis, a new concept ? Signal processing, vol. 36, no. 3, pages 287–314, 1994. (cited in page 22)

[Darmois 1953] George Darmois. Analyse générale des liaisons stochastiques : étude particulière de l'analyse factorielle linéaire. Revue de l'Institut international de statistique, pages 2–8, 1953. (cited in pages 21 et 22)

[Daubechies 1988] Ingrid Daubechies. Orthonormal bases of compactly supported wavelets. Communications on pure and applied mathematics, vol. 41, no. 7, pages 909–996, 1988. (cited in page 26)

[Davies 2004] Mike Davies. Identifiability issues in noisy ICA. IEEE Signal processing letters, vol. 11, no. 5, pages 470–473, 2004. (cited in pages 23 et 163)

[Davis *et al.* 2016] Damek Davis, Brent Edmunds et Madeleine Udell. The sound of APALM clapping : Faster nonsmooth nonconvex optimization with stochastic asynchronous PALM. In Advances in Neural Information Processing Systems, pages 226–234, 2016. (cited in pages 104 et 107)

[Deledalle *et al.* 2014] Charles-Alban Deledalle, Samuel Vaiter, Jalal Fadili et Gabriel Peyré. Stein Unbiased GrAdient estimator of the Risk (SUGAR) for

multiple parameter selection. SIAM Journal on Imaging Sciences, vol. 7, no. 4, pages 2448–2487, 2014. (cited in page 69)

[Deville & Duarte 2015] Yannick Deville et Leonardo Tomazeli Duarte. An overview of blind source separation methods for linear-quadratic and post-nonlinear mixtures. In International Conference on Latent Variable Analysis and Signal Separation, pages 155–167. Springer, 2015. (cited in pages 130 et 157)

[Do & Vetterli 2005] Minh N Do et Martin Vetterli. The contourlet transform : an efficient directional multiresolution image representation. IEEE Transactions on image processing, vol. 14, no. 12, pages 2091–2106, 2005. (cited in page 26)

[Dobigeon *et al.* 2014] Nicolas Dobigeon, Jean-Yves Tourneret, Cédric Richard, José Carlos M Bermudez, Stephen McLaughlin et Alfred O Hero. Nonlinear unmixing of hyperspectral images : Models and algorithms. IEEE Signal Processing Magazine, vol. 31, no. 1, pages 82–94, 2014. (cited in page 21)

[Donoho & Stodden 2004] David Donoho et Victoria Stodden. When does non-negative matrix factorization give a correct decomposition into parts ? In Advances in neural information processing systems, pages 1141–1148, 2004. (cited in page 23)

[Donoho *et al.* 2006] David L Donohoet al. Compressed sensing. IEEE Transactions on information theory, vol. 52, no. 4, pages 1289–1306, 2006. (cited in page 24)

[Douglas & Rachford 1956] Jim Douglas et Henry H Rachford. On the numerical solution of heat conduction problems in two and three space variables. Transactions of the American mathematical Society, vol. 82, no. 2, pages 421–439, 1956. (cited in page 38)

[Duarte & Jutten 2014] Leonardo Tomazeli Duarte et Christian Jutten. Design of smart ion-selective electrode arrays based on source separation through nonlinear independent component analysis. Oil & Gas Science and Technology–Revue d'IFP Energies nouvelles, vol. 69, no. 2, pages 293–306, 2014. (cited in page 21)

[Duarte *et al.* 2012] Leonardo T Duarte, Rafael A Ando, Romis Attux, Yannick Deville et Christian Jutten. Separation of sparse signals in overdetermined linear-quadratic mixtures. In International Conference on Latent Variable Analysis and Signal Separation, pages 239–246. Springer, 2012. (cited in page 132)

[Duarte *et al.* 2015] Leonardo Tomazeli Duarte, Ricardo Suyama, Romis Attux, João Marcos Travassos Romano et Christian Jutten. A sparsity-based method for blind compensation of a memoryless nonlinear distortion : Application to ion-selective electrodes. IEEE Sensors Journal, vol. 15, no. 4, pages 2054–2061, 2015. (cited in pages 132 et 157)

[Duong *et al.* 2010] Ngoc QK Duong, Emmanuel Vincent et Rémi Gribonval. Under-determined reverberant audio source separation using a full-rank

spatial covariance model. IEEE Transactions on Audio, Speech, and Language Processing, vol. 18, no. 7, pages 1830–1840, 2010. (cited in pages 7 et 15)

[Easley *et al.* 2008] Glenn Easley, Demetrio Labate et Wang-Q Lim. Sparse directional image representations using the discrete shearlet transform. Applied and Computational Harmonic Analysis, vol. 25, no. 1, pages 25–46, 2008. (cited in page 26)

[Eches & Guillaume 2013] Olivier Eches et Mireille Guillaume. A bilinear–bilinear nonnegative matrix factorization method for hyperspectral unmixing. IEEE Geoscience and Remote Sensing Letters, vol. 11, no. 4, pages 778–782, 2013. (cited in page 144)

[Eckart & Young 1936] Carl Eckart et Gale Young. The approximation of one matrix by another of lower rank. Psychometrika, vol. 1, no. 3, pages 211–218, 1936. (cited in page 22)

[Ehsandoust *et al.* 2016] Bahram Ehsandoust, Bertrand Rivet, Christian Jutten et Massoud Babaie-Zadeh. Nonlinear blind source separation for sparse sources. In 2016 24th European Signal Processing Conference (EUSIPCO), pages 1583–1587. IEEE, 2016. (cited in pages 22, 132, 133 et 147)

[Ehsandoust *et al.* 2017] Bahram Ehsandoust, Massoud Babaie-Zadeh, Bertrand Rivet et Christian Jutten. Blind source separation in nonlinear mixtures : separability and a basic algorithm. IEEE Transactions on Signal Processing, vol. 65, no. 16, pages 4339–4352, 2017. (cited in pages 130, 146, 167 et 168)

[Elad & Aharon 2006] Michael Elad et Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. IEEE Transactions on Image processing, vol. 15, no. 12, pages 3736–3745, 2006. (cited in page 24)

[Elad 2010] Michael Elad. Sparse and redundant representations : from theory to applications in signal and image processing. Springer Science &amp ; Business Media, 2010. (cited in page 24)

[Eldar 2009] Yonina C Eldar. Generalized SURE for exponential families : Applications to regularization. IEEE Transactions on Signal Processing, vol. 57, no. 2, pages 471–481, 2009. (cited in pages 69, 162 et 183)

[Feng & Kowalski 2018] Fangchen Feng et Matthieu Kowalski. Revisiting sparse ICA from a synthesis point of view : Blind source separation for over and underdetermined mixtures. Signal Processing, vol. 152, pages 165–177, 2018. (cited in page 69)

[Févotte *et al.* 2009] Cédric Févotte, Nancy Bertin et Jean-Louis Durrieu. Nonnegative matrix factorization with the Itakura-Saito divergence : With application to music analysis. Neural computation, vol. 21, no. 3, pages 793–830, 2009. (cited in pages 7 et 15)

[Fletcher *et al.* 2008] P Thomas Fletcher, Suresh Venkatasubramanian et Sarang Joshi. Robust statistics on Riemannian manifolds via the geometric median.

In 2008 IEEE Conference on Computer Vision and Pattern Recognition, pages 1–8. IEEE, 2008. (cited in page 110)

[Gidel *et al.* 2019] G. Gidel, F. Bach et S. Lacoste-Julien. Implicit Regularization of discrete gradient dynamics in deep linear neural networks. In ArXiv :1904.13262v1, 2019. (cited in page 123)

[Gillis & Glineur 2012] Nicolas Gillis et François Glineur. Accelerated multiplicative updates and hierarchical ALS algorithms for nonnegative matrix factorization. Neural Computation, vol. 24, no. 4, pages 1085–1105, 2012. (cited in pages 21, 23, 88 et 89)

[Giryes *et al.* 2011] Raja Giryes, Michael Elad et Yonina C Eldar. The projected GSURE for automatic parameter tuning in iterative shrinkage methods. Applied and Computational Harmonic Analysis, vol. 30, no. 3, pages 407–422, 2011. (cited in pages 69, 162 et 183)

[Golub *et al.* 1979] Gene H Golub, Michael Heath et Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. Technometrics, vol. 21, no. 2, pages 215–223, 1979. (cited in page 69)

[Gribonval & Schnass 2010] Rémi Gribonval et Karin Schnass. Dictionary Identification—Sparse Matrix-Factorization via ℓ_1-Minimization. IEEE Transactions on Information Theory, vol. 56, no. 7, pages 3523–3539, 2010. (cited in pages 21, 29, 30 et 156)

[Gribonval *et al.* 2015] Rémi Gribonval, Rodolphe Jenatton et Francis Bach. Sparse and spurious : dictionary learning with noise and outliers. IEEE Transactions on Information Theory, vol. 61, no. 11, pages 6298–6319, 2015. (cited in pages 21, 156 et 158)

[Gunasekar *et al.* 2017] G. Gunasekar, B. Woodworth, S. Bhojanapalli, B. Neyshabur et N. Srebro. Implicit Regularization in Matrix Factorization. In Proceedings NIPS 2017, 2017. (cited in pages 121 et 123)

[Haar 1910] Alfred Haar. Zur theorie der orthogonalen funktionensysteme. Mathematische Annalen, vol. 69, no. 3, pages 331–371, 1910. (cited in page 26)

[Hamzaoui & Bobin 2018] I El Hamzaoui et J Bobin. Sparse component separation from Poisson measurements. arXiv preprint arXiv :1812.04370, 2018. (cited in page 29)

[Hansen & O'Leary 1993] Per Christian Hansen et Dianne Prost O'Leary. The use of the L-curve in the regularization of discrete ill-posed problems. SIAM Journal on Scientific Computing, vol. 14, no. 6, pages 1487–1503, 1993. (cited in page 69)

[Hardt *et al.* 2015] Moritz Hardt, Benjamin Recht et Yoram Singer. Train faster, generalize better : Stability of stochastic gradient descent. arXiv preprint arXiv :1509.01240, 2015. (cited in page 123)

[Hastie *et al.* 2015] Trevor Hastie, Rahul Mazumder, Jason D Lee et Reza Zadeh. Matrix completion and low-rank SVD via fast alternating least squares. The Journal of Machine Learning Research, vol. 16, no. 1, pages 3367–3402, 2015. (cited in page 105)

[Hien *et al.* 2019] Le Thi Khanh Hien, Nicolas Gillis et Panagiotis Patrinos. Inertial Block Mirror Descent Method for Non-Convex Non-Smooth Optimization. arXiv preprint arXiv :1903.01818, 2019. (cited in page 44)

[Hochreiter & Schmidhuber 1997] Sepp Hochreiter et Jürgen Schmidhuber. Flat minima. Neural Computation, vol. 9, no. 1, pages 1–42, 1997. (cited in page 124)

[Honkela *et al.* 2007] Antti Honkela, Harri Valpola, Alexander Ilin et Juha Karhunen. Blind separation of nonlinear mixtures by variational Bayesian learning. Digital Signal Processing, vol. 17, no. 5, pages 914–934, 2007. (cited in pages 130, 147 et 158)

[Hosseini & Deville 2003] Shahram Hosseini et Yannick Deville. Blind separation of linear-quadratic mixtures of real sources using a recurrent structure. In International Work-Conference on Artificial Neural Networks, pages 241–248. Springer, 2003. (cited in page 157)

[Hotelling 1933] Harold Hotelling. Analysis of a complex of statistical variables into principal components. Journal of educational psychology, vol. 24, no. 6, page 417, 1933. (cited in page 22)

[Hoyer 2004] Patrik O Hoyer. Non-negative matrix factorization with sparseness constraints. Journal of machine learning research, vol. 5, no. Nov, pages 1457–1469, 2004. (cited in page 24)

[Huang *et al.* 2017] Gao Huang, Zhuang Liu, Laurens Van Der Maaten et Kilian Q Weinberger. Densely connected convolutional networks. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2261–2269. IEEE, 2017. (cited in page 145)

[Hyvarinen & Morioka 2017] AJ Hyvarinen et Hiroshi Morioka. Nonlinear ICA of temporally dependent stationary sources. Proceedings of Machine Learning Research, 2017. (cited in pages 130 et 154)

[Hyvarinen *et al.* 1998] Aapo Hyvarinen, Erkki Oja, Patrik Hoyer et Jarmo Hurri. Image feature extraction by sparse coding and independent component analysis. In Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No. 98EX170), volume 2, pages 1268–1273. IEEE, 1998. (cited in page 24)

[Ivezic *et al.* 2008] Zeljko Ivezic, JA Tyson, B Abel, E Acosta, R Allsman, Y AlSayyad, SF Anderson, J Andrew, R Angel, G Angeliet al. LSST : from science drivers to reference design and anticipated data products. arXiv preprint arXiv :0805.2366, 2008. (cited in page 15)

[Jenatton *et al.* 2010] Rodolphe Jenatton, Julien Mairal, Guillaume Obozinski et Francis R Bach. Proximal Methods for Sparse Hierarchical Dictionary Learning. In ICML, volume 1, page 2. Citeseer, 2010. (cited in page 35)

[Jezierska *et al.* 2012] Anna Jezierska, Emilie Chouzenoux, Jean-Christophe Pesquet et Hugues Talbot. A primal-dual proximal splitting approach for restoring data corrupted with Poisson-Gaussian noise. In 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1085–1088. IEEE, 2012. (cited in page 35)

[Jimenez 2006] Guillermo Bedoya Jimenez. Non-linear blind signal separation for chemical solid-state sensor arrays. PhD thesis, Universitat Politècnica de Catalunya, 2006. (cited in page 21)

[Jolliffe 1986] IT Jolliffe. Principal component analysis. 1986. Spring-verlag, New York, vol. 2, page 29, 1986. (cited in page 22)

[Jung *et al.* 2000] Tzyy-Ping Jung, Scott Makeig, Colin Humphries, Te-Won Lee, Martin J Mckeown, Vicente Iragui et Terrence J Sejnowski. Removing electroencephalographic artifacts by blind source separation. Psychophysiology, vol. 37, no. 2, pages 163–178, 2000. (cited in pages 7 et 15)

[Kampffmeyer 2015] Michael Christian Kampffmeyer. Parallelization of the alternating-least-squares algorithm with weighted regularization for efficient gpu execution in recommender systems. Master's thesis, UiT Norges arktiske universitet, 2015. (cited in page 105)

[Kervazo *et al.* 2018] C Kervazo, Jérôme Bobin et Cecile Chenot. Blind separation of a large number of sparse sources. Signal Processing, vol. 150, pages 157–165, 2018. (cited in page 72)

[Keskar *et al.* 2016] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy et Ping Tak Peter Tang. On large-batch training for deep learning : Generalization gap and sharp minima. arXiv preprint arXiv :1609.04836, 2016. (cited in page 123)

[Kim & Park 2008] Jingu Kim et Haesun Park. Sparse nonnegative matrix factorization for clustering. Rapport technique, Georgia Institute of Technology, 2008. (cited in page 24)

[Kim *et al.* 2008] Dongmin Kim, Suvrit Sra et Inderjit S Dhillon. Fast Projection-Based Methods for the Least Squares Nonnegative Matrix Approximation Problem. Statistical Analysis and Data Mining : The ASA Data Science Journal, vol. 1, no. 1, pages 38–51, 2008. (cited in pages 46, 172 et 173)

[Koldovsky *et al.* 2006] Zbynek Koldovsky, Petr Tichavsky et Erkki Oja. Efficient variant of algorithm FastICA for independent component analysis attaining the Cramér-Rao lower bound. IEEE Transactions on neural networks, vol. 17, no. 5, pages 1265–1277, 2006. (cited in pages 24 et 77)

[Lanaras *et al.* 2015] Charis Lanaras, Emmanuel Baltsavias et Konrad Schindler. Hyperspectral super-resolution by coupled spectral unmixing. In Proceedings

of the IEEE International Conference on Computer Vision, pages 3586–3594, 2015. (cited in page 58)

[Le Pennec & Mallat 2000] Erwan Le Pennec et Stéphane Mallat. Image compression with geometrical wavelets. In Proceedings 2000 International Conference on Image Processing (Cat. No. 00CH37101), volume 1, pages 661–664. IEEE, 2000. (cited in page 24)

[Le Pennec & Mallat 2005] Erwan Le Pennec et Stéphane Mallat. Sparse geometric image representations with bandelets. IEEE transactions on image processing, vol. 14, no. 4, pages 423–438, 2005. (cited in page 26)

[Le Roux *et al.* 2015] Jonathan Le Roux, Felix J Weninger et John R Hershey. Sparse NMF–half-baked or well done ? Mitsubishi Electric Research Labs (MERL), Cambridge, MA, USA, Tech. Rep., no. TR2015-023, 2015. (cited in page 22)

[LeCun *et al.* 2012] Yann A LeCun, Léon Bottou, Genevieve B Orr et Klaus-Robert Müller. Efficient backprop. In Neural networks : Tricks of the trade, pages 9–48. Springer, 2012. (cited in page 123)

[Lee & Seung 1999] Daniel D Lee et H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. Nature, vol. 401, no. 6755, page 788, 1999. (cited in page 23)

[Lee *et al.* 2016] Jason D Lee, Max Simchowitz, Michael I Jordan et Benjamin Recht. Gradient descent converges to minimizers. arXiv preprint arXiv :1602.04915, 2016. (cited in page 123)

[Li *et al.* 2006] Yuanqing Li, Shun-Ichi Amari, Andrzej Cichocki, Daniel WC Ho et Shengli Xie. Underdetermined blind source separation based on sparse representation. IEEE Transactions on Signal Processing, vol. 54, no. 2, pages 423–437, 2006. (cited in pages 22 et 24)

[Liu & Wang 2013] Qingju Liu et Wenwu Wang. Show-through removal for scanned images using non-linear NMF with adaptive smoothing. In 2013 IEEE China Summit and International Conference on Signal and Information Processing, pages 650–654. IEEE, 2013. (cited in page 144)

[Longo *et al.* 2017] Giuseppe Longo, Massimo Brescia et Stefano Cavuoti. The astronomical data deluge : the template case of photometric redshifts. In CEUR Workshop Proceedings, volume 2022, pages 27–29, 2017. (cited in page 15)

[Lukas 2006] Mark A Lukas. Robust generalized cross-validation for choosing the regularization parameter. Inverse Problems, vol. 22, no. 5, page 1883, 2006. (cited in page 69)

[Maas *et al.* 2013] Andrew L Maas, Awni Y Hannun et Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In Proc. ICML, volume 30, page 3, 2013. (cited in page 145)

[Madrolle *et al.* 2018] Stéphanie Madrolle, Pierre Grangeat et Christian Jutten. A Linear-Quadratic Model for the Quantification of a Mixture of Two Diluted Gases with a Single Metal Oxide Sensor. Sensors, vol. 18, no. 6, page 1785, 2018. (cited in page 21)

[Mairal *et al.* 2009] Julien Mairal, Francis Bach, Jean Ponce et Guillermo Sapiro. Online dictionary learning for sparse coding. In Proceedings of the 26th annual international conference on machine learning, pages 689–696. ACM, 2009. (cited in pages 104, 105, 107 et 114)

[Mairal *et al.* 2014] Julien Mairal, Francis Bach, Jean Ponceet al. Sparse modeling for image and vision processing. Foundations and Trends® in Computer Graphics and Vision, vol. 8, no. 2-3, pages 85–283, 2014. (cited in pages 20, 24, 26 et 27)

[Mallat 1999] Stéphane Mallat. A wavelet tour of signal processing. Academic Press, 1999. (cited in page 24)

[Meganem *et al.* 2011] Ines Meganem, Philippe Deliot, Xavier Briottet, Yannick Deville et Shahram Hosseini. Physical modelling and non-linear unmixing method for urban hyperspectral images. In 2011 3rd Workshop on Hyperspectral Image and Signal Processing : Evolution in Remote Sensing (WHISPERS), pages 1–4. IEEE, 2011. (cited in page 144)

[Meganem *et al.* 2014] Ines Meganem, Yannick Deville, Shahram Hosseini, Philippe Deliot et Xavier Briottet. Linear-quadratic blind source separation using NMF to unmix urban hyperspectral images. IEEE Transactions on Signal Processing, vol. 62, no. 7, pages 1822–1833, 2014. (cited in page 144)

[Mensch *et al.* 2018] Arthur Mensch, Julien Mairal, Bertrand Thirion et Gaël Varoquaux. Stochastic subsampling for factorizing huge matrices. IEEE Transactions on Signal Processing, vol. 66, no. 1, pages 113–128, 2018. (cited in pages 26, 58, 104, 105, 107 et 162)

[Merrikh-Bayat *et al.* 2011] Farnood Merrikh-Bayat, Massoud Babaie-Zadeh et Christian Jutten. Linear-quadratic blind source separating structure for removing show-through in scanned documents. International Journal on Document Analysis and Recognition (IJDAR), vol. 14, no. 4, pages 319–333, 2011. (cited in pages 18 et 21)

[Mianjy *et al.* 2018] Poorya Mianjy, Raman Arora et Rene Vidal. On the implicit bias of dropout. arXiv preprint arXiv :1806.09777, 2018. (cited in page 162)

[Nascimento & Dias 2005] José MP Nascimento et Jose MB Dias. Does independent component analysis play a role in unmixing hyperspectral data ? IEEE Transactions on Geoscience and Remote Sensing, vol. 43, no. 1, pages 175–187, 2005. (cited in page 23)

[Neelakantan *et al.* 2015] Arvind Neelakantan, Luke Vilnis, Quoc V Le, Ilya Sutskever, Lukasz Kaiser, Karol Kurach et James Martens. Adding gradient noise improves learning for very deep networks. arXiv preprint arXiv :1511.06807, 2015. (cited in page 124)

[Negro *et al.* 2016] Francesco Negro, Silvia Muceli, Anna Margherita Castronovo, Ales Holobar et Dario Farina. Multi-channel intramuscular and surface EMG decomposition by convolutive blind source separation. Journal of neural engineering, vol. 13, no. 2, page 026027, 2016. (cited in pages 7 et 15)

[Nesterov 2005] Yu Nesterov. Smooth minimization of non-smooth functions. Mathematical programming, vol. 103, no. 1, pages 127–152, 2005. (cited in page 111)

[Neu & Rosasco 2018] Gergely Neu et Lorenzo Rosasco. Iterate averaging as regularization for stochastic gradient descent. arXiv preprint arXiv :1802.08009, 2018. (cited in page 113)

[Neyshabur *et al.* 2017] B. Neyshabur, R. Tomioka, R. Salakhutdinov et N. Srebro. Geometry of Optimization and Implicit Regularizationin Deep Learning. In ArXiv :1705.03071v1, 2017. (cited in page 121)

[Neyshabur 2017] B. Neyshabur. Implicit Regularization in Deep Learning. PhD thesis, Toyota Technological Institute Chicago, https ://arxiv.org/abs/1709.01953, 2017. (cited in pages 30, 121 et 123)

[Olshausen & Field 1996] Bruno A Olshausen et David J Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature, vol. 381, no. 6583, page 607, 1996. (cited in page 26)

[Ozerov & Févotte 2010] Alexey Ozerov et Cédric Févotte. Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation. IEEE Transactions on Audio, Speech, and Language Processing, vol. 18, no. 3, pages 550–563, 2010. (cited in pages 7 et 15)

[Paatero & Tapper 1994] Pentti Paatero et Unto Tapper. Positive matrix factorization : A non-negative factor model with optimal utilization of error estimates of data values. Environmetrics, vol. 5, no. 2, pages 111–126, 1994. (cited in pages 23, 41 et 46)

[Parikh *et al.* 2014] Neal Parikh, Stephen Boydet al. Proximal algorithms. Foundations and Trends® in Optimization, vol. 1, no. 3, pages 127–239, 2014. (cited in page 35)

[Patrascu & Necoara 2015] Andrei Patrascu et Ion Necoara. Efficient random coordinate descent algorithms for large-scale structured nonconvex optimization. Journal of Global Optimization, vol. 61, no. 1, pages 19–46, 2015. (cited in page 91)

[Pereyra *et al.* 2015] Marcelo Pereyra, José M Bioucas-Dias et Mário AT Figueiredo. Maximum-a-posteriori estimation with unknown regularisation parameters. In 2015 23rd European Signal Processing Conference (EUSIPCO), pages 230–234. IEEE, 2015. (cited in page 69)

[Pierre *et al.* 2015] Fabien Pierre, J-F Aujol, Aurélie Bugeau, Nicolas Papadakis et V-T Ta. Luminance-chrominance model for image colorization. SIAM Journal on Imaging Sciences, vol. 8, no. 1, pages 536–563, 2015. (cited in page 58)

[Pock & Sabach 2016] Thomas Pock et Shoham Sabach. Inertial proximal alternating linearized minimization (iPALM) for nonconvex and nonsmooth problems. SIAM Journal on Imaging Sciences, vol. 9, no. 4, pages 1756–1787, 2016. (cited in page 44)

[Poh *et al.* 2010] Ming-Zher Poh, Daniel J McDuff et Rosalind W Picard. Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. Optics express, vol. 18, no. 10, pages 10762–10774, 2010. (cited in pages 7 et 15)

[Polyak & Juditsky 1992] Boris T Polyak et Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. SIAM Journal on Control and Optimization, vol. 30, no. 4, pages 838–855, 1992. (cited in page 113)

[Powell 1973] Michael JD Powell. On search directions for minimization algorithms. Mathematical programming, vol. 4, no. 1, pages 193–201, 1973. (cited in page 42)

[Puigt *et al.* 2012] Matthieu Puigt, Anthony Griffin et Athanasios Mouchtaris. Nonlinear blind mixture identification using local source sparsity and functional data clustering. In Sensor Array and Multichannel Signal Processing Workshop (SAM), 2012 IEEE 7th, pages 481–484. IEEE, 2012. (cited in pages 132, 133 et 154)

[Raguet *et al.* 2013] Hugo Raguet, Jalal Fadili et Gabriel Peyré. A generalized forward-backward splitting. SIAM Journal on Imaging Sciences, vol. 6, no. 3, pages 1199–1226, 2013. (cited in pages 38 et 40)

[Rapin *et al.* 2013] Jérémy Rapin, Jérôme Bobin, Anthony Larue et Jean-Luc Starck. Sparse and non-negative BSS for noisy data. IEEE Transactions on Signal Processing, vol. 61, no. 22, pages 5620–5632, 2013. (cited in pages 29, 37, 47 et 144)

[Rapin *et al.* 2014] Jérémy Rapin, Jérôme Bobin, Anthony Larue et Jean-Luc Starck. NMF with sparse regularizations in transformed domains. SIAM Journal on Imaging Sciences, vol. 7, no. 4, pages 2020–2047, 2014. (cited in pages 16, 88 et 144)

[Rapin 2014] Jérémy Rapin. Décompositions parcimonieuses pour l'analyse avancée de données en spectrométrie pour la Santé. PhD thesis, Université Paris Sud-Paris XI, 2014. (cited in pages 17, 18, 24, 26, 27, 124, 144 et 177)

[Repetti *et al.* 2015] Audrey Repetti, Mai Quyen Pham, Laurent Duval, Emilie Chouzenoux et Jean-Christophe Pesquet. Euclid in a Taxicab : Sparse Blind Deconvolution with Smoothed ℓ_1/ℓ_2 Regularization. IEEE Signal Processing Letters, vol. 22, no. 5, pages 539–543, 2015. (cited in page 58)

[Ruppert 1988] David Ruppert. Efficient estimations from a slowly convergent Robbins-Monro process. Rapport technique, Cornell University Operations Research and Industrial Engineering, 1988. (cited in page 113)

[Sato *et al.* 2017] Hiroyuki Sato, Hiroyuki Kasai et Bamdev Mishra. Riemannian stochastic variance reduced gradient. arXiv preprint arXiv :1702.05594, 2017. (cited in page 113)

[Smith *et al.* 2017] Samuel L Smith, Pieter-Jan Kindermans, Chris Ying et Quoc V Le. Don't decay the learning rate, increase the batch size. arXiv preprint arXiv :1711.00489, 2017. (cited in pages 123 et 124)

[Starck *et al.* 2007] Jean-Luc Starck, Jalal Fadili et Fionn Murtagh. The undecimated wavelet decomposition and its reconstruction. IEEE Transactions on Image Processing, vol. 16, no. 2, pages 297–309, 2007. (cited in pages 29 et 99)

[Starck *et al.* 2010] Jean-Luc Starck, Fionn Murtagh et Jalal M Fadili. Sparse Image and Signal Processing : Wavelets, Curvelets, Morphological Diversity. Cambridge University Press, 2010. (cited in pages 24, 26, 48, 120 et 139)

[Stein 1981] Charles M Stein. Estimation of the mean of a multivariate normal distribution. The annals of Statistics, pages 1135–1151, 1981. (cited in pages 69 et 183)

[Teflioudi *et al.* 2012] Christina Teflioudi, Faraz Makari et Rainer Gemulla. Distributed matrix completion. In 2012 ieee 12th international conference on data mining, pages 655–664. IEEE, 2012. (cited in page 105)

[Theis & Amari 2004] Fabian J Theis et Shun-ichi Amari. Postnonlinear overcomplete blind source separation using sparse sources. In International Conference on Independent Component Analysis and Signal Separation, pages 718–725. Springer, 2004. (cited in page 132)

[Thouvenin *et al.* 2015] Pierre-Antoine Thouvenin, Nicolas Dobigeon et Jean-Yves Tourneret. Estimation de variabilité pour le démélange non-supervisé d'images hyperspectrales. In 25eme Colloque Groupe de Recherche et d'Etudes du Traitement du Signal et des Images (GRETSI 2015), pages pp–1, 2015. (cited in page 58)

[Thouvenin *et al.* 2016] Pierre-Antoine Thouvenin, Nicolas Dobigeon et Jean-Yves Tourneret. Online unmixing of multitemporal hyperspectral images accounting for spectral variability. IEEE Transactions on Image Processing, vol. 25, no. 9, pages 3979–3990, 2016. (cited in page 104)

[Thouvenin *et al.* 2018] Pierre-Antoine Thouvenin, Nicolas Dobigeon et Jean-Yves Tourneret. Partially asynchronous distributed unmixing of hyperspectral images. IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 4, pages 2009–2021, 2018. (cited in page 104)

[Toumi *et al.* 2013] Ichrak Toumi, Bruno Torrésani et Stefano Caldarelli. Effective processing of pulse field gradient NMR of mixtures by blind source separation. Analytical Chemistry, vol. 85, no. 23, pages 11344–11351, 2013. (cited in page 98)

[Tripuraneni *et al.* 2018] Nilesh Tripuraneni, Nicolas Flammarion, Francis Bach et Michael I Jordan. Averaging stochastic gradient descent on Riemannian manifolds. arXiv preprint arXiv :1802.09128, 2018. (cited in page 113)

[Tseng 2001] Paul Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. Journal of Optimization Theory and Applications, vol. 109, no. 3, pages 475–494, 2001. (cited in pages 9, 41, 42, 88 et 171)

[Van Vaerenbergh & Santamaría 2006] Steven Van Vaerenbergh et Ignacio Santamaría. A spectral clustering approach to underdetermined postnonlinear blind source separation of sparse sources. IEEE Transactions on Neural Networks, vol. 17, no. 3, pages 811–814, 2006. (cited in page 132)

[Vandaele *et al.* 2016] Arnaud Vandaele, Nicolas Gillis, François Glineur et Daniel Tuyttens. Heuristics for exact nonnegative matrix factorization. Journal of Global Optimization, vol. 65, no. 2, pages 369–400, 2016. (cited in page 58)

[Vidal & Pereyra 2018] Ana Fernandez Vidal et Marcelo Pereyra. Maximum Likelihood Estimation of Regularisation Parameters. In 2018 25th IEEE International Conference on Image Processing (ICIP), pages 1742–1746. IEEE, 2018. (cited in page 69)

[Vincent *et al.* 2003] Emmanuel Vincent, Cédric Févotte, Rémi Gribonval, Laurent Benaroya, Xavier Rodet, Axel Röbel, Eric Le Carpentier et Frédéric Bimbot. A tentative typology of audio source separation tasks. In 4th Int. Symp. on Independent Component Analysis and Blind Signal Separation (ICA), pages 715–720, 2003. (cited in pages 7 et 15)

[Vincent *et al.* 2006] Emmanuel Vincent, Rémi Gribonval et Cédric Févotte. Performance measurement in blind audio source separation. IEEE transactions on audio, speech, and language processing, vol. 14, no. 4, pages 1462–1469, 2006. (cited in pages 146 et 167)

[Vincent *et al.* 2011] Emmanuel Vincent, Maria G Jafari, Samer A Abdallah, Mark D Plumbley et Mike E Davies. Probabilistic modeling paradigms for audio source separation. In Machine Audition : Principles, Algorithms and Systems, pages 162–185. IGI Global, 2011. (cited in pages 7 et 15)

[Xing *et al.* 2018] C. Xing, D. Arpit, C. Tsirigotis et Y. Bengio. A walk with sgd. In arXiv :1802.08770, 2018. (cited in pages 13, 123 et 124)

[Xu & Yin 2013] Yangyang Xu et Wotao Yin. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. SIAM Journal on imaging sciences, vol. 6, no. 3, pages 1758–1789, 2013. (cited in pages 27, 35, 42, 44 et 177)

[Xu & Yin 2014] Yangyang Xu et Wotao Yin. A globally convergent algorithm for nonconvex optimization based on block coordinate update. arXiv preprint arXiv :1410.1386, 2014. (cited in pages 44 et 88)

[Xu *et al.* 2019] Jiaxin Xu, Jerome Bobin, Anne De de Vismes Ott et Christophe Bobin. Spectral unmixing for activity estimation in Gamma-Ray Spectrometry. 2019. (cited in page 120)

[Zangwill 1969] Willard I Zangwill. Nonlinear programming : a unified approach, volume 196. Prentice-Hall Englewood Cliffs, NJ, 1969. (cited in page 42)

[Zhang *et al.* 2016] Hongyi Zhang, Sashank J Reddi et Suvrit Sra. Riemannian SVRG : Fast stochastic optimization on Riemannian manifolds. In Advances in Neural Information Processing Systems, pages 4592–4600, 2016. (cited in page 113)

[Zhou *et al.* 2008] Yunhong Zhou, Dennis Wilkinson, Robert Schreiber et Rong Pan. Large-scale parallel collaborative filtering for the netflix prize. In International conference on algorithmic applications in management, pages 337–348. Springer, 2008. (cited in page 105)

[Zibulevsky & Pearlmutter 2001] Michael Zibulevsky et Barak A Pearlmutter. Blind source separation by sparse decomposition in a signal dictionary. Neural Computation, vol. 13, no. 4, pages 863–882, 2001. (cited in pages 9, 22 et 24)

[Zibulevsky 2003] Michael Zibulevsky. Blind source separation with relative Newton method. In Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Source Separation (ICA 2003), Nara, Japan, pages 897–902, 2003. (cited in pages 21, 24, 77 et 88)

université PARIS-SACLAY
ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Stratégies d'optimisation pour la séparation aveugle de sources parcimonieuses grande échelle

Mots clés : Séparation Aveugle de Sources Grande Échelle, Représentations Parcimonieuses, Optimisation Multi-Convexe, Choix de Paramètres de Régularisation, Agrégation d'Estimateurs sur Variétés Riemanniennes, Séparation Aveugle de Sources Non-Linéaire.

Résumé : Lors des dernières décennies, la Séparation Aveugle de Sources (BSS) est devenue un outil de premier plan pour le traitement de données multi-valuées. L'objectif de ce doctorat est cependant d'étudier les cas *grande échelle*, pour lesquels la plupart des algorithmes classiques obtiennent des performances dégradées. Ce document s'articule en quatre parties, traitant chacune un aspect du problème: i) l'introduction d'algorithmes robustes de BSS parcimonieuse ne nécessitant qu'un seul lancement (malgré un choix d'hyper-paramètres délicat) et fortement étayés mathématiquement; ii) la proposition d'une méthode permettant de maintenir une haute qualité de séparation malgré un nombre de sources important: iii) la modification d'un algorithme classique de BSS parcimonieuse pour l'application sur des données de grandes tailles; et iv) une extension au problème de BSS parcimonieuse non-linéaire.
Les méthodes proposées ont été amplement testées, tant sur données simulées que réalistes, pour démontrer leur qualité. Des interprétations détaillées des résultats sont proposées.

Title : Optimization framework for large-scale sparse blind source separation

Keywords : Large-Scale Blind Source Separation, Sparse Representations, Multi-Convex Optimization and Block Coordinate Methods, Regularization Parameter Choice, Estimator Aggregation on Riemannian Manifolds, Non-Linear Blind Source Separation.

Abstract : During the last decades, Blind Source Separation (BSS) has become a key analysis tool to study multi-valued data. The objective of this thesis is however to focus on *large-scale* settings, for which most classical algorithms fail. More specifically, it is subdivided into four sub-problems taking their roots around the large-scale sparse BSS issue: i) introduce a mathematically sound robust sparse BSS algorithm which does not require any relaunch (despite a difficult hyper-parameter choice); ii) introduce a method being able to maintain high quality separations even when a large-number of sources needs to be estimated; iii) make a classical sparse BSS algorithm scalable to large-scale datasets; and iv) an extension to the non-linear sparse BSS problem.
The methods we propose are extensively tested on both simulated and realistic experiments to demonstrate their quality. In-depth interpretations of the results are proposed.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France