Détection et classification de cibles multispectrales dans l'infrarouge

Florian Maire

THESE DE DOCTORAT TELECOM SUDPARIS et UNIVERSITE PIERRE ET MARIE CURIE

Spécialité : Probabilités et Statistiques

Ecole doctorale : Informatique, Télécommunications et Electronique de Paris

Présentée par

Florian MAIRE

Pour obtenir le grade de
DOCTEUR DE TELECOM SUDPARIS

DETECTION ET CLASSIFICATION DE CIBLES MULTISPECTRALES DANS L'INFRAROUGE

dirigée par Randal DOUC, Professeur à Telecom SudParis,
co-dirigée par Eric MOULINES, Professeur à Telecom ParisTech,
encadrée par Sidonie LEFEBVRE, Ingénieur de Recherche à l'ONERA.

soutenue le : 14 Février 2014

devant le jury composé de :

M^{me} Stéphanie ALLASSONNIÈRE	Ecole Polytechnique	examinateur
M. Jacques BLANC-TALON	DGA	examinateur
M. Randal DOUC	Telecom SudParis	directeur
M^{me} Sidonie LEFEBVRE	ONERA	encadrant
M. Jean-Michel MARIN	Université de Montpellier 2	rapporteur
M. Eric MOULINES	Telecom ParisTech	co-directeur
M. Christian ROBERT	Université Paris-Dauphine	examinateur
M. Olivier WINTENBERGER	Université Paris 6	examinateur

Thèse n° 2014 TELE0007

École Doctorale EDITE ED130
Case courrier 168, 4 place Jussieu
75 252 Paris cedex 05

ONERA - The French Aerospace Lab
Chemin de la Hunière
91 123 Palaiseau

Telecom SudParis
9 rue Charles Fourier
91 011 Evry

Remerciements

Ne vous y trompez pas ! Les concepts, raisonnements ou autres théorèmes ne peuvent sembler anonymes, immuables, froids et indigestes que si l'on oublie qu'ils émergent en réalité de belles histoires, très humaines, ponctuées de doutes, d'errances, de rebondissements palpitants, d'exhaspérations inavouables, de satisfaction, d'allegresse... bref tout un tas de sentiments qui font intervenir tout un tas de (brillants !) acteurs que j'ai eu la chance de côtoyer tout au long de ces trois ans... Je souhaiterai ici les remercier très chaleureusement pour leur role essentiel dans ce travail.

A Sidonie, Randal et Eric pour votre patience, votre confiance et votre clairvoyance. A Sidonie, merci pour la liberté que tu m'as délibérement offerte dans mon travail et qui m'a permis d'explorer un certains nombres de problématiques passionnantes, pas vraiment prévues au départ. Merci pour tes précieux conseils, ton soutien et tes attentions, même quand cela va jusqu'à de m'interdir de Matlab ! ! *Pseudo-prior*, *Pseudo-marginal*... tout sauf un *pseudo-directeur de thèse* : merci Randal d'avoir laissé le rôle de voix-off à d'autres occupations. Pour ton sens de l'esthétisme dans les mathématiques, ton humilité "*on ne découvre pas vraiment les choses, elles nous préexistent, on ne fait que de les constater, et parfois on finit par se les approprier...*". J'espère avoir hérité de quelques unes de tes qualités ! Au delà de tout ce que tu as pu m'apporter scientifiquement parlant, merci Eric pour ton humanité, ton honnêteté et pour toutes ces discussions mathématiques que tu rendais presque aussi amusante que *de faire chabrot devant un match Girondin de Bordeaux-Créteil Lusitanos* ! C'était un vrai plaisir de travailler avec une personne aussi énergique et investie que toi !

Merci aux rapporteurs ainsi qu'aux membres du jury pour avoir fait l'effort de se plonger dans le cadre physique lié à la problèmatique de ma thèse et pour l'intéressante discussion lors de la soutenance. Je remercie également Stéphanie Allassonnière pour la base solide qu'a constituée l'ensemble de ses travaux sur les modèles à prototype deformable. I would like to thank Jimmy Olsson for giving me the opportunity to warm up my brain ahead of the PhD with this very nice internship in Lund University, it was again a pleasure to work with you recently ! Je souhaite aussi exprimer ma reconnaissance à Wojciech Pieczynski, Directeur du département Communications Images et Traitement de l'Information (CITI) à Telecom SudParis et à Vathana Ly Vath, Professeur à l'ENSIIE, pour la confiance qu'ils m'ont témoignée en me permettant de faire des heures d'enseignements, extrêmement formatrices et qui m'ont motivées à poursuivre dans cette voie.

L'aspect bucolique du centre de recherche de l'ONERA à Palaiseau offre un cadre particulièrement propice pour travailler, préparer un doctorat, gagner des tournois de foot, manger sur la terrasse de la cantine l'été, faire des batailles de boules de neige l'hiver... une belle l'harmonie y rassemble les générations de thésards, les équipes de chercheurs, le personnel administratif et d'intendance. Je tiens à les remercier tous et plus particulièrement le département du DOTA et l'équipe MPSO. Merci à mes deux *associées*, les deux soeurs siamoises Salima aka *The Artiste*, "*sa7a enti jiblou kulli tebsima wa istaqbile*

min djazair fil buro" et Tatiana aka *Yo !* aka *Double-T*, pour tes prestations bluffantes au jeu de l'échange de personnalité : votre complicité m'a beaucoup inspirée durant tout ce temps, *Mazal souvenir andi*. A Martin aka *L'Ingénieur-Bout*, pour les bons moments du jeu de la géographie, à Julien aka *Le Compte de Rizzac* pour m'avoir initié (ou du moins essayé !) à Kaamelot, à Alex aka *L'Homme du Champ de la Chèvre* pour les picnics extraordinaires au lac de l'X, à Edouard aka *L'Homme qui valait mille audio-guides de Versailles* : banc "Prop" ou "Brazil" peu importe, quand tu as fait PTT Armoric, tu réussis à tous les coups ! A William (et ses vrais pots du J3) pour les team-desserts, à Christophe (et son humour !), à Julien I., Matthieu et Adrian. A l'équipe de foot : Fred, Thomas, Jeff, Romain, Nico "*A défaut de publier, on ramène des trophées !!*" et aux stagiaires, dont la présence a toujours été synonyme d'été et de bons délires : Laura, Melinda, Kenny, Gael, Alice, les deux Kevin et Philipe. Je remercie aussi Yohan, Yazid, Mustapha, Cyrille et les autres doctorants de Telecom SudParis pour avoir rendu très agréable mes séjours à Evry.

A mon père "- *Bien le faire ou ne pas l'entreprendre...*", à ma mère "*...* - *Ouai mais bon faut savoir s'arrêter aussi...*", à ma soeur "*...* - *Vas-y, l'écoute pas, je vais te préparer un mi-cuit chocolat, ça va te requinquer !!*", comme toujours mes points de repères, merci pour votre Amour. A mes deux grands-mères pour m'avoir transmis leur gout pour le combat et (j'espère) un epsilon de leur sagesse ainsi qu'à mes regrettés grand-pères. Merci à mes oncles et mes tantes, à mes cousins et cousines, pour leur soutient et leur présence tout simplement. A mes frères, ceux de Fresnes, rois de l'évasion de mes problèmes de maths : Irfane, Joachim, Kevin, Lionel, Mattias, Mustapha, Nhat-Iep, Oualid, Polak, Samba, Samy, "*l'Amitié, c'est l'Amour véritable la famille !!*" ainsi qu'à leurs familles pour les précieux bols d'air qu'ils m'ont régulièrement servis. A Malik "*tu m'aurais dit on l'aurait fait à deux cette thèse mon vieux*", à Houssam, Sarafou, Sofiane, Walid, Lisa, Aurélie, Delphine, Lucile, Carole, Kahina et leurs familles respectives.

Enfin, je remercie le Metteur en scène, pour le rôle très agréable qu'Il m'a confié et pour m'avoir facilité cette belle expérience...

Résumé

Les dispositifs de protection de sites sensibles doivent permettre de détecter des menaces potentielles suffisamment à l'avance pour pouvoir mettre en place une stratégie de défense. Dans cette optique, les méthodes de détection et de reconnaissance d'aéronefs se basant sur des images infrarouge multispectrales doivent être adaptées à des images **faiblement résolues** et être robustes à la **variabilité spectrale et spatiale** des cibles. Nous mettons au point dans cette thèse, des méthodes statistiques de détection et de reconnaissance d'aéronefs satisfaisant ces contraintes.

Tout d'abord, nous spécifions une méthode de **détection d'anomalies** pour des images multispectrales, combinant un calcul de vraisemblance spectrale avec une étude sur les ensembles de niveaux de la transformée de Mahalanobis de l'image. Cette méthode ne nécessite aucune information *a priori* sur les aéronefs et nous permet d'identifier les images contenant des cibles. Ces images sont ensuite considérées comme des réalisations d'un modèle statistique d'observations fluctuant spectralement et spatialement autour de **formes caractéristiques** inconnues. L'estimation des paramètres de ce modèle est réalisée par une nouvelle méthodologie d'**apprentissage séquentiel non supervisé** pour des modèles à données manquantes que nous avons développée. La mise au point de ce modèle nous permet *in fine* de proposer une méthode de reconnaissance de cibles basée sur l'estimateur du maximum de vraisemblance *a posteriori* .

Les résultats encourageants, tant en détection qu'en classification, justifient l'intérêt du développement de dispositifs permettant l'acquisition d'images multispectrales. Ces méthodes nous ont également permis d'identifier les regroupements de bandes spectrales optimales pour la détection et la reconnaissance d'aéronefs faiblement résolus en infrarouge.

Mots-clefs

Signature infrarouge, Imagerie multispectrale, Détection d'anomalies, Reconnaissance de formes, Modèles à prototype déformable, Algorithme Expectation-Maximization, Apprentissage séquentiel, Méthodes de Monte Carlo par chaînes de Markov

Detection and Classification of Multispectral Infrared Targets

Abstract

Surveillance systems should be able to detect potential threats far ahead in order to put forward a defence strategy. In this context, detection and recognition methods making use of multispectral infrared images should cope with **low resolution signals** and handle both **spectral and spatial variability** of the targets. We introduce in this PhD thesis a novel statistical methodology to perform aircraft detection and classification which take into account these constraints.

We first propose an **anomaly detection** method designed for multispectral images, which combines a spectral likelihood measure and a level set study of the image Mahalanobis transform. This technique allows to identify images which feature an anomaly without any prior knowledge on the target. In a second time, these images are used as realizations of a statistical model in which the observations are described as random spectral and spatial deformation of **prototype shapes**. The model inference, and in particular the prototype shape estimation, is achieved through a novel **unsupervised sequential learning algorithm** designed for missing data models. This model allows to propose a classification algorithm based on maximum *a posteriori* probability.

Promising results in detection as well as in classification, justify the growing interest surrounding the development of multispectral imaging devices. These methods have also allowed us to identify the optimal infrared spectral band regroupments regarding the low resolution aircraft IRS detection and classification.

Keywords

Infrared Signature, Multispectral Imagery, Anomaly Detection, Shape Recognition, Deformable Template models, Expectation-Maximization Algorithm, Sequential Inference, Markov chain Monte Carlo methods

Table des matières

Introduction

Pour identifier la présence d'un objet dans une scène optique, de nombreuses applications civiles et militaires ont recours au rayonnement infrarouge. En effet, tout corps réfléchit et émet un rayonnement infrarouge, nommé Signature InfraRouge (SIR), caractéristique de ses propriétés physico-chimiques et de sa température. Lorsque l'objet d'intérêt possède un différentiel de température important avec son environnement, comme c'est le cas pour un aéronef en vol, sa SIR est peu sensible au contexte, ce qui représente un avantage par rapport au domaine du visible. Parmi les différentes façons d'exploiter la SIR, on peut choisir d'étudier simultanément plusieurs bandes spectrales infrarouge. On parle alors de signature infrarouge multispectrale, respectivement hyperspectrale, quand il s'agit d'une dizaine de bandes, respectivement d'une centaine de bandes. Par rapport à des signatures en bande large, disposer de SIR multispectrales permet d'accéder à des informations plus précises, potentiellement discriminatoires, sur les caractéristiques spectrales de l'objet en question. D'ailleurs, le développement de caméras infrarouge multispectrales, capables de filmer une même scène optique simultanément dans plusieurs bandes spectrales, fait l'objet d'intenses travaux de recherche. Toutefois, l'utilisation de tels dispositifs expérimentaux requiert le développement de méthodes efficaces de traitement, ce qui fait l'objet de cette thèse. On s'intéresse plus particulièrement à des SIR multispectrales dans la bande II de l'infrarouge, ce qui correspond à la plage de longueurs d'onde comprises entre 3 et 5 μm, pour la détection et la classification d'avions militaires.

Dans notre approche, la scène optique est connue de façon statistique et les sources d'incertitudes sur les paramètres la caractérisant sont multiples. Par conséquent, les SIR d'aéronefs sont considérées comme des variables aléatoires. Ainsi, pour être performantes, les méthodes de détection et de classification doivent intégrer les phénomènes de dispersion affectant les SIR. En particulier, l'enjeu principal sera de modéliser simultanément les deux types de dispersion, spatiale et spectrale, ce qui, jusqu'à présent, n'a pas été abordé dans la littérature. La dispersion spatiale résulte du fait que les SIR d'avions ne possèdent pas une unique géométrie de référence, mais sont descriptibles par une enveloppe de formes qui varient en fonction de nombreux paramètres tels que le type d'avion, le régime moteur, l'angle d'approche, etc. La dispersion spectrale désigne, quant à elle, les variations possibles du spectre d'un aéronef dans la bande II. Par ailleurs, travailler sur des SIR multispectrales, nous permet de choisir les bandes prises en compte dans notre étude. Il conviendra donc de spécifier une méthode de sélection des bandes pertinentes pour nos applications.

Des études ont été menées sur la détection d'avions à partir de leur SIR monospectrale, c'est à dire leur signature infrarouge intégrée en bande large [M6]. Parmi les méthodes proposées, la plus performante consiste à étudier les aires de certains ensembles de niveaux des SIR bien choisis. Toutefois, cette solution ne prend pas en compte la dispersion spectrale des cibles et n'est donc pas adaptable au cas des SIR multispectrales. En revanche, le détecteur d'anomalies RX [RY90], considéré en imagerie multispectrale, comme un détecteur de référence, intègre quant à lui ce type de dispersion. L'idée générale de cet algorithme est

de quantifier la différence entre les propriétés statistiques de chaque pixel et celles du reste de l'image, considéré comme un fond. Cette méthode semble adaptée à notre problématique car, au delà d'exploiter l'information spectrale véhiculée par une SIR multispectrale, elle ne nécessite que peu d'informations *a priori* sur une cible pour être détectée. Toutefois, bien que la dispersion spatiale nous empêche de connaître avec exactitude la taille des cibles, nous savons que celles-ci présentent une certaine cohérence spatiale, qui n'est pas exploitée par le détecteur RX.

Dès lors qu'une cible est détectée, le problème essentiel est de pouvoir la classifier. La plupart des algorithmes de classification classiques tels que les Machines à Support de Vecteurs (SVM) ou les méthodes d'apprentissage de type K-moyenne (K-means clustering) ne sont pas adaptés à des données présentant des similitudes mais également des variations qu'il convient d'analyser. L'hypothèse que ces données sont en réalité des mesures d'une même expérience (ou d'expériences similaires) nous conduit à une famille de modèles statistiques appelée modèles à prototype (ou template) déformable [d'A63, Gre93]. Dans un modèle à prototype déformable, différentes observations d'un même phénomène physique sont décrites par un prototype ou un ensemble de prototypes originaux ayant subi différents types d'altérations aléatoires, telles que des déformations dans le temps, dans l'espace, en intensité etc... Dans le cas des SIR monospectrales, les observations sont des images et dans le cas des SIR multispectrales, ce sont des cubes d'images. Des méthodes d'apprentissage permettant l'estimation des prototypes ainsi que des déformations associées ont été proposées pour ce type de modèles [AK10, KG92]. Il est alors possible de classer toutes les observations, en estimant pour chacune d'elle le prototype dont elle dérive le plus vraisemblablement. Suivant cette idée, une méthode a été proposée pour classifier des SIR monospectrales et a conduit à des résultats de classification satisfaisants [LAJ$^+$12]. Toutefois, le modèle à prototype déformable utilisé dans cette étude n'est pas adapté au cas de SIR multispectrales, parce qu'il n'intègre pas la modélisation simultanée des dispersions spatiale et spectrale. De plus, cette méthode d'estimation des prototypes s'avère trop coûteuse (en temps et en capacité de calcul) pour pouvoir être utilisée dans des dispositifs expérimentaux opérant en temps réel.

L'ensemble de ces études montrent que des méthodes de détection et de classification existent en traitement du signal mais qu'aucune n'a comme spécificité d'intégrer simultanément les phénomènes de dispersions spatiale et spectrale caractéristiques des aéronefs, ce qui justifie notre étude. Par ailleurs, les méthodologies que nous proposons nous permettront de spécifier les bandes spectrales qui optimisent les performances de détection et de classification. Celles-ci pourront être valorisées dans le cadre de la conception de caméras multispectrales, par l'ONERA ou par des industriels.

La démarche adoptée suit l'enchaînement logique suggéré par la problématique : nous commençons par présenter une méthode de détection ne nécessitant quasiment aucune connaissance *a priori* sur les cibles d'intérêt, en adoptant une approche de type détection d'anomalies. Nous exposons ensuite comment modéliser puis estimer les différentes dispersions, le but étant d'approcher les formes prototypes intervenant dans un modèle à prototype déformable adapté à notre problématique. Ceci nous permet *in fine* de proposer une méthode de classification adaptée à ce type de données. Le cas d'application type de cette étude consiste à détecter et classifier un aéronef situé à une vingtaine de kilomètres du point d'observation, au moyen d'un capteur infrarouge de veille situé au niveau du sol. A cette distance, la signature infrarouge est faiblement résolue spatialement : la cible n'apparaît que sur une dizaine de pixels de l'image, ce qui complexifie cette étude.

Dans un premier temps, nous avons mis au point un détecteur d'anomalies adapté au cas des SIR multispectrales. Pour cela, nous avons exploité les propriétés spectrales des signatures infrarouge en nous inspirant du détecteur d'anomalies RX [RY90]. Nous basons notre étude sur une image intermédiaire n'ayant aucune réalité physique. Elle est obtenue à partir d'une SIR multispectrale, et fournit une cartographie des zones de la scène optique contenant potentiellement une anomalie. Le détecteur d'anomalies que nous proposons consiste à étudier les périmètres de certains ensembles de niveaux de cette image intermédiaire, ayant des valeurs peu vraisemblables pour le modèle de fond. Si de tels ensembles de niveaux existent et ont un périmètre *anormalement* grand, alors une anomalie est détectée. L'originalité de cette méthode réside dans le fait que la dispersion affectant les composantes spatiales et spectrales des SIR, *a priori* considérée comme une nuisance, est ici exploitée comme un atout. Les résultats montrent que, comparée à des détecteurs de type RX, cette méthode améliore les performances de détection et en particulier réduit considérablement le taux de fausses alarmes pour notre application. Une fois établi, ce détecteur permet de sélectionner les bandes spectrales qui optimisent la détection. Il ressort de cette étude que, pour ces bandes optimales, il y a un avantage évident à détecter la présence d'un aéronef dans une scène optique à partir de SIR multispectrales plutôt que monospectrales. Par ailleurs, ceci nous permet de *localiser* dans la bande II l'information nécessaire pour détecter une cible. Nous limitons ainsi la redondance contenue dans les bandes spectrales, ce qui a un intérêt appréciable vis-à-vis du stockage des données, notamment comparé aux images hyperspectrales. Cette démarche ainsi que les résultats obtenus sont exposés dans le Chapitre I.

Toutefois, le manque de connaissance devient critique dès lors que l'on cherche à classifier les SIR multispectrales pour lesquelles on a détecté une anomalie. Pour cela, nous avons adapté un modèle à prototype déformable au contexte des signatures infrarouge multispectrales d'aéronefs. Sans ce modèle, les dispersions spatiales et spectrales sur les observations brutes sont telles qu'elles rendent impossible leur étude en vue d'une classification. Le modèle à prototype déformable est paramétré par les caractéristiques de chaque type d'avions : leur géométrie de référence, leur photométrie spectrale ou encore leur déformation typique. *A priori*, toutes ces données sont inconnues. Nous avons donc mis en place une méthode d'estimation séquentielle (ou en ligne) de ces paramètres basée sur un algorithme de type Expectation Maximization (EM). Dans un cadre d'apprentissage séquentiel, les observations (dans notre cas les SIR) sont traitées au fur et à mesure de leur acquisition et chaque nouvelle donnée améliore l'estimation des paramètres du modèle (dans notre cas les prototypes). Ce type d'apprentissage permet en particulier le traitement en temps réel des observations et dispense de les stocker. Parmi les méthodes d'estimation en ligne de paramètres dans des modèles à données manquantes, l'EM en ligne (Online EM) [CM07] est une solution pertinente qui conserve la structure originale de l'algorithme EM et donc sa simplicité ainsi que sa flexibilité d'implémentation. La difficulté est d'adapter cette méthode à un cadre dans lequel l'espérance de la vraisemblance complète sous la loi jointe *a posteriori* n'est pas calculable analytiquement. La solution mise en œuvre pour approcher cette espérance fait intervenir un algorithme de type Monte Carlo à chaînes de Markov (MCMC) basé sur les travaux de sélection de modèles dans un cadre bayésien menés par Carlin et Chib [CC95]. Cette méthodologie générale d'apprentissage dans des modèles à données manquantes est détaillée dans le Chapitre II. Nous appliquons cette méthode au cas spécifique des SIR monospectrales et étendons ensuite le modèle au cas des signatures multispectrales. Nous proposons une méthode de classification de ces signatures adaptée du maximum *a posteriori* et, comme pour la détection, nous montrons, dans le Chapitre III, que les performances de classification obtenues en

multispectral sont meilleures que dans le cas monospectral. De plus, cette méthode permet de comparer la pertinence de différents regroupements de bandes spectrales pour le problème de la classification d'aéronefs. Nous avons ainsi établi une méthode d'apprentissage de formes prototypes décrivant les différents aéronefs, qui nous ont permis de mettre en oeuvre une technique de classification de signatures infrarouge multispectrales robuste aux dispersions spatiale et spectrale des cibles.

Le Chapitre IV présente notre contribution théorique sur les algorithmes MCMC utilisant des chaînes de Markov non homogènes. Plus précisément, nous prouvons un théorème de comparaison pour des algorithmes MCMC utilisant comme critère la variance asymptotique de l'estimateur de Monte-Carlo. Ce résultat étend le théorème de comparaison de Tierney [Tie95] pour des chaînes inhomogènes alternant entre deux noyaux de transition. Ceci justifie, entre autres, l'efficacité du noyau MCMC dérivé de Carlin et Chib [CC95] proposé dans l'algorithme d'apprentissage, par rapport à d'autres méthodes classiques telles que l'échantillonneur de Gibbs. Le théorème est énoncé, démontré et des exemples d'application sont également proposés.

Bibliographie personnelle

Ce travail a été l'occasion de publier un certain nombre de contributions à des workshops, conférences ou revues à comité de lecture. Certains de ces travaux sont soumis et en attente de revue.

[M1] Maire, F., Lefebvre, S. et Moulines, É., *Online EM for Functional Data*, **soumis à** Computational Statistics and Data Analysis, 2013.

[M2] Maire, F. et Lefebvre, S. *Wavelenght Bands Selection for Aircraft Detection with a low resolution Multispectral Infrared Sensor*, **accepté, deuxième révision,** IEEE Transactions on Image Processing, 2013.

[M3] Maire, F., Douc, R. et Olsson, J., *Partial Ordering of Inhomogeneous Markov chains and applications to Markov chains Monte Carlo methods*, Annals of Statistics, 2014.

[M4] Maire, F., Lefebvre, S., Douc, R. et Moulines, É., Aircraft classification with low infrared sensor. *In Proceedings of IEEE Statistical Signal Processing workshop*, pages 761-765, 2011.

[M5] Maire, F., Lefebvre, S., Douc, R. et Moulines, É., An online learning algorithm for mixture models of deformable templates. *In Proceedings of IEEE Machine Learning for Signal Processing workshop*, pages 1-6, 2012.

[M6] Jakubowicz, J., Lefebvre, S., Maire, F. et Moulines, É,. *Detecting Aircraft With a Low-Resolution Infrared Sensor*, IEEE Transactions on Image Processing, 21 :3034–3041,2012.

[M7] Allassonnière, S., Glaunès, J. A., Bigot, J., Maire, F. et Richard, F J-P., *Statistical models for deformable templates in image and shape analysis*, Annales Mathématiques Blaise Pascal, 2013.

Notations

Ensembles, vecteurs et matrices

- $\mathbb{R}$ désigne l'ensemble des nombres réels,
- $\mathbb{N}$ l'ensemble des entiers naturels et $\mathbb{N}^* = \mathbb{N} \setminus \{0\}$,
- Soit $E \subset \mathbb{R}$ et $(m, n) \in \mathbb{N}^{*2}$, $\mathcal{M}_{m,n}(E)$ désigne l'ensemble des matrices de taille $m \times n$ dont les éléments appartiennent à E.
- Soit $E \subset \mathbb{R}^d$, $d > 0$, pour tout $x \in E$, $x = (x_1, \ldots, x_d)$, notons pour $1 \leq i, j \leq d$:

$$x_{i:j} = (x_i, \ldots, x_j), \quad x_{-i} = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_d) \ .$$

Pour toute matrice $M \in \mathcal{M}_{m,n}(E)$, $\det(M)$ désigne son déterminant, M^T sa transposée et $\mathrm{Tr}(M)$ sa trace. Si $n = m$, $\mathrm{diag}(M)$ est le vecteur composé des éléments de la diagonale de M et M^{-1} désigne, quand elle existe, la matrice inverse de M. Pour tout $n \in \mathbb{N}^*$, Id_n denote la matrice identité d'ordre n.

Soit $\mathbb{U}$ un sous ensemble de $\mathbb{R}^d$ $(d > 0)$

- $\mathcal{C}^p(\mathbb{U})$ désigne l'ensemble des fonctions $\mathbb{U} \to \mathbb{R}$, p fois dérivables et de dérivée p-ième continue,
- $\mathcal{L}^p(\mathbb{U})$ désigne l'ensemble des fonctions $\mathbb{U} \to \mathbb{R}$ dont la puissance p est intégrable

$$\int_{\mathbb{U}} |f(u)|^p \mathrm{d}u < \infty \ .$$

Soit une fonction $f : \mathbb{U} \to \mathbb{R}$

- Lorsque $f \in \mathcal{C}^1$, $\nabla_u f$ désigne le gradient de f : $\nabla_u f = (\frac{\partial f}{\partial u_1}, \ldots, \frac{\partial f}{\partial u_d})^T$ et pour tout $u^* \in \mathbb{U}$, $\nabla_u f(u^*)$ désigne la valeur du gradient en u^*.
- Avec les mêmes conventions, lorsque $f \in \mathcal{C}^2$, $\nabla_u^2 f$ désigne la matrice Hessienne de f dont les coefficients sont $[\nabla_u^2 f]_{(i,j)} = \frac{\partial^2 f}{\partial u_i \partial u_j}$, pour tout $(i, j) \in \{1, \ldots, d\}^2$.

Variables aléatoires

Soit X et Y deux variables aléatoires définies respectivement sur les espaces mesurables $(\mathsf{X}, \mathcal{X})$ et $(\mathsf{Y}, \mathcal{Y})$. Dans la plupart des applications considérées, les espaces mesurables seront tels que $\mathsf{X} \subset \mathbb{R}^d$ et $\mathcal{X} = \mathcal{B}(\mathsf{X})$ où $\mathcal{B}(\mathsf{X})$ est la tribu borélienne associée à X. Pour tout $x \in \mathsf{X}$, on désigne par $\mathrm{d}x$ un élément infinitésimal de $\mathcal{B}(\mathsf{X})$ correspondant à un voisinage de x. Dans la mesure du possible, les variables aléatoires seront désignées par des lettres majuscules $X, Y, \ldots$ et leurs réalisations par les minuscules de la même lettre $x, y, \ldots$. On note f_X la densité de X par rapport à une mesure λ définie sur $(\mathsf{X}, \mathcal{X})$, $f_{X|Y}$ la densité de X conditionnellement à Y (par rapport à λ) et $f_{X,Y}$ la densité jointe des variables X et Y par rapport à une mesure ν définie sur l'espace produit $(\mathsf{X} \times \mathsf{Y}, \mathcal{X} \otimes \mathcal{Y})$. Les mesures λ et ν sont typiquement les mesures de Lebesgue sur les espaces mesurables concernés.

Soit $A \in \mathcal{X}$, on définit :

- la probabilité que $X \in A$ (respectivement $X \in A$ sachant que $Y = y$)

$$\mathbb{P}[X \in A] = \int_A f_X(x)\lambda(\mathrm{d}x) \ , \quad \mathbb{P}[\, X \in A \,|\, y \,] = \int_A f_{X|Y}(x \,|\, y)\lambda(\mathrm{d}x) \ ,$$

- l'espérance de X (respectivement l'espérance conditionnelle de X sachant que $Y = y$)

$$\mathbb{E}[X] = \int_{\mathsf{X}} x f_X(x)\lambda(\mathrm{d}x) \ , \quad \mathbb{E}[\, X \,|\, y \,] = \int_{\mathsf{X}} x f_{X|Y}(x \,|\, y)\lambda(\mathrm{d}y) \ .$$

Par ailleurs,

- $X \sim f_X$ signifie que X est une variable aléatoire de densité f_X,
- $x \sim f_X$ signifie que l'on simule une réalisation de la variable aléatoire X ayant f_X pour densité et que l'on appelle x cette réalisation.
- $X_1, \ldots, X_n \overset{i.i.d.}{\sim} f_X$ signifie que $X_1, \ldots, X_n$ sont des variables aléatoires indépendantes et identiquement distribuées de densité f_X.

Soit M une variable aléatoire discrète prenant ses valeurs dans un ensemble M de cardinal fini, telle que $\mathbb{P}[M = m] = w_m$, $\sum_{m \in \mathsf{M}} w_m = 1$, alors :

- $M \propto w_m$ signifie que M est une variable aléatoire discrète telle que $\mathbb{P}[M = m] = w_m$,
- $m \propto w_m$ signifie que l'on simule une réalisation de la variable aléatoire M telle que $\mathbb{P}[M = m] = w_m$ et que l'on appelle m cette réalisation.

Mesures, noyaux de transition

Soit ν une mesure sur $(\mathsf{X}, \mathcal{X})$ et $f : \mathsf{X} \to \mathbb{R}$. f est dit λ-intégrable si

$$\int_{\mathsf{X}} |f(x)|\lambda(\mathrm{d}x) < \infty \ ,$$

et on désigne pour tout $p \in \mathbb{N}^*$, $\mathcal{L}^p(\nu)$ l'ensemble des fonctions $f : \mathsf{X} \to \mathbb{R}$ telles que f^p est λ-intégrable. Les écritures suivantes sont équivalentes

$$\mathbb{E}_\lambda[f(X)] = \int_{\mathsf{X}} |f(x)|\lambda(\mathrm{d}x) = \int |f| \mathrm{d}\lambda = \lambda f \ .$$

Soit $K : \mathsf{X} \times \mathcal{X} \to [0, 1]$ un noyau de transition sur X, tel que

- pour tout $x \in \mathsf{X}$, $K(x, \cdot\,)$ définit une mesure sur $(\mathsf{X}, \mathcal{X})$,
- pour tout $A \in \mathcal{X}$, $x \to K(x, A)$ définit une fonction mesurable sur X.

Soit K_1 et K_2 deux noyaux définis sur $(\mathsf{X}, \mathcal{X})$. On note $K_1 K_2$ le noyau produit défini sur $(\mathsf{X}, \mathcal{X})$ par :

$$\forall\, (x, A) \in (\mathsf{X} \times \mathcal{X}), \quad K_1 K_2(x, A) = \int_{\mathsf{X}} K_1(x, \mathrm{d}\tilde{x}) K_2(\tilde{x}, A) \ ,$$

et pour tout $p \in \mathbb{N}^*$, K^p le noyau K itéré p fois défini récursivement par

$$\forall\, (x, A) \in (\mathsf{X} \times \mathcal{X}), \quad K^p(x, A) = \int_{\mathsf{X}} K^{p-1}(x, \mathrm{d}\tilde{x}) K(\tilde{x}, A) \ .$$

On rappelle que λK et Kf sont respectivement la mesure sur $(\mathsf{X}, \mathcal{X})$ et l'application mesurable sur X définis par

$$\forall\, A \in \mathcal{X}, \quad \lambda K(A) = \int_{\mathsf{X}} \lambda(\mathrm{d}x) K(x, A) \;,$$
$$\forall\, x \in \mathsf{X}, \quad Kf(x) = \int_{\mathsf{X}} K(x, \mathrm{d}x') f(x') \;.$$

Pour tout $A \in \mathcal{X}$, $x \to \mathbb{1}_A(x)$ désigne la fonction caractéristique de l'ensemble A :

$$\mathbb{1}_A(x) = \begin{cases} 0 & \text{si} \quad x \notin A \;, \\ 1 & \text{si} \quad x \in A \;. \end{cases}$$

Pour tout $x \in \mathsf{X}$, δ_x désigne la mesure de dirac chargée en le singleton $\{x\}$ telle que pour toute fonction $f : \mathsf{X} \to \mathbb{R}$:

$$\int_{\mathsf{X}} f(\tilde{x}) \delta_x(\mathrm{d}\tilde{x}) = f(x) \;.$$

Simulation de SIR

Cette première section a pour objectif de présenter les données sur lesquelles seront appliqués les algorithmes de détection et de classification d'aéronefs proposés dans cette thèse.

1 Simulation d'aéronefs

1 .1 Préliminaire

On peut définir la signature infrarouge (SIR) d'un aéronef comme étant l'ensemble des quantités nécessaires à la prédiction du signal qui serait mesuré par un capteur infrarouge situé dans le champ de visé de l'appareil. La connaissance de la SIR d'un aéronef est essentielle pour deux types d'acteurs différents :

- pour les concepteurs d'avions, la connaissance de la SIR d'un avion permet d'évaluer sa probabilité de détection et donc sa capacité de survie en milieu hostile. Disposer d'un modèle de SIR permet également de pouvoir tester différentes solutions de furtivité : l'objectif, pour un industriel, est de diminuer la signature de l'avion afin d'empêcher ou de retarder sa détection par un système de surveillance adverse.
- pour les concepteurs de capteurs, cela permet d'évaluer les performances de détection et de classification de leur système de veille.

Pour des raisons pratiques, il n'est pas possible de mesurer ces signatures de façon expérimentale : problèmes de disponibilité des différents types d'avions, dangerosité des configurations, répétitions d'approches pour divers scénarios, etc...

C'est pourquoi, l'ONERA a développé depuis une trentaine d'année un code de simulation, CRIRA (Calcul de Rayonnement Infra-Rouge d'Avions), initié par [Gau81], permettant de calculer les SIR de plusieurs aéronefs dans différents contextes. Il prend en entrée, pour chaque simulation de SIR, une soixantaine de variables décrivant les caractéristiques de l'aéronef et de son environnement.

Parmi les données disponibles en sortie de CRIRA, nous nous intéressons particulièrement à l'éclairement différentiel ΔE entre le fond de ciel et l'aéronef, qui s'exprime par :

$$\Delta E = \frac{1}{D^2} \left(L_a - L_c\right) \tau_{atm} \, S_a \ ,$$

où

- L_a et L_c sont respectivement la luminance de l'aéronef et du fond de ciel,
- τ_{atm} est le coefficient d'absorption de l'atmosphère entre l'avion et le capteur,
- S_a est la surface apparente de l'avion vue du capteur,
- D est la distance capteur-avion.

La luminance (qui s'exprime en $\mathrm{W.m^{-2}.sr^{-1}}$) indique l'intensité lumineuse d'une source dans une direction donnée, divisée par son aire apparente dans cette direction. Une scène

FIGURE 1 – SIR d'un avion de combat à faible distance simulée par CRIRA

optique est, dans son ensemble, la juxtaposition de sources élémentaires, chacune correspondant à une zone de l'espace. Par conséquent, la luminance d'une scène optique est une image où chaque pixel a pour valeur l'éclairement émis par la (ou les) source(s) à laquelle (auxquelles) il correspond. Dans toute cette étude, nous nous intéressons à l'éclairement émis dans le spectre de bande II de l'infrarouge, ce qui correspond à la plage de longueurs d'ondes comprises entre 3 et 5 μm. Nous assimilerons, dans ce travail, la SIR d'un aéronef avec l'image formée par les éclairements différentiels. Par ailleurs, comme éclairement et luminance sont des grandeurs proportionnelles, nous utiliserons alternativement les deux formulations. Enfin, pour des raisons de confidentialité, l'échelle des valeurs d'éclairement est modifiée et les valeurs sont exprimées en unité arbitraire (u.a.).

Il est possible d'obtenir deux types de SIR :

– **SIR monospectrale** (ou en bande large) : la luminance différentielle est intégrée sur l'ensemble de la bande II, dénotée b_{II}. La signature est donc une image de résolution $P \times P$, telle que la valeur du n-ième pixel ($n \in \{1, \ldots, P^2\}$) s'écrit :

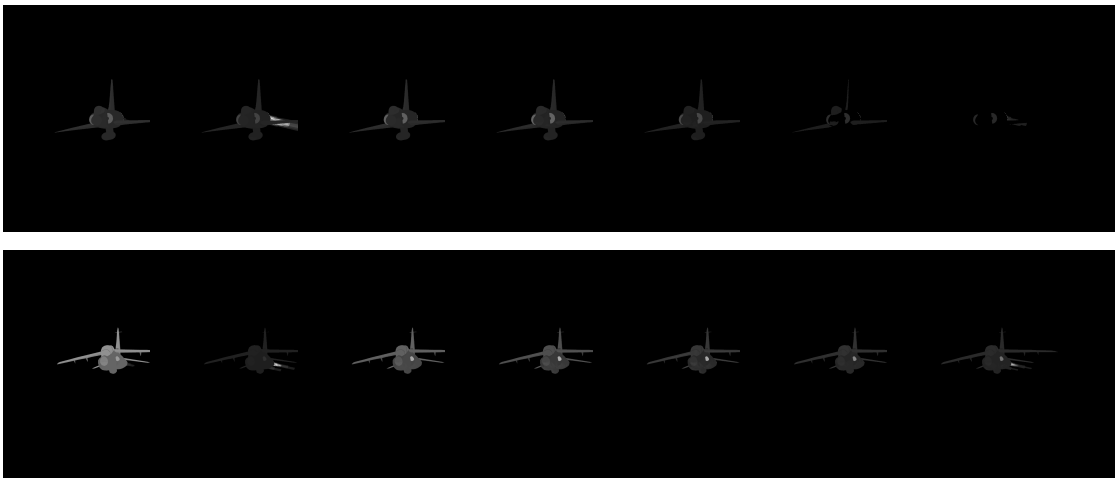
$$y_n = \int_{b_{\text{II}}} \ell_n(\nu) \mathrm{d}\nu \ ,$$

où ℓ_n est la luminance spectrale différentielle d'une zone de la scène optique correspondant au n-ième pixel. La figure 1 montre un exemple de deux SIR monospectrales fortement résolues (1024×1024) d'un avion à très faible distance avec deux angles d'approche différents.

– **SIR multispectrale** : supposons que la bande II soit décomposée en K sous-bandes $\{b_k\}_{k=1}^K$ de sorte à ce que $b_{\text{II}} = \cup_{k=1}^K b_k$ et $\cap_{k=1}^K b_k = \{\emptyset\}$. CRIRA peut également fournir la signature infrarouge d'un aéronef simultanément dans les K sous-bandes. Le n-ième pixel s'écrit alors comme un vecteur $y_n \in \mathbb{R}^K$ tel que

$$y_n = (y_n^{(1)}, \ldots, y_n^{(K)}) \quad \text{où} \quad \forall\, k \in \{1, \ldots, K\}\,, \quad y_n^{(k)} = \int_{b_k} \ell_n(\nu) \mathrm{d}\nu \ .$$

La figure 2 montre un exemple de SIR multispectrale fortement résolue (1024×1024 pixels) de deux avions de combat différents. Dans cette simulation, le spectre 2000 - 3500 cm^{-1} est décomposé en $K = 7$ bandes de largeur spectrale 200 cm^{-1} pour les 6 premières bandes et 300 cm^{-1} pour la dernière.



2000-2200 2200-2400 2400-2600 2600-2800 2800-3000 3000-3200 3200-3500

Figure 2 – SIR multispectrales de deux avions de combat à faible distance simulées par CRIRA dans les bandes spectrales indiquées (en cm^{-1})

Remarque 1. Nous distinguons dans notre scénario deux principales contributions à la SIR d'un aéronef :

- Les émissions intrinsèquement liées à l'avion, avec en particulier l'émission du fuselage et des ailes chauffées par les effets aérodynamiques, les gaz chauds du jet, les entrées d'air, qui sont composées d'une cavité avec une source interne assimilée aux premiers étages d'un compresseur basse pression, le nez et les autres parties métalliques chauffées par les gaz de combustion et enfin les composants mécaniques et électriques qui sont des sources de chaleur.
- Les réflexions sur l'avion de l'atmosphère, du sol et du soleil.

Remarque 2. Représentation des Signatures Infrarouge sous forme d'images

- Une SIR monospectrale Y de dimension $P \times P$ est une collection de nombres réels $(y_1, \ldots, y_{P^2})$. Lorsqu'on donne la représentation d'une SIR monospectrale (*e.g.* figures 1 et 4), le pixel ayant la valeur la plus élevée (resp. la plus basse) est affiché en blanc (resp. en noir). Les pixels intermédiaires sont colorés par une échelle linéaire de niveaux de gris suivant leur valeur.
- Une SIR multispectrale Y s'écrit comme un ensemble de K images $Y = \{Y_1, \ldots, Y_K\}$ où chaque image Y_k est une collection de nombres réels $(y_1^{(k)}, \ldots, y_{P^2}^{(k)})$. Il y a deux représentations possibles pour Y : soit chaque image Y_k est représentée de façon indépendante, *i.e.* le pixel de l'image Y_k ayant la valeur le plus élevée (resp. la plus basse) est affiché en blanc (resp. en noir), soit l'image multispectrale Y est affichée dans son ensemble, *i.e.* le pixel ayant la valeur la plus élevée (resp. basse) dans l'ensemble des images $\{Y_1, \ldots, Y_K\}$ est affiché en blanc (resp. noir). La figure 5(a) montre une même SIR représentée par ces deux méthodes. Bien entendu, l'intérêt étant d'étudier les variations en intensité dans chaque bande d'une SIR multispectrale, nous privilégions l'illustration d'une image multispectrale par la seconde approche. Par défaut, les SIR multispectrales seront représentées de cette façon.

Dans le cadre de cette thèse, nous nous plaçons dans un scénario typique d'attaque frontal air-sol d'un site sensible protégé par un capteur de veille, se déroulant de jour par un avion militaire volant à basse altitude. Dans le but de pouvoir détecter l'avion le plus tôt possible et ainsi pouvoir prendre les mesures nécessaires, l'avion doit être détecté à grande distance du capteur (une vingtaine de kilomètres environ). Ainsi, à la différence des figures 1 et 2 représentant respectivement des SIR monospectrales et multispectrales d'avions de combat situés à faible distance du capteur, les SIR correspondant au scénario de référence ont une faible résolution spatiale : la cible s'étend typiquement sur une dizaine de pixels. Bien que se basant sur ce scénario, les méthodes de détection et de classification proposées dans cette étude sont toutefois généralisables à d'autres configurations.

1 .2 Dispersion des SIR

La spécification de ce scénario fixe une trentaine de variables d'entrée de CRIRA parmi lesquelles une approche typique (de face) ainsi qu'une certaine catégorie d'aéronefs (notre étude concerne des avions militaires). Cependant, la simulation d'une SIR par CRIRA nécessite que toute les variables d'entrée soient spécifiées. De plus, un grand nombre d'informations telles que les conditions météorologiques ou certaines caractéristiques de l'avion, ayant un impact sur la SIR, ne sont pas déterminées par ce scénario. Par conséquent, il n'est pas possible d'évaluer les performances d'un capteur sans prendre en compte les incertitudes sur les caractéristiques de l'aéronef et sur son environnement.

Le tableau 1 liste l'ensemble des paramètres qui demeurent incertains dans le cas de notre scénario : 9 décrivent les propriétés optiques des surfaces de l'avion, 7 concernent les circonstances de vol (altitude exacte, vitesse, angles d'approche, régime moteur, ...) et 12 autres sont relatifs aux conditions atmosphériques comme la visibilité, le taux d'humidité, la température, la présence de nuages, etc. Les SIR correspondantes à ce scénario sont typiquement des images de taille 15×15 comme par exemple celles présentées dans les figures 4 et 5. Il est clair que dans un tel scénario, les détails des images de haute résolution 1024×1024 (figures 1 et 2) ne sont plus observables mais les SIR conservent toutefois des caractéristiques géométriques et spectrales qu'il est possible d'analyser. Le cas extreme, que nous ne traiterons pas dans cette étude, correspond aux SIR non résolues *i.e.* dont l'information est contenue en un seul scalaire (resp. vecteur) dans le cas monospectral (resp. multispectral).

Ces incertitudes sont à l'origine de la dispersion des SIR observées *in fine* . Nous distinguons deux types de dispersion :

- **dispersion spatiale des SIR** : l'incertitude sur le modèle d'avion ainsi que sur les paramètres de présentation de l'aéronef, tels que les angles d'approche, se traduit par une méconnaissance de la géométrie des SIR. En conséquence, pour le scénario typique de cette thèse, des formes spatiales de SIR aussi diverses que celles illustrées par la figure 4 peuvent être observées.
- **dispersion spectrale des SIR** : comme le montre la figure 3, le rayonnement infrarouge des cibles a deux sources principales ; le rayonnement produit par les gaz de combustion et celui émis et/ou réfléchi par le fuselage de l'appareil. Le rayonnement issu du gaz dépend essentiellement du type d'appareil et du régime moteur, tandis que celui du fuselage dépend des propriétés optiques de l'aéronef ainsi que des conditions météorologiques. En conséquence, pour un scénario donné, (i) des profils spectraux de SIR aussi variés que ceux présentés dans la figure 5 peuvent être observés et (ii) des différences dans le niveau d'intensité moyen de la signature existent. Nous ferons référence à ce phénomène par dispersion spectrale dans la suite de cette étude.

Tableau 1 – Description des variables d'entrée de CRIRA incertaines pour notre scénario

Description	
Altitude de vol,	Angle de gisement de l'avion,
Angle d'assiette de l'avion,	Angle de gîte de l'avion,
Régime moteur,	Mach de vol,
Cap de la trajectoire,	Modèle d'aérosol,
Écart de temperature du sol par rapport à la moyenne,	Visibilité au niveau du sol,
Émissivité des peintures des entrées d'air,	Modèle d'atmosphère,
Émissivité des peintures de verrières,	Humidité relative au niveau du sol,
Émissivité des peintures du bord d'attaque,	Température de l'air au niveau du sol,
Émissivité des peintures du nez intrados,	Numéro du jour,
Émissivité des peintures du bord de fuite,	Heure solaire,
Émissivité des peintures du fuselage,	Épaisseur de la couche de nuage,
Émissivité des peintures du fond des tuyères,	Hauteur de la base des nuages,
Émissivité des peintures du fond des tuyères internes,	Présence de nuages,
Émissivité des peintures du fond des tuyères externes,	Albedo du sol.

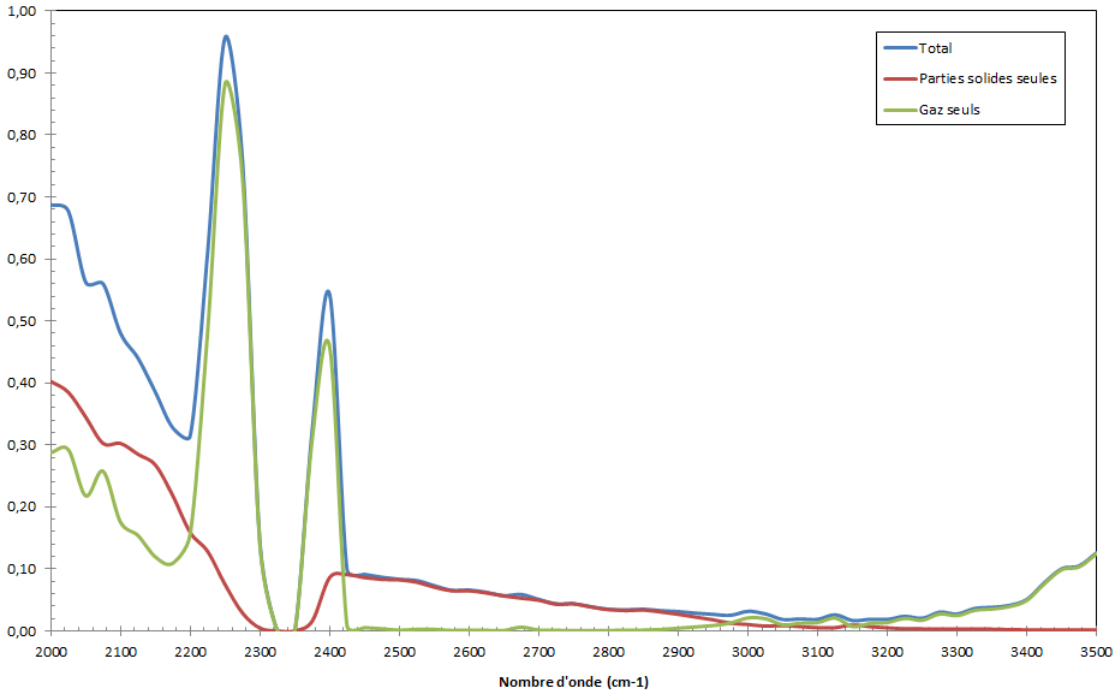


Figure 3 – Spectre caractéristique d'un aéronef non résolu

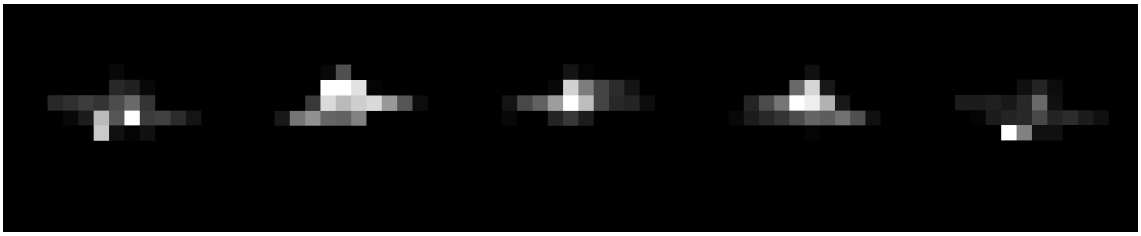
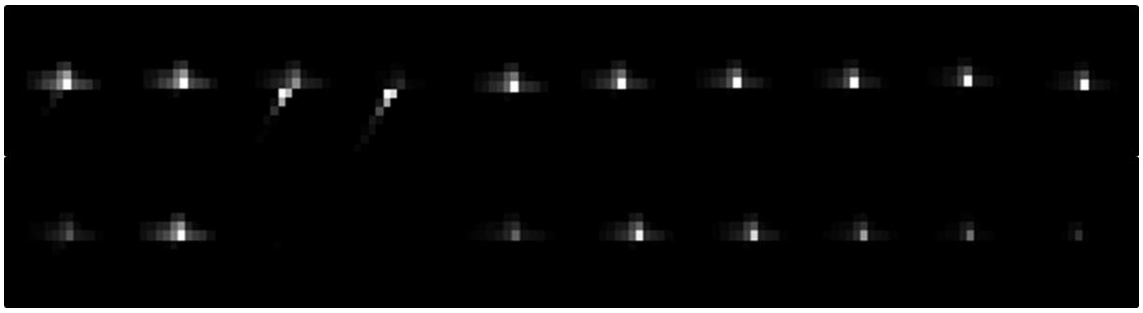
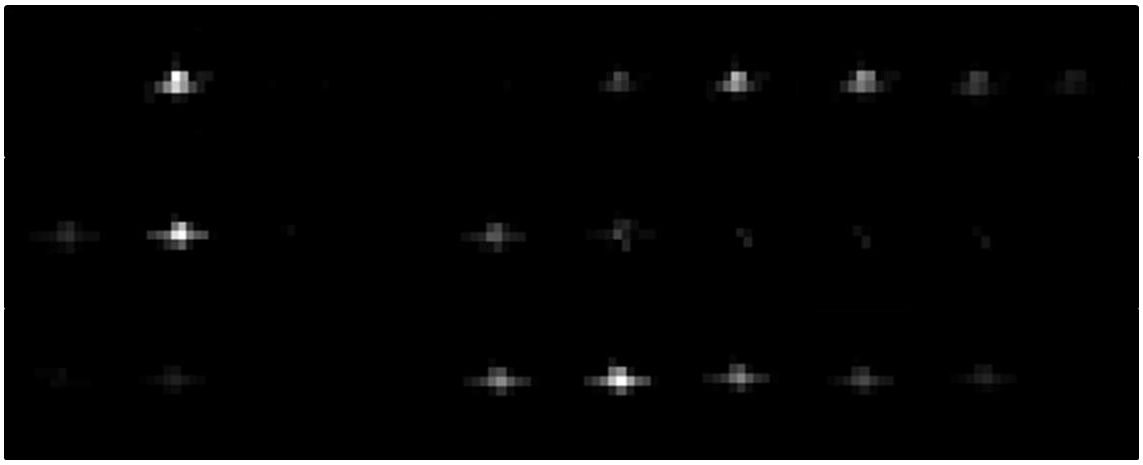


Figure 4 – Cinq SIR monospectrales faiblement résolues correspondant au scénario de référence



(a) Exemple d'une SIR multispectrale faiblement résolue correspondant à notre scénario. Les deux lignes correspondent à la même SIR : dans la 1ère ligne, les images dans chaque sous-bande sont affichées de façon indépendantes et dans la seconde l'image multispectrale est représentée dans son ensemble, Cf. remarque 2. Les sous-bandes spectrales sont données en cm^{-1}.



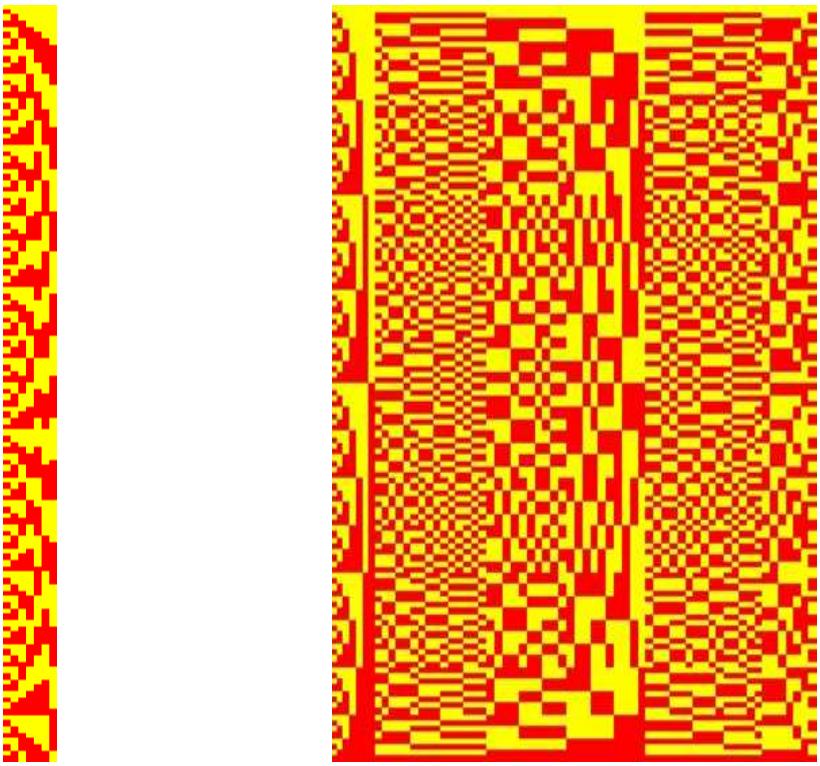
(b) Trois autres SIR multispectrales faiblement résolues. Les 10 bandes spectrales utilisées sont les mêmes que celles spécifiées dans la figure 5(a).

FIGURE 5 – SIR multispectrales correspondant au scénario de référence

Notons que la plupart des simulateurs de SIR existants [JD06, NKSS91, RM05, Gau81] ne prennent pas en compte ces dispersions : par conséquent, ils ne peuvent pas être utilisés pour spécifier des méthodes de détection et de classification car, dans ce cas, à un scénario correspond une SIR unique. Pour compenser cette limite, les industriels privilégient souvent l'approche du *pire des cas* qui consiste à fixer chaque paramètre inconnu à sa valeur la plus défavorable, afin de connaître une borne inférieure des performances de leur système de surveillance. Toutefois, cette technique donne des résultats très pessimiste et offre une fausse certitude. Ces simulateurs doivent donc être couplés avec des méthodes de propagation d'incertitudes afin d'approcher la performance *réelle* des systèmes optroniques.

Une analyse de sensibilité sur les variables d'entrées de CRIRA a été menée dans [LRVD10a][M6], dans le but d'identifier les paramètres qui influent le plus sur la variabilité de la SIR, et pour lesquelles il est important de prendre en compte l'incertitude associée. Les autres variables, qui ont peu d'influence sur la variabilité de la sortie, peuvent être fixées à une valeur constante pour la suite de l'étude. Les méthodes traditionnelles d'analyse de sensibilité basées sur une étude de la variance des paramètres et sur le cal-

cul d'indices de sensibilité, appelés indices de Sobol, ne sont pas applicables en raison du nombre élevé de variables d'entrées et du caractère qualitatif de certaines d'entre elles. Une approche par plan d'expériences à deux niveaux (les deux niveaux des variables sont ses valeurs minimale et maximale) a été mise en œuvre dans [Var10] afin d'estimer l'influence des variables d'entrées sur la SIR obtenue en sortie du code et ainsi permettre d'identifier les interactions existantes entre elles. Un plan d'expérience factoriel complet consiste à tester toutes les combinaisons possibles des deux niveaux de toutes les variables. En raison du nombre de calculs à mener (2^{28} dans notre cas), une approche par plan d'expérience fractionnaire a été privilégiée. Contrairement au plan factoriel complet, un plan factoriel fractionnaire ne permet d'estimer que des groupes d'interactions appelés *contrastes*. L'avantage de ce type de plan est que l'on peut évaluer l'influence des variables tout en tenant compte des interactions entre deux et trois facteurs, avec un coût qui reste envisageable, bien qu'élevé. La figure 6 illustre la différence de ces deux méthodes à travers leur matrice d'expériences et montre comment l'approche par plan d'expérience fractionnaire permet de gérer près de 10 fois plus de facteurs que l'approche factorielle avec le même nombre de calculs.



(a) (b)

Figure 6 – Plans d'expérience : (a) factoriel avec 7 facteurs et donc $2^7 = 128$ calculs - (b) fractionnaire avec 64 facteurs et 128 calculs. Les facteurs sont les colonnes et les configurations sont les lignes de ces matrices : les niveaux hauts sont représentés en rouge et les niveaux bas en jaune.

Pour simuler des SIR, les variables les moins influentes sont laissées constantes. Les plus influentes sont elles échantillonnées par des méthodes de type Quasi-Monte Carlo (QMC) [MC94]. A la différence des méthodes de Monte Carlo basées sur des tirages aléatoires, les méthodes de QMC utilisent des points de tirage répartis le plus uniformément possible

dans l'espace à échantillonner. A cet effet, des suites déterministes de nombres, appelées suites à faible discrépance, sont utilisées pour explorer l'espace de définition de ces variables incertaines. La discrépance est une mesure de l'uniformité de la dispersion des points : à la limite quand les points sont répartis de façon régulière, la discrépance de l'ensemble tend vers 0. En pratique, du fait de leur simplicité d'implémentation, les suites de Faure [Thi00] ont été utilisées mais d'autres suites à faible discrépance comme celle de Sobol peuvent aussi bien être choisies [Tuf97]. Par ailleurs, la procédure de *scrambling* proposée par Faure et Tezuka [TF03] est intégrée dans le processus afin d'ajouter un aspect aléatoire et avoir une meilleure répartition des points sur les projections en deux dimensions tout en conservant la faible discrépance de l'ensemble.

2 Simulation des fonds

Les SIR simulées par CRIRA ne peuvent pas être directement utilisées car il s'agit d'éclairements différentiels, qui ne prennent donc pas en compte la texture du fond de ciel (figures 4 et 5) . Il faut donc ajouter un modèle de fond de ciel réaliste aux sorties CRIRA.

2 .1 Cas monospectral

Dans [M6], deux modèles de fond de ciel ont été proposés pour des SIR monospectrales. Le premier est un modèle de fond non corrélé : un bruit blanc Gaussien qui est particulièrement adapté au cas des ciels clairs mais que nous utiliserons également comme première approximation grossière d'un fond de ciel nuageux. Le second est un modèle de fond texturé : un processus Brownien fractionnaire en deux dimensions, aussi appelé drap ou champ Brownien fractionnaire, [ST94] qui convient spécifiquement aux ciels nuageux. Un champ Brownien fractionnaire $\{B_H(u),\, u \in \mathbb{R}^2\}$ est un champ Gaussien continu sur $\mathbb{R}^2$, indicé par $u \in \mathbb{R}^2$, centré ($\forall u \in \mathbb{R}^2$, $\mathbb{E}[B_H(u)] = 0$), tel que $B_H(0,0) = 0$ et dont la covariance est donnée pour tout $(u_1, u_2) \in \mathbb{R}^2$ par :

$$\mathbb{E}[B_H(u_1)B_H(u_2)] = \frac{1}{2}\left(|u_1|^{2H} + |u_2|^{2H} - |u_1 - u_2|^{2H}\right) , \tag{1}$$

où H est le coefficient de Hurst $H \in [0, 1]$.

Ces deux modèles requièrent la spécification de paramètres, la variance σ^2_{II} pour le bruit blanc Gaussien ainsi que le paramètre de Hurst H pour le processus Brownien fractionnaire. Une centaine d'images de fond de ciel, spectralement intégrées dans la bande 3-5 μm et issues de la campagne de mesure MIRAMER menée par l'ONERA, ont permis d'estimer ces paramètres. Plus précisément, trois écart-types ont été estimés pour le cas non corrélé :

- $\sigma^{(1)}_{\text{II}} = 0.025$ est l'écart type moyen pour un fond de ciel clair,
- $\sigma^{(2)}_{\text{II}} = 0.058$ est l'écart type moyen pour un fond de ciel nuageux,
- $\sigma^{(3)}_{\text{II}} = 0.18$ est l'écart type maximal observé pour un fond nuageux.

Ces mêmes images ont permis d'estimer un paramètre de Hurst H dans l'intervalle

$$0.4 \leq H \leq 0.65 ,$$

qui correspond à des fonds de ciels nuageux sous diverses hypothèses (conditions météorologiques, types de nuages, etc...).

Il est donc possible de simuler une image multispectrale réaliste. Pour toute SIR monospectrale Y obtenue en sortie de CRIRA, l'image monospectrale finale $\tilde{Y}$ s'écrit sous l'hypothèse d'un fond modélisé par un bruit blanc pour $i \in \{1, 2, 3\}$:

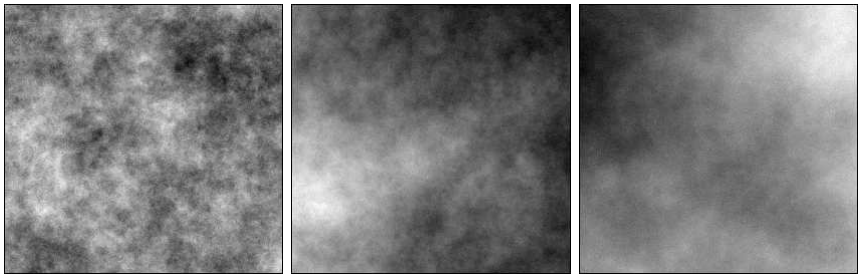
$$\tilde{Y} = Y + \sigma_{\text{II}}^{(i)} \epsilon \ , \quad \text{avec} \quad \epsilon \sim \mathcal{N}(0, \text{Id}_{P^2}) \ , \tag{2}$$

où pour tout $n \in \mathbb{N}^*$, Id_n est la matrice identité $n \times n$.

De la même façon, sous l'hypothèse de ciel nuageux, l'image monospectrale finale s'écrit :

$$\tilde{Y} = Y + \sigma_{\text{II}}^{(2)} W \ , \quad \text{avec} \quad W \sim B_H \ , \tag{3}$$

où B_H est le processus Brownien fractionnaire de paramètre de Hurst H. En pratique, pour générer de tels processus, nous utilisons l'algorithme de Stein [Ste02] qui fournit une méthode de simulation rapide et exacte de draps Brownien fractionnaires (Cf. figure 7).



(a) (b) (c)

Figure 7 – Images de fonds nuageux, obtenues par simulation d'un drap Brownien fractionnaire de paramètre de Hurst $H = 0.4$ (a), $H = 0.5$ (b) et $H = 0.65$ (c)

Remarque 3. Comme le processus de bruit additif représente la signature infrarouge du ciel comprise entre le capteur et l'infini, les modèles de SIR d'aéronefs (2) et (3), illustrés par l'image de gauche de la Figure 8, sont approximatifs. En effet, la présence d'un aéronef masque la section du fond qui se trouve entre la cible et l'infini. Ainsi, bruiter de la même façon l'ensemble de l'image reviendrait à dire que la signature infrarouge du ciel entre le capteur et la cible et entre le capteur et l'infini est identique, ce qui n'est pas justifiable physiquement. Nous proposons donc de ne pas ajouter de bruit là où la cible est présente. Ceci revient à faire l'hypothèse, justifiable physiquement, que la signature infrarouge de la colonne d'atmosphère comprise entre le détecteur et la cible est négligeable comparée à la la signature du ciel comprise entre le détecteur et l'infini. Toutefois, les SIR simulées par CRIRA possèdent de nombreux pixels ayant une valeur négligeable si bien qu'en ne bruitant que les zones de l'image où la cible a une luminance nulle, nous aboutissons à des images où, comme le montre l'image au centre de la Figure 8, la proportion des pixels bruités est faible. Nous proposons donc un compromis entre ces deux approches, illustré par l'image de droite de la Figure 8, qui consiste à bruiter les pixels qui sont inférieurs au pixel médian de la cible.

La figure 9 présente quelques SIR monospectrales obtenues en ajoutant une réalisation du modèle de fond de ciel clair (fig. 9(a)) et une réalisation du modèle de fond de ciel nuageux (fig. 9(b)) aux images différentielles d'aéronefs simulées par CRIRA. C'est à partir de ces données que nos méthodes devront diagnostiquer la présence d'un aéronef dans la scène optique et le cas échéant l'identifier parmi une liste de plusieurs avions.

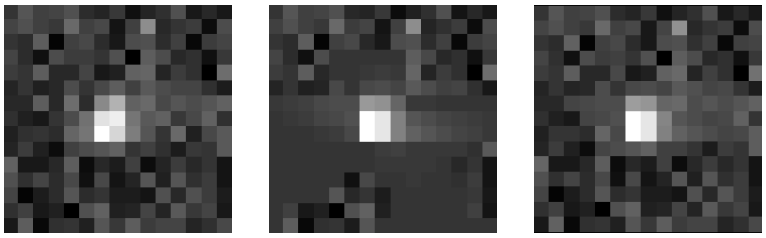
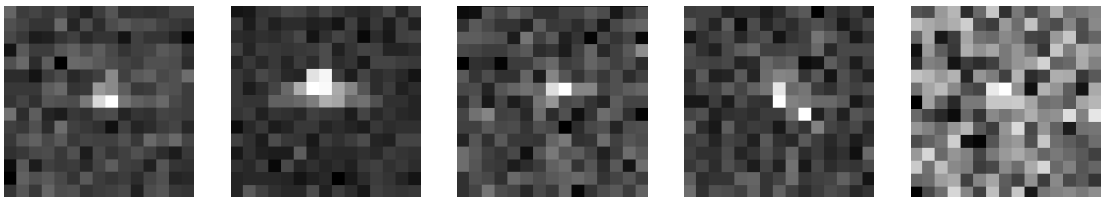
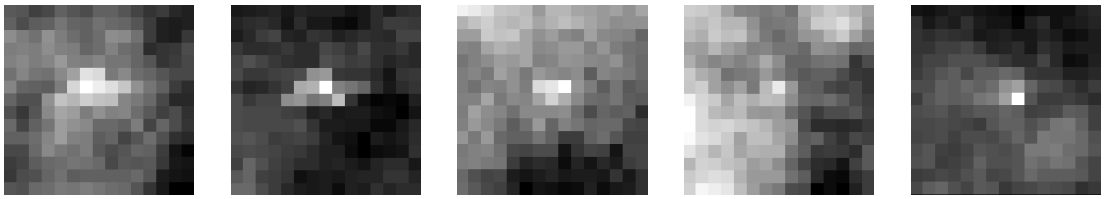


FIGURE 8 – SIR d'un aéronef bruitée de trois façons différentes



(a) SIR monospectrale d'aéronef sous un fond de ciel clair



(b) SIR monospectrale d'aéronef sous un fond de ciel nuageux

FIGURE 9 – Cinq SIR monospectrales faiblement résolues correspondant au scénario de référence

2 .2 Cas multispectral

Soit $Y = \{Y_1, \ldots, Y_K\}$, une image multispectrale à K bandes, simulée par CRIRA. Sous les hypothèses

(i) le modèle de bruit Gaussien reste valide dans chaque sous-bande b_k,

(ii) les fonds de ciels sont indépendants dans chaque sous-bande b_k,

l'image bruitée dans la sous-bande k s'écrit pour tout $k \in \{1, \ldots, K\}$:

$$\tilde{Y}_k = Y_k + \sigma_k \epsilon \ , \quad \text{avec} \quad \epsilon \sim \mathcal{N}(0, \mathrm{Id}_{P^2}) \ , \tag{4}$$

où σ_k^2 est la variance du bruit dans la sous-bande k. Nous proposons un modèle physique permettant de calculer les variances $\{\sigma_k^2\}_{k=1}^K$.

Soit $\{X_{p,k}(t)\}_{t>0}$ un processus stochastique indiquant le nombre de photons dont le nombre d'onde appartient à la bande spectrale b_k, $k \in \{1, \ldots, K\}$ et incident sur une surface équivalente à un pixel $p \in \{1, \ldots, P^2\}$ du détecteur au temps t. Sous l'hypothèse que l'arrivée des photons sur le détecteur est un processus de Poisson, nous avons :

$$\mathbb{P}[X_{p,k}(t + \Delta t) - X_{p,k}(t) = n] = \frac{e^{-N_{p,k}} {N_{p,k}}^n}{n!} \ , \tag{5}$$

où $N_{p,k}$ est le nombre moyen de photons de nombre d'onde dans la bande k incidents sur le pixel p durant Δt. En conséquence, sous l'hypothèse que $\{X_{p,k}(t)\}_{t>0}$ est un processus stationnaire, nous avons : $\mathrm{Var}[X_{p,k}(t)] = N_{p,k}$ qui ne dépend pas de t.

Soit ℓ la fonction de $\mathbb{R} \to \mathbb{R}$ qui, pour tout nombre d'onde ν associe la luminance spectrale du fond $\ell(\nu)$ correspondante. Pour tout pixel $p \in \{1, \ldots, P^2\}$, caractérisé par une surface équivalente sur le détecteur S_p et un angle solide Ω_p, le nombre total de photons $N_p(\mathrm{d}\nu)$ de nombre d'onde dans un voisinage infinitésimal à ν et incident sur p s'écrit :

$$N_p(\mathrm{d}\nu) = \frac{\ell(\nu)\tau(\nu)\Omega_p S_p \Delta t}{hc\nu}\mathrm{d}\nu \ , \tag{6}$$

où la fonction $\tau : \mathbb{R} \to \mathbb{R}$ est le filtre du détecteur et la quantité $hc\nu$ l'énergie d'un photon de nombre d'onde ν. $N_{p,k}$ s'écrit alors :

$$N_{p,k} = \int_{\nu_k^-}^{\nu_k^+} \frac{\ell(\nu)\tau(\nu)\Omega_p S_p \Delta t}{hc\nu}\mathrm{d}\nu \ , \tag{7}$$

où la bande b_k correspond au spectre compris entre ν_k^- et ν_k^+. En faisant les approximations que la fonction de filtre τ est égale sur toute la bande II à $\bar{\tau}$ et que la luminance spectrale ℓ est constante sur chaque bande et égale à $\bar{\ell}_k$, il vient l'approximation de $N_{p,k}$ suivante :

$$\bar{N}_{p,k} = \rho \, \frac{\bar{l}_k(\nu_k^+ - \nu_k^-)}{\bar{\nu}_k} \ , \tag{8}$$

où $\rho = \frac{\bar{\tau}\Omega_p S_p \Delta t}{hc}$.

Chaque photon incident peut potentiellement libérer un électron. Suivant le modèle proposé par [MRBL08], nous considérons qu'un photon a une probabilité Q de libérer un électron et qu'il a lui même une probabilité η de conduire le courant. La valeur associée au pixel p dans la bande k à l'instant t, dénotée $\{\tilde{X}_{p,k}(t),\, t > 0\}$, est supposée être reliée au nombre d'électrons conducteurs émis par un photon incident sur la zone correspondante au pixel p et dont le nombre d'onde appartient à la bande k par un gain G. Il vient :

$$\tilde{X}_{p,k}(t) = G\eta Q X_{p,k}(t) \ . \tag{9}$$

L'écart type des pixels de fond σ_k est donc exactement l'écart type de $\tilde{X}_{p,k}(t)$ que nous exprimons par :

$$\sigma_k = \varrho \sqrt{\frac{\bar{l}_k(\nu_k^+ - \nu_k^-)}{\bar{\nu}_k}} \ , \tag{10}$$

avec $\varrho = \sqrt{G\eta Q \rho}$. Comme $b_{\mathrm{II}} = \cup_{k=1}^{K-1}[\nu_k \,;\, \nu_{k+1}]$, nous pouvons écrire :

$$\sigma_k = \sigma_{\mathrm{II}} \sqrt{\frac{\bar{l}_k(\nu_k^+ - \nu_k^-)/\bar{\nu}_k}{\sum_{k=1}^{K} \bar{l}_k(\nu_k^+ - \nu_k^-)/\bar{\nu}_k}} \ , \tag{11}$$

ce qui permet de connaître la variance dans n'importe quelle sous-bande pourvu que σ_{II} et $\{\bar{\ell}_k\}_{k=1}^K$ soient connues. Trois valeurs possibles pour σ_{II} ont été proposées dans le cas monospectral (Cf. Section 2 .1) et les luminances moyennes sont estimées à partir du logiciel MATISSE [SBC+02], développé par l'ONERA, permettant le calcul de luminances spectrales de fonds naturels ; voir la figure 11. La figure 10 présente trois SIR obtenues en additionnant l'image multispectrale issue de CRIRA avec un fond de ciel multispectral simulé avec la méthode indiquée. Dans cette simulation, nous avons utilisé l'écart type moyen pour les fonds nuageux $\sigma_{\mathrm{II}}^{(2)}$.

Remarque 4. La Remarque 3 s'applique également dans le cas multispectral : pour tout $k \in \{1, \ldots, K\}$, seuls les pixels de l'image Y_k ayant une valeur inférieure au pixel médian de la cible dans la bande k sont bruités.

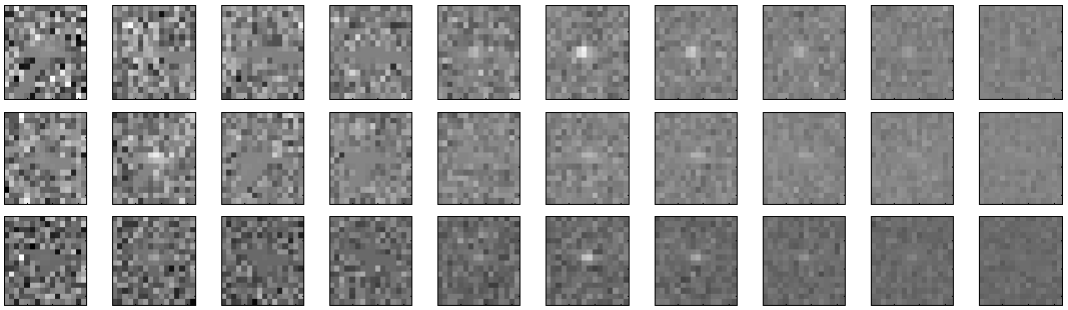


FIGURE 10 – Trois SIR multispectrales sur fond de ciel clair

Remarque 5. Cas des fonds texturés en multispectral
Pour les fonds de ciel nuageux en multispectral, les hypothèses précédentes ne sont plus valides. En effet, le modèle doit nécessairement prendre en compte les corrélations des textures dans les différentes sous-bandes spectrales. Toutefois, pour estimer les paramètres d'un tel modèle il est nécessaire de disposer d'images multispectrales de fonds de ciel nuageux dans ces bandes spectrales. L'acquisition de ce type de données au moyen d'un imageur multi ou hyperspectral en bande II permettrait à terme d'obtenir une méthode de simulation de fonds texturés en multispectral.

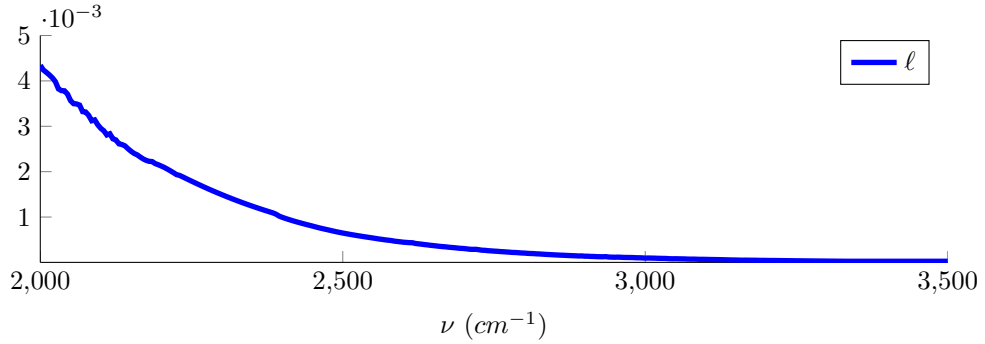


FIGURE 11 – Profil de luminance spectral obtenu par MATISSE

Préambule

Par soucis pédagogique, nous proposons dans ce Préambule une étude poussée de l'état de l'art des quatre thèmes majeurs abordés dans cette thèse suivie, pour chaque partie, d'un résumé de notre contribution sur ces différents points. L'idée est de resituer les problématiques dans leur contexte, de justifier l'intérêt mais aussi les limites des différentes solutions existantes et de décrire succinctement la valeur ajoutée de nos contributions. Les Chapitres I, II, III et IV donnent davantage de précisions, d'illustrations et de résultats concernant nos contributions.

A Méthodes de détection

A .1 Détection d'anomalies

Comme le montrent les Figures 9 et 10, détecter la présence d'un aéronef sur un fond de ciel à partir d'images faiblement résolues s'avère être une tâche délicate. En plus de la présence de bruit et de la faible résolution des images, l'incertitude sur certaines variables d'entrée du simulateur induit une forte dispersion sur les signatures infrarouge que l'on cherche à détecter. Par conséquent, il est difficile de détecter les avions à partir de leur SIR par une approche directe, étant donné qu'*a priori* on ne sait ni quelle géométrie, ni quelle photométrie permettront de les identifier.

La détection d'anomalies est un autre moyen d'envisager une telle tâche. Sous l'hypothèse que le fond est un processus prévisible, homogène et majoritaire, la détection de cibles équivaut alors à la recherche d'éléments qui se démarquent du fond. Cette démarche est particulièrement bien adaptée à notre contexte : d'une part les SIR d'aéronefs sont *a priori* mal connues et d'autre part si une anomalie est détectée dans une image de fond de ciel, il s'agit vraisemblablement d'un objet d'intérêt (*e.g.* aéronef, drone, etc...). Les algorithmes de détection d'anomalies sont utilisées dans des domaines aussi variés que la santé [SPS01], la vidéo surveillance [SM04, PLL07], la cyber-sécurité [RSRD07, LEK$^+$03] ou encore les fraudes à la carte bancaire [AFR97]. Les rapports [PP07] et [CBK09] fournissent une analyse détaillée des enjeux et des techniques employées dans les méthodes de détection d'anomalies appliquées à ces différents domaines. Nous reprenons dans le paragraphe suivant les éléments pertinents de cette approche sous l'angle de notre problématique. Le formalisme qui sera utilisé dans le Chapitre I est également introduit.

Soit $Y_1, \ldots, Y_n$ un ensemble d'observations (ou de mesures) d'un phénomène d'intérêt et désignons par $\mathsf{Y} = \mathsf{H}_0 \cup \mathsf{H}_1$ l'espace des observations, où H_0 et H_1 sont respectivement les sous-espaces des observations *normales* et *contenant une anomalie*. Les observations peuvent être des nombres réels $\mathsf{Y} = \mathbb{R}$, des courbes $\mathsf{Y} = \mathbb{R}^p$, des images $\mathsf{Y} = \mathbb{R}^\ell \times \mathbb{R}^m$ ou plus généralement toute fonction définie sur un espace U à valeurs dans un espace V, $\mathsf{Y} = \mathsf{V}^\mathsf{U}$. Un algorithme de détection d'anomalies spécifie une fonction ϕ telle que

$$\phi : \mathsf{Y} \to \{0, 1\} \ . \tag{12}$$

où pour toute observation $Y \in \mathsf{Y}$, $\phi(Y) = 1$ indique que Y est une anomalie et $\phi(Y) = 0$ est une observation normale.

Plusieurs éléments compliquent la mise au point d'une fonction ϕ capable de détecter une cible pour des images mono ou multispectrales : tout d'abord, les observations sont généralement bruitées et par conséquent certaines zones d'une image de fond peuvent être perçues, à tord, comme des anomalies. Les propriétés statistiques du fond peuvent évoluer dans le temps et dans ce cas des séquences d'images de fond ne correspondant pas au modèle initial peuvent être suspectées de contenir une cible. Enfin, dans les situations où un agent ennemi essaie de contourner la vigilance d'un dispositif de contrôle, il procédera de sorte à ce que l'anomalie résultant de son activité soit la plus difficile possible à détecter. Dans notre scénario, l'ennemi utilisera typiquement un aéronef dont la SIR Y aura une image par ϕ le plus proche possible de H_0 (*e.g.* utilisation d'aéronefs furtifs).

Les méthodes de détection d'anomalies peuvent être classées en deux familles :

- les méthodes dites *supervisées* : il existe une base d'apprentissage constituée d'observations $\{Y_k \in \mathsf{Y},\, 1 \leq k \leq n\}$ et d'un ensemble de labels $\{I_k \in \{0,1\},\, 1 \leq k \leq n\}$ tels que I_k indique si Y_k est une anomalie ou non. Dans une approche bayésienne, des modèles statistiques d'observations normales et anormales peuvent être établis à partir des données labellisées. Dans un second temps, à chaque nouvelle observation inconnue est attribuée la classe qui maximise la vraisemblance *a posteriori* de l'un ou l'autre des deux modèles. Le choix de ϕ peut également se faire par des techniques d'apprentissage traditionnelles utilisant des réseaux de neurones [DSSV00], des réseaux bayésiens [BWJ01] ou encore des machines à support de vecteurs [DG02]. Pour certaines applications, il est possible de définir plusieurs classes d'observations normales ($I_k \in \{0, 1, \ldots, C\}$) et dans ce cas autant de classifieurs que de classes sont apprises [Amb03]. Une nouvelle observation sera identifiée comme une anomalie si elle n'est pas classée comme normale par l'un des ces classifieurs. D'autres stratégies combinant plusieurs classifieurs sont également envisageables.
- les méthodes dites *non-supervisées* : il n'y a pas de base d'apprentissage disponible. Il est néanmoins possible d'obtenir un modèle statistique pour les observations normales, en particulier sous l'hypothèse que la majorité des observations ne sont pas des anomalies. Des approches paramétriques [Esk00] (*e.g.* hypothèse de données Gaussiennes [SL05] ou modèles de regression [KTAT03]) et non paramétriques [DJC98] (*e.g.* étude d'histogrammes) ont été proposées. Dans ce cas, les observations normales se trouvent dans les régions de fortes probabilités de la distribution apprise du fond, tandis que les anomalies sont dans ses zones de faibles probabilités.

Dans notre situation, il est difficile d'établir une base d'apprentissage. En effet, les labels doivent être précisés *à la main* ce qui, en plus d'être fastidieux, est délicat car pour certaines observations il n'est pas possible d'affirmer à l'oeil nu si elles contiennent une anomalie (on peut se référer par exemple à la seconde image multispectrale de la figure 10). En revanche, comme les SIR sont mesurées par un capteur de veille, l'hypothèse que la grande majorité des observations ne contienne pas d'anomalie est vérifiée. En effet, l'attaque frontale d'un site sensible par un aéronef est supposé être un événement rare. Il est donc possible d'établir un modèle réaliste pour les observations normales. Dans le modèle de simulation des SIR présenté dans la partie précédente, les observations ne contenant pas d'anomalies sont des réalisations de processus stochastiques connus (Gaussien *i.i.d.* ou Brownien fractionnaire). Par conséquent, des méthodes d'estimation de paramètres permettent, à partir d'échantillons, d'avoir accès à une bonne connaissance des modèles de fond.

Dans la suite, nous ferons l'hypothèse (réaliste) que, dans le cas fonds Gaussiens, le modèle de fond est parfaitement connu : la variance peut, en pratique, être approchée par les estimateurs de maximum de vraisemblance. Des méthodes plus techniques (voir par exemple [Lég00, Chapitre 5]) permettent en théorie d'estimer le paramètre de Hurst d'un drap Brownien fractionnaire. Toutefois, comme elles n'ont pas été testées à partir d'images de fonds simulées, nous ne ferons pas d'hypothèse particulière dans le cas d'un fond texturé. Nous verrons que dans le cas Gaussien, cette hypothèse n'est pas indispensable à l'établissement des méthodes de détection, mais qu'elle nous permet de nous placer dans un cadre où les calculs sont plus légers.

La détection *a contrario* est une approche non paramétrique partageant certaines similitudes avec la détection d'anomalies. Dans ce contexte, les observations sont des images, $\mathsf{Y} = \mathbb{R}^\ell \times \mathbb{R}^m$. Initialement proposée pour détecter des alignements dans une image [DMM00], des généralisations à d'autres structures [DMM01] ou formes [DMM03] ont été proposées. Les méthodes *a contrario* sont conceptuellement basées sur le principe d'Helmholtz [Gre07] énonçant que le cerveau humain perçoit d'autant mieux une forme géométrique dans une image que cette dernière se manifeste par un écart important par rapport à un modèle naïf, non structuré, appelé modèle *a contrario* . Ainsi un événement qui, probabilistiquement parlant, a peu de chance de se réaliser dans une image est considéré comme d'autant plus significatif s'il est répété plusieurs fois dans l'image. Le test statistique ϕ^{ac} correspondant à l'approche *a contrario* ne consiste pas à dire si une observation $Y \in \mathsf{Y}$ provient d'un modèle de H_0 ou de H_1 mais si des caractéristiques contenues dans Y (alignements, formes identiques, couleurs redondantes, tailles similaires) ont pu se réaliser « par chance » ou si elles constituent un indice significatif traduisant la présence d'une anomalie.

Ainsi, tout comme les méthodes de détection d'anomalies, les approches *a contrario* ne nécessitent pas de modèle de cible *a priori* H_1. Toutefois, à la différence du modèle H_0 présent dans les méthodes de détection d'anomalies, le modèle *a contrario* ne concerne pas nécessairement le fond mais peut être défini pour les structures que l'on cherche à identifier. Ainsi, pour des alignements de points, le modèle naïf est une loi uniforme sur $[0\,;\,\pi]$ pour l'orientation principale de la photométrie d'un voisinage de pixels [DMM00] ; pour la détection de clusters dans des images binaires (pixels noirs ou blancs), le modèle naïf utilisé dans [DMM03] est une loi uniforme pour la distribution des pixels noirs... Le choix du modèle *a contrario* autorise une plus grande liberté que le modèle de H_0 : dans la plupart des cas il est établi par le praticien sans nécessiter d'apprentissage particulier. A la différence des méthodes bayésiennes classiques qui nécessitent la spécification quantitative de lois *a priori* , l'approche *a contrario* requiert des *a priori* qualitatifs : lignes, courbes, formes convexes, etc... L'implémentation de ces méthodes est donc plus simple.

En effet, le seul paramètre à fixer est la mesure de significativité d'une caractéristique : pour tout élément géométrique E, on définit par N_E la variable aléatoire qui compte le nombre d'occurrences de E dans une image $Y \in \mathsf{Y}$. E est dit ε-significatif si l'espérance sous le modèle *a contrario* de N_E est inférieure ou égale à ε. Pour toute image $Y \in \mathsf{Y}$, le test statistique pour la détection *a contrario* peut se mettre sous la forme

$$\phi^{\text{ac}}(Y) = \begin{cases} 0 & \text{si aucun élément } \varepsilon\text{-significatif n'est présent dans Y ,} \\ 1 & \text{si au moins un élément } \varepsilon\text{-significatif est présent dans Y .} \end{cases}$$

Des études ont mis à profit cette approche pour détecter des objets mobiles dans une séquence d'images [DPK05], pour repérer des changements dans des images IRM multimodales 3D ou satellitaires [RFH$^+$07, RMHM$^+$10] et une adaptation au cas où le modèle

a contrario est un fond structuré a été proposé dans [GM09]. Bien que cette approche soit adaptée à notre problématique, la faible résolution des signatures infrarouge que nous utilisons (typiquement $\ell = m = 15$) nous contraint dans l'implémentation d'un détecteur *a contrario* . De plus, l'élément E le mieux adapté serait une forme convexe (Cf. figure 9) : la variabilité de la taille de la forme ainsi que de sa photométrie risquerait de se traduire par un taux de fausse alarme élevé. Enfin, la méthode *a contrario* ne permet pas de prendre en compte la corrélation entre les bandes spectrales qu'il convient d'exploiter dans une méthodologie de détection d'anomalies pour des images multispectrales.

A .2 Éléments de détection dans des images multispectrales

La télédétection regroupe l'ensemble des techniques permettant l'acquisition d'informations sur un objet sans entrer physiquement en contact avec lui, dans le but de le détecter à distance. En particulier, la mesure du profil spectral est un renseignement précieux car représentatif de la composition physico-chimique et de la température de l'objet d'intérêt. Le développement de caméras multi / hyperspectrales capables de filmer une scène optique simultanément dans une dizaine / plusieurs centaines de bandes spectrales a rendu possible l'acquisition de telles informations. Toutefois, le traitement de ces données volumineuses demeure une problématique majeure, notamment en ce qui concerne la mise au point d'algorithmes de détection de cibles dans des images multi / hyperspectrales.

On rappelle les notations introduites dans la section Simulation pour des images multispectrales :

- On considère le découpage d'un spectre B en K sous-bandes spectrales $b_1, \ldots, b_K$ tel que : $\cup_{k=1}^{K} b_k = B$ et $\cap_{k=1}^{K} b_k = \emptyset$,
- Une image multispectrale à K bandes de dimension $\ell \times m$ est une observation $Y \in \mathsf{Y}$ où $\mathsf{Y} = \mathbb{R}^K \times \mathbb{R}^\ell \times \mathbb{R}^m$ telle que $Y = \{Y_1, \ldots, Y_K\}$,
- Pour tout $k \in \{1, \ldots, K\}$, Y_k est l'image de la scène optique intégrée dans la bande b_k, qui peut être considérée comme un vecteur de taille $P = \ell m$ dont les coordonnées correspondent à chacun de ses pixels,

$$Y_k = (y_1^{(k)}, \ldots, y_P^{(k)}) \ ,$$

- Avec un léger abus de notation, nous définissons pour tout $p \in \{1, \ldots, P\}$ par $y_p \in \mathbb{R}^K$ le p-ième pixel spectral de Y *i.e.*

$$y_p = (y_p^{(1)}, \ldots, y_p^{(K)}) \ ,$$

Comme toute méthode de détection, un algorithme de détection pour des données multi/hyperspectrales spécifie une fonction $\phi : \mathsf{Y} \to \{0, 1\}$ qui pour chaque observation détermine si elle contient une cible. Dans certains cas, ϕ fait intervenir une seconde fonction $\psi : \mathbb{R}^K \to \{0, 1\}$ qui détermine si un pixel spectral appartenant à une image multispectrale est une anomalie. Utilisant la même convention que pour ϕ, $\psi(y_p) = 0$ indique que le pixel spectral p de Y est une anomalie et inversement pour $\psi(y_p) = 0$. Enfin, nous définissons par Φ et Ψ les ensembles respectifs de ces deux types de tests.

Les erreurs commises par un test $\phi \in \Phi$ (resp. $\psi \in \Psi$) sur une observation $Y \in \mathsf{Y}$ peuvent être de deux types :

(i) fausse alarme : $\phi(Y) = 1$ et $Y \in \mathsf{H}_0$ - mesuré par la probabilité de fausse alarme $\mathrm{P_{FA}}$, aussi appelée niveau du test,

$$\mathrm{P_{FA}}(\phi) = \mathbb{P}[\phi(Y) = 1 \,|\, Y \in \mathsf{H}_0] \ ,$$

(ii) non détection : $\phi(Y) = 0$ et $Y \in \mathsf{H}_1$ - mesuré par la probabilité de non détection $\mathrm{P_{ND}}$,

$$\mathrm{P_{ND}}(\phi) = \mathbb{P}[\phi(Y) = 0 \,|\, Y \in \mathsf{H}_1] \,.$$

Remarque A.1. La probabilité de détection $\mathrm{P_D}$, aussi appelée puissance du test, définie par

$$\mathrm{P_D}(\phi) = \mathbb{P}[\phi(Y) = 1 \,|\, Y \in \mathsf{H}_1] \,,$$

sera parfois utilisée en lieu et place de $\mathrm{P_{ND}}$, ces deux quantités apportant la même information $\mathrm{P_D}(\phi) = 1 - \mathrm{P_{ND}}(\phi)$.

L'objectif consiste à trouver un test bénéficiant d'un compromis entre une faible $\mathrm{P_{FA}}$ et une forte $\mathrm{P_D}$. Le test de Neyman-Pearson [NP92] $\psi_{\mathrm{NP}} \in \Psi$, aussi appelé test du rapport de vraisemblance (*Likelihood Ratio Test, LRT*), s'écrit pour un pixel spectral $y \in \mathbb{R}^K$

$$\psi_{\mathrm{NP}}(y) = \left\{ \begin{array}{lll} 1 & \text{si} & \Lambda(y) \leq \eta \\ 0 & \text{si} & \Lambda(y) > \eta \end{array} \right. , \qquad \text{où} \qquad \Lambda(y) = \frac{f_{\theta_1}(y \,|\, \mathsf{H}_1)}{f_{\theta_0}(y \,|\, \mathsf{H}_0)} \,, \tag{13}$$

où $\eta > 0$ est le seuil du test, $f_{\theta_0}(\,\cdot \,|\, \mathsf{H}_0)$ et $f_{\theta_1}(\,\cdot \,|\, \mathsf{H}_1)$ sont respectivement les densités de probabilité du modèle de fond et du modèle de cible, de paramètre respectif $(\theta_1, \theta_2) \in \Theta_1 \times \Theta_2$. Pour un seuil η fixé, les probabilités de fausse alarme et de non détection s'écrivent dans ce cas :

$$\mathrm{P_{FA}}(\psi) = \int_{\{y \,;\, \Lambda(y) \leq \eta\}} f_{\theta_0}(y \,|\, \mathsf{H}_0) \mathrm{d}y \quad \text{et} \quad \mathrm{P_{ND}}(\psi) = \int_{\{y \,;\, \Lambda(y) > \eta\}} f_{\theta_1}(y \,|\, \mathsf{H}_1) \mathrm{d}y \,.$$

Soit $\alpha \in [0, 1]$ le niveau (probabilité de fausse alarme) de ψ_{NP}. La particularité de ψ_{NP} est que pour tout autre test $\psi \in \Psi$ tel que $\mathrm{P_{FA}}(\psi) = \alpha$, nous avons par construction

$$\mathrm{P_D}(\psi) \leq \mathrm{P_D}(\psi_{\mathrm{NP}}) \,, \tag{14}$$

ce qui justifie que le test du rapport de vraisemblance est le test le plus puissant pour un niveau α donné [1].

Remarque A.2. Pour toute fonction $g : \mathbb{R} \to \mathbb{R}$, strictement croissante, on peut montrer que le détecteur ψ_g analogue à ψ_{NP} mais dans lequel le rapport de vraisemblance Λ est remplacé par $g \circ \Lambda$ reste optimal au sens de Neyman-Pearson [Kay98].

Définition A.3. Dans la suite du document, nous dénommerons par *test de detection*, l'ensemble des fonctions $\phi \in \Phi$ (resp. $\psi \in \Psi$) et par *statistique du test*, les fonctions de $\mathsf{Y} \to \mathbb{R}$ (resp $\mathbb{R}^K \to \mathbb{R}$) associées aux tests. Typiquement, la statistique associée au test de Neyman-Pearson ψ_{NP} est le rapport des vraisemblances Λ (13).

Exemple A.4. Le détecteur à filtre adapté (*Matched Filter*) [MMS03]
Considérons l'hypothèse sous laquelle les pixels spectraux du fond et de la cible suivent respectivement une loi normale multivariée de paramètre $\theta_0 = (\mu_0, \Sigma)$ et $\theta_1 = (\mu_1, \Sigma)$ *i.e.* pour $i \in \{0, 1\}$

$$f_{\theta_i}(y \,|\, \mathsf{Y}_i) = \frac{1}{(2\pi)^K \sqrt{|\Sigma|}} \exp\left(-\frac{1}{2} (y - \mu_i)^T \Sigma^{-1} (y - \mu_i) \right) .$$

1. Cette propriété est parfois appelée *optimalité au sens de Neyman-Pearson*

Le détecteur à filtre adapté ψ_{MF} est un test de Neyman-Pearson (13) dont la statistique Λ est composée par la fonction g strictement croissante sur $\mathbb{R}$ définie par

$$g : \Lambda \to \log(\Lambda) + \frac{1}{2}(\mu_1 - \mu_0)^T \Sigma^{-1}(\mu_1 - \mu_0) \ ,$$

tel que

$$\psi_{\mathrm{MF}}(y) = \left\{ \begin{array}{lll} 1 & \text{si} & \Lambda_{\mathrm{MF}}(y) \leq \eta \\ 0 & \text{si} & \Lambda_{\mathrm{MF}}(y) > \eta \end{array} \right. , \qquad \text{où} \qquad \Lambda_{\mathrm{MF}}(y) = (\mu_1 - \mu_0)^T \Sigma^{-1}(y - \mu_0) \ . \quad (15)$$

L'intérêt de ψ_{MF} est d'obtenir un détecteur relativement simple à implémenter et qui, d'après la remarque A.2, conserve l'optimalité du test de Neyman-Pearson. Cependant, les hypothèses liées à ce test sont restrictives. Dans un cadre plus général, le signal du fond et de la cible peuvent avoir des matrices de covariance Σ_0 et Σ_1 différentes et le rapport de vraisemblance revient, moyennant une composition par une transformation strictement croissante bien choisie, à :

$$\tilde{\Lambda}_{\mathrm{MF}}(y) = (\mu_0 - y)^T \Sigma_0^{-1}(\mu_0 - y) - (\mu_1 - y)^T \Sigma_1^{-1}(\mu_1 - y) \ . \quad (16)$$

On reconnaît dans cette expression la distance de Mahalanobis D_{M} qui est une statistique fréquemment utilisée pour mesurer la vraisemblance entre des données $x \in \mathbb{R}^n$ et le modèle d'une distribution normale multivariée $\mathcal{N}(\mu, \Sigma)$:

$$D_{\mathrm{M}}(x; \mu, \Sigma) = (x - \mu)^T \Sigma^{-1}(x - \mu) \ . \quad (17)$$

Ainsi le test général $\tilde{\Lambda}_{\mathrm{MF}}$ (16) revient à comparer les distances de Mahalanobis entre un pixel spectral y avec le modèle de fond $\mathcal{N}(\mu_0, \Sigma_0)$ et avec le modèle de cible $\mathcal{N}(\mu_1, \Sigma_1)$.

Toutefois, même sous l'hypothèse de l'exemple A.4, il est rare de connaître *a priori* les paramètres (θ_0, θ_1) et les tests (15) et (16) sont donc difficilement implémentables en pratique.

Dans le cas général, lorsque les paramètres des densités $\{f_{\theta_i}(\cdot \,|\, \mathsf{Y}_i)\}_{i \in \{0,1\}}$ ne sont pas ou que partiellement connus, le test du rapport de vraisemblance peut être remplacé en utilisant les estimateurs de maximum de vraisemblance de θ_0 et θ_1. Le test résultant $\psi_{\mathrm{GLRT}} \in \Psi$, appelé test du rapport de vraisemblance généralisé (*Generalized Likelihood Ratio Test, GLRT*) [ML06] s'écrit alors pour tout $y \in \mathbb{R}^K$ comme

$$\psi_{\mathrm{GLRT}}(y) = \left\{ \begin{array}{lll} 1 & \text{si} & \hat{\Lambda}(y) \leq \eta \\ 0 & \text{si} & \hat{\Lambda}(y) > \eta \end{array} \right. , \qquad \text{où} \qquad \hat{\Lambda}(y) = \frac{\max_{\theta \in \Theta_1} f_\theta(y \,|\, \mathsf{H}_1)}{\max_{\theta \in \Theta_0} f_\theta(y \,|\, \mathsf{H}_0)} \ . \quad (18)$$

Contrairement à ψ_{NP} ou ψ_{MF}, ψ_{GLRT} ne dispose pas de la propriété d'optimalité (14) mais donne en pratique de bons résultats [MS02].

A défaut d'optimalité au sens Neyman-Pearson, certains tests statistiques ont la propriété d'avoir un taux de fausse alarme constant (*Constant False Alarm Rate, CFAR*) : un taux de fausse alarme souhaité détermine un seuil de détection η qui est indépendant de la la densité $f_{\theta_0}(\cdot \,|\mathsf{H}_0)$. C'est notamment le cas des tests pour lesquels la statistique S a, sous l'hypothèse $Y \in \mathsf{H}_0$, une distribution r_0 connue analytiquement et qui ne dépend pas de paramètres inconnus. Dans ce cas, nous avons :

$$\mathrm{P}_{\mathrm{FA}}(\phi) = \mathbb{P}(S < \eta \,|\, \mathsf{H}_0) = \int_{-\infty}^{\eta} r_0(s) \mathrm{d}s \ , \quad (19)$$

et le seuil η est une fonction de $\mathrm{P_{FA}}(\phi)$. Cette fonction est connue analytiquement dans le cas où la fonction de répartition de S sous l'hypothèse $Y \in \mathsf{H}_0$ est inversible. Dans le cas contraire, d'autres heuristiques (*e.g.* des méthodes numériques telles que les méthodes de Monte-Carlo) permettent d'approcher numériquement η pour un taux de false alarme donné.

Remarque A.5. Dans l'equation (19), l'écriture $\mathbb{P}(S < \eta \,|\, \mathsf{H}_0)$ la statistique S est considérée comme une variable aléatoire. En effet, les observations Y sont des variables aléatoires sur $\mathbb{R}^\ell \times \mathbb{R}^m \times \mathbb{R}^K$ et S désigne en toute rigueur la variable aléatoire $S(Y)$.

Détecteur RX

Introduit par Reed et Yu [RY90], le détecteur RX est un test de détection de type *Generalized Likelihood Ratio Test* (18), qui fait figure de référence dans le domaine de détection d'anomalies pour des images multi/hyperspectrales. Il est à l'origine de nombreux détecteurs qui ont été proposés, dans le but de l'adapter à divers contextes ; voir les articles de synthèse [Man05, MDC10]. Le détecteur RX recouvre de fait deux approches légèrement différentes, l'une adaptée à des cibles étendues spatialement et dénotée $\phi_{\mathrm{RX}} \in \Phi$ et l'autre $\psi_{\mathrm{RX}} \in \Psi$ adéquate pour des cibles dont la taille caractéristique ne dépasse pas un pixel.

• Les travaux initiaux de Reed et Yu [RY90, YHR$^+$97] font l'hypothèse que l'on dispose *a priori* d'un motif spatial de la cible $M \in \mathbb{R}^\ell \times \mathbb{R}^m$ et que le fond est un processus Gaussien de moyenne nulle et de matrice de covariance inconnue. Dans ce modèle, la distribution d'un pixel spectral y_p, $p \in \{1, \ldots, P\}$ s'écrit

$$\begin{cases} y_p \sim \mathcal{N}(0, \Sigma) & \text{si } Y \in \mathsf{H}_0 \ , \\ y_p \sim \mathcal{N}(\upsilon M(p), \Sigma) & \text{si } Y \in \mathsf{H}_1 \ , \end{cases} \tag{20}$$

où $\upsilon = (\upsilon_1, \ldots, \upsilon_K) \in \mathbb{R}^K$ est un vecteur de coefficient donnant l'intensité du signal dans les K sous-bandes spectrales et $\Sigma \in \mathcal{M}_K^+(\mathbb{R})$ est la matrice de covariance du fond. υ et Σ sont supposés inconnus. Avec un léger abus de notation, M est considéré dans (20) comme un vecteur de $\mathbb{R}^P$. La statistique Λ_{RX} du test $\phi_{\mathrm{RX}} \in \Phi$ proposé par [RY90, eq. 12] se déduit du rapport de vraisemblance généralisé pour ce modèle et s'écrit :

$$\Lambda_{\mathrm{RX}}(Y) = \frac{\Big[\, \langle y_1, M\rangle \ldots \langle y_P, M\rangle \,\Big] \Gamma(Y) \Big[\, \langle y_1, M\rangle \ldots \langle y_P, M\rangle \,\Big]^T}{\|M\|^2} \ , \tag{21}$$

où $\Gamma(Y) \in \mathcal{M}_P^+(\mathbb{R})$ est la matrice dont les coefficients sont définis pour tout $(i,j) \in \{1, \ldots, P\}^2$ par $\{\Gamma(Y)\}_{i,j} = \sum_{k=1}^K y_i^{(k)} y_j^{(k)}$. La popularité de ce détecteur s'explique par le fait qu'il dispose de la propriété *CFAR*. En effet la statistique Λ_{RX} a une distribution connue : elle suit une loi bêta qui ne dépend que de K et de P dans le cas où $Y \in \mathsf{H}_0$ et une loi bêta décentrée quand $Y \in \mathsf{H}_1$, [RY90, eq. (49), eq. (50)].

• La majorité des études liées à ce détecteur ont depuis travaillé à l'élaboration d'un test RX au niveau pixel $\psi_{\mathrm{RX}} \in \Psi$ [MZHS08, DS10, TGB10]. Cette approche est pertinente dans les cas où la cible recherchée a une taille typique de l'ordre d'un pixel. Ainsi, l'hypothèse de Reed et Yu sur la connaissance du motif spatial M n'est plus requise et la distribution d'un pixel spectral s'écrit pour tout $p \in \{1, \ldots, P\}$ dans ce cas :

$$\begin{cases} y_p \sim \mathcal{N}(\mu_0, \Sigma) & \text{si } Y \in \mathsf{H}_0 \ , \\ y_p \sim \mathcal{N}(\mu_1, \Sigma) & \text{si } Y \in \mathsf{H}_1 \ , \end{cases} \tag{22}$$

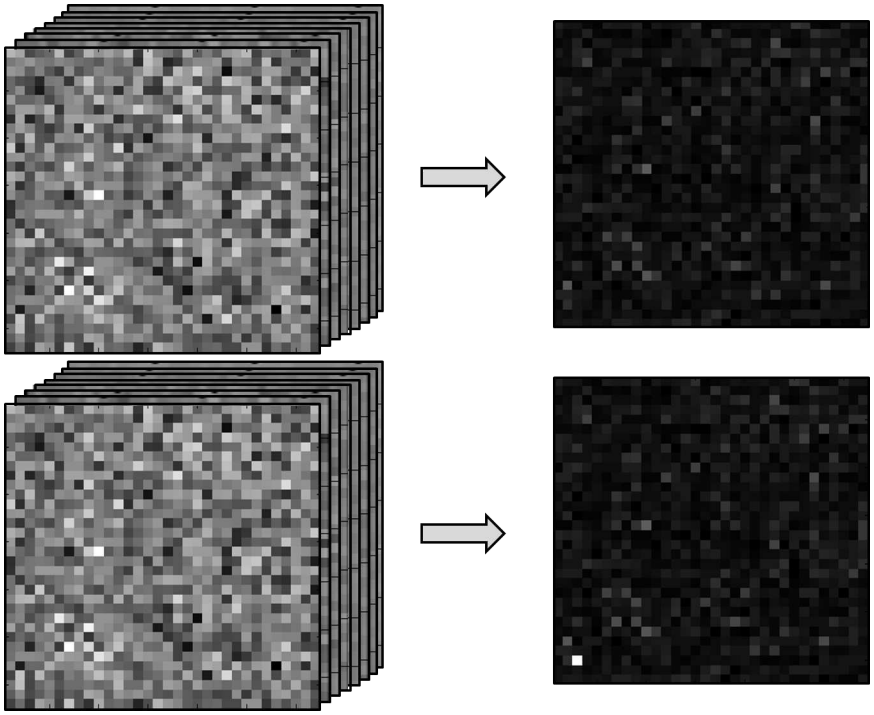


FIGURE 12 – Illustration du détecteur RX appliqué à une image multispectrale $Y \in \mathsf{Y}$ de $K = 30$ bandes : la première ligne correspond au cas $Y \in \mathsf{H}_0$ et la seconde au cas $Y \in \mathsf{H}_1$.

où $\mu_0 \in \mathbb{R}^K$ est supposé connu tandis que $\mu_1 \in \mathbb{R}^K$ et $\Sigma \in \mathcal{M}_K^+(\mathbb{R})$ sont inconnus. La statistique λ_{RX} de ψ_{RX} est déduite du rapport de vraisemblance généralisé et peut se mettre sous la forme :

$$\lambda_{\mathrm{RX}}^n(y_p) = (y_p - \mu_0)^T \left(\frac{n}{n+1}\hat{\Sigma} + \frac{1}{n+1}(x-\mu_0)(x-\mu_0)^T \right)^{-1} (y_p - \mu_0) \;, \tag{23}$$

où $\hat{\Sigma}$ est la matrice de covariance empirique, estimée à partir de n pixels de référence constituant une base d'apprentissage. En pratique, la plupart des travaux considèrent un très grand nombre de pixels de référence n, de sorte à ce que la statistique λ_{RX} s'écrit plus généralement :

$$\begin{aligned} \lambda_{\mathrm{RX}}^{\infty}(y_p) &= (y_p - \mu_0)^T \hat{\Sigma}^{-1} (y_p - \mu_0) \;, \\ &= D_{\mathrm{M}}(y_p; \mu_0, \hat{\Sigma}) \;, \end{aligned} \tag{24}$$

où D_M est la distance de Mahalanobis (17). La popularité de ce test s'explique en partie par sa simplicité d'implémentation. De plus, la distribution de la statistique $\lambda_{\mathrm{RX}}^{\infty}$ est connue : il s'agit d'une loi du chi-deux à K degrés de liberté sous l'hypothèse H_0 et d'une loi du chi-deux décentrée à K degrés de liberté, de moyenne $(\mu_1-\mu_0)^T\hat{\Sigma}^{-1}(\mu_1-\mu_0)$ sous l'hypothèse H_1 [SBH+02]. En conséquence, de la même façon que pour ϕ_{RX}, ψ_{RX} possède la propriété *CFAR*.

Exemple A.6. La figure 12 illustre le détecteur RX sur un modèle jouet. On considère une image multispectrale Y à $K = 30$ bandes, telle que pour tout $(k,p) \in \{1,\dots,K\} \times \{1,\dots,P\}$, le pixel p de la bande k est simulé par

$$y_p^{(k)} \sim \mathcal{N}(0, k^{-1}) \;.$$

Une seconde image $\bar{Y}$ est créée à partir de Y dans laquelle une anomalie sur le pixel p appartenant à la 3-ème colonne et à l'antépénultième ligne a été artificiellement ajoutée : pour $k \in \{15, \ldots, 20\}$ $\bar{y}_p^{(k)} = 0.1$. Le modèle de fond est supposé connu et correspond à la distribution de Y. Les deux images de la colonne de droite de la figure 12 correspondent de haut en bas respectivement à la collection de pixels $\{\lambda_{\mathrm{RX}}^{\infty}(y_p),\ 1 \leq p \leq P\}$ et $\{\lambda_{\mathrm{RX}}^{\infty}(\bar{y}_p),\ 1 \leq p \leq P\}$, où $\lambda_{\mathrm{RX}}^{\infty}$ est la fonction définie dans l'équation (24). L'anomalie, bien que présente dans uniquement 6 bandes spectrales d'un seul pixel, est clairement révélée par le détecteur RX.

De nombreuses variantes de l'algorithme RX (24) ont été proposées dans la littérature, citons entre autres :

- le *local RX* [MDC10] : le détecteur est appliqué séquentiellement à tous les pixels spectraux de Y. Pour chaque pixel spectral y_p, les estimateurs empiriques $\hat{\Sigma}_p$ et $\hat{\mu}_{0,p}$ de Σ et μ_0 sont calculés à partir des pixels se trouvant dans un voisinage de y_p en utilisant, par exemple, une fenêtre centrée sur ce pixel.
- le *subspace RX* [Sch07] dans lequel Σ est estimée après avoir enlevé un certain nombre de bandes correspondantes aux principales composantes (*i.e.* ayant des variances élevées), obtenues par une Analyse en Composante Principale.
- le *cluster based RX* [BSH00, SS97] pour les cas où le fond est constitué de plusieurs éléments (forêt, désert, mer, ...). L'idée est dans un premier temps de segmenter l'image est d'apprendre pour chacune des classes les statistiques de fond, en utilisant par exemple un mélange de gaussiennes, puis de comparer la distance de Mahalanobis entre chaque pixel et les différents fonds.

A .3 Liens avec notre problématique

Malgré les avantages qu'elles présentent (facilité d'implémentation et contrôle théorique du niveau de test), ces différentes approches ne sont adaptées à notre problématique. En effet, les remarques de la Section A .1 montrent que, dans un contexte de détection de SIR multispectrales, l'implémentation d'un détecteur à filtre adapté ou de type RX (exemple A.4 et paragraphe Détecteur RX) n'est pas pertinente. Tout d'abord, le modèle Gaussien pour l'hypothèse $Y \in \mathsf{H}_1$ supposé dans le filtre adapté n'est pas réaliste pour décrire la signature infrarouge des aéronefs. Par ailleurs, les avions occupant, en moyenne, une dizaine de pixels de l'image multispectrale, le test RX au niveau des pixels ψ_{RX} ne permet pas de prendre en compte la dispersion spatiale des SIR et n'est donc pas pertinent. Enfin, le test RX au niveau des observations ϕ_{RX} comporte lui aussi des hypothèses trop restrictives. D'une part, dans notre cas la dispersion spatiale des SIR ne permet pas de disposer du modèle de motif M intervenant dans le calcul de la statistique Λ_{RX} et d'autre part, l'hypothèse que le vecteur υ, bien que supposé inconnu, suffit à rendre compte de la dispersion spectrale des SIR multispectral n'est, là encore, pas réaliste. En effet, cela reviendrait à faire l'hypothèse que tous les aéronefs ont des spectres qui ne varient qu'à une constante de proportionnalité près ; hypothèse qui n'est pas vérifiée dans la majorité des cas (voir par exemple la figure 5(b)).

Une démarche tenant compte simultanément des informations spatiale et spectrale contenues dans des images hyperspectrales a toutefois été proposée dans [CVGCMM$^+$06]. L'objectif de ce travail n'est pas de détecter des anomalies mais de classifier des zones d'images hyperspectrales dans différentes catégories. La technique de classification employée fait intervenir l'apprentissage de noyaux. Le principe des méthodes à noyau est de plonger les données d'apprentissage dans un espace où elles sont linéairement séparables, afin d'établir plus simplement un algorithme de classification. L'un des intérêts majeurs

de cette approche est de pouvoir composer différents noyaux élémentaires permettant de passer d'un espace de représentation à un autre dans le but de mieux discriminer les classes. Plusieurs types de noyaux composites combinant des noyaux travaillant sur les caractéristiques spectrales et d'autres sur les caractéristiques spatiales sont proposés dans [CVGCMM$^+$06], et permettent d'exploiter conjointement ces deux types d'informations. Cependant la faible résolution des SIR d'aéronefs, typiquement de dimension 15×15 pixels, rend l'application de ces méthodes impossible dans le cadre de la détection d'anomalies.

Par ailleurs, utilisant les acquis de la morphologie mathématique [Ser82], des travaux concernant le partitionnement d'images hyperspectrales utilisent une approche prenant en compte les caractéristiques spatiales et spectrales [PMPP05, BPS05]. Plus précisément, une analyse en composante principale est réalisée dans un premier temps sur l'image hyperspectrale (ou tout autre algorithme de réduction de données), prenant ainsi en compte les informations spatiales qu'elle véhicule. Dans un second temps, un profil morphologique étendu est créé à partir de la composante principale issue de l'ACP : basé sur la théorie de la morphologie mathématique, ce profil permet d'extraire des informations sur les formes, les tailles et les structures présentes dans l'image [Fau07]. Enfin, un noyau composite utilisant ces deux types d'informations permet de classifier les pixels de l'image. Toutefois, ces méthodes ne s'appliquent, là encore, qu'à des images fortement résolues spatialement et ne sont donc pas adaptables à notre situation.

Ces différentes remarques justifient le développement d'une méthodologie adaptée à la détection d'aéronefs dans des images multispectrales de faible résolution et exploitant simultanément les informations spatiales et spectrales.

A .4 Détection d'aéronefs à partir de SIR monospectrale

Une contribution récente a permis d'établir plusieurs détecteurs adaptés à la détection d'aéronefs à partir de signatures infrarouge monospectrales [M6]. Contrairement aux travaux cités ci-dessus, cette étude traite des données similaires à celles que nous utilisons dans cette thèse, simulées par la méthode décrite dans la section Simulation de SIR, mais en monospectral et non multispectral (*e.g.* figure 9(a)). La méthodologie proposée ne tient par conséquent compte ni de la dispersion spectrale des observations ni de l'information, potentiellement discriminatoire, transmise par les SIR multispectrales.

On rappelle que dans le cas de SIR monospectrales, une observation $Y \in \mathsf{Y} = \mathbb{R}^\ell \times \mathbb{R}^m$ peut-être identifiée à une collection de pixels $\{y_1, \ldots, y_P\}$ où $P = \ell m$. Supposant que la SIR d'aéronef ait un rayonnement dans l'infrarouge plus important que le fond, un test de détection naturel est de considérer le test $\phi_1 \in \Phi$ qui compare la statistique

$$S_1(Y) = \max_{p \in \{1,\ldots,P\}} y_p$$

à un seuil $\eta_1 \in \mathbb{R}$ défini comme étant un quantile élevé de la distribution du fond. Bien que ce test donne de bons résultats sur des images « faciles », comme par exemple les quatre premières images de la figure 9(a), il échoue typiquement sur des images « plus difficiles », comme par exemple la dernière image de la même figure, où le contraste entre la cible et le fond est faible. En conséquent ϕ_1 présente un taux de fausses alarmes élevé. Il est possible de réduire ce taux en considérant, en plus de la photométrie de chaque pixel, son information contextuelle. En effet, un aéronef présentant une certaine cohérence spatiale, les pixels le décrivant appartiennent généralement à un ensemble connexe de pixels ayant une photométrie plus élevée que la moyenne. Dans cette optique, les techniques étudiant les ensembles de niveaux dans des images [OP03] sont des outils précieux permettant

d'exploiter cette information. Deux tests $(\phi_2, \phi_3) \in \Phi^2$ basés sur ces méthodes ont ainsi été proposés.

Rappelons qu'à toute image $Y \in \mathsf{Y}$, il est possible d'associer une fonction $I(Y) : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ définie comme l'interpolation sur $\mathbb{R}^2$ de Y (par exemple en utilisant une interpolation bi-cubique). Le test ϕ_2, étudie les ensembles convexes de $\mathbb{R}^2$ ayant une photométrie supérieure ou égale à un seuil $\nu \in \mathbb{R}$. La statistique de ce test s'écrit :

$$S_2(Y) = \max_{E \in \mathrm{cc}(Y,\nu)} |E| \,,$$

où pour tout $E \subset \mathbb{R}^2$, $|E|$ désigne l'aire de E et pour tout $Y \in \mathsf{Y}$, $\nu \in \mathbb{R}$

$$\mathrm{cc}(Y, \nu) = \left\{ E \subset \mathbb{R}^2,\ E \text{ est convexe},\ \forall\, u \in E,\ I \circ Y(u) \geq \nu \right\} .$$

L'avantage de prendre en compte l'environnement contextuel de chaque pixel est de pouvoir diminuer le seuil η_1 du test ϕ_1, afin de permettre la détection des cibles ayant un faible contraste avec le fond, sans pour autant augmenter le taux de fausse alarme. Pour le modèle de fond Gaussien *i.i.d.* (2), si $Y \in \mathsf{H}_0$, $S_2(Y)$ sera probablement très faible lorsque $\nu \gg \sigma_{\mathrm{II}}$. La figure 13 illustre le détecteur ϕ_2 appliqué sur une SIR monospectrale ayant un fond gaussien *i.i.d.* . Toutefois, pour le modèle de fond texturé, cette approche n'est pas aussi efficace car en raison de la corrélation spatiale, ces fonds nuageux peuvent présenter de larges zones où les pixels ont un niveau un peu plus élevé que la moyenne. En conséquence, le taux de fausse alarme ne diminue que faiblement par rapport à ϕ_1 dans ce cas.

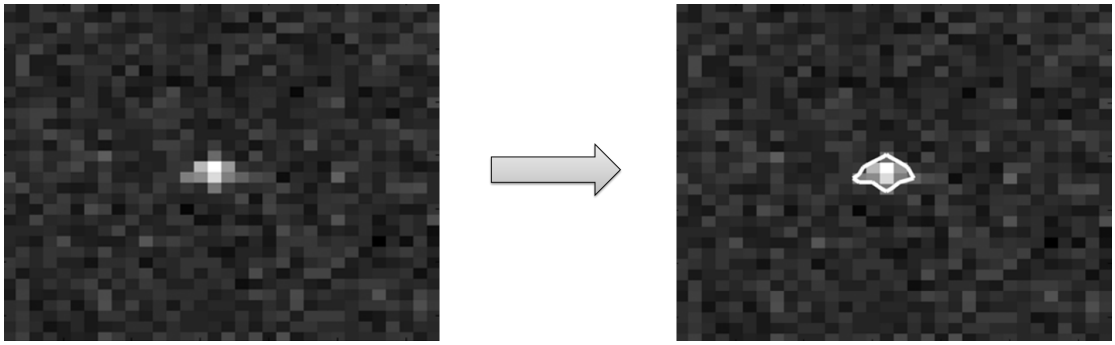


Figure 13 – Ensembles de niveau λ correspondant au quantile 0.99 de la distribution des pixels de l'image

En plus de l'information spatiale exploitée par ϕ_2, le dernier test ϕ_3 permet de prendre en compte la photométrie des pixels. Plus précisément, pour toute fonction $f \in \mathcal{C}(\mathbb{R}^2)$ à support compact, on définit la norme en variation totale par :

$$\|f\|_{\mathrm{TV}} = \int_{\mathbb{R}^2} |\nabla f| \,, \quad \text{où} \quad |\nabla f| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \,.$$

La statistique S_3 du test ϕ_3 utilise la norme en variation totale et est définie pour tout $Y \in \mathsf{Y}$ par :

$$S_3(Y) = \int_{A_\nu(Y)} |\nabla I(Y)| \,, \quad \text{où} \quad A_\nu(Y) = \{u \in \mathbb{R}^2,\ I \circ Y(u) \geq \nu\} \,.$$

Cette statistique calcule les variations de photométries dans des zones où l'image a un niveau supérieur à un seuil ν. Celle-ci s'avère particulièrement pertinente lorsque le fond

est texturé : pour une image $Y \in \mathsf{H}_0$, la statistique $S_3(Y)$ sera faible car les variations de la photométrie sont faibles, contrairement à $S_2(Y)$.

Pour $i \in \{1,2,3\}$, les tests $\phi_i \in \Phi$ sont définis par :

$$\phi_i(Y) = \begin{cases} 0 & \text{si} \quad S_i(Y) \leq \eta_i \\ 1 & \text{si} \quad S_i(Y) > \eta_i \end{cases} , \tag{25}$$

où $\{\eta_i\}_{i\in\{1,2,3\}}$ sont des seuils à spécifier, permettant d'atteindre une $\mathrm{P_{FA}}$ souhaitée. Il a été montré dans [M6] que, dans le cas du modèle de fond Gaussien, pour le test ϕ_1, le seuil η_1 est une fonction analytique du taux de fausse alarme souhaité et dispose de fait de la propriété *CFAR*. En revanche, dans le cas du modèle de fond texturé, tout comme pour les détecteurs ϕ_2 et ϕ_3, les seuils ne sont plus des fonctions analytiques de la $\mathrm{P_{FA}}$ souhaitée. Dans ces cas, des tirages de Monte-Carlo de $\{Y_k,\, k \in \mathbb{N}\}$ sous le modèle de H_0 permettent d'obtenir une valeur approchée d'un quantile q de $S(Y) \,|\, Y \in \mathsf{H}_0$ et ainsi de fixer le seuil η_i pour avoir $\mathrm{P_{FA}}(\phi_i) \approx 1 - q$.

Le Tableau 2 présente les résultats de détection obtenus dans [M6] sur une base de 90 000 avions simulés par la méthode présentée dans la section Simulation de SIR. Ce tableau confirme que pour le modèle de fond Gaussien *i.i.d.* , le détecteur ϕ_2 est plus efficace que le détecteur ϕ_3 et inversement dans le cas des fonds texturés.

Tableau 2 – Pourcentage de détection d'aéronefs à partir de SIR multispectrales avec $\mathrm{P_{FA}} = 0.01$

(a) fond Gaussien - $\sigma_{\mathrm{II}}^{(1)}$

avion	ϕ_1	ϕ_2	ϕ_3
1	86.3	89.1	80.5
2	93.3	91.2	85.8

(b) fond Brownien fractionnaire - $\sigma_{\mathrm{II}}^{(2)}$ et $H = 0.55$

avion	ϕ_1	ϕ_2	ϕ_3
1	59.2	14.1	90.7
2	84.6	14.6	92.1

(c) fond Brownien fractionnaire - $\sigma_{\mathrm{II}}^{(3)}$ et $H = 0.55$

avion	ϕ_1	ϕ_2	ϕ_3
1	6.5	2.9	56.6
2	35.1	3.3	68.6

Résumé de la contribution

L'intérêt des détecteurs multispectraux et hyperspectraux a été prouvé dans le domaine de la télé-détection [HCYW96, CC02] et plusieurs études soulignent leur application potentielle à la détection de cibles [KR02, MS02]. Cependant à l'heure actuelle, peu de détecteurs multispectraux existent dans le domaine spectral correspondant à l'infrarouge. Nous avons donc recours aux SIR simulées par CRIRA (voir la section sur la Simulation des SIR) pour spécifier un capteur multispectral infrarouge à faible résolution spatiale, capable de détecter des avions situés à grande distance. Les incertitudes sur les données d'entrées du simulateur induisent une dispersion sur les SIR à la fois spectrale et spatiale que la méthodologie doit intégrer. Par conséquent, étant donné la faible information dont nous disposons sur les cibles *a priori* , nous adoptons une démarche de type détection d'anomalies (voir Préambule A .1).

Dans cette étude, nous nous plaçons dans un cadre où le fond est modélisé par un processus stationnaire aléatoire dont la distribution est connue. Cette hypothèse permet de simplifier les calculs mais n'est pas déterminante dans l'application de la méthodologie. Par ailleurs, en l'absence de modèle de fond texturé en multispectral, les observations sur lesquelles nous travaillons dans l'hypothèse d'un ciel nuageux, sont ici des SIR $Y = (Y_1, \ldots, Y_K)$ avec un fond gaussien multispectral d'écart type $\{\sigma_1, \ldots, \sigma_K\}$ (11) équivalent à $\sigma_{\mathrm{II}}^{(2)}$.

Une nouvelle statistique de détection, tenant compte des deux types de dispersion est proposée : elle consiste à calculer dans un premier temps la transformée de Mahalanobis (17) de l'image multispectrale, puis, dans un second temps, à étudier les ensembles de niveau de cette image intermédiaire afin de révéler l'éventuelle présence d'anomalies. Ces deux étapes combinent (i) la démarche adoptée par le détecteur RX pour des cibles dont la taille typique n'excède pas un pixel [MZHS08, DS10, TGB10] (voir le Paragraphe Détecteur RX) et (ii) l'étude des ensembles de niveau proposée dans [M6], pour la détection de cibles infrarouge en bande large (voir le Préambule A .4). La combinaison de ces deux approches permet en effet d'exploiter respectivement les caractéristiques spectrales et spatiales des cibles. Notons que dans les méthodes de détection directe, la dispersion des SIR est généralement considérée comme une nuisance, tandis qu'elle est ici exploitée comme un atout permettant de distinguer une cible du fond. Les seuils intervenant dans ce test ne pouvant être calculés explicitement en fonction d'une probabilité de fausse alarme souhaitée sont estimés à partir de simulation d'images de fond de ciel par des méthodes de Monte-Carlo. Enfin, l'utilisation des ensembles de niveau permet au détecteur de localiser spatialement une cible dans une image, ce qui permet, entre autre, d'effectuer une analyse *a posteriori* sur la géométrie des anomalies détectées.

Le détecteur est appliqué à une base de test composée d'images multispectrales, certaines contenant une cible et d'autres étant uniquement des images de fond de ciel. Ces images multispectrales sont constituées de 10 bandes spectrales élémentaires, *arbitrairement* choisies, de même largeur spectrale 100 cm^{-1} et régulièrement réparties dans l'intervalle 2000-3000 cm^{-1} ; voir par exemple les trois images de la figure 14. La comparaison de courbes caractéristiques de plusieurs détecteurs (figure I-6) illustre deux principaux résultats :

- le détecteur proposé alliant études des caractéristiques spectrales des données et ensembles de niveaux possède de meilleures performances de détection que le détecteur RX, qui exploite simplement les caractéristiques spectrales de l'image,
- les cibles pouvant avoir un contraste inversée par rapport au fond, il est important de considérer à la fois des quantiles faibles et des quantiles élevés de la transformée de

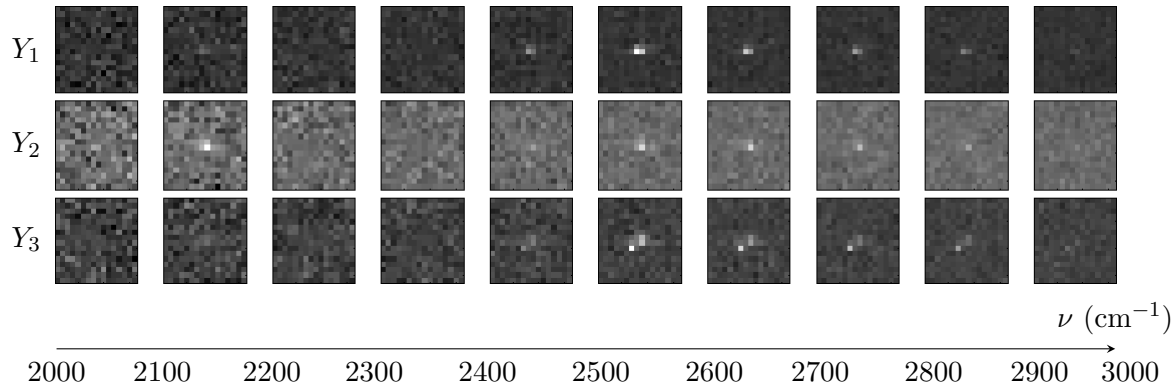


FIGURE 14 – Trois images multispectrales du même avion dans le même scénario

Mahalanobis des images multispectrales pour la spécification des seuils intervenant dans les tests statistiques.

Il est toutefois possible d'améliorer ces performances en ne prenant en compte que certains regroupements de bandes dans l'intervalle 2000-3000 cm^{-1}. En effet, en raison de l'absorption de certains gaz composant l'atmosphère, la SIR d'un aéronef est presque toujours nulle dans certaines bandes élémentaires qu'il est donc inutile de conserver ; voir par exemple les deux bandes 2200-2300 cm^{-1} et 2300-2400 cm^{-1} des images multispectrales de la figure 14.

Les bandes spectrales les plus pertinentes pour la détection sont sélectionnées au moyen d'un algorithme génétique qui optimise soit la probabilité de détection du test pour un taux de fausse alarme fixé soit l'aire sous la courbe caractéristique du détecteur. Pour ces deux critères, la même combinaison optimale comprenant deux regroupements de bandes disjointes est obtenue : le premier aux alentours de 2000 cm^{-1} et le second au voisinage de 2500 cm^{-1}. D'un point de vue physique, ce résultat est cohérent avec le spectre de rayonnement typique dans la bande II d'un aéronef illustré par la figure 3. On constate par ailleurs que le premier regroupement a une largeur spectrale plus faible que le second : ceci s'explique par le fait que la luminance spectrale du fond diminue lorsque le nombre d'ondes augmente (voir Figure 11). Or, sous le modèle (11), la variance du fond est proportionnelle à la luminance et diminue donc avec le nombre d'onde. Par conséquent, un regroupement *large* en début de la bande II aura un niveau de bruit plus élevé qu'un regroupement de même largeur au milieu ou à la fin de la bande II. Cependant, plus la bande est large, plus le niveau du signal est élevé et plus il est possible de détecter une cible. L'optimisation revient donc à trouver un compromis entre la largeur des bandes spectrales sélectionnées et le rapport signal à bruit associé.

La figure I-12 illustre les performances de notre détecteur sur des images multispectrales constituées de 2, 3 ou 4 regroupements de bandes élémentaires optimaux et sur des images intégrées en bande II. Les résultats montrent que, pour ces bandes spectrales bien choisies, les performances de détection en multispectral sont meilleures qu'en monospectral. Ceci justifie, dans cette application, l'intérêt du multispectral.

Cette méthodologie de détection peut aisément être adaptée à d'autres scénarios et à d'autres cibles faiblement résolues sur des fonds naturels de type forêt, désert, mer... Toutefois, pour des fonds réels, l'hypothèse d'un modèle de fond gaussien peut-être remise en question. Dans ce cas, un estimateur de la matrice de covariance intervenant dans le calcul de la transformée de Mahalanobis plus approprié que celui proposé par la matrice de covariance empirique consisterait à calculer un M-estimateur [Hub64, Mar76] de la matrice

de covariance. Cet estimateur serait en outre plus robuste aux aberrations pouvant exister dans des données réelles.

Il serait également intéressant d'étudier les performances de l'algorithme de détection dans le cas d'un modèle de fond nuageux en multispectral plus réaliste que le bruit Gaussien en multispectral d'écart-type équivalent à $\sigma_{\mathrm{II}}^{(2)}$. Dans une première approximation, un modèle de fond texturé en multispectral peut consister à modéliser les images dans chaque bande comme des draps browniens ou des champs de Markov non indépendants. L'acquisition d'images multispectrales de nuages permettraient d'estimer les paramètres d'un tel modèle et de proposer un modèle de corrélation inter-bande adéquat.

B Motifs caractéristiques d'un ensemble de données et modèles de déformation

B .1 Position du problème

L'accroissement des dispositifs permettant l'acquisition de données de différente nature (courbes, images, vidéos, etc...) et généralement de grandes dimensions, soulève la question de leur traitement. Prises individuellement, ces données ne sont, la plupart du temps, pas significatives et il est nécessaire de les comparer, de confronter les unes avec les autres afin de pouvoir dégager des caractéristiques communes et d'identifier des sources de variabilité. Nous nous intéressons plus particulièrement au cas où les données d'intérêt sont des mesures d'une même expérience se déroulant dans un contexte variable et présentant de fait une structure sous-jacente commune. Un grand nombre de données entrent dans ce cadre, en particulier dans le domaine de l'imagerie médicale [MT96], en météorologie [ZC07, VP11], en génétique [HZ00], en finance [LPP02] ou encore en traitement de la parole [KG92, DPH00]. Les signatures infrarouge d'avions acquises par la méthode proposée dans la section simulation de SIR sont également concernées par cette approche et justifient l'intérêt porté à ces méthodes. Dans ce cas précis, l'expérience consiste à mesurer la signature infrarouge d'un aéronef dans plusieurs bandes spectrales. Le contexte expérimental pouvant varier (dans notre étude, en raison de l'incertitude sur les données d'entrée du simulateur), le signal observé diffère d'une simulation à l'autre.

Pour extraire des informations de ces bases de données, chaque observation est perçue comme la version perturbée d'une observation moyenne, hypothétique, inconnue et caractéristique de l'ensemble des mesures possibles liées à l'expérience. L'estimation de cette observation caractéristique permet, entre autres, de reconnaître ou classifier une nouvelle observation en la comparant à cette donnée moyenne. Nous considérons le cas d'une base de données composée d'images de chiffres trois manuscrits, issus de la base US Postal [Hul94] (figure 15) et pour se placer dans un cadre expérimental, un bruit Gaussien *i.i.d.* est ajouté à ces images initialement débruitées. Comme le montre la figure 15, les différentes images ont des caractéristiques communes comme par exemple la présence dans la structure géométrique de trois barres horizontales reliées par des lignes incurvées, ou encore la photométrie de l'image qui est nulle partout sauf au centre de l'image où les pixels appartenant au chiffre valent 1. Elles sont toutefois toutes différentes et rentrent de ce fait dans notre cadre d'étude.

Nous montrons dans cette sous-partie que la plupart des méthodes classiques d'analyse de données ne sont pas adaptées au cas d'observations présentant de fortes variations et ne permettent par conséquent pas d'extraire une observation caractéristique d'un tel ensemble de données.

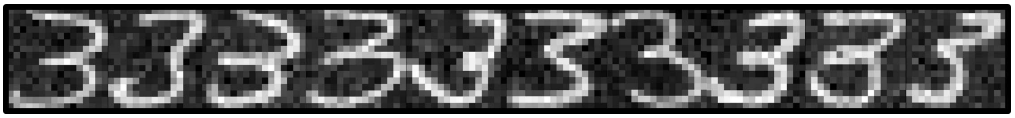


FIGURE 15 – Exemples d'images bruitées de chiffres trois manuscrits issus de la base US Postal.

Pour accéder à une observation caractéristique, une idée naturelle consiste à calculer la moyenne empirique des $n = 1000$ observations de la base $\{Y_k,\ k \leq n\}$, $\bar{Y}_n = n^{-1} \sum_{k=1}^{n} Y_k$. Comme le montre la figure 16, cette image moyenne, floue, ne permet pas d'appréhender les principales caractéristiques des chiffres trois.

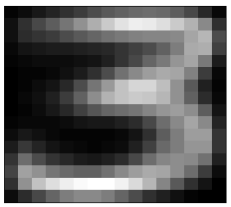
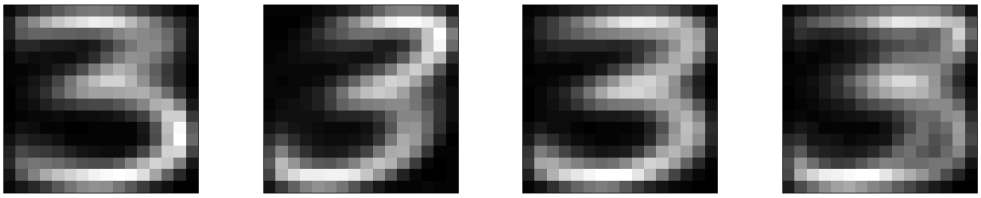


Figure 16 – Moyenne empirique $\bar{Y}_n$ de $n = 1000$ images de chiffres trois manuscrits illustrés dans la figure précédente.

L'Analyse en Composante Principale (ACP) [Jol05] est une technique fréquemment utilisée en traitement d'images dans le but de décrire et de visualiser les caractéristiques d'une collection d'images. L'idée est de diagonaliser la matrice de covariance empirique $\Gamma = n^{-1} \sum_{k=1}^{n} (Y_k - \bar{Y}_n)(Y_k - \bar{Y}_n)^T$, afin d'extraire les directions où la variabilité contenue dans les images est la plus importante. La figure 17 illustre les deux principaux modes de variations obtenus en appliquant une ACP à $n = 1000$ images de chiffres trois manuscrits. Tout comme pour la moyenne empirique, les composantes principales de l'ACP ne fournissent qu'une tendance approximative des géométries représentatives des images de chiffres trois manuscrits.



(a) $\bar{Y}_n + \sqrt{\lambda_1}\omega_1$ (b) $\bar{Y}_n - \sqrt{\lambda_1}\omega_1$ (c) $\bar{Y}_n + \sqrt{\lambda_2}\omega_2$ (d) $\bar{Y}_n - \sqrt{\lambda_2}\omega_2$

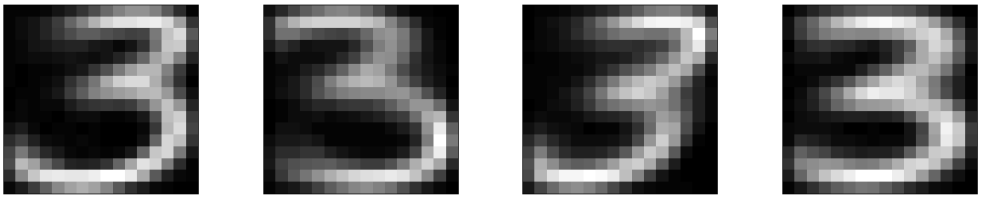
Figure 17 – Premier et second mode de variation obtenus par ACP sur la base des chiffres trois où pour $j \in \{1, 2\}$ ω_j est le vecteur propre associé à la j-ème plus grande valeur propre λ_j de Γ.

L'algorithme des K-moyennes (*K-means*) [Har75] est une méthode pour partitionner des données en K classes $\{\mathcal{C}_1, \ldots, \mathcal{C}_K\}$. L'objectif est d'attribuer chaque observation Y_k à l'une des K classes de sorte à ce que la fonction

$$Q(Y_1, \ldots, Y_n, \mathcal{C}_1, \ldots, \mathcal{C}_K) = \sum_{i=1}^{K} \sum_{Y_k \in \mathcal{C}_i} \|Y_k - \mu_i\|^2 ,$$

soit minimale. Pour tout $i \in \{1, \ldots, K\}$, μ_i est la moyenne des observations $Y_k \in C_i$. Cet algorithme a été appliqué à $n = 1000$ images de chiffres manuscrits dans le but d'obtenir $K = 4$ classes représentatives. Bien que cet algorithme permette de dégager des géométries caractéristiques des chiffres trois de façon assez satisfaisante, voir la figure 18, il ne permet pas d'estimer quantitativement les variations entre les observations. De plus, l'exemple suivant prouve qu'il n'est pas adapté à des données présentant davantage de variations.

Les limites de ces méthodes sont plus apparentes dans le cas où les données font l'objet de plus de variabilité. En effet les chiffres manuscrits de la base US Postal présentent une certaine régularité : ils sont centrés, leur photométrie est régulière etc... Nous appliquons

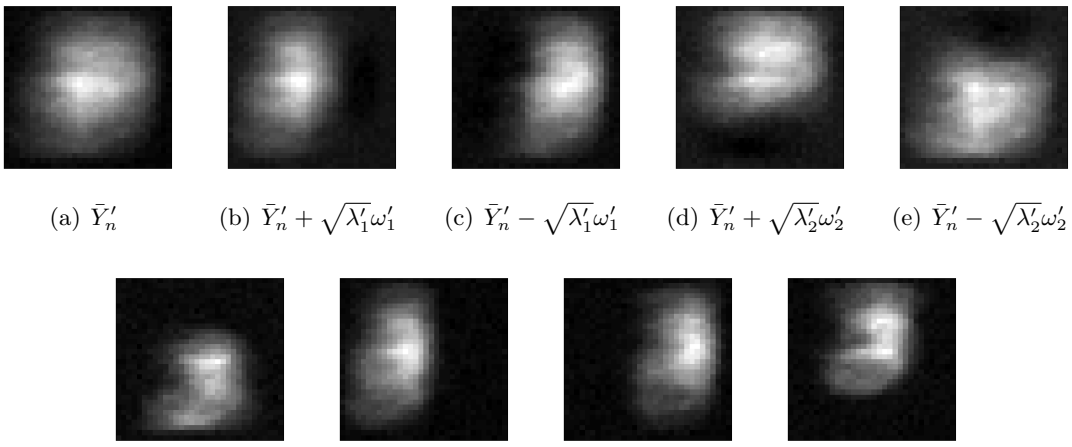


(a) μ_1 (b) μ_2 (c) μ_3 (d) μ_4

FIGURE 18 – Centres des classes obtenues en appliquant l'algorithme des K-moyennes.

les méthodes précédentes dans le cas, artificiel mais réaliste pour des applications expérimentales, où les chiffres ne sont plus nécessairement centrés. Comme le montre la figure 19, nous ajoutons une translation aléatoire à chaque image de la base.

FIGURE 19 – Exemples d'images bruitées de chiffres trois manuscrits non centrés.



(a) $\bar{Y}'_n$ (b) $\bar{Y}'_n + \sqrt{\lambda'_1}\omega'_1$ (c) $\bar{Y}'_n - \sqrt{\lambda'_1}\omega'_1$ (d) $\bar{Y}'_n + \sqrt{\lambda'_2}\omega'_2$ (e) $\bar{Y}'_n - \sqrt{\lambda'_2}\omega'_2$

(f) μ'_1 (g) μ'_2 (h) μ'_3 (i) μ'_4

FIGURE 20 – Images caractéristiques de chiffres trois non centrés, obtenues par diverses méthodes. (a) moyenne empirique, (b-e) ACP, (f-i) K-moyennes

La figure 20 montre que cette légère perturbation a des conséquences importantes sur l'estimation de l'image caractéristique : contrairement aux figures 17 et 18, il est presque impossible de reconnaître la forme d'un chiffre trois dans ces images moyennes. En effet, ces méthodes statistiques usuelles traitent les différents pixels d'une image comme autant d'informations indépendantes. Les éléments contextuels propres à chaque pixel ne sont donc pas pris en compte dans ces analyses. En conséquence, dans un cadre expérimental réaliste où de telles déformations peuvent se produire, l'apprentissage des structures géométriques caractéristiques de ces images s'avère délicat.

Un exemple encore plus frappant consiste à considérer des images de quatre chiffres manuscrits (*e.g.* zéro, trois, cinq et huit) que l'on place dans l'un des quatre coins d'images

d'une base d'apprentissage. Nous comparons les centres des classes obtenus par l'algorithme des K-moyennes ($K = 4$) appliqué à ces images dans deux configurations :

- cas A : Chaque chiffre est systématiquement dans le même coin,
- cas B : Chaque chiffre est placé aléatoirement dans l'un des coins (figure 21).

Figure 21 – Chiffres (zéro, trois, cinq et huit) issus de la base US Postal positionnés aléatoirement de façon uniforme dans l'un des coins de l'image.

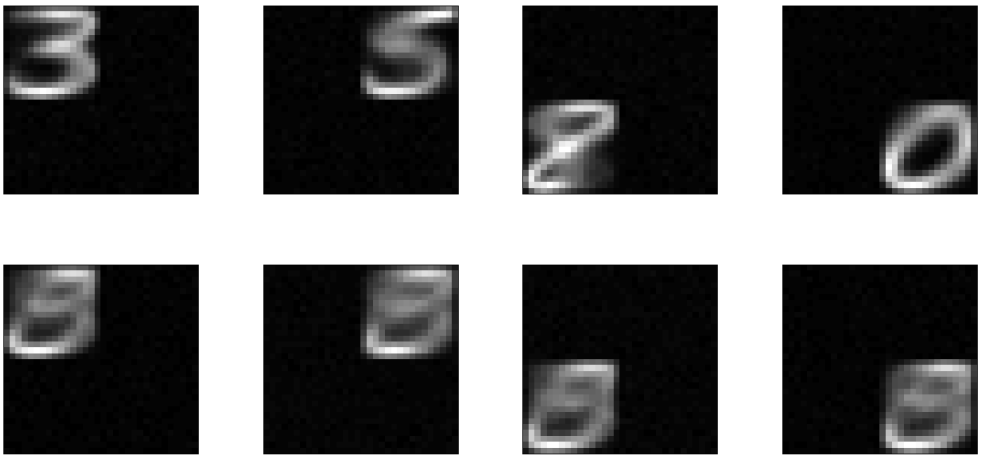


Figure 22 – Images caractéristiques de chiffres zero, trois, cinq et huit obtenues par l'algorithme K-moyennes : le 1-ère ligne correspond à la base d'apprentissage du cas A et la 2-ème ligne à celle du cas B

Alors que les quatre images caractéristiques attendues dans cet exemple correspondraient à une image de chaque chiffre, la figure 22 montre que dans le cas B, les centres des classes fournies par l'algorithme des K-moyennes tiennent avant tout compte de la position des chiffres dans l'image plutôt que de la forme caractéristique de chacun. Quand les chiffres occupent systématiquement la même place (cas A), les centres des classes sont plus proches des formes caractéristiques attendues.

Remarque B.1. Nous avons utilisé la position relative de chaque chiffre pour montrer que dans de telles situations, les algorithmes d'analyse de données classiques (ACP, K-moyennes, etc...) ne sont pas capables d'extraire des formes caractéristiques contenues dans une base de données d'apprentissage. Nous aurions alternativement pu travailler avec une base de chiffres qui diffèrent par une simple rotation, par un effet de zoom ou par toute autre transformation du plan qui conserve la structure géométrique de ces images. A la place d'images, nous aurions pu établir les mêmes conclusions en utilisant une base de courbes présentant l'évolution d'un phénomène quelconque et s'intéresser dans ce cas à l'extraction de motifs caractéristiques mais se produisant à des instants différents...

En résumé, l'analyse de ce type de données doit nécessairement prendre en compte le signal (courbe, image, etc...) dans sa globalité afin de pouvoir établir des relations entre

les caractéristiques des différentes observations. L'objectif est d'extraire de ces données des motifs typiques, représentatifs de la diversité des observations en faisant abstraction des paramètres de présentation (translation, rotation, effet d'échelle, déformation dans l'espace et le temps), considérés comme des sources de variabilités superficielles et non informatives. Cette remarque nous amène à considérer les techniques de recalage de signal que nous présentons dans la sous-partie suivante.

B .2 Recalage temporel de courbes et recalage spatial d'images

Les méthodes de recalage de courbes ou d'images, communément désignées sous le terme *curve registration / image warping methods* dans la littérature [Bro92, ZF03, Gos05], concernent l'ensemble des techniques permettant de faire correspondre deux observations d'un même phénomène ayant été acquises à des instants différents, à des lieux différents ou par des dispositifs différents... Dans ce contexte, deux types d'approches sont généralement distinguées : l'une, *feature-based registration*, qui est basée sur l'identification dans les différents signaux de caractéristiques attendues *a priori* [LMM95, Bod09] et l'autre, *intensity-based registration*, qui a pour objectif de faire correspondre les intensités (mesures dans le cas de courbes ou niveaux de gris dans le cas d'images) de plusieurs observations [RL98, LY09, AAT07, Cha11a]. Notre cas d'étude faisant intervenir des données dont nous avons une faible connaissance *a priori* , nous nous intéresserons dans cette partie aux méthodes du second type. Par ailleurs, la faible résolution des signatures infrarouge rendrait la recherche de caractéristiques délicate. Les techniques de recalage que nous étudions consistent donc à modéliser et à quantifier les transformations qu'il est nécessaire d'appliquer pour faire correspondre les mesures ou niveau de gris de deux observations. Ceci permet d'estimer l'écart entre une nouvelle donnée et une observation de référence ou encore à aligner les caractéristiques de plusieurs données.

Dans le cas général, une donnée $\mathcal{Y} \in \mathsf{Y}$ est une fonction de $\mathbb{U}$, ($\mathbb{U}$ est un ouvert de $\mathbb{R}$ dans le cas de courbes, un ouvert de $\mathbb{R}^2$ dans le cas d'images, etc...) et $\mathsf{Y} = \mathcal{L}^2(\mathbb{U})$ peut être *a priori* considéré comme l'espace des observations. Cependant, dans la plupart des cas, $\mathcal{Y}$ est observée sur une grille de discrétisation déterministe (ou *design*) $\Omega \subset \mathbb{U}^{|\Omega|}$ où $|\Omega|$ est la résolution de l'acquisition. Ω est définie comme le vecteur $\Omega = (u_1, \ldots, u_{|\Omega|})$ tel que pour tout $p \in \{1, \ldots, |\Omega|\}$, $u_p \in \mathbb{U}$. On désignera dans ce cas une observation $Y \in \mathbb{U}^{|\Omega|}$ par le vecteur

$$Y = (\mathcal{Y}(u_1), \ldots, \mathcal{Y}(u_{|\Omega|})) \ .$$

Soit G l'ensemble des applications de $\mathbb{U}$ dans lui même, paramétré par $\beta \in \mathsf{B}$ où B est un sous ensemble ouvert de $\mathbb{R}^d$ ($d > 0$) tel que :

$$\mathsf{G} = \{G_\beta : \mathbb{U} \to \mathbb{U},\ \beta \in \mathsf{B}\} \ .$$

Dans le cas où les observations sont discrétisées, on considère pour tout $G_\beta \in \mathsf{G}$, l'application $\mathrm{T}_{G_\beta} : \mathbb{U}^{|\Omega|} \to \mathbb{U}^{|\Omega|}$ telle que pour tout $Y \in \mathsf{Y}$:

$$\mathrm{T}_{G_\beta} : Y \to \left(\tilde{Y} \circ G_\beta(u_1), \ldots, \tilde{Y} \circ G_\beta(u_{|\Omega|})\right) \ , \tag{26}$$

où $\tilde{Y}$ est une interpolation de l'observation discrète Y sur $\mathbb{U}$, obtenue par exemple au moyen d'une interpolation bicubique.

Le recalage d'une observation $\mathcal{Y}_1 \in \mathsf{Y}$ (ou $Y_1 \in \mathsf{Y}$) sur $\mathcal{Y}_2 \in \mathsf{Y}$ (respectivement sur $Y_2 \in \mathsf{Y}$) consiste à approcher la transformation $G^\star \in \mathsf{G}$ définie par :

- lorsque $\mathsf{Y} = \mathcal{L}^2(\mathbb{U})$,

$$G^\star = \underset{G_\beta \in \mathsf{G}}{\arg\min} \int_{\mathbb{U}} |\mathcal{Y}_1 \circ G_\beta(u) - \mathcal{Y}_2(u)|^2 \mathrm{d}u + r(G_\beta) \;, \tag{27}$$

- lorsque $\mathsf{Y} = \mathbb{U}^{|\Omega|}$,

$$G^\star = \underset{G_\beta \in \mathsf{G}}{\arg\min} \|\mathrm{T}_{G_\beta}(Y_1) - Y_2\|^2 + r(G_\beta) \;, \tag{28}$$

où r est une métrique sur G telle que le terme $r(G_\beta)$ pénalise les « grandes » déformations. Nous présentons dans cette sous-section une liste (non-exhaustive) de modèles de déformations utilisés en statistique pour le recalage de courbes ou d'images ainsi que quelques méthodes permettant de les estimer.

Les transformations solides (translations, homothéties et éventuellement rotations dans le cas où $\mathbb{U} \subset \mathbb{R}^2$) agissent sur l'intégralité d'une observation et sont de faible dimension dans le sens où elles sont caractérisées par un petit nombre de paramètres [Goo91, GM96, Boo97, GLM07]. Dans le cas général, ces transformations prennent la forme d'applications affines et s'expriment quand $\mathbb{U} \subset \mathbb{R}$ par

$$\forall\, u \in \mathbb{U}\,, \quad G_\beta(u) = \lambda u + \tau \;,$$

où $\beta = (\lambda, \tau) \in \mathbb{R} \times \mathbb{R}$ et quand $\mathbb{U} \subset \mathbb{R}^2$ par

$$\forall\, u \in \mathbb{U}\,, \quad G_\beta(u) = \lambda R_\alpha u + \tau \;, \quad \text{où } R_\alpha = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \;,$$

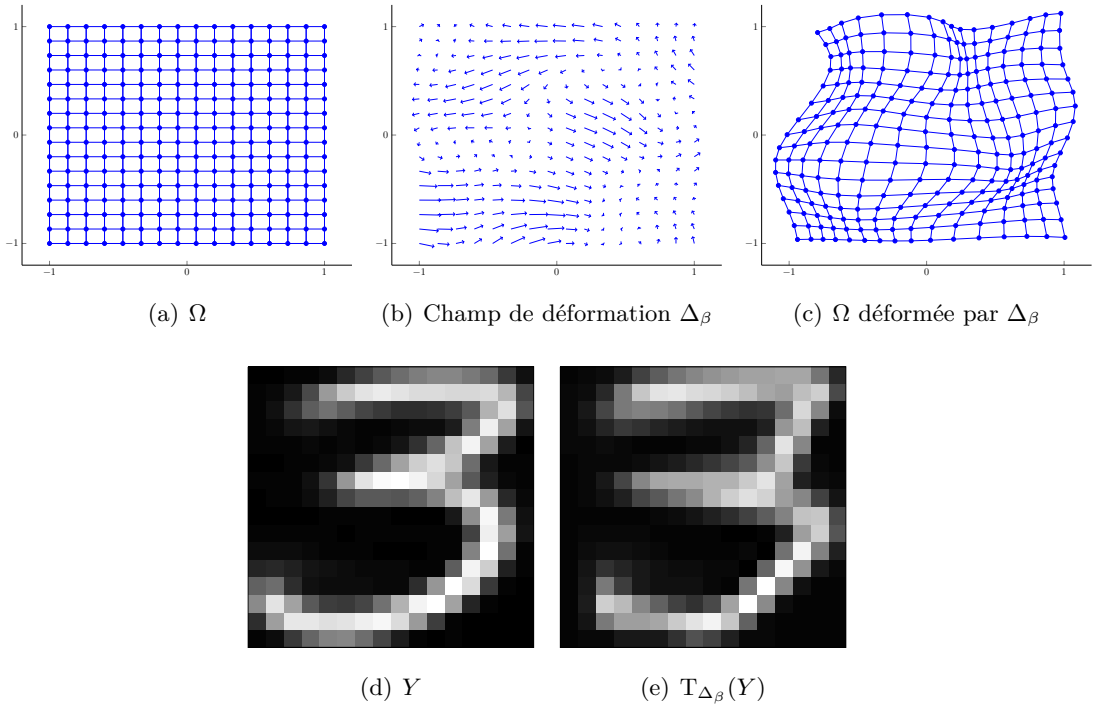
et $\beta = (\lambda, \alpha, \tau) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}^2$.

En complément de ces transformations, des déformations locales permettent « d'ajuster » ponctuellement les intensités de deux observations différentes [AAT07]. Contrairement aux transformations solides, les déformations locales peuvent avoir une dimension élevée (virtuellement infinie), suivant le degré de liberté du modèle de déformation utilisé. La figure 23 illustre l'action d'un champ de déformation local sur une image de chiffre manuscrit. Dans cette exemple, G_β s'écrit pour tout $u \in \mathbb{R}^2$

$$G_\beta(u) = u + \Delta_\beta(u)\,, \quad \text{avec} \quad \Delta_\beta(u) = \sum_{i=1}^{d} \beta_i \psi(r_k, u) \;, \tag{29}$$

où $u \to \Delta_\beta(u)$ est un champ de vecteur continu, contrôlé par un paramètre $\beta \in \mathsf{B} = \mathbb{R}^d$ avec $d = 50$. $\{r_i,\ 1 \leq i \leq d\} \subset \mathbb{U}$ est un ensemble de points de contrôle du champ de déformation et pour tout $r \in \mathbb{U}$, $\psi(r, \cdot)$ est un noyau Gaussien centré en r.

La principale limite à l'utilisation de ce type de modèles provient de la non inversibilité de G_β. Il est en effet souvent nécessaire de pouvoir passer d'une observation à une autre en composant à droite ou à gauche par G_β ou ${G_\beta}^{-1}$. Tant que les déformations restent « petites », le modèle de déformation (29) préserve globalement la topologie et une approximation satisfaisante de ${G_\beta}^{-1}$ est donnée par la transformation $u \to u - \Delta_\beta(u)$. Toutefois, ces approximations deviennent fausses dès lors que les déformations ont une grande amplitude. Comme en témoigne la figure 23(c), G_β ne semble pas bijective : la zone où le maillage se resserre laisse supposer l'existence de points avec plusieurs antécédents. Pour parer à cet inconvénient, les déformations non rigides peuvent être modélisées par des difféomorphismes [CRM96, JM00, TY05, BMTY05, Ash07, MMTY08]. L'idée est



(a) Ω (b) Champ de déformation Δ_β (c) Ω déformée par Δ_β

(d) Y (e) $\mathrm{T}_{\Delta_\beta}(Y)$

FIGURE 23 – Déformation locale par action d'un champ de vecteur

de considérer un processus $\{t \to \tilde{G}_\beta(t),\ t \in (0,1),\ \tilde{G}_\beta(t) \in \mathsf{G}\}$ dont les variations infinitésimales sont des champs de vecteurs du même type que Δ_β (29). Plus précisément, $\tilde{G}_\beta(t)$ est solution de l'équation différentielle :

$$\frac{\mathrm{d}\tilde{G}_\beta(t)}{\mathrm{d}t} = \Delta_\beta \circ \tilde{G}_\beta(t) \qquad \text{tel que} \qquad \tilde{G}_\beta(0) = \mathrm{Id}\ , \tag{30}$$

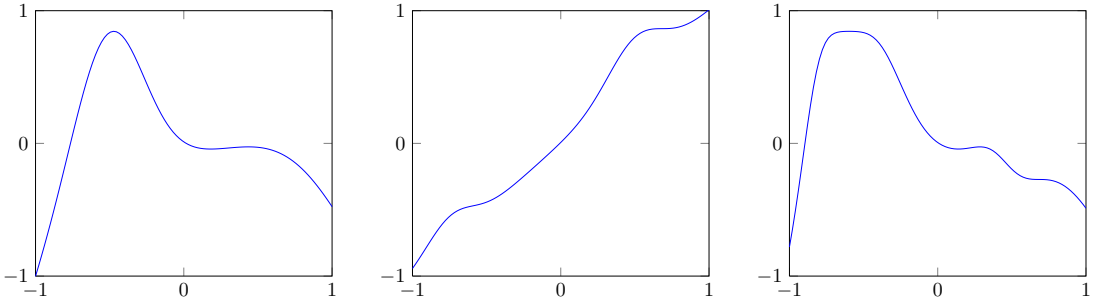
où Δ_β est le champ de vecteur défini dans (29) et est indépendant de t. Par ailleurs, de sorte à ce que pour tout $t \in (0,1)$ $\tilde{G}_\beta(t)(\mathbb{U}) \subseteq \mathbb{U}$, il est nécessaire d'imposer que Δ_β soit nul sur les bords de $\mathbb{U}$. Enfin, il suffit de poser $G_\beta = \tilde{G}_\beta(1)$ pour définir un difféomorphisme tel que ${G_\beta}^{-1}$ existe et ${G_\beta}^{-1} = G_{-\beta}$.

Remarque B.2. ***Stochastic Time***
Lorsque les observations sont des fonctions du temps, il est généralement souhaité que le modèle de déformations locales conserve la chronologie du motif [KG92, RL98]. Initié par [SC78], ces modèles de transformation du temps, dénommés dans la littérature par *Dynamic time warping* ou *Stochastic time*, imposent que G_β soit monotone croissante et continue. Supposant de plus que G_β est de classe $\mathcal{C}^2$ sur $(u_i, u_f) \subset \mathbb{U}$ et admettant pour conditions aux limites $G_\beta(u_i) = u_i$ et $G_\beta(u_f) = \tilde{u}_f$, un tel modèle de déformation du temps a été proposé dans [RL98]. Il s'agit des fonctions $G_\beta \in \mathsf{G}$ solutions de l'équation différentielle

$$\frac{\mathrm{d}^2 G_\beta}{\mathrm{d}u^2} = w_\beta \frac{\mathrm{d}G_\beta}{\mathrm{d}u}\ ,$$

où la fonction $u \to w_\beta(u)$ définit la courbure relative de G_β. Intégrant cette équation, il vient

$$G_\beta(u) = \tilde{u}_f \frac{\int_0^u \exp\left(\int_0^u w_\beta(v)\mathrm{d}v\right)\mathrm{d}u}{\int_0^{u_f} \exp\left(\int_0^u w_\beta(v)\mathrm{d}v\right)\mathrm{d}u}\ .$$



(a) $u \to \mathcal{Y}(u)$ (b) $u \to G_\beta(u)$ (c) $u \to \mathcal{Y} \circ G_\beta(u)$

Figure 24 – Déformation locale du temps

De façon analogue au modèle de déformations locales dans le cas des images, w_β est paramétrée par un vecteur $\beta \in \mathsf{B}$ où $\mathsf{B} \subset \mathbb{R}^d$ et qui peut se mettre sous la forme :

$$w_\beta(u) = \sum_{i=1}^{d} \psi(r_i, u)\beta_i \ ,$$

où $\{r_i,\ 1 \leq i \leq d\}$ est un ensemble de points de contrôle de la déformation du temps et pour tout $i \in \{1, \ldots, d\}$, les fonctions $K(r_i, \cdot)$ sont choisies comme étant une base de B-spline d'ordre 1 [Ram98]. La fonction $\{u \to G_\beta(u)\}$ est souvent appelée temps stochastique qui, en opposition au temps physique linéaire, s'écoule de façon non déterministe de sorte à pouvoir synchroniser les variations des courbes. Une illustration de ce modèle est proposée dans la figure 24, avec $u_0 = -1$, $u_f = 1$, $\tilde{u}_f = 1$ et $d = 10$. Un paramètre $\beta \in \mathsf{B}$ quelconque a spécifié le temps stochastique G_β présenté dans la figure 24(b).

Remarque B.3. Déformations dans l'espace de mesure
Dans certaines situations, en particulier dans le domaine du recalage de courbes [LY09, Cha11a], un autre type de transformation complétant les transformations de $\mathbb{U}$, est nécessaire : ce type de déformation agit sur l'espace de mesure, généralement $\mathbb{R}$, afin d'ajuster les intensités en amplitude *i.e.* « verticalement ». Une telle déformation S_υ, paramétrée par un vecteur $\upsilon = (\lambda, \mu) \in \mathbb{R}^2$, s'écrit dans le cas d'une observation discrète $Y \in \mathbb{R}^{|\Omega|}$,

$$S_\upsilon : Y \to \left(\lambda \tilde{Y}(u_1) + \mu, \ldots, \lambda \tilde{Y}(u_{|\Omega|}) + \mu\right) \ ,$$

où $u \to \tilde{Y}(u)$ est définie comme dans (26).

Tous ces modèles de recalage produisent des déformations qui sont contrôlées par un nombre fini de paramètres $\beta \in \mathsf{B}$. Même dans le cas des modèles de difféomorphismes, qui sont théoriquement des déformations de dimension infinie, la paramétrisation du champ de vecteur Δ_β (29) permet de se ramener à un problème en dimension finie. Ainsi, estimer la déformation $G^\star \in \mathsf{G}$ permettant de recaler les données revient à un problème d'optimisation sur B. Les équations (27) et (28) deviennent alors :

$$\begin{aligned} &G^\star = G_{\beta^\star} \ , \\ &\quad \text{où} \quad \beta^\star = \arg\min_{\beta \in \mathsf{B}} \begin{cases} \int_{\mathbb{U}} |\mathcal{Y}_1(u) - \mathcal{Y}_2 \circ G_\beta(u)|^2 \mathrm{d}u + r(G_\beta) \text{ si } \mathsf{Y} = \mathcal{L}^2(\mathbb{U}), \\ \|Y_1 - \mathrm{T}_{G_\beta}(Y_2)\|^2 + r(G_\beta) \text{ si } \mathsf{Y} = \mathbb{U}^{|\Omega|} \, . \end{cases} \end{aligned} \tag{31}$$

Ce problème peut-être résolu en utilisant des méthodes d'optimisation classiques pour estimer $\beta^\star$, comme par exemple l'algorithme de descente de gradient [JM00, RC03], la

méthode de Newton [KTNU00], l'algorithme de Levenberg-Marquardt [TRU98], des algorithmes génétiques [JR95, CAM04] ou encore des algorithmes de programmations dynamiques [SC78, WG97].

Toutefois, quand la dimension de B devient élevée, une autre approche consiste à considérer le paramètre β comme une variable aléatoire [AGP91, AAT07, MMTY08, LY09]. Dans cette optique, la loi de β conditionnellement à deux observations $(Y_1, Y_2) \in \mathsf{Y}^2$ s'écrit

$$\mathbb{P}(\,\beta\,|\,Y_1, Y_2\,) \propto \mathbb{P}(\,Y_2\,|\,\beta, Y_1\,)\,\mathbb{P}(\,\beta\,)\;, \tag{32}$$

où $\mathbb{P}(\,\beta\,)$ désigne la loi de β *a priori* . La loi *a posteriori* de Y_2 sachant β et Y_1, est généralement approchée par une distribution Gaussienne du type $\mathcal{N}(\mathrm{T}_{G_\beta}(Y_1), \sigma^2 Id_{|\Omega|})$ où $\sigma > 0$. Des méthodes de type Markov chain Monte Carlo (voir [RC04] et section D) telles que l'algorithme de Metropolis-Hastings [MRR$^+$53, Has70] permettent d'obtenir des échantillons $\beta_1, \ldots, \beta_n$ de la distribution $\mathbb{P}(\,\cdot\,|\,Y_1, Y_2\,)$. Le paramètre $\beta^\star$ peut par exemple être estimé comme étant le β_i le plus probable, ou bien à partir de la moyenne empirique des réalisations de β. Il est aussi possible d'utiliser un estimateur de maximum *a posteriori* (MAP) de β associé au modèle (32) pour définir $\beta^\star$ [Ash07].

B .3 Éléments sur les modèles à prototype déformable

La section B .1 a montré les limites des méthodes d'analyse de données classiques dès lors que les observations présentent de fortes variations. Considérant les modèles de recalage décrit dans la section B .2, une approche naturelle consiste à étudier la moyenne de Fréchet de cet ensemble d'observations [Fré48, Cha11a, Big11]. La notion de moyenne de Fréchet est une généralisation à des espaces métriques non euclidiens de la moyenne habituelle définie sur des espaces euclidiens, comme par exemple la moyenne sur $\mathbb{R}^{|\Omega|}$ des observations $\bar{Y}_N = N^{-1}\sum_{n=1}^N Y_k$ (Cf. Figure 16). Supposons qu'un modèle de déformation $G_\beta \in \mathsf{G}$ soit spécifié, l'equation (31) induit une mesure de dissimilarité $d_{\mathcal{M}}$ entre deux observations $\mathcal{Y}_1$ et $\mathcal{Y}_2$:

$$d^2_{\mathcal{M}}(\mathcal{Y}_1, \mathcal{Y}_2) = \inf_{\beta\in\mathsf{B}} \left\{ \int_{\mathbb{U}} |\mathcal{Y}_1(u) - \mathcal{Y}_2 \circ G_\beta(u)|^2 \,\mathrm{d}u + r(G_\beta) \right\} . \tag{33}$$

La moyenne de Fréchet $\mathcal{F}^{(N)}$ de N observations $\{\mathcal{Y}_1, \ldots, \mathcal{Y}_N\}$ sur l'espace $(\mathbb{R}^{|\Omega|}, d_{\mathcal{M}})$ est définie par :

$$\mathcal{F}^{(N)} = \operatorname*{arg\,min}_{\mathcal{F}\in\mathcal{L}^2(\mathbb{U})} \left\{ \frac{1}{N}\sum_{n=1}^N d^2_{\mathcal{M}}(\mathcal{F}, \mathcal{Y}_n) \right\} , \tag{34}$$

$$= \operatorname*{arg\,min}_{\mathcal{F}\in\mathcal{L}^2(\mathbb{U})} \left\{ \frac{1}{N}\sum_{n=1}^N \inf_{\beta_n\in\mathsf{B}} \left[\int_{\mathbb{U}} |\mathcal{F} - \mathcal{Y}_n \circ G_{\beta_n}(u)|^2 \,\mathrm{d}u + r(G_{\beta_n}) \right] \right\} . \tag{35}$$

Or, pour des paramètres $\beta_1, \ldots, \beta_N$ fixés, la fonction $\mathcal{F}$ minimisant (35) est trivialement $\mathcal{F}^{(N)} = N^{-1}\sum_{n=1}^N \mathcal{Y}_n \circ G_{\beta_n}$. Ainsi en pratique, le calcul de la moyenne de Fréchet d'un ensemble d'observation $\{\mathcal{Y}_n,\, n \leq N\}$ s'opère en deux étapes :

(i) recaler les données *i.e.* estimer les paramètres $\beta_1^\star, \ldots, \beta_N^\star$ tels que :

$$(\beta_1^\star, \ldots, \beta_N^\star) = \operatorname*{arg\,min}_{(\beta_1,\ldots,\beta_N)\in\mathsf{B}^N} \left\{ \frac{1}{N}\sum_{n=1}^N \int_{\mathbb{U}} \left| \mathcal{Y}_n \circ G_{\beta_n}(u) - \frac{1}{N}\sum_{\ell=1}^N \mathcal{Y}_\ell \circ G_{\beta_\ell}(u) \right|^2 \mathrm{d}u \right.$$
$$\left. + r(G_{\beta_n}) \right\} , \tag{36}$$

(ii) estimer la moyenne de Fréchet comme la moyenne des données recalées :

$$\mathcal{F}^{(N)} = \frac{1}{N}\sum_{n=1}^{N} \mathcal{Y}_n \circ G_{\beta_n^\star} \ . \tag{37}$$

En plus d'être une solution non paramétrique élégante au problème d'estimation d'une observation caractéristique, les travaux se basant sur l'étude de la moyenne de Fréchet bénéficient de propriétés théoriques appréciables (existence, unicité, convergence, etc...) [Zie77, BC11, Cha11b, BG13]. Toutefois, cette méthode s'avère difficilement implémentable en raison de la complexité de l'optimisation nécessaire à la réalisation de l'étape (i), surtout lorsque le paramètre β est de grande dimension.

Remarque B.4. Nous n'avons considéré dans le paragraphe précédent que le cas d'observations fonctionnelles *i.e.* $\mathsf{Y} = \mathcal{L}^2(\mathbb{U})$. Bien entendu, les méthodes basées sur l'étude des moyennes de Fréchet sont également applicables à des données discrètes ($\mathsf{Y} = \mathbb{U}^{|\Omega|}$).

Une autre approche, désignée sous le nom de modèle à prototype déformable, a récemment motivé un nombre important de contributions [AGP91, TY05, AAT07, AKT10a][M7]. Ces méthodes spécifient un modèle d'observation à travers un cadre statistique, permettant ainsi l'étude rigoureuse des caractéristiques de ces types de données. Les premiers modèles à prototype déformable ont été proposée par D'Arcy [d'A63], dans le but d'étudier les différences anatomiques chez certaines espèces d'animaux . Un cadre mathématique étendant les travaux de D'Arcy dans le domaine du traitement de l'image a été formalisé par Grenander [Gre93].

Le principe des modèles à prototype déformable est de considérer l'ensemble des observations $\{\mathcal{Y}_n,\ n \in \mathbb{N}\}$ comme une collection de variables aléatoires variant toutes autour d'une fonction déterministe $\mathcal{T} \in \mathcal{L}^2(\mathbb{U})$, *a priori* inconnue, caractéristique des observations et dénommée dans ce contexte *prototype* ou *template*. Deux sources de variabilités sont considérées :

- Des déformations aléatoires $G_\beta \in \mathsf{G}$ qui n'affectent pas le prototype $\mathcal{T}$ directement mais l'espace $\mathbb{U}$ sur lequel il est défini.
- Un processus de bruit additif, typiquement dû au processus de mesure $\mathcal{W} \in \mathsf{Y}$.

Ainsi, pour tout $n \in \mathbb{N}$, l'observation $\mathcal{Y}_n$ s'écrit :

$$\mathcal{Y}_n = \mathcal{T} \circ G_{\beta_n} + \mathcal{W}_n \ . \tag{38}$$

Le développement de ces modèles a pour objectif de caractériser statistiquement le prototype $\mathcal{T} \in \mathcal{L}^2(\mathbb{U})$ associé à une certaine population $\{\mathcal{Y}_n,\ n \in \mathbb{N}\}$ ainsi que la variabilité, donnée par les lois de β et de W, autour de celui-ci [DLJ04, GJ06, MMTY08, AK10].

Remarque B.5. La dénomination « *famille de modèles à prototype déformable* » justifie que cette approche, relativement abstraite, permet de modéliser des observations issues d'un nombre important de situations. En effet, il y a virtuellement autant de modèles à prototype déformable qu'il y a de modèles de deformation $G_\beta \in \mathsf{G}$ et de modèles de bruit additif $\mathcal{W} \in \mathsf{Y}$. Chaque type de données ayant ses propres caractéristiques, il est nécessaire de spécifier un modèle de déformation et de bruit adéquats. Ces choix définissent *de facto* un modèle à prototype déformable en particulier.

Remarque B.6. Bien que les modèles de déformation $G_\beta \in \mathsf{G}$ et de bruit $\mathcal{W} \in \mathsf{Y}$ soient fixés *a priori* , l'origine de l'aléa expliquant la diversité des observations $\{Y_n,\ n \in \mathbb{N}\}$ vient du fait que les paramètres $\{\beta_n,\ n \in \mathbb{N}\}$ et $\{\mathcal{W}_n,\ n \in \mathbb{N}\}$ (38) sont des réalisations *i.i.d.* des

variables aléatoires β et $\mathcal{W}$. Notons que dans certains cas, les lois de β et $\mathcal{W}$ ne peuvent être, *a priori* , que partiellement connues. Ainsi, à chaque observation $\mathcal{Y}_n$, il correspond de façon sous-jacente un paramètre de déformation β_n et un terme de bruit $\mathcal{W}_n$. A la différence des méthodes non paramétriques se basant sur les moyennes de Fréchet, l'étape intermédiaire intervenant dans l'estimation du prototype dans ces modèles déformables n'est pas de recaler individuellement toutes les observations (*i.e.* de connaître les paramètres des déformations $\{\beta_n^\star,\, n \leq N\}$ qui optimisent (36)) mais d'approcher la loi de probabilité de β et de $\mathcal{W}$.

Remarque B.7. L'apprentissage de la distribution des paramètres de déformations β, du modèle de bruit additif $\mathcal{W}$ et du prototype $\mathcal{T} \in \mathcal{L}^2(\mathbb{U})$, permet d'obtenir de nouvelles observations $\{\mathcal{Y}_n,\, n \in \mathbb{N}\}$ en simulant des réalisations *i.i.d.* du processus stochastique défini dans l'équation (38). Le modèle à prototype déformable est un donc modèle génératif.

Dans le cas d'observations discrétisées, Y_n s'écrit dans le modèle à prototype déformable comme le vecteur de pixels

$$Y_n = \left(Y_{n,1}, \ldots, Y_{n,|\Omega|}\right) \ ,$$

tel que pour tout $i \in \{1, \ldots, |\Omega|\}$,

$$Y_{n,i} = \mathcal{T} \circ G_{\beta_n}(u_i) + W_{n,i} \ ,$$

où $W_n = \left(W_{n,1}, \ldots, W_{n,|\Omega|}\right)$ est une réalisation de la variable aléatoire W définie sur $(\mathbb{R}^{|\Omega|}, \mathcal{B}(\mathbb{R}^{|\Omega|}))$.

Exemple B.8. Le modèle à prototype déformable proposé par Allassonnière et al.[AAT07, AK10, AKT10a]
Dans le contexte de la reconnaissance de formes, un modèle à prototype déformable dans le cas où les observations $\{Y_n,\ n \in \mathbb{N}\}$ sont des images de chiffres manuscrits a été proposé dans [AAT07]. L'image Y_n est observée sur une grille de pixels $(u_1, \ldots, u_{|\Omega|})$ et $\mathsf{Y} = \mathbb{R}^{|\Omega|}$. Pour tout $s \in \{1, \ldots, |\Omega|\}$, le pixel s de l'observation Y_n s'écrit :

$$Y_{n,s} = \mathcal{T}_\alpha(u_s - \Delta_{\beta_n}(u_s)) + \sigma W_{n,s} \ , \tag{39}$$

telle que :

- le prototype $\mathcal{T}_\alpha \in \mathcal{L}^2(\mathbb{U})$ s'écrit comme une combinaison linéaire de m noyaux Gaussiens centrés en m points de $\mathbb{U} = \mathbb{R}^2$, $(r_{p,1}, \ldots, r_{p,m}) \in \mathbb{U}^m$, contrôlant la photométrie. La fonction $\mathcal{T}_\alpha$ est paramétrée par un vecteur $\alpha = (\alpha_1, \ldots, \alpha_m) \in \mathbb{R}^m$ inconnu tel que :

$$\mathcal{T}_\alpha(u) = \sum_{v=1}^{m} \alpha_v \, \phi_v(u) \ , \tag{40}$$

  où pour tout $v \in \{1, \ldots, m\}$, $\phi_v : \mathbb{U} \to \mathbb{R}$ est un noyau Gaussien centré en $r_{p,v}$ et de variance σ_p^2. Les points de contrôles de la photométrie $\{r_{p,v},\, v \leq m\}$ et la variance σ_p^2 sont supposés connus.
- le modèle de déformation $G_\beta \in \mathsf{G}$ est un champ de déformation composé d'une combinaison linéaire de d déformations locales, centrées en d points de $\mathbb{U}$, $(r_{g,1}, \ldots, r_{g,d}) \in \mathbb{U}^d$, dont l'action sur la géométrie du prototype est contrôlée par un paramètre inconnu $\beta = [\beta^{(1)}\ \beta^{(2)}] \in \mathsf{B}$, $(\mathsf{B} \subseteq \mathbb{R}^d \times \mathbb{R}^d)$. Pour toute fonction $\mathcal{T} \in \mathcal{L}^2(\mathbb{U})$, la fonction $\mathcal{T} \circ G_\beta$ est définie pour tout $u \in \mathbb{U}$ par :

$$\begin{aligned} (\mathcal{T} \circ G_\beta)(u) \quad &= \mathcal{T}\left(u - \Delta_\beta(u)\right) \ , \\ &= \sum_{v=1}^{m} \alpha_v \phi_v \left(u - \sum_{q=1}^{d} [\beta_q^{(1)} \beta_q^{(2)}] \, \psi_q(u) \right) \ , \end{aligned} \tag{41}$$

où pour tout $q \in \{1, \ldots, d\}$, $\psi_q : \mathbb{U} \to \mathbb{R}$ est un noyau Gaussien centré en $r_{g,q}$ et de variance σ_g^2. De même que pour la photométrie, les points de contrôles de la géométrie $\{r_{g,q},\ q \leq d\}$ et la variance σ_g^2 sont supposés connus. A chaque observation Y_n est attaché un paramètre de déformation β_n, inconnu, modélisé comme étant une réalisation d'une variable aléatoire β de une loi normale de moyenne nulle et de matrice de covariance Γ inconnue.

- le processus de bruit utilisé dans cette étude est un bruit blanc Gaussien indépendant de variance σ^2 inconnue, tel que pour tout $n \in \mathbb{N}$, $s \in \{1, \ldots, |\Omega|\}$, $W_{n,s} \sim \mathcal{N}(0, 1)$.

Étant donnée une base d'apprentissage $\{Y_n,\ n \in \mathbb{N}\}$, le développement d'algorithmes (déterministes ou stochastiques) permettant d'estimer les paramètres α, Γ et σ constitue une problématique majeure (voir Section C , Exemple C.10, [AKT10a, AAT07]). Une version stochastique de l'algorithme Expectation Maximization a été proposée dans [AKT10a]. Appliqué à une base de d'images de chiffres manuscrits US Postal, cet algorithme a permis d'extraire avec succès des formes caractéristiques de chiffres.

Une adaptation de ce modèle dans le cas où les observations $\{Y_n,\ n \in \mathbb{N}\}$ dérivent de différentes formes prototypes a été proposée dans [AAT07, AK10] : l'objectif étant dans ce cas d'estimer une forme prototype par chiffre. On parle alors d'un *mélange de modèles à prototype deformable* : chaque observation $Y_n \in \mathsf{Y}$ est supposée appartenir à une classe $J_n \in \{1, \ldots, C\}$ qui, tout comme le paramètre de déformation $\beta_n \in \mathsf{B}$, est modélisée par une variable aléatoire J dont la réalisation n'est pas observée. La loi *a priori* de J est une loi de probabilité discrète $\{\omega_1, \ldots, \omega_C\}$, où le poids de chaque classe est inconnu.

Exemple B.9. Modèle de recalage de courbe [LY09]
Un modèle à prototype déformable a été proposé par [LY09] dans le cas où les observations sont des courbes *i.e.* $\mathbb{U} = \mathbb{R}$. A la différence du cas des images (Cf. exemple B.8), les instants de mesures peuvent varier d'une observation à l'autre, si bien que pour tout $n \in \mathbb{N}$ $u_n = (u_{n,1}, \ldots, u_{n,\ell_n}) \in \mathbb{U}^{\ell_n}$ sont les ℓ_n instants de mesures correspondant à l'observation $Y_n = (Y_{n,1}, \ldots, Y_{n,\ell_n})$. Le modèle des observations fait intervenir deux types de déformations aléatoires :

- une translation horizontale (*i.e.* variation du temps) β,
- une translation verticale (*i.e.* variation de l'intensité) υ.

Comme dans l'exemple B.8, la courbe Y_n appartient à une classe $J_n \in \{1, \ldots, C\}$, non observée. Ainsi, pour tout $(n, s) \in \mathbb{N} \times \{1, \ldots, \ell_n\}$, la s-ième mesure de la n-ième observation appartenant à la classe $J_n = j$ s'écrit :

$$Y_{n,s} = \mathcal{T}_{\alpha_j}(u_{n,s} + \beta_n) + \upsilon_n + \sigma W_{n,s} \ , \tag{42}$$

où β_n et υ_n sont respectivement les paramètres de translation en temps et en intensité associés à Y_n et $W_{n,s} \sim \mathcal{N}(0, 1)$. J_n, β_n et υ_n sont des réalisations des variables aléatoires J, β et υ de loi *a priori* respective $J \sim (\omega_1, \ldots, \omega_C)$, $\beta \sim \mathcal{N}(0, \sigma_\beta^2)$ et $\upsilon \sim \mathcal{N}(0, \sigma_\upsilon^2)$. Pour tout $j \in \{1, \ldots, C\}$, le prototype associé à la classe j est paramétré par le vecteur $\alpha_j = (\alpha_{j,1}, \ldots, \alpha_{j,m}) \in \mathbb{R}^m$ et la fonction $\mathcal{T}_{\alpha_j}$ est développée sur une base de B-spline cubic $\{B_v,\ v \leq m\}$. L'approximation au premier ordre $B_v(u_{n,s} + \beta_n) \approx B_v(u_{n,s}) + \beta_n B_v'(u_{n,s})$ permet d'exprimer (42) par :

$$W_{n,s} = \sum_{v=1}^{m} \alpha_{j,v} \left(B_v(u_{n,s}) + \beta_n B_v'(u_{n,s}) \right) + \upsilon_n + \sigma W_{n,s} \ .$$

L'algorithme SACK (*Simultaneously Aligning and Clustering K-clusters*) proposé dans [LMM95] permet d'estimer les paramètres du modèle $(\alpha_1, \ldots, \alpha_C, \sigma)$ ainsi que ceux des distributions *a priori* $(\sigma_\beta, \sigma_\upsilon, \omega_1, \ldots, \omega_C)$.

Résumé de la contribution

Nous avons présenté dans le Préambule A , différentes approches possibles permettant de détecter une cible dans une image multispectrale, sur laquelle nous avons *a priori* peu de connaissance. Le résumé de notre contribution sur ce point ainsi que le Chapitre I montrent que la méthodologie que nous proposons permet de détecter, avec une probabilité supérieure à 99% et un taux de fausse alarme inférieur à 1‰, un aéronef dans des situations réalistes à partir d'images multispectrales d'une scène optique. Nous nous intéressons à présent à la question de la reconnaissance d'une cible détectée. Pour classifier des données, il est utile de disposer de différents modèles de référence permettant de calculer, par exemple, la classe qui maximise la vraisemblance *a posteriori* d'une observation inconnue. Toutefois, dans notre étude, de tels modèles n'existent pas *a priori* et nous utiliserons en pratique les images pour lesquelles une cible a été détectée pour spécifier ces modèles. Dans ce travail, nous utilisons les SIR monospectrales et multispectrales simulées par CRIRA comme base d'observations.

Notre étude concerne des données présentant de fortes variations spatiales et spectrales et la section B .1 a montré que les méthodes d'analyse de données classiques ne permettent pas d'extraire des formes caractéristiques de ce type d'observations : une étape de recalage est donc nécessaire. Nous proposons un nouveau modèle d'observation adapté à l'étude des SIR multispectrales à K bandes. Dans cette optique, une observation est une fonction $\mathcal{Y} : \mathbb{U} \to \mathbb{R}^K$ avec $\mathbb{U} \subset \mathbb{R}^2$. Nous avons présenté dans la section B .2 un certain nombre de modèles de déformation. Dans le cas des SIR multispectrales, nous considérons les déformations $\{G_\beta,\, \beta \in \mathsf{B}\}$ agissant sur $\mathbb{U}$ prenant en compte :

- un champ de déformation locale $D_\delta \in \mathsf{G}$ paramétré par $\delta \in \mathsf{D}$ (D est un sous-ensemble de $\mathbb{R}^{d-5}$) et défini comme dans l'Exemple B.8 (41),
- une rotation $R_\rho \in \mathsf{G}$ paramétrée par $\rho \in [0, 2\pi[\, \times \mathbb{U}$,
- une translation $T_\tau \in \mathsf{G}$ paramétrée par $\tau \in \mathbb{R}^2$.

Ainsi, dans cette configuration, $\beta = (\delta, \rho, \tau)$ et $\mathsf{B} = \mathsf{D} \times [0, 2\pi[\, \times \mathbb{U} \times \mathbb{R}^2$. Les difféomorphismes (30) sont en effet des transformations trop sophistiquées pour des données si faiblement résolues et les effets de zoom ne sont d'aucune utilité dans notre étude car les cibles se trouvent toutes à la même distance du capteur. Nous prenons également en compte un paramètre d'échelle dans l'espace de mesure $\lambda > 0$ qui permet d'ajuster les intensités entre les différentes observations. Ce paramètre est particulièrement important car les SIR d'avions ont des niveaux de luminance qui varient sensiblement.

L'objectif étant de spécifier un modèle d'observation, nous considérons un mélange de modèles à prototype déformable, similaire à l'Exemple B.8, permettant de caractériser simultanément les géométries et photométries typiques de différentes cibles. Nous considérons un mélange de C classes dans lequel une image multispectrale à K bandes $\mathcal{Y} = (\mathcal{Y}^{(1)}, \ldots, \mathcal{Y}^{(K)})$ appartenant à la classe j s'écrit

$$\forall k \in \{1, \ldots, K\}, \quad \mathcal{Y}^{(k)}(u) = \lambda \mathcal{T}_j^{(k)} \circ [R_\rho(u) + D_\delta(u) + \tau] + \sigma_k \mathcal{W}(u)\ . \tag{43}$$

Ce modèle revient à effectuer les hypothèses suivantes :

(i) Pour chaque classe $j \in \{1, \ldots, C\}$, il existe des *prototypes spectraux* $\mathcal{T}_j^{(1)}, \ldots, \mathcal{T}_j^{(K)}$, *a priori* inconnus, paramétrés comme dans l'Exemple B.8 par le modèle (40).

(ii) Les différents templates spectraux sont altérés par une même déformation G_β et par un même paramètre d'ajustement de l'intensité $\lambda > 0$, indépendants de la bande spectrale k.

(iii) Toutes les classes ont le même paramètre de bruit additif qui dépend cependant de la bande spectrale k.

Les hypothèses (i) et (ii) sont justifiées car les variations entre les images des différentes sous-bandes sont avant tout d'ordre photométrique : la géométrie de la cible est en effet, à quelques exceptions près, commune à toutes les bandes spectrales. Le bruit additif modélise dans cette application (1) l'incertitude induite par le système d'acquisition des données et (2) la signature infrarouge du fond. Par conséquent, il est justifiable de faire varier le niveau de bruit suivant la bande spectrale (voir le modèle physique de la Partie Simulation des fonds, (11)) mais pas nécessaire de le faire dépendre de la classe d'une observation, d'où l'hypothèse (iii).

Le modèle à prototype déformable de l'Exemple B.8 convient pour des données ayant une géométrie fortement structurée, comme des chiffres manuscrits [AK10, AAT07], ou des données fortement résolues telles que des images médicales [AKT10b]. Les SIR d'avions sont trop faiblement résolues pour pouvoir espérer extraire des déformations caractéristiques. Dans notre approche, les prototypes sont des fonctions déterministes contenant une information géométrique et photométrique, tandis que les déformations et le paramètre d'échelle sont considérées comme des nuisances aléatoires non informatives : les déformations ont pour vocation d'offrir un certain nombre de degrés de liberté aux prototypes. Cette différence, en apparence superficielle, a des conséquences au niveau des lois *a priori* que nous proposons pour les paramètres de déformation. Alors que dans l'Exemple B.8, le paramètre β a une loi *a priori* faisant intervenir une matrice de covariance complète inconnue, nous supposons dans notre modèle que β a une matrice de covariance diagonale paramétrée par un unique scalaire, inconnu, qui dépend de la classe. Ce paramètre renseigne davantage sur la taille typique des déformations locales associées à une classe donnée que sur de réelles structures de déformations. Une des difficultés concernant l'implémentation du modèle de l'Exemple B.8 est la nécessité de spécifier des *hyperpriors*, *i.e.* des lois *a priori* sur les paramètres inconnus telles que la matrice de covariance de β. En effet, sans ces lois *a priori* , l'estimation des paramètres inconnus du modèle, et en particulier de la matrice de covariance complète, soulève des problèmes de régularité (voir Préambule C et Chapitre II), dont nous nous affranchissons avec cette approche.

La procédure d'estimation des paramètres du modèle est présentée dans un cadre général dans le Préambule C et dans le Chapitre II. Dans un second temps, elle est appliquée au cas spécifique des SIR monospectrales et multispectrales d'aéronefs dans le Chapitre III. Les résultats d'apprentissage (*e.g.* Figures III-4 et III-6) montrent que les paramètres sont correctement estimés comme en témoignent les observations que le modèle permet de générer ; voir Figure III.10(c). Une méthode de classification se basant sur le modèle le plus vraisemblable *a posteriori* est proposée. Deux types différents d'aéronefs sont testés et notre algorithme permet d'obtenir un taux de bonne classification supérieur à 95% dans le cas d'un ciel clair et supérieur à 90% dans le cas d'un ciel nuageux. Les résultats sont meilleurs avec les SIR multispectrales que monospectrales, ce qui confirme les résultats de la détection concernant l'intérêt des signatures multispectrales pour la détection et la classification d'aéronefs. Ces résultats s'expliquent pour deux raisons principales :

(i) Suivant les regroupements de bandes spectrales sélectionnés, le rapport signal à bruit peut être plus fort qu'en bande large.

(ii) La prise en compte de prototypes spectraux $(\mathcal{T}^{(1)}, \ldots, \mathcal{T}^{(K)})$ offre une description plus précise sur les variations d'intensités existant dans les sous-bandes tandis qu'en bande large, le seul paramètre scalaire d'ajustement est parfois brutal.

Ces deux raisons ont pour conséquence de faciliter l'apprentissage des paramètres du modèle dans le cas multispectral, d'où les meilleures performances de classification dans ce cas. Toutefois, l'intérêt du multispectral aurait été encore plus évident si les deux aéronefs avaient eu des profils spectraux distincts, non seulement en terme d'intensité, mais

également en terme de variation. Il aurait alors été possible de proposer un modèle d'observation tenant compte de déformations spatiales mais également spectrales, comme le suggère la section 3 .2 du chapitre III. Notre étude montre qu'il existe trois profils typiques d'intensité spectrale mais qu'ils ne sont pas discriminatoires car aussi représentatifs des deux aéronefs. Cette approche pourrait toutefois s'avérer intéressante lorsque la base d'apprentissage contient des aéronefs militaires ainsi que des avions de ligne ou d'autres leurres caractérisés par des profils spectraux différents.

S'inspirant du détecteur d'anomalies multispectrales proposé dans le chapitre I, il serait intéressant d'appliquer les algorithmes d'apprentissage et de classification, non pas directement aux SIR multispectrales, mais à leur transformée de Mahalanobis. L'estimation des prototypes serait facilitée car le rapport signal à bruit, qui joue un rôle prépondérant dans l'apprentissage, est meilleur dans les transformées de Mahalanobis que dans les images multispectrales (voir l'illustration de la Figure I-1). Les prototypes ne seraient dans ce cas plus les géométries/photométries typiques des aéronefs mais les cartographies caractéristiques des anomalies. Toutefois, dans un tel modèle et sous les mêmes hypothèses de fond, le processus de bruit additif ne suit plus une loi gaussienne mais une loi du chi-deux. A la différence du cas gaussien, la densité de cette loi a un support correspondant à $\mathbb{R}^+$, ce qui se traduit par la présence de la fonction indicatrice $\mathbb{1}_{\{Y>\lambda\Phi_X\alpha\}}$ dans l'expression de la densité jointe. Par conséquent, le modèle n'est plus exponentiel et la procédure d'apprentissage doit être étendue à un cas plus général pour que cette approche prometteuse puisse être développée.

De la même façon que pour la méthodologie de détection, la spécification d'un modèle de fond texturé en multispectral permettrait d'évaluer les performances de classification sur un ciel nuageux avec un modèle de fond plus réaliste que le modèle gaussien d'écart-type équivalent à $\sigma_{\mathrm{II}}^{(2)}$. Pour traiter ce problèmes deux alternatives sont possibles :

(i) spécifier un modèle de corrélation spectro-spatial du processus de bruit qui préserve le caractère exponentiel du modèle d'observation,

(ii) si un tel modèle de bruit n'est pas envisageable, adapter l'algorithme d'apprentissage au cas des modèles non exponentiels.

Enfin, contrairement à la méthodologie de détection, notre méthode de classification ne permet d'identifier les bandes spectrales optimales pour la classification au moyen d'un algorithme d'optimisation. En effet, les paramètres du modèle sont estimés à partir d'images dont les bandes spectrales ont été spécifiées et la classification est réalisée *a posteriori* . L'implémentation d'un algorithme qui optimise le taux de bonne classification serait prohibitif en terme de temps. Notre méthodologie permet néanmoins de comparer l'efficacité vis à vis de la classification de différents regroupements de bandes. Il resort de notre étude que le regroupement permettant la meilleure classification est celui qui est optimal pour la détection. Toutefois, ceci ne garantit pas l'optimalité de ce regroupement pour la classification. Une perspective intéressante consisterait à combiner des classifieurs basés sur des regroupements de bandes *prometteurs* par des méthodes de boosting [Sch03]. Ceci permettrait (i) d'obtenir un classifieur final plus performant et (ii) de connaître les poids respectifs de chaque regroupement prometteur spécifiant ainsi de nouvelles combinaisons de bandes à évaluer.

C Estimation de paramètres dans des modèles à données manquantes

C .1 Apprentissage

En statistique, l'apprentissage (*Machine Learning*) désigne l'ensemble des techniques permettant d'estimer, à partir d'observations, les paramètres inconnus intervenant dans un modèle. Dans de nombreuses situations, comme par exemple dans le cas des modèles à prototype déformable (Cf. Section B .3), les observations sont considérées comme des réalisations d'une variable aléatoire Y ayant une distribution *a priori* inconnue. Spécifier un modèle équivaut à proposer une distribution de probabilité susceptible d'avoir généré les données observées. Dans la plupart des cas, un modèle est une famille de distributions $\{\mathbb{P}_\theta,\ \theta \in \Theta\}$ où Θ est un ouvert de $\mathbb{R}^{\mathrm{d}_\Theta}$ ($\mathrm{d}_\Theta > 0$). θ est typiquement le vecteur contenant les paramètres inconnus du modèle et l'objectif des méthodes d'apprentissage statistique est d'estimer θ à partir des observations. Nous supposons que Y est définie sur l'espace mesurable $(\mathsf{Y}, \mathcal{Y})$ où Y est un ouvert de $\mathbb{R}^{\mathrm{d}_\mathsf{Y}}$ ($\mathrm{d}_\mathsf{Y} > 0$) et $\mathcal{Y}$ une tribu associée à Y. Enfin, nous nous plaçons dans le cas où la famille de distributions $\{\mathbb{P}_\theta,\ \theta \in \Theta\}$ admet pour densité la famille de fonctions $\{p_\theta,\ \theta \in \Theta\}$ par rapport à la mesure de Lebesgue sur $(\mathsf{Y}, \mathcal{Y})$, dont nous omettrons, en l'absence d'ambiguïté, la notation :

$$\mathbb{P}_\theta(Y \leq y) = \int_{-\infty}^{y} p_\theta(y')\mathrm{d}y' \,.$$

On définit de plus la fonction de vraisemblance $\mathrm{L} : \Theta \to \mathbb{R}$ telle que, pour un ensemble d'observation $\{y_1, \ldots, y_n\}$,

$$\mathrm{L}(\theta \,|\, y_1, \ldots, y_n) = \prod_{k=1}^{n} p_\theta(y_n)\,, \tag{44}$$

ainsi que la fonction de log-vraisemblance $\ell : \Theta \to \mathbb{R}$ associée :

$$\ell(\theta \,|\, y_1, \ldots, y_n) = \sum_{k=1}^{n} \log p_\theta(y_k)\ . \tag{45}$$

Toute méthode d'apprentissage fournit un estimateur $\hat{\theta} \in \Theta$ de θ tel que la distribution $\mathbb{P}_{\hat{\theta}}$ décrive au mieux les données $\{y_1, \ldots, y_n\}$. Lorsque p_θ est une fonction dont l'expression analytique est connue, des méthodes classiques telles que celles des moments ou des moindres carrés fournissent des estimateurs possibles de θ [Vap98]. Une autre possibilité est de calculer l'estimateur qui maximise la vraisemblance, appelé *l'estimateur du maximum de vraisemblance* θ^{MV}, défini par

$$\theta^{\mathrm{MV}} = \arg\max_{\theta \in \Theta} \mathrm{L}(\theta \,|\, y_1, \ldots, y_n)\ . \tag{46}$$

Le problème d'apprentissage revient dans cette approche à un problème d'optimisation.

Remarque C.1. La distribution $\mathbb{P}_\theta$ est un modèle pour les données $(y_1, \ldots, y_n)$. Dans certaines situations, ces dernières peuvent avoir une distribution « réelle » $\mathbb{P}_\star$ qui diffère du modèle et qui est trop complexe pour être décrite par une famille de distributions $\{\mathbb{P}_\theta,\ \theta \in \Theta\}$ telle que Θ soit un espace de dimension fini. Dans ce cas le modèle $\mathbb{P}_\theta$ est dit *mal spécifié*. A l'inverse, $\mathbb{P}_\theta$ est un modèle *bien spécifié* si il existe $\theta^\star \in \Theta$, correspondant au « vrai » paramètre, tel que $\mathbb{P}_{\theta^\star}$ et $\mathbb{P}_\star$ coïncident *i.e.* $(y_1, \ldots, y_n)$ sont des réalisations de $\mathbb{P}_{\theta^\star}$.

C .2 Apprentissage en bloc dans les modèles à données manquantes

Dans cette partie, nous faisons l'hypothèse que nous disposons d'un *bloc* fixe de n observations $y_1, \ldots, y_n$ disponible tout au long de la procédure d'apprentissage. Nous considérons les situations où la distribution de Y est définie comme la marginale de la distribution jointe du couple (X, Y) tel que X est une variable aléatoire définie sur $(\mathsf{X}, \mathcal{X})$ avec X un ouvert de $\mathbb{R}^{\mathrm{d}_\mathsf{X}}$ ($\mathrm{d}_\mathsf{X} > 0$) et $\mathcal{X}$ une tribu associée à X. X est appelé dans la littérature le vecteur des données manquantes ou cachées du modèle. Dans ce cadre, p_θ s'écrit pour tout $y \in \mathsf{Y}$:

$$p_\theta(y) = \int_\mathsf{X} f_\theta(x, y)\mathrm{d}x \,, \tag{47}$$

où $\{f_\theta,\, \theta \in \Theta\}$ est une famille de fonctions définies sur $\mathsf{X} \times \mathsf{Y}$, non-négatives et intégrables sur X. Dans le contexte des modèles à données manquantes, (44) est appelée vraisemblance incomplète et on appelle

$$\mathrm{L}\left(\theta \,|\, (y_1, x_1), \ldots, (y_n, x_n)\right) = \prod_{k=1}^{n} f_\theta(x_n, y_n) \,, \tag{48}$$

la vraisemblance des données complètes. Remarquons qu'avec ces notations

$$\mathrm{L}(\theta \,|\, y_1, \ldots, y_n) = \int \cdots \int \mathrm{L}\left(\theta \,|\, (y_1, x_1), \ldots, (y_n, x_n)\right) \mathrm{d}x_1 \ldots \mathrm{d}x_n \,.$$

Enfin, définissons par $\{x \to \pi_\theta(x \,|\, y)\,;\, \theta \in \Theta, y \in \mathsf{Y}\}$ la famille de densités de X conditionnellement à $Y = y$, définies et intégrables sur X. π_θ est généralement désignée sous le nom de densité *a posteriori* . Dans cette approche, la recherche du maximum de vraisemblance s'écrit :

$$\theta^{\mathrm{MV}} = \arg\max_{\theta \in \Theta} \; \ell(\theta \,|\, y_1, \ldots, y_n) = \arg\max_{\theta \in \Theta} \; \sum_{k=1}^{n} \log \int_\mathsf{X} f_\theta(x, y_k)\mathrm{d}x \,. \tag{49}$$

Notons que $y \to \int_\mathsf{X} f_\theta(x, y)\mathrm{d}x$ n'est parfois pas calculable : c'est le cas notamment des modèles à prototype déformable (exemples B.9 et C.10). Dans ces situations, les algorithmes d'optimisations classiques ne permettent pas de calculer l'estimateur du maximum de vraisemblance en maximisant directement (49). Des méthodes permettant de maximiser la vraisemblance de façon itérative s'avèrent particulièrement adaptées à ce contexte. Avant de poursuivre le développement, nous présentons un modèle statistique élémentaire faisant intervenir des données manquantes.

Exemple C.2. Mélange de gaussiennes - (1)
Dans ce modèle, Y suit une loi gaussienne dont la moyenne et la variance dépendent d'une variable discrète non observée $X \in \{1, \ldots, C\}$, indiquant la classe de l'observation. Considérons le cas $C = 2$, *i.e.* Y suit un modèle de mélange à deux composantes :

$$Y \sim \begin{cases} \mathcal{N}(\mu_1, \sigma_1^2) & \text{si} \quad X = 1, \\ \mathcal{N}(\mu_2, \sigma_2^2) & \text{si} \quad X = 2, \end{cases} \tag{50}$$

où $(\mu_1, \mu_2) \in \mathbb{R}^2$ et $(\sigma_1, \sigma_2) \in \mathbb{R}^{+2}$. On suppose que les classes $X = 1$ et $X = 2$ ont respectivement des poids *a priori* $\omega_1 > 0$ et $\omega_2 > 0$, tels que $\omega_1 + \omega_2 = 1$. La densité jointe f_θ s'écrit :

$$f_\theta(x, y) = \omega_1 \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{1}{2{\sigma_1}^2}(y-\mu_1)^2} \mathbb{1}_{\{x=1\}}(x) + \omega_2 \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{1}{2{\sigma_2}^2}(y-\mu_2)^2} \mathbb{1}_{\{x=2\}}(x) \,,$$

Les paramètres du modèle à estimer sont $\theta = (\mu_1, \mu_2, \sigma_1, \sigma_2, \omega_1, \omega_2)$. Notons que, dans le cas où les données manquantes du modèle sont des variables discrètes prenant leurs valeurs dans un ensemble de cardinal fini, p_θ est toujours calculable explicitement : la marginale (47) est dans ce cas une somme finie. Dans cet exemple :

$$p_\theta(y) = f_\theta(1, y) + f_\theta(2, y) \; .$$

Définition C.3. Dans cette section, pour toute fonction $G : (\mathsf{X}, \mathsf{Y}) \to \mathbb{R}$ intégrable sur $\mathsf{X} \times \mathsf{Y}$, nous notons par

$$\mathbb{E}_\theta[G(X, Y)] = \int_{\mathsf{X}} \int_{\mathsf{Y}} G(x, y) f_\theta(x, y) \mathrm{d}x \mathrm{d}y$$

l'espérance de la variable aléatoire $G(X, Y)$ sous la loi jointe des données complètes (X, Y) paramétrée par $\theta \in \Theta$ et respectivement par

$$\mathbb{E}_\theta[G(X, Y) \,|\, Y] = \int_{\mathsf{X}} G(x, Y) \pi_\theta(x \,|\, Y) \mathrm{d}x \quad \text{et} \quad \mathbb{E}_\theta[G(X, Y) \,|\, y] = \int_{\mathsf{X}} G(x, y) \pi_\theta(x \,|\, y) \mathrm{d}x$$

la variable aléatoire définie comme l'espérance conditionnelle sous la loi *a posteriori* des données manquantes paramétrée par $\theta \in \Theta$ et une réalisation de cette variable.

Algorithme du gradient, Méthode de Newton

Une première approche fait intervenir des algorithmes d'optimisation itératifs. Les identités de Louis et de Fisher [DLR77, discussion p.29] permettent d'exprimer, sous de faibles hypothèses, le gradient $\nabla_\theta p_\theta$ et la matrice Hessienne $\nabla^2_\theta p_\theta$ comme des espérances sous la loi *a posteriori* du gradient $\nabla_\theta \log f_\theta$ et de la matrice Hessienne $\nabla^2_\theta \log f_\theta$ de la densité complète :

$$\nabla_\theta \ell(\theta' \,|\, y_1, \ldots, y_n) = \sum_{k=1}^{n} \mathbb{E}_{\theta'} \left[\nabla_\theta \log f_{\theta'}(X_k, Y_k) \,\middle|\, y_k \right] \; ,$$

$$H(\theta') = \sum_{k=1}^{n} \mathbb{E}_{\theta'} \left[\nabla^2_\theta \log f_{\theta'}(X_k, Y_k) \,\middle|\, y_k \right] - \mathbb{E}_{\theta'} \left[\nabla^2_\theta \log \pi_{\theta'}(X_k \,|\, Y_k) \,\middle|\, y_k \right] \; .$$

Dans le cas où ces espérances sont calculables, des techniques d'optimisation usuelles telles que l'algorithme du gradient ou la méthode de Newton [Fle87, Lue03] sont des solutions implémentables au problème d'optimisation (49). Initialisées par un paramètre $\hat{\theta}_0 \in \Theta$ quelconque, ces deux méthodes construisent une séquence d'estimateurs $\{\hat{\theta}_i, \; i \in \mathbb{N}\}$ définie respectivement par :

$$\hat{\theta}_{i+1} = \hat{\theta}_i + \rho_i \, \nabla_\theta \, \ell(\hat{\theta}_i \,|\, y_1, \ldots, y_n) \; , \quad \hat{\theta}_0 \in \Theta \; , \tag{51}$$

$$\hat{\theta}_{i+1} = \hat{\theta}_i - \rho_i \, H^{-1}(\hat{\theta}_i) \, \nabla_\theta \, \ell(\hat{\theta}_i \,|\, y_1, \ldots, y_n) \; , \quad \hat{\theta}_0 \in \Theta \; , \tag{52}$$

où $\{\rho_i \geq 0, \, i \in \mathbb{N}\}$ est le coefficient multiplicatif contrôlant la descente de gradient. Ces méthodes nécessitent une charge importante de calculs (coefficient multiplicatif, calculs du gradient et de la matrice Hessienne de la vraisemblance,...) et sont de fait délicates à implémenter en pratique. Par ailleurs, dans certains modèles, en particulier lorsque Θ est un espace de grande dimension, la matrice Hessienne peut être mal conditionnée et dans certains cas non inversible. Des alternatives plus robustes telles que les méthodes du gradient conjugué ou de quasi-Newton existent mais sont encore plus lourdes à implémenter.

Algorithme EM

L'algorithme Expectation-Maximization (EM), introduit par [DLR77], fait figure de référence parmi les méthodes d'apprentissage dans les modèles à données manquantes. Tout comme les méthodes d'optimisation précédemment présentées, son objectif est de maximiser de façon itérative la fonction de log-vraisemblance ℓ (44) étant donné un ensemble d'observations $\{y_1, \ldots, y_n\}$. L'algorithme EM s'en démarque toutefois par sa simplicité d'implémentation, ce qui explique la mise au point de nombreuses méthodes d'apprentissage statistique qui lui sont liées [CD85, WT90, CLM95, DI96, QLdP99, DLM99].

A l'image des méthodes (51) et (52), une idée naturelle est de faire intervenir dans l'optimisation (49) la vraisemblance complète qui, à la différence de la vraisemblance incomplète, a une expression analytique connue. Toutefois, la vraisemblance complète faisant apparaître les données manquantes $x_1, \ldots, x_n$, inconnues, on considère son espérance conditionnellement aux observations $(y_1, \ldots, y_n)$ que l'on normalise par n :

$$\theta \to \frac{1}{n} \sum_{k=1}^{n} \mathbb{E}_\theta \left[\log f_\theta(X_k, Y_k) \,\middle|\, y_k \right] . \tag{53}$$

L'idée centrale de l'algorithme EM est de considérer l'espérance (53) conditionnée par un paramètre quelconque $\theta_0 \in \Theta$. Ce conditionnement spécifie une fonction $\mathcal{Q} : \Theta \times \Theta \to \mathbb{R}$ telle que :

$$\mathcal{Q}(\theta_0, \theta) = \frac{1}{n} \sum_{k=1}^{n} \mathbb{E}_{\theta_0} \left[\log f_\theta(X_k, Y_k) \,\middle|\, y_k \right] = \frac{1}{n} \sum_{k=1}^{n} \int_{\mathsf{X}} \log f_\theta(x_k, y_k) \pi_{\theta_0}(x_k | y_k) \mathrm{d}x_k . \tag{54}$$

La convergence de l'algorithme se base sur la Proposition suivante :

Proposition C.4. Pour tout $(\theta_0, \theta) \in \Theta^2$, si $\mathcal{Q}(\theta_0, \theta) \geq \mathcal{Q}(\theta_0, \theta_0)$ alors

$$\ell(\theta \,|\, y_1, \ldots, y_n) \geq \ell(\theta_0 \,|\, y_1, \ldots, y_n) . \tag{55}$$

Démonstration. Remarquant que $f_\theta(x, y) = \pi_\theta(x \,|\, y) p_\theta(y)$, il vient pour tout $(\theta_0, \theta) \in \Theta^2$

$$\begin{aligned} \mathcal{Q}(\theta_0, \theta) \quad &= n^{-1} \left\{ \sum_{k=1}^{n} \mathbb{E}_{\theta_0} \left[\log \pi_\theta(X_k \,|\, Y_k) \,\middle|\, y_k \right] + \sum_{k=1}^{n} \log p_\theta(y_k) \right\} , \\ &= n^{-1} \left\{ \sum_{k=1}^{n} \int_{\mathsf{X}} \log \pi_\theta(x_k \,|\, y_k) \pi_{\theta_0}(x_k \,|\, y_k) \, \mathrm{d}x_k + \ell(\theta \,|\, y_1, \ldots, y_n) \right\} . \end{aligned}$$

Avec la condition $\mathcal{Q}(\theta_0, \theta) \geq \mathcal{Q}(\theta_0, \theta_0)$, il vient

$$\sum_{k=1}^{n} \int_{\mathsf{X}} \log \frac{\pi_\theta(x_k \,|\, y_k)}{\pi_{\theta_0}(x_k \,|\, y_k)} \pi_{\theta_0}(x_k \,|\, y_k) \, \mathrm{d}x_k + \ell(\theta \,|\, y_1, \ldots, y_n) - \ell(\theta_0 \,|\, y_1, \ldots, y_n) \geq 0 .$$

L'inégalité de Jensen pour les fonctions concaves permet de montrer que

$$\sum_{k=1}^{n} \int_{\mathsf{X}} \log \frac{\pi_\theta(x_k \,|\, y_k)}{\pi_{\theta_0}(x_k \,|\, y_k)} \pi_{\theta_0}(x_k \,|\, y_k) \, \mathrm{d}x_k \leq 0 ,$$

ce qui conclut la preuve. □

EM est un algorithme itératif qui, partant d'un paramètre initial quelconque $\hat{\theta}_0 \in \Theta$, propose à chaque itération $i \in \mathbb{N}$ un nouvel estimateur $\hat{\theta}_{i+1} \in \Theta$ tel que $\mathcal{Q}(\theta_i, \theta_{i+1}) \geq \mathcal{Q}(\theta_i, \theta_i)$. Ainsi, en vertu de la Proposition C.4, la suite $\{\ell(\hat{\theta}_i \,|\, y_1, \ldots, y_n),\, i \in \mathbb{N}\}$ est monotone croissante. Pour $i \in \mathbb{N}^*$, on définit la fonction $Q_i : \Theta \to \mathbb{R}$, $Q_i(\theta) = \mathcal{Q}(\hat{\theta}_{i-1}, \theta)$. Étant donné $\hat{\theta}_i \in \Theta$, l'estimateur suivant $\hat{\theta}_{i+1}$ est obtenu à la i+1-ème itération de l'algorithme EM , décomposée classiquement en deux étapes :

(i) étape E : Calcul d'espérance

$$\theta \to Q_{i+1}(\theta) = \frac{1}{n} \sum_{k=1}^{n} \mathbb{E}_{\hat{\theta}_i} \left[\log f_\theta(X_k, Y_k) \;\middle|\; y_k \right] , \tag{56}$$

(ii) étape M : Étape de Maximisation

$$\hat{\theta}_{i+1} = \arg\max_{\theta \in \Theta} Q_{i+1}(\theta) \, . \tag{57}$$

Dans les cas où la séquence d'estimateurs converge *i.e.* il existe $i \in \mathbb{N}$, $\hat{\theta}_i = \hat{\theta}_{i+1} = \theta^\star$, alors $\theta^\star$ est un point stationnaire de la vraisemblance : $\nabla \ell(\theta \,|\, y_1, \ldots, y_n)|_{\theta=\theta^\star} = 0$. Des hypothèses supplémentaires permettent de garantir la convergence de $\{\hat{\theta}_i,\ i \in \mathbb{N}\}$ à partir de n'importe quel état initial $\hat{\theta}_0 \in \Theta$ [CMR05, Théorème 10.5.3]. Toutefois, il n'y a pas de garantie sur la nature du point stationnaire : il peut être un maximum local de ℓ [Wu83], auquel cas $\theta^\star \neq \theta^{\mathrm{MV}}$ (46).

Remarque C.5. La Proposition C.4 montre que maximiser (56) n'est pas nécessaire pour augmenter la vraisemblance : il suffit de trouver un paramètre θ' tel que $Q_i(\theta') \geq Q_i(\hat{\theta}_i)$. L'algorithme EM est donc un cas particulier d'un ensemble d'algorithmes plus général, parfois appelé GEM (Generalized EM), pour lesquels l'étape de maximisation est remplacée par une étape d'*accroissement* de la fonction Q_i.

Remarque C.6. Le cas des familles exponentielles
L'algorithme EM est particulièrement adapté au cas où la vraisemblance complète définit une famille exponentielle de fonctions positives sur $\mathsf{X} \times \mathsf{Y}$ *i.e.* le logarithme de la densité jointe f_θ peut se mettre sous la forme :

$$\log f_\theta(x, y) = t(\theta) + \langle S(x, y), r(\theta) \rangle \ , \tag{58}$$

où $S : \mathsf{X} \times \mathsf{Y} \to \mathsf{S}$ est le vecteur des statistiques exhaustives du modèle (*complete data sufficient statistics vector*), r et s sont des fonctions telles que $r : \Theta \to \mathsf{S}$ et $t : \Theta \to \mathbb{R}$. S est dans la plupart des modèles un espace euclidien ou éventuellement un produit d'espaces euclidiens de dimension finie. Définissons les fonctions $\bar{s} : \Theta \to \mathsf{S}$ et $\bar{\theta} : \mathsf{S} \to \Theta$:

$$\bar{s}(\theta) = \frac{1}{n} \sum_{k=1}^{n} \mathbb{E}_\theta \left[S(X_k, Y_k) \mid y_k \right] \ , \qquad \bar{\theta}(s) = \arg\max_{\theta \in \Theta} t(\theta) + \langle s, r(\theta) \rangle \ . \tag{59}$$

Sous ces hypothèses, pour $\hat{\theta}_0 \in \Theta$,

- l'étape E de la i-ème itération de l'algorithme se résume au calcul de l'espérance sous la loi *a posteriori* du vecteur des statistiques exhaustives

$$\hat{s}_{i+1} = \bar{s}(\hat{\theta}_i) \ , \tag{60}$$

qui est indépendante de θ.

- l'étape M de la i-ème itération de l'algorithme est la maximisation de la fonction $\bar{\theta}$:

$$\hat{\theta}_{i+1} = \bar{\theta}(\hat{s}_{i+1}) \; . \tag{61}$$

  Dans la plupart des modèles exponentiels, cette étape d'optimisation peut être réalisée de façon exacte.

Notons que dans le cas des modèles exponentiels, la convergence de l'algorithme EM peut être repérée de façon équivalente par l'existence de $s^\star \in \mathsf{S}$ ou $\theta^\star \in \Theta$ telle que

$$s^\star = \bar{s} \circ \bar{\theta}(s^\star) \quad \text{et} \quad \theta^\star = \bar{\theta} \circ \bar{s}(\theta^\star) \; . \tag{62}$$

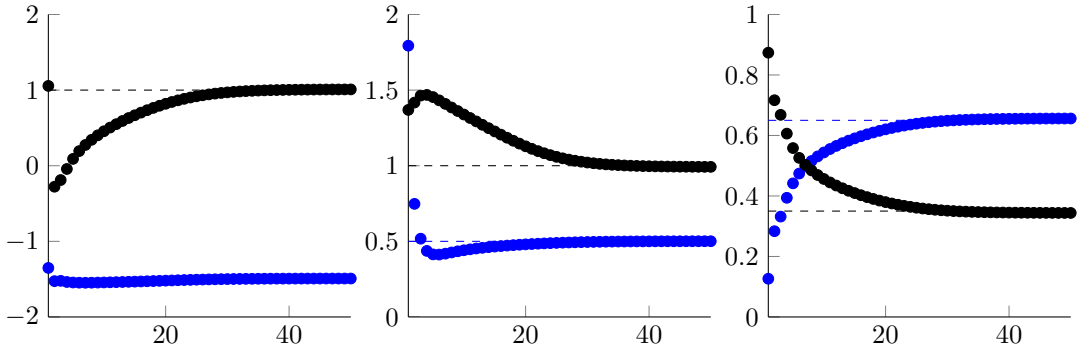
Exemple C.7. Mélange de gaussiennes - (2)
Le modèle de l'exemple C.2 est exponentiel et l'étape E de l'algorithme se résume au calcul des poids *a posteriori* des classes $X = 1$ et $X = 2$ pour les n observations $\{y_1, \ldots, y_n\}$. Pour $j \in \{1, 2\}$, $k \in \{1, \ldots, n\}$ et une valeur de l'estimateur $\hat{\theta} \in \Theta$, ces poids s'écrivent :

$$\hat{\pi}_{\hat{\theta},j,k} = \mathbb{P}_{\hat{\theta}}[X = j \,|Y = y_k] \propto f_{\hat{\theta}}(j, y_k) = \frac{\hat{\omega}_k}{2\pi\hat{\sigma}_k^2} \exp\{-\frac{1}{2\hat{\sigma}_k^2}\|y_k - \hat{\mu}_{j,k}\|^2\} \; . \tag{63}$$

Les statistiques exhaustives de ce modèle sont les fonctions $x \to \mathbb{1}_{\{x=1\}}(x)$ et $x \to \mathbb{1}_{\{x=2\}}(x)$. La mise à jour des estimateurs lors de l'étape M de la i-ième itération est réalisée de façon exacte :

$$\hat{\mu}_{j,i+1} = \frac{\sum_{k=1}^n y_k \hat{\pi}_{\hat{\theta}_i,j,k}}{\sum_{k=1}^n \hat{\pi}_{\hat{\theta}_i,j,k}}, \; \hat{\sigma}^2_{j,i+1} = \frac{\sum_{k=1}^n (y_k - \hat{\mu}_{j,i+1})^2 \hat{\pi}_{\hat{\theta}_i,j,k}}{\sum_{k=1}^n \hat{\pi}_{\hat{\theta}_i,j,k}}, \; \hat{\omega}_{j,i+1} = \frac{\sum_{k=1}^n \hat{\pi}_{\hat{\theta}_i,j,k}}{\sum_{k=1}^n \hat{\pi}_{\hat{\theta}_i,1,k} + \hat{\pi}_{\hat{\theta}_i,1,k}}.$$

$n = 30000$ réalisations d'un modèle de mélange Gaussien sont simulées avec les paramètres



estimation des moyennes $\{\hat{\mu}_{1,i}, \hat{\mu}_{2,i},\ 1 \le i \le 50\}$ | estimation des écarts types $\{\hat{\sigma}_{1,i}, \hat{\sigma}_{2,i},\ 1 \le i \le 50\}$ | estimation des poids $\{\hat{\omega}_{1,i}, \hat{\omega}_{2,i},\ 1 \le i \le 50\}$

FIGURE 25 – Estimation des paramètres du modèle en utilisant l'algorithme EM

$\mu_1 = -1.5$, $\mu_2 = 1$, $\sigma_1 = 0.5$, $\sigma_2 = 1$, $\omega_1 = 0.65$ et $\omega_2 = 0.25$. L'algorithme EM est ensuite appliqué, partant de paramètres initiaux aléatoires, et fournit une séquence d'estimateurs $\{\hat{\theta}_i,\ i \in \mathbb{N}\}$ où $\hat{\theta}_i = (\hat{\mu}_{1,i}, \hat{\mu}_{2,i}, \hat{\sigma}_{1,i}, \hat{\sigma}_{2,i}, \hat{\omega}_{1,i}, \hat{\omega}_{2,i})$. 50 itérations de l'algorithme sont effectuées et la convergence de $\hat{\theta}$ (pointillés) vers $\theta^\star$ (tirets) est illustrée par la figure 25 ; les classes 1 et 2 étant respectivement représentées en bleu et en noir.

Des variantes de l'algorithme EM existent lorsque l'une des deux étapes E ou M n'est pas réalisable directement.

Remarque C.8. Approximation de l'étape E
Nous nous intéressons plus particulièrement au cas où l'espérance conditionnelle n'est pas calculable explicitement, situation que l'on retrouve dans de nombreux modèles, en particulier les modèles à prototype déformable (Exemple B.8). Nous considérons le cas des modèles exponentiels afin d'alléger l'écriture de ces algorithmes. Notons que la plupart des résultat théoriques de convergence portant sur les algorithmes dérivés de l'EM implémentables lorsque l'étape E n'est pas calculable, ont été prouvés pour des modèles exponentiels [FM03, DLM99, KL04].

Une première alternative, proposée par l'algorithme Monte Carlo EM (MCEM) [WT90, FM03], consiste à remplacer le calcul de l'espérance conditionnelle $\bar{s}(\hat{\theta}_i)$ intervenant dans l'étape E par une approximation numérique de Monte-Carlo (voir Section D .2). L'étape E de la i-ème itération du MCEM s'écrit alors :

$$\tilde{s}_{m_i}(\hat{\theta}_i) = \frac{1}{nm_i} \sum_{k=1}^{n} \sum_{j=1}^{m_i} S(x_{i,k}^{(j)}, y_k) \;, \tag{64}$$

où pour tout $k \in \{1, \ldots, n\}$, $\{x_{i,k}^{(j)},\, j \leq m_i\}$ sont des tirages *i.i.d.* de $\pi_{\hat{\theta}_i}(\,\cdot \,|\, y_k)$. Toutefois, dans de nombreux modèles, comme dans le cas du modèle à prototype déformable de l'exemple B.8, simuler des échantillons *i.i.d.* de la loi *a posteriori* n'est pas possible. Dans ce cas, une alternative consiste à simuler pour tout $k \in \{1, \ldots, n\}$ des échantillons $\{\tilde{x}_{i,k}^{(j)},\, j \leq m_i\}$ d'une chaîne de Markov $\pi_{\hat{\theta}_i}(\,\cdot \,|\, y_k)$-réversible (Cf. Section D .1 et [RC04] pour une introduction sur les algorithmes MCMC).

Ce type d'approximation de l'étape E soulèvent la question du nombre $\{m_i,\ i \in \mathbb{N}\}$ de simulations des données manquantes intervenant dans (64). En effet, cette étape de simulation induit une charge de calcul importante, en particulier dans le cas où un noyau de transition d'une chaîne de Markov est utilisé pour échantillonner $\pi_{\hat{\theta}_i}(\,\cdot \,|\, y_k)$. Il est généralement conseillé d'augmenter m_i à mesure que l'estimateur se rapproche de la convergence [CMR05, Section 11.1.2]. Toutefois, le compromis entre la précision de l'estimateur $\tilde{s}_{m_i}(\hat{\theta}_i)$ de $\bar{s}(\hat{\theta}_i)$ et le coût de calcul reste une question ouverte et différentes méthodes permettant d'automatiser le choix de m_i en fonction du nombre d'itération i ont été proposées [BH99, FM03, LF04].

La procédure de Robbins Monroe [RM51] permet de trouver les zéros de toute fonction $h : \mathsf{X} \to \mathsf{X}$, à partir d'observations bruitées $\tilde{h}(x)$ de $h(x)$, pour $x \in \mathsf{X}$. Cette méthode construit récursivement une séquence d'estimateurs $\{\hat{x}_i \in \mathsf{X},\, i \in \mathbb{N}\}$ telle que

$$\hat{x}_{i+1} = \hat{x}_i + \rho_i \tilde{h}(\hat{x}_i) \;, \quad \hat{x}_0 \in \mathsf{X} \;,$$

où $\{\rho_i,\ i \in \mathbb{N}\}$ est une suite décroissante de nombres positifs. Sous certaines hypothèses sur $\tilde{h}$ et sur $\{\rho_i,\ i \in \mathbb{N}\}$, la séquence $\{\hat{x}_i,\, i \in \mathbb{N}\}$ converge vers les zéros de h [DLM99, AMP05].

L'algorithme Stochastic Approximation EM (SAEM) [DLM99] utilise une procédure de Robbins Monro pour trouver les points fixes de la fonction $\bar{s} \circ \bar{\theta} : \mathsf{S} \to \mathsf{S}$,

$$\bar{s} \circ \bar{\theta}(s) = \frac{1}{n} \sum_{k=1}^{n} \mathbb{E}_{\bar{\theta}(s)} \left[S(X_k, Y_k) \mid y_k \right] \;,$$

à partir d'observations de $\tilde{s}_m \circ \bar{\theta} : \mathsf{S} \to \mathsf{S}$ telles que

$$\tilde{s}_m \circ \bar{\theta}(s) = \frac{1}{nm} \sum_{k=1}^{n} \sum_{j=1}^{m} S(x_k^{(j)}, y_k), \qquad x_k^{(j)} \overset{i.i.d.}{\sim} \pi_{\bar{\theta}(s)}(\,\cdot \,|\, y_k) \;.$$

L'algorithme SAEM construit une séquence d'estimateurs $\{\hat{s}_i,\ i \in \mathbb{N}\}$ qui converge vers $s^\star$, par la récursion

$$\hat{s}_{i+1} = \hat{s}_i + \rho_i(\tilde{s}_{m_i} \circ \bar{\theta}(\hat{s}_i) - \hat{s}_i), \quad \hat{s}_0 \in \mathsf{S}\ ,$$

Le calcul de de la fonction $\tilde{s}_{m_i} \circ \bar{\theta}(\hat{s}_i)$ faisant intervenir une étape de maximisation, le SAEM conserve la structure des deux étapes de l'algorithme EM :

$$\left\{ \begin{array}{l} \hat{s}_{i+1} = \hat{s}_i + \rho_i(\tilde{s}_{m_i}(\hat{\theta}_i) - \hat{s}_i)\ , \\ \hat{\theta}_{i+1} = \bar{\theta}(\hat{s}_{i+1})\ . \end{array} \right. \tag{65}$$

Contrairement au MCEM , l'espérance conditionnelle n'est pas approchée de façon indépendante à chaque itération mais de façon continue : chaque nouvelle itération du SAEM améliore l'approximation. De cette façon, à chaque itération, toutes les données manquantes simulées au cours des itérations précédentes sont prises en compte avec un oubli progressif contrôlé par le facteur $\{\rho_i,\ i \in \mathbb{N}\}$. En conséquence, comparé au MCEM , le SAEM permet non seulement de réduire considérablement la charge de calcul induite à l'étape E mais également de s'affranchir du problème du nombre de simulations $\{m_i,\ i \in \mathbb{N}\}$, qui de fait peut être considéré comme constant, généralement pris égal à 1. Une adaptation du SAEM dans les cas où il n'est pas possible de simuler des échantillons *i.i.d.* de la loi *a posteriori* a été proposée dans [KL04] : l'algorithme SAEM-MCMC utilise des réalisations d'une chaîne de Markov admettant $\pi_\theta(\,\cdot\,|\,y)$ comme distribution stationnaire pour calculer la fonction $\bar{s}_m$; se référer à la Section D .3 pour une implémentation dans le cas des modèles à prototype déformable.

Remarque C.9. Approximation de l'étape M
Les versions précédentes de l'algorithme EM sont particulièrement utiles lorsque l'étape de maximisation M n'est pas coûteuse en calcul. Dans le cas contraire, des méthodes alternatives telles que l'algorithme ECM (Expectation Conditional Maximization), conservant la structure initiale de l'EM , existent [MR93]. Dans cette approche, lorsque Θ est de grande dimension, la maximisation globale de $\mathcal{Q}$ (54) est remplacée pas la maximisation successive de tous les paramètres de façon individuelle en tenant compte de ceux qui ont déjà été maximisés (à la manière d'un échantillonneur de Gibbs dans le contexte des algorithmes ; Section MCMC D .2).

Enfin, l'étape M peut également être approchée par une étape de l'algorithme de Newton (52). Contrairement à l'algorithme de Newton, l'algorithme résultant, dénommé EM Gradient [Lan95], est adapté au cas des modèles à données manquantes et remplace la maximisation de $\theta \to \mathcal{Q}(\hat{\theta}_i, \theta)$ par :

$$\hat{\theta}_{i+1} = \hat{\theta}_i + \rho_i\, J^{-1}(\hat{\theta}_i) \nabla_\theta \sum_{k=1}^{n} \log p_{\hat{\theta}_i}(y_k)\ , \tag{66}$$

tel que

$$J(\theta') = -\sum_{k=1}^{n} \mathbb{E}_{\theta'} \left[\nabla_\theta^2 \log f_{\theta'}(X_k, Y_k) \,\middle|\, y_k \right]\ . \tag{67}$$

Bien qu'en apparence différent de l'EM [DLR77], cet algorithme en partage certaines propriétés. En particulier, l'EM Gradient converge sous certaines conditions vers les maximum locaux de la fonction de vraisemblance et, pour un nombre d'itérations suffisamment important, la Proposition C.4 est vérifiée *i.e.* chaque itération améliore la vraisemblance [Lan95].

Exemple C.10. Apprentissage dans les modèles à template déformable
Le mélange de modèles à prototype déformable (Exemple B.8) est également un exemple de modèle à données manquantes. Pour tout $k \in \{1,\dots,n\}$, les données manquantes associées à l'observation Y_k sont la variable de classe J_k et le paramètre de déformation β_k. Soit $X_k = (J_k, \beta_k)$ le vecteur des données manquantes. Le modèle (39) est équivalent au modèle hiérarchique suivant :

(i) $J_k \sim (\omega_1,\dots,\omega_C)$,
(ii) $\beta_k \,|\, J_k = j \sim g_\theta(\cdot \,|\, j)$,
(iii) $Y_k \,|\, \beta_k, J_k = j \sim p_\theta(\cdot \,|\, j, \beta_k)$,

tel que

$$g_\theta(\cdot \,|\, j) = \mathcal{N}(0, \Gamma_j) \quad \text{et} \quad p_\theta(\cdot \,|\, j, \beta_k) = \mathcal{N}(\Phi_{\beta_k}\alpha_j, \sigma_j^2 \mathrm{Id}_{|\Omega|}) \,. \tag{68}$$

En effet, le modèle (39) peut être réécrit (en utilisant (41)) sous une forme vectorielle :

$$Y_k = \Phi_{\beta_k}\alpha_j + \sigma_j^2 W_k \,, \quad \text{où} \quad W_k \sim \mathcal{N}_{|\Omega|}(0, \mathrm{Id}_{|\Omega|}) \,,$$

et Φ_{β_k} est la matrice de taille $|\Omega| \times m$ dont les coefficients sont données par

$$\forall\,(s,v) \in \{1,\dots,|\Omega|\} \times \{1,\dots,m\}, \quad [\Phi_{\beta_n}]_{s,v} = \phi_v\left(u_s - \sum_{q=1}^{d} [\,\beta_{k,q}^{(1)}\,\beta_{k,q}^{(2)}\,]\,\psi_q(u_s)\right) \,.$$

Le logarithme de la densité des données complètes de ce modèle s'écrit :

$$\log f_\theta(j,\beta,y) \propto -|\Omega|\log(\sigma_j^2) - \frac{1}{\sigma_j^2}\|y - \Phi_\beta\alpha_j\|^2 - \log\det\Gamma_j - \beta^T\Gamma_j^{-1}\beta + \log(\omega_j) \,. \tag{69}$$

On peut montrer (Cf. *e.g.* [AKT10a]) que la densité des données complètes (69) définit une famille exponentielle de fonctions (58). Le modèle est exponentiel : l'algorithme EM est donc une méthode appropriée pour estimer le vecteur de paramètres

$$\theta = \left\{\alpha_j, \Gamma_j, \omega_j, \sigma_j^2 \,|\, j \in \{1,\dots,C\}\right\} \,. \tag{70}$$

Le vecteur des statistiques exhaustives intervenant dans (58) s'écrit pour tout $(X,Y) \in (\mathsf{X} \times \mathsf{Y})$:

$$S(X,Y) = \{S_1(X,Y),\dots,S_C(X,Y)\} \,, \quad S_j : \mathsf{X} \times \mathsf{Y} \to \tilde{\mathsf{S}} \,,$$

où $\tilde{\mathsf{S}}$ est un sous ensemble d'un espace euclidien spécifié par le modèle et $\mathsf{S} = \tilde{\mathsf{S}}^C$.

Pour chaque itération i de l'EM, les n espérances conditionnelles $\mathbb{E}[S(X_k,Y_k) \,|\, y_k; \hat{\theta}_i\,]$, $k \in \{1,\dots,n\}$, ne sont pas calculables explicitement. En effet, la loi *a posteriori* des données manquantes a pour densité

$$\pi_\theta(\,j,\beta \,|\, y) = \frac{\omega_j \det\Gamma_j^{-1/2} \exp\left\{-1/2\sigma_j^2\|y - \Phi_\beta\alpha_j\|^2 - 1/2\beta^T\Gamma_j^{-1}\beta\right\}}{\sum_{\ell=1}^{C}\omega_\ell \det\Gamma_\ell^{-1/2}\int \exp\left\{-1/2\sigma_\ell^2\|y - \Phi_{\beta'}\alpha_\ell\|^2 - 1/2\beta'^T\Gamma_\ell^{-1}\beta'\right\}\mathrm{d}\beta'} \tag{71}$$

et ne permet pas le calcul exact de l'espérance. Une approximation par les méthodes de Monte Carlo n'est pas envisageable car la simulation de couples *i.i.d.* $\{(i_1,\beta_1),\dots,(i_k,\beta_k)\}$ par des méthodes directes (inversion de la fonction de répartition ou tout autre algorithme de simulation de loi connue) n'est pas possible. Une première solution proposée dans [AAT07] consiste à remplacer π_θ par la densité discrète

$$\pi_\theta(\,j,\beta \,|\, y) \approx \sum_{\ell=1}^{C} \pi_\theta(\ell \,|\, \beta_\ell^\star(y), y)\mathbb{1}_{\{j,\beta\}}(\ell, \beta_\ell^\star(y)) \,, \quad \text{où} \quad \left\{\begin{array}{l} \beta_\ell^\star(y) = \arg\max_\beta \pi_\theta(\beta \,|\, \ell, y) \,, \\ \pi_\theta(\ell \,|\, \beta, y) \propto \pi_\theta(\ell, \beta \,|\, y) \,. \end{array}\right.$$

Cette alternative représente une charge de calcul importante. Il s'agit en effet de calculer à chaque itération i de l'algorithme EM les paramètres de déformation $(\beta_j^\star(y_1), \ldots, \beta_j^\star(y_n))$ pour toute les classes $j \in \{1, \ldots, C\}$. Cette étape revient donc à effectuer à chaque itération de l'EM nC recalages entre les prototypes des différentes classes et les différentes observations (Cf. Section B .2) : c'est un problème d'optimisation non linéaire dont la résolution, en plus d'être coûteuse en ressources, peut mener à un minimum local. Par ailleurs, il a été prouvé dans [AAT07] qu'avec une telle approximation de l'étape E, l'algorithme EM résultant ne converge pas vers l'estimateur de maximum de vraisemblance. Dans [AKT10a], un algorithme MCMC a été proposé pour approcher cette espérance et la convergence de l'EM stochastique résultant a été prouvé (Cf. Section D .3).

C .3 Apprentissage séquentiel dans des modèles à données manquantes

A la différence des méthodes d'apprentissage en bloc (Section C .2), l'apprentissage séquentiel permet d'estimer les paramètres d'un modèle au fur et à mesure que les données sont collectées. De tels algorithmes sont en effet indispensables lorsque les observations sont des vecteurs de très grande dimension et qu'il est virtuellement impossible de pouvoir les stocker simultanément [LACM06]. Ils permettent également de suivre l'évolution temporelle des paramètres lorsque les observations sont acquises de façon séquentielle et ainsi permettre le traitement de l'information en temps réel. Enfin, comparée aux méthodes en blocs, l'approche séquentielle permet un allégement considérable du temps de calcul, notamment lorsque des méthodes d'approximation numérique (optimisation, MCMC etc...) sont inclues dans l'algorithme. Dans cette partie, l'algorithme EM classique [DLR77] et ses variantes (MCEM (64), SAEM (65) etc...) seront désignés sous le nom générique d'EM en bloc, par opposition aux algorithmes EM séquentiels que nous présentons à présent.

Une procédure d'apprentissage séquentielle est caractérisée par une fonction $\mathcal{F} : \Theta \times \mathsf{Y} \to \Theta$ telle que :

$$\forall\, n \in \mathbb{N}, \quad \hat{\theta}_{n+1} = \mathcal{F}(\hat{\theta}_n, y_{n+1}), \quad \hat{\theta}_0 \in \Theta\ . \tag{72}$$

Notons que l'indice n qui correspondait au nombre d'observations (constant) dans l'algorithme EM en bloc permet dans une approche séquentielle d'identifier les itérations. En effet, la n-ième observation permet de procéder au calcul de l'estimateur $\hat{\theta}_n$. Dans le contexte des modèles à données manquantes, l'algorithme EM en bloc ne peut être directement appliqué dans un cadre séquentiel. En effet, par construction (Cf. Proposition C.4), l'inégalité (55) n'est valide que si l'ensemble des données reste identique d'une itération à l'autre. Des adaptations sont donc nécessaires.

Recursive EM

L'algorithme Recursive EM [Tit84] peut être analysé comme une version séquentielle de l'EM Gradient [Lan95] (66) légèrement modifié. En effet, une des limites de l'EM Gradient est que la matrice J (67) n'est pas nécessairement définie positive ce qui impose des restrictions en terme d'implémentation. Dans le Recursive EM, la matrice J est remplacée par la matrice de Fischer des données complètes I (73)

$$\forall\, \hat{\theta} \in \Theta\ , \quad I(\hat{\theta}) = -\mathbb{E}_{\hat{\theta}} \left[\nabla_\theta^2 \log f_{\hat{\theta}}(X, Y) \right]\ , \tag{73}$$

et une itération du Recursive EM s'écrit

$$\hat{\theta}_{n+1} = \hat{\theta}_n + \rho_n\, I^{-1}(\hat{\theta}_n) \nabla_\theta \log p_{\hat{\theta}_n}(y_{n+1}), \quad \hat{\theta}_0 \in \Theta\ . \tag{74}$$

avec $\{\rho_n,\ n \in \mathbb{N}\}$ une suite décroissante de nombres positifs. D'après (74), l'estimateur $\hat{\theta}_{n+1}$ ne nécessite que l'observation y_{n+1} et $\hat{\theta}_n$ pour être calculé : il entre de ce fait dans le cadre des algorithmes séquentiels (72). Bien que structurellement éloigné de l'algorithme EM classique, il en partage le nom car c'est un algorithme d'apprentissage itératif adapté aux modèles à données manquantes. Chaque itération n'est pas composée des deux étapes E et M mais nécessite les calculs du gradient de la vraisemblance des données incomplètes et de la matrice de Fisher associée. Notons toutefois qu'en vertu de l'identité de Fisher [DLR77, discussion p.29], le gradient de la vraisemblance incomplète (parfois appelé vecteur de score) peut s'écrire :

$$\nabla_\theta \log p_{\hat{\theta}}(y_{n+1}) = \mathbb{E}_{\hat{\theta}} \left[\nabla_\theta \log f_{\hat{\theta}}(X_{n+1}, Y_{n+1}) \,\middle|\, y_{n+1} \right] . \tag{75}$$

Dans le cas des modèles exponentiels (58), le calcul du gradient devient :

$$\nabla_\theta \log p_{\hat{\theta}}(y_{n+1}) = \nabla_\theta t(\hat{\theta}) + \left\langle \mathbb{E}_{\hat{\theta}} \left[S(X_{n+1}, Y_{n+1}) \,\middle|\, y_{n+1} \right], \nabla_\theta r(\hat{\theta}) \right\rangle ,$$

qui revient au calcul de l'espérance conditionnelle du vecteur des statistiques exhaustives sous l'estimateur courant $\hat{\theta}$, présent dans l'algorithme EM en bloc (60). Toutefois dans les cas où cette espérance n'est pas calculable exactement, le Recursive EM sera moins coûteux en calcul car il requiert d'approcher une seule espérance $\mathbb{E}_{\hat{\theta}}[S(X_{n+1}, Y_{n+1}) \,|\, y_{n+1}]$ contre n espérances $\{\, \mathbb{E}_{\hat{\theta}}[S(X_k, Y_k) \,|\, y_k] \,,\ 1 \leq k \leq n \,\}$ dans l'algorithme EM en bloc (60).

Lorsque la vraisemblance complète suit un modèle exponentiel et que le modèle est bien spécifié (Remarque C.1), une preuve de convergence du Recursive EM a été proposée dans [WZ06]. Sous des hypothèses raisonnables, le Théorème 1 de [WZ06] démontre que l'estimateur $\theta^\star = \lim_{n \to \mathbb{N}} \hat{\theta}_n$, $\{\hat{\theta}_n,\ n \in \mathbb{N}\}$ étant la séquence d'estimateurs obtenue par (74), vérifie

$$\theta^\star = \left\{ \theta \in \Theta,\ \nabla_\theta \mathrm{K}\left(p_{\theta^\star} \,\|\, p_\theta \right)\Big|_{\theta=\theta^\star} = 0 \right\} , \quad \text{presque-sûrement,} \tag{76}$$

où $\theta^\star \in \Theta$ est la « vraie » valeur du paramètre. Pour toutes distributions $\mathbb{P}_1$ et $\mathbb{P}_2$ définies sur un même espace de probabilité $(\mathsf{X}, \mathcal{X})$, admettant des densités p_1 et p_2 (par rapport à une mesure de domination commune), $\mathrm{K}(\, p_1 \,\|\, p_2 \,)$ désigne la divergence de Kullback-Leibler définie par

$$\mathrm{K}(\, p_1 \,\|\, p_2 \,) = \mathbb{E}_{p_1} \left[\log \frac{p_1(X)}{p_2(X)} \right] = \int_{\mathsf{X}} \log \frac{p_1(x)}{p_2(x)}\, p_1(x)\, \mathrm{d}x \,. \tag{77}$$

En conséquence, sous l'hypothèse que la matrice de Fisher (73) est calculable (ce qui suppose, entre autre, que l'espérance $\mathbb{E}_{\hat{\theta}}[\, S(X, Y)]$ soit connue) et est définie positive, le Recursive EM est un candidat potentiel pour l'estimation séquentielle des paramètres dans des modèles à données manquantes. Il a notamment été appliqué avec succès en traitement d'image : détection de changement dans des scènes de fonds [KTKPB02] ; en traitement du signal : identification d'utilisateurs dans des canaux multi-utilisateurs [LGW04], estimation des directions d'arrivée d'un signal (DOA) [CB05] et modélisation de traffic IP [LACM06].

Online EM

Une autre approche séquentielle, plus proche de la formulation de l'EM [DLR77], a été proposée dans [CM07] : l'Online EM . Généralisant [NH98], cet algorithme construit, à partir d'un paramètre initial $\hat{\theta}_0 \in \Theta$, une séquence d'estimateurs $\{\hat{\theta}_n,\ n \in \mathbb{N}\}$ telle que

$$\forall\, n \in \mathbb{N}\ , \quad \hat{\theta}_{n+1} = \arg\max_{\theta\in\Theta} \hat{Q}_{n+1}(\theta)\ , \quad \hat{\theta}_0 \in \Theta\ , \tag{78}$$

où la suite de fonctions $\{\theta \to \hat{Q}_n(\theta),\, n \in \mathbb{N}\}$ est définie récursivement par :

$$\begin{cases} \hat{Q}_1(\theta) = \mathbb{E}_{\hat{\theta}_0}\left[\log f_\theta(X_1, Y_1) \,\middle|\, y_1\right]\ , \quad \hat{\theta}_0 \in \Theta\ , \\ \hat{Q}_{n+1}(\theta) = \hat{Q}_n(\theta) + \rho_n\left(\mathbb{E}_{\hat{\theta}_n}\left[\log f_\theta(X_{n+1}, Y_{n+1}) \,\middle|\, y_{n+1}\right] - \hat{Q}_n(\theta)\right), \end{cases} \tag{79}$$

avec $\{\rho_n,\ n \in \mathbb{N}\}$ une suite décroissante de nombres positifs.

Dans le cas des modèles exponentiels (58), les étapes E (79) et M (78) de l'Online EM s'écrivent simplement pour tout $\hat{\theta}_0 \in \Theta$ et $n \in \mathbb{N}$:

$$\begin{cases} \hat{s}_{n+1} = \hat{s}_n + \rho_n\left(\mathbb{E}_{\hat{\theta}_n}\left[S(X_{n+1}, Y_{n+1}) \,\middle|\, y_{n+1}\right] - \hat{s}_n\right), \quad \hat{s}_1 = \mathbb{E}_{\hat{\theta}_0}\left[S(X_1, Y_1) \,\middle|\, y_1\right], \\ \hat{\theta}_{n+1} = \bar{\theta}(\hat{s}_{n+1})\,. \end{cases} \tag{80}$$

Tout comme pour le SAEM (65), l'étape E est une approximation stochastique. Les finalités de cette approximation sont toutefois différentes :

- Dans le SAEM , étant donné un ensemble d'observations $y_1, \ldots, y_n$, l' objectif de l'approximation stochastique est de trouver les points fixes de la fonction

$$s \to \bar{s} \circ \bar{\theta}(s) = \frac{1}{n}\sum_{k=1}^{n} \mathbb{E}_{\bar{\theta}(s)}[\, S(X_k, Y_k) \,|\, y_k\,] \tag{81}$$

  qui n'est pas calculable explicitement en raison de l'espérance conditionnelle. Pour cela, la fonction

$$s \to \tilde{s} \circ \bar{\theta}(s) = \frac{1}{n}\sum_{k=1}^{n} S(X_k, y_k)\ , \quad X_k \sim \pi_{\bar{\theta}(s)}(\,\cdot\,|\, y_k)\ , \tag{82}$$

  est considérée comme une version bruitée de $s \to \bar{s} \circ \bar{\theta}(s)$

- Dans l'Online EM , on suppose qu'à chaque itération une nouvelle observation $Y \sim \mathbb{P}_\star$ est disponible et l'approximation stochastique permet de trouver les points fixes de la fonction :

$$s \to s_\star \circ \bar{\theta}(s) = \mathbb{E}_\star\left[\mathbb{E}_{\bar{\theta}(s)}[\, S(X, Y) \,|\, Y\,]\right] \tag{83}$$

  où $\mathbb{E}_\star$ désigne l'espérance sous la loi $\mathbb{P}_\star$. Cette fonction n'est pas calculable car la distribution exacte des données $\mathbb{P}_\star$ n'est pas connue. L'approximation stochastique de l'Online EM considère donc la fonction

$$s \to \tilde{s}_\star \circ \bar{\theta}(s) = \mathbb{E}_{\bar{\theta}(s)}[\, S(X, Y) \,|\, Y\,]\ , \quad Y \sim \mathbb{P}_\star\ , \tag{84}$$

  comme une version bruitée de $s \to s_\star \circ \bar{\theta}(s)$.

Un théorème analogue à celui garantissant la convergence du Recursive EM a été prouvé dans [CM07, Théorème 5] pour les modèles exponentiels. Soit $\theta^\star = \lim_{n\to\infty} \hat{\theta}_n$, $\{\hat{\theta}_n,\ n \in \mathbb{N}\}$ étant la séquence d'estimateurs obtenue par l'Online EM (80), alors

$$\theta^\star \in \left\{\theta \in \Theta,\ \nabla_\theta \mathrm{K}\left(p_\star \,\|\, p_\theta\right)\big|_{\theta=\theta^\star} = 0\right\}, \quad \text{presque-sûrement,} \tag{85}$$

où $p_\star$ désigne la densité de la distribution réelles des données $\mathbb{P}_\star$ (Cf. Remarque C.1).

Comparé au Recursive EM [Tit84], l'Online EM [CM07] dispense du calcul de la matrice de Fisher I (73) ainsi que de son inverse (qui peut être coûteux en temps de calcul quand I^{-1} doit être approchée). L'Online EM se rapproche de la méthodologie de l'EM [DLR77] avec l'alternance des étapes E et M et en conserve donc la simplicité d'implémentation. Enfin, la convergence de l'Online EM est garantie même dans des situations où le modèle est mal spécifié ce qui n'est pas le cas pour le Recursive EM : dans (85), la densité $p_\star$ peut-être différent de $p_{\theta^\star}$. Pour ces différentes raisons, l'Online EM permet de couvrir un domaine d'application plus important.

Un cas particulier de l'Online EM permettant l'apprentissage séquentiel de paramètres dans un réseau gaussien normalisé avait été proposé dans [SI00], avec le choix spécifique $\rho_n = (1 + \lambda_n/\rho_{n-1})^{-1}$ où $0 \leq \lambda_n \leq 1$. Ce modèle de réseau gaussien normalisé présente des similitudes avec le modèle hiérarchique présenté dans le cadre des mélanges de prototype déformable (Remarque C.10). En particulier, dans ces deux modèles, la structure des statistiques exhaustives est identique (dénoté ϕ dans [SI00, Section 4.3 p.7] et S dans [AK10, Appendix p.16]). Toutefois, la seule donnée manquante du modèle de réseau gaussien normalisé est la variable de classe (discrète), ce qui permet de calculer l'espérance du vecteur des statistiques exhaustives de façon exacte. A l'inverse, la présence de la variable cachée de déformation dans l'exemple C.10 nécessite une approximation supplémentaire.

Deux résultats ont été prouvés pour cet algorithme [SI00, Section 4] :

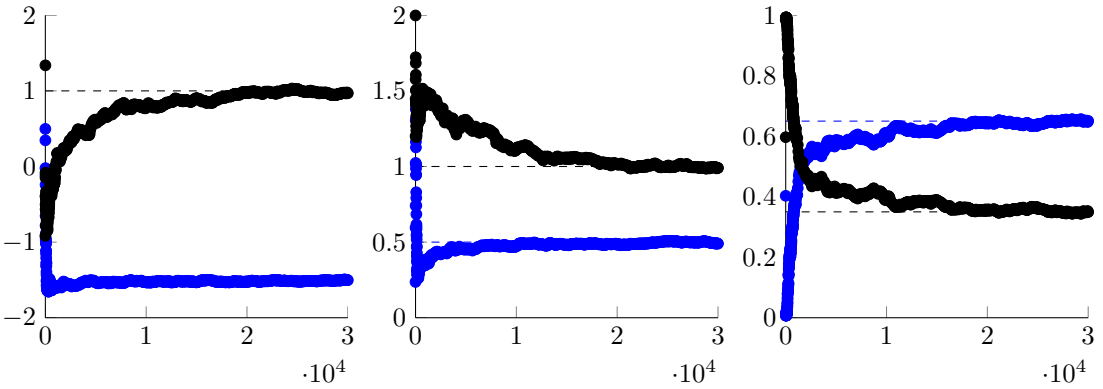
- Pour un certain choix de $\{\lambda_n,\ n \in \mathbb{N}\}$, lorsque l'étape de maximisation n'intervient qu'après n iterations durant lesquelles les n mêmes observations $y_1, \ldots, y_n$ sont traitées (*i.e.* l'estimateur est laissé constant durant cette période), alors la séquence d'estimateurs proposée par cet algorithme est la même que celle obtenue par l'EM en bloc avec les données $y_1, \ldots, y_n$.
- L'étape E est équivalente à une approximation stochastique de Robbins-Monro [RM51], ce qui justifie que la séquence $\{\hat{\theta}_n,\ n \in \mathbb{N}\}$ converge vers l'estimateur de maximum de vraisemblance.

Enfin, une idée intéressante proposée dans [SI00, Section 6] permet de supprimer, d'ajouter ou de diviser les classes estimées par l'algorithme d'apprentissage. En effet, dans l'approche séquentielle, les observations peuvent connaître des variations importantes tout au long de l'apprentissage. Par conséquent, la représentativité des différentes classes est potentiellement amenée à changer au cours du temps et certaines classes peuvent devenir inutiles et d'autres trop importantes.

L'Online EM [CM07] a été implémenté avec succès pour détecter les directions d'arrivée d'un signal sur un capteur [CCM06] et une version proche de l'algorithme de Sato et al. [SI00] a permis de classifier des données issues d'un modèle de mélange gaussien [HG09].

Exemple C.11. Mélange de gaussiennes - (3)
L'Online EM [CM07] est implémenté pour le modèle de mélange de deux gaussiennes (Exemple C.2). La base d'apprentissage, composée de 30000 observations simulées par le modèle (50), et utilisée pour l'estimation de paramètres par l'EM en bloc (Exemple C.7), est à présent traitée de façon séquentielle. Initialisée aléatoirement, la séquence d'estimateurs $\{\hat{\theta}_n,\ n \in \mathbb{N}\}$ est mise à jour par la récursion (80) avec $\rho_n = n^{-\nu}$ où $\nu = 0.75$. La

figure 26 montre que l'estimateur converge au bout d'environ 20000 iterations vers les vrais paramètres du modèles $\theta^\star$ (représentés en pointillés).



estimation des moyennes $\{\hat{\mu}_{1,n}, \hat{\mu}_{2,n},\ 1 \leq n \leq 3.10^4\}$

estimation des écarts types $\{\hat{\sigma}_{1,n}, \hat{\sigma}_{2,n},\ 1 \leq n \leq 3.10^4\}$

estimation des poids $\{\hat{\omega}_{1,n}, \hat{\omega}_{2,n},\ 1 \leq n \leq 3.10^4\}$

FIGURE 26 – Estimation des paramètres par l'Online EM

La figure 27 montre que la variance de l'estimateur $\{\hat{\mu}_{1,n}, \hat{\mu}_{2,n},\ n \in \mathbb{N}\}$ obtenu dans le cas de l'algorithme EM en bloc (en haut) et dans l'Online EM (en bas) sont similaires : ceci confirme que la convergence au bout de 50 itérations de l'EM en bloc utilisant 30000 données (Figure 25) est comparable avec celle observée lorsque ces mêmes données sont traitées de façon séquentielle par l'Online EM (Figure 26). Toutefois, l'EM en bloc nécessite que l'intégralité des données soit traitée 50 fois ce qui implique une charge de calcul plus importante. La figure 28 représente l'évolution des estimateurs obtenus par ces deux algorithmes en fonction du temps : l'avantage de l'approche séquentielle est évident. Cette différence serait encore plus grande dans le cas où l'étape E nécessiterait d'être approchée numériquement : en effet, pour atteindre la convergence le calcul des poids *a posteriori* (63) est effectuée environ $30 \times 30000 = 900000$ dans le cas de l'EM en bloc contre environ 20000 pour l'Online EM , soit un rapport de 1 à 45. En revanche, ce constat n'est plus valable lorsque c'est l'étape M qui doit être approchée : 30 étapes M sont nécessaires pour atteindre la convergence dans le cas de l'EM en bloc contre 20000 pour l'Online EM .

Enfin, la figure 29 illustre les propriétés d'adaptabilité de l'Online EM . Nous considérons le scénario suivant : jusqu'à $n = 30000$, les données y_n sont simulées comme précédemment puis, au delà, les valeurs des paramètres de la classe 1 sont changés ($\mu_1' = -1.0$, $\sigma_1' = 0.35$ et $\omega_1' = 0.55$). Environ 10000 données sont nécessaires pour que la séquence d'estimateurs converge vers les nouveaux paramètres.

Dans cet exemple, la matrice de Fisher des données complètes $I(\hat{\theta}_n)$ (73) est diagonale et le vecteur de score (75) s'écrit simplement comme une fonction de y_{n+1}, $\hat{\theta}_n$ et des poids *a posteriori* (63). En conséquence l'EM séquentiel [Tit84], (74) est implémentable. La mise à jour des paramètres suit la récursion :

$$\hat{\omega}_{1,n+1} = \hat{\omega}_{1,n} + \rho_n(\pi_{1,n+1} - \hat{\omega}_{1,n}) \ , \quad \hat{\mu}_{j,n+1} = \hat{\mu}_{j,n} + \rho_n \left\{ \frac{\pi_{j,n+1}}{\hat{\omega}_{j,n}}(y_{n+1} - \hat{\mu}_{j,n}) \right\} \ ,$$

$$\hat{\sigma}_{j,n+1} = \hat{\sigma}_{j,n} + \rho_n \left\{ \frac{\pi_{j,n+1}}{2\hat{\sigma}_{j,n}\hat{\omega}_{j,n}}[(y_{n+1} - \hat{\mu}_{j,n})^2 - \hat{\sigma}_{j,n}^2] \right\} \ , \quad \hat{\omega}_{2,n+1} = 1 - \hat{\omega}_{2,n+1} \ .$$

où $\pi_{j,n+1} = \mathbb{P}_{\hat{\theta}_n}[X = j \,|\, y_{n+1}]$ sont les poids *a posteriori* (63). Dans cette implémentation, le choix de $\{\rho_n,\ n \in \mathbb{N}\}$ est identique à celui de l'Online EM . En pratique, la convergence du Recursive EM est similaire à celle de l'Online EM , comme le montre la figure 28.

C .4 Relation entre l'approche en bloc et l'approche en ligne des algorithmes de type EM pour les modèles exponentiels

L'EM , le SAEM et l'Online EM sont tous des cas particuliers de l'algorithme général suivant, dont l'objectif est de trouver les points fixes de la fonction $h : \mathsf{S} \to \mathsf{S}$:

$$h : s \to \mathbb{E}_{\bar{\theta}(s)}[\, S(X,Y)\,] = \int_{\mathsf{X}} \int_{\mathsf{Y}} S(x,y) f_{\bar{\theta}(s)}(x,y) \mathrm{d}x \mathrm{d}y \tag{86}$$

en alternant entre une étape de calcul d'espérance $h(s)$ et une étape de maximisation $\bar{\theta}(s)$. Utilisant la relation $f_\theta(x,y) = \pi_\theta(x \,|\, y) p_\star(y)$, h peut se mettre sous la forme de deux espérances

$$h : s \to \mathbb{E}_\star \left[\mathbb{E}_{\bar{\theta}(s)} \left[\, S(X,Y) \,|\, Y \,\right] \right] . \tag{87}$$

La convergence de cette procédure itérative est assurée par la Proposition 3 de [CM07] qui prouve que pour tout $(s_0, s) \in \mathsf{S}^2$

$$h(s) \geq h(s_0) \Longrightarrow \mathrm{K}\left(p_\star \,\|\, p_{\bar{\theta}(s)} \right) \leq \mathrm{K}\left(p_\star \,\|\, p_{\bar{\theta}(s_0)} \right) . \tag{88}$$

Les points fixes de h correspondent donc au sous ensemble de S défini par

$$s^\star \in \left\{ s \in \mathsf{S},\ \nabla_s \mathrm{K}\left(p_\star \,\|\, p_{\bar{\theta}(s)} \right)\Big|_{s=s^\star} = 0 \right\} . \tag{89}$$

Approche en bloc Lorsque qu'un ensemble fixe d'observations $(y_1, \ldots, y_n) \in \mathsf{Y}^n$ est disponible, la distribution empirique des observations s'écrit pour tout $A \in \mathcal{Y}$

$$\mathbb{P}_n[Y \in A] = \frac{1}{n} \sum_{k=1}^{n} \delta_{y_k}(A) .$$

Les approches en bloc correspondent au cas particulier $\mathbb{P}_\star = \mathbb{P}_n$. Lorsque l'espérance *a posteriori* sous la loi des données manquantes est calculable, la recherche des points fixes de h (87) correspond exactement à l'EM proposé par [DLR77]. En effet, dans ce cas

$$h(s) = \mathbb{E}_n \left[\mathbb{E}_{\bar{\theta}(s)}[\, S(X,Y) \,|\, Y \,] \right] = \frac{1}{n} \sum_{k=1}^{n} \mathbb{E}_{\bar{\theta}(s)}[\, S(X,Y) \,|\, y_k \,] = \bar{s} \circ \bar{\theta}(s) ,$$

où $\mathbb{E}_n$ est l'espérance sous la loi $\mathbb{P}_n$. Lorsque l'espérance *a posteriori* sous la loi des données manquantes n'est pas calculable, le calcul de $\bar{s} \circ \bar{\theta}(s)$ (81) est approché, pour tout $s \in \mathsf{S}$, par $\tilde{s} \circ \bar{\theta}(s)$ (82) et une procédure de Robbins-Monro est nécessaire pour compenser cette approximation : cette approche correspond au SAEM .

Notons que lorsque $\mathbb{P}_\star = \mathbb{P}_n$, la divergence de Kullback-Leibler entre $\mathbb{P}_n$ et $\mathbb{P}_{\bar{\theta}(s)}$ s'écrit :

$$\mathrm{K}\left(p_n \,\|\, p_{\bar{\theta}(s)} \right) = -\frac{1}{n} \sum_{k=1}^{n} \log p_{\bar{\theta}(s)}(y_k) .$$

et correspond, à un facteur près, à la fonction de vraisemblance ℓ du paramètre $\bar{\theta}(s)$ pour les données $(y_1, \ldots, y_n)$ (45). Ainsi dans ce cas particulier, la propriété (88) est équivalente à la propriété de monotonicité de l'EM (Proposition C.4) et l'ensemble des points stationnaires (89) correspond au maximum de vraisemblance.

Approche en ligne Les approches en ligne supposent qu'à chaque itération, une nouvelle observation $Y \sim \mathbb{P}_\star$ est disponible. A la différence des approches en bloc dans lesquelles l'espérance sous la loi des observations est calculable car $\mathbb{P}_\star = \mathbb{P}_n$ est une loi discrète, cette dernière ne l'est généralement pas dans les approches en ligne car la distribution $\mathbb{P}_\star$ est le plus souvent inconnue. L'Online EM propose donc d'approcher la fonction $s_\star \circ \bar{\theta}(s)$ par $\tilde{s}_\star \circ \bar{\theta}(s)$ et une procédure de Robbins-Monro est nécessaire pour compenser cette approximation.

	EM	SAEM	Online EM
loi des observations	$\mathbb{P}_n$	$\mathbb{P}_n$	$\mathbb{P}_\star$
espérance sous la loi des observations	connue	connue	inconnue
espérance sous la loi des données manquantes	connue	inconnue	connue

Tableau 3 – Comparaison des algorithmes EM , SAEM , Online EM

Le Tableau 3 compare les cas d'implémentation des algorithmes EM , SAEM , Online EM . Alors que l'EM est un algorithme déterministe, le SAEM et l'Online EM ont recours à une procédure d'approximation stochastique pour compenser l'impossibilité de calculer l'espérance sous la loi des données manquantes et sous la loi des observations respectivement.

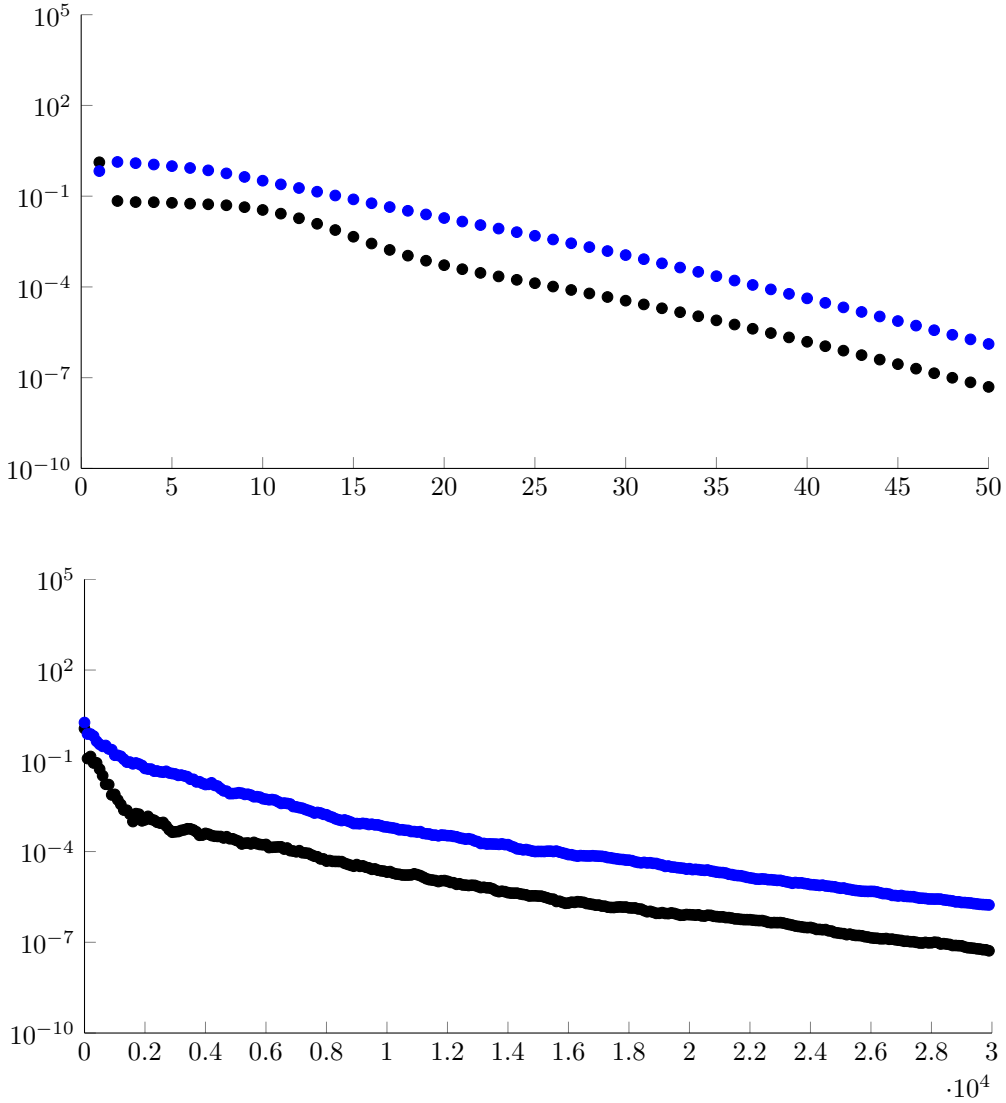


Figure 27 – Variance de le la séquence d'estimateurs $\{\hat{\mu}_{1,n}, \hat{\mu}_{2,n}, n \in \mathbb{N}\}$ obtenue par l'EM en bloc (en haut) et par l'Online EM (en bas)

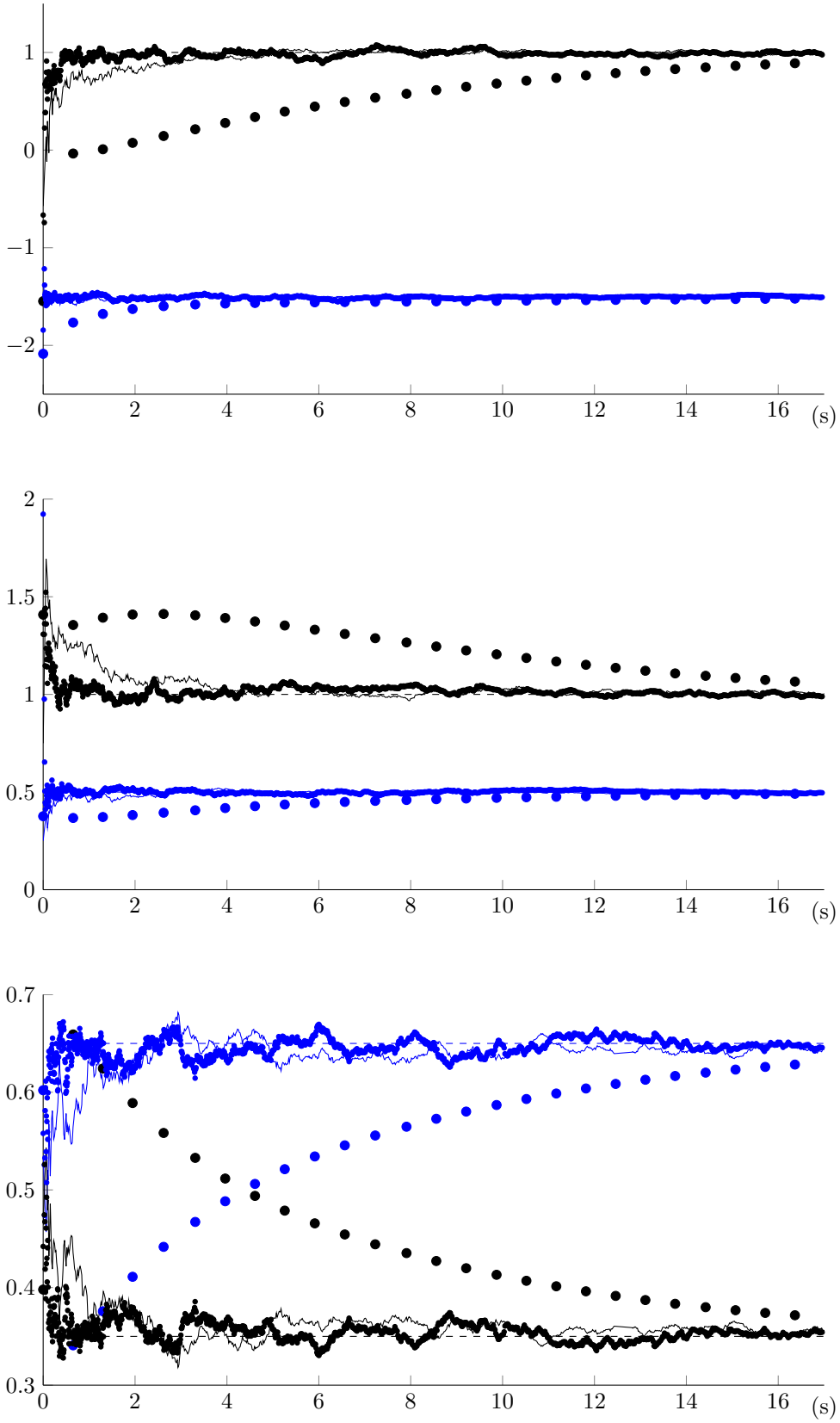


FIGURE 28 – Estimation des paramètres par l'EM en bloc (points), par l'Online EM (pointillés) et par le Recursive EM (trait plein) en fonction du temps (en secondes) : $\{\hat{\mu}_{1,t}, \hat{\mu}_{2,t},\, t > 0\}$ (en haut), $\{\hat{\sigma}_{1,t}, \hat{\sigma}_{2,t},\, t > 0\}$ (au centre) et $\{\hat{\omega}_{1,t}, \hat{\omega}_{2,t},\, t > 0\}$ (en bas)

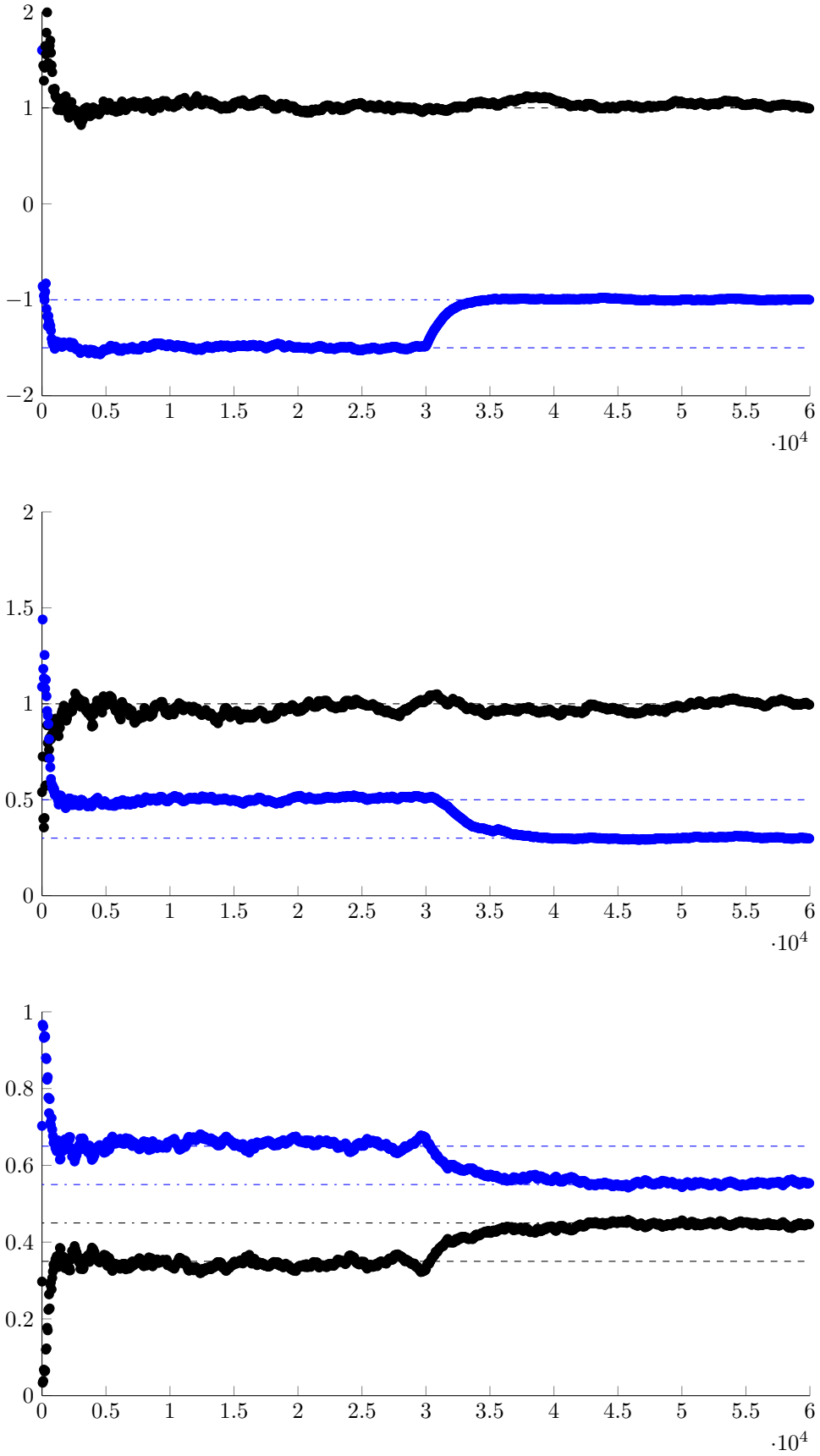


Figure 29 – Estimation des paramètres par l'Online EM dans le cas de données évoluant au cours du temps : $\{\hat{\mu}_{1,n}, \hat{\mu}_{2,n},\ n \in \mathbb{N}\}$ (en haut), $\{\hat{\sigma}_{1,n}, \hat{\sigma}_{2,n},\ n \in \mathbb{N}\}$ (au centre) et $\{\hat{\omega}_{1,n}, \hat{\omega}_{2,n},\ n \in \mathbb{N}\}$ (en bas)

Résumé de la contribution

Dans le Préambule B et le Chapitre III, nous proposons un modèle d'observation, similaire à celui de l'Exemple C.10, décrivant la variabilité spectrale et spatiale des SIR monospectrales et multispectrales d'aéronefs autour de formes caractéristiques inconnues. Le Préambule C expose différentes méthodes permettant d'estimer les paramètres de ce genre de modèle. Dans notre modèle, le choix des lois *a priori* des variables manquantes permet de garantir l'appartenance de la fonction de vraisemblance complète à la famille des exponentielles (voir Chapitre II-3). En conséquence, un algorithme de type Expectation-Maximization EM (Préambule C .2) est bien approprié à l'estimation des paramètres de ce modèle. Toutefois, les méthodes *en bloc* ne permettent pas de résoudre efficacement notre problème d'apprentissage, principalement pour les deux raisons suivantes :

- **performances** - pour un nombre important de données, l'approche en ligne est moins coûteuse en temps et en ressource que les méthodes en bloc (voir l'exemple C.11) et permet un traitement en temps réel de l'information.
- **contraintes expérimentales** - notre méthode doit être adaptée à un contexte où les données sont traitées séquentiellement et ne peuvent être stockées.

Nous nous tournons donc vers les approches séquentielles.

Plusieurs algorithmes d'estimation séquentielle de paramètres dans des modèles à données manquantes ont été proposés ; voir le Préambule C .3. Les deux principales approches, l'Online EM [CM07] et le Recursive EM [Tit84], construisent de façon différente une séquence d'estimateur convergeant presque-sûrement vers un ensemble où la divergence de Kullback-Leibler (77) entre la distribution du modèle et celle des observations est stationnaire. Nous privilégions la démarche de l'Online EM qui permet de couvrir un éventail plus large de situations et en particulier les cas où le modèle est mal spécifié ; voir la Remarque C.1 pour plus de détails. Cependant, cet algorithme n'est pas implémentable dans notre contexte : en effet, l'espérance conditionnelle intervenant dans l'étape E (80) n'est pas calculable explicitement. Dans le cas des algorithmes en bloc, différentes méthodes dérivées de l'EM (MCEM , SAEM , SAEM-MCMC ; voir la Remarque C.8), offrent des solutions alternatives à cette situation, fréquente en pratique. Il a été prouvé dans le cas de modèles exponentiels que l'approximation de l'espérance conditionnelle par des méthodes de Monte Carlo ou MCMC n'altère pas la convergence de la séquence d'estimateurs construite pas ces algorithmes, qui est similaire à celle de l'EM [FM03, DLM99, KL04]. Toutefois, aucun algorithme alternatif ne permettant de traiter ce problème n'a été proposé dans les contextes séquentiels.

Nous avons montré dans le Préambule C .4 que dans le cadre des modèles exponentiels, les méthodes d'apprentissage telles que l'EM , le SAEM ou l'Online EM avaient pour objectif la recherche des points stationnaires de la fonction définie sur S, l'espace des statistiques exhaustives du modèle, par

$$h : s \to \mathbb{E}_{\bar{\theta}(s)}[S(X,Y)] \ ,$$

où S est le vecteur des statistiques exhaustives. Notons que l'implémentation de l'Online EM nécessite de pouvoir calculer, entre autre, la constante

$$p_\theta(y) = \int_{\mathsf{X}} f_\theta(x,y)\mathrm{d}x$$

intervenant dans le calcul de $\mathbb{E}_\theta[S(X,Y) \,|\, Y]$, ce qui, en dehors des situations particulières où les données manquantes sont des variables discrètes, est rarement possible. Nous proposons un nouvel algorithme, le Monte Carlo online EM (MCoEM), qui étend l'Online EM aux situations dans lesquelles l'espérance conditionnelle sous la loi *a posteriori* des

données manquantes n'est pas calculable. Dans le MCoEM , l'ensemble des points fixes de h est approché récursivement par la procédure d'approximation stochastique définie par :

$$\hat{s}_{n+1} = \hat{s}_n + \rho_n \left(\frac{1}{L} \sum_{\ell=1}^{L} S(X_\ell, Y_n) - \hat{s}_n \right), \quad \left\{ \begin{array}{l} Y_n \sim \mathbb{P}_\star \\ X_\ell \sim \pi_{\bar{\theta}(\hat{s}_n)}(\cdot \,|\, Y_n) \end{array} \right.$$

Le tableau 4 compare ces différents algorithmes et détaille les situations où ils sont implémentables.

	EM	SAEM	Online EM	MCoEM
loi des observations	$\mathbb{P}_n$	$\mathbb{P}_n$	$\mathbb{P}_\star$	$\mathbb{P}_\star$
esp. sous la loi des observations	connue	connue	inconnue	inconnue
esp. sous la loi des données manquantes	connue	inconnue	connue	inconnue
approximation stochastique de la fonction $s \to h(s)$	$h(s)$	$\frac{1}{n}\sum_{k=1}^{n} S(X_k, Y_k)$	$\mathbb{E}_{\bar{\theta}(s)}[S(X,Y) \,|\, Y]$	$\frac{1}{L}\sum_{\ell=1}^{L} S(X_\ell, Y)$
		$X_k \sim \pi_{\bar{\theta}(s)}(\cdot \,|\, Y_k)$	$Y \sim \mathbb{P}_\star$	$\left\{ \begin{array}{l} Y \sim \mathbb{P}_\star \\ X_\ell \sim \pi_{\bar{\theta}(s)}(\cdot \,|\, Y) \end{array} \right.$

Tableau 4 – Comparaison des algorithmes EM , SAEM , Online EM et MCoEM

Des techniques de simulation telles que la méthode de rejet ou l'utilisation d'une chaîne de Markov permettent d'obtenir des échantillons de la loi *a posteriori* , sans nécessiter la connaissance de la constante $p_\theta(y)$. Le MCoEM est donc implémentable dans des situations où l'Online EM ne l'est pas. Notons qu'à la différence des méthodes en bloc, il n'est pas possible dans un contexte séquentiel de coupler les itérations de la chaîne de Markov avec celles de l'EM comme c'est le cas dans l'algorithme SAEM-MCMC ; voir Préambule D .3. En effet, à chaque itération la loi cible de la chaîne de Markov change brutalement car la loi *a posteriori* des données manquantes s'écrit conditionnellement à une nouvelle observation qui n'a jamais été traitée auparavant. Il est donc nécessaire de simuler à chaque itération une nouvelle chaîne de Markov visant la loi des données manquantes conditionnellement à la nouvelle observation et à l'estimateur courant des paramètres.

Dans le contexte des mélanges de modèle à prototype déformable, il n'est pas possible d'obtenir des échantillons *i.i.d.* de la loi *a posteriori* des données manquantes. Par conséquent, le MCoEM doit être couplé avec une méthode appropriée permettant de simuler les données manquantes conditionnellement aux observations. Le MCoEM est particulièrement sensible à cette étape de simulation qui, à chaque itération, s'apparente à une procédure simultanée de clustering/recalage d'une nouvelle donnée. Comme l'approximation stochastique des statistiques exhaustives est mise à jour après chaque nouvelle observation, un mauvais échantillonnage des données manquantes conditionnellement à une seule observation suffit à perturber l'estimation des paramètres. Les algorithmes en bloc tels que le SAEM sont plus robustes à ce problème dans la mesure où la mise à jour des paramètres n'intervient qu'après que les données manquantes de toutes les observations aient été simulées. Par conséquent, si certaines données sont délicates à clusteriser/recaler, l'influence

sur l'estimation sera moindre que dans une méthode en ligne. Une attention particulière doit donc être apportée à cette étape de simulation dans un contexte séquentiel. Nous proposons une méthode MCMC, basée sur l'algorithme de Carlin et Chib (voir le Préambule D .3 pour plus de détails) qui permet d'obtenir des échantillons *exactement* sous la loi *a posteriori* . L'utilisation de cet échantillonneur ne se limite pas au cas des mélanges de modèle à prototype déformable et s'adapte aisément aux problèmes de simulation de lois mixtes dont l'une des variables est l'indicateur de classe. L'association du MCoEM avec l'algorithme de Carlin et Chib permet

(i) d'obtenir des résultats d'estimation similaires à ceux du SAEM ,

(ii) d'améliorer considérablement les performances en terme de *coût* de calcul comparé au SAEM ,

(iii) d'adapter le traitement des données à un contexte séquentiel.

Notre algorithme est tout d'abord appliqué à un modèle jouet de mélange de régression Gaussienne, utilisé dans [CMR05], mais dans un contexte où l'Online EM n'est pas implémentable. Quantitativement, l'estimateur des paramètres de regression proposé par le MCoEM couplé avec l'échantillonneur de Carlin et Chib converge vers les valeurs exactes avec une variance acceptable (Chapitre II-5). Dans un second temps, des illustrations sur des images de chiffres manuscrits, sur des courbes de croissance (Chapitre II-6) et sur des SIR d'aéronefs mono et multispectrales (Chapitre III) prouvent l'efficacité de notre méthode sur des données réelles.

Nous menons actuellement des travaux dont le but est de prouver que, dans le cas des modèles exponentiels et sous certaines hypothèses relatives à l'approximation de l'espérance, la séquence d'estimateurs construite par le MCoEM converge presque sûrement vers un ensemble où la divergence de Kullback-Leibler entre la distribution du modèle et la distribution réelle des observations est stationnaire. Sous ces hypothèses, le MCoEM aurait alors une convergence analogue à celle de l'Online EM et du Recursive EM tout en couvrant un éventail plus large de situations.

Enfin comme remarqué dans le Préambule B , une perspective avec de nombreuses applications serait de généraliser le MCoEM à des situations où le modèle n'est pas exponentiel. Très peu de travaux dans la littérature traitent le problème d'estimation des paramètres par un algorithme de type EM dans le cas des modèles non exponentiels, à la fois d'un point de vue théorique et méthodologique. Afin de rester dans l'esprit de l'algorithme EM et de la Proposition C.4, une idée serait de construire une séquence de paramètres $\{\hat{\theta}_n \in \Theta,\, n \in \mathbb{N}\}$ convergeant vers $\theta^\star = \arg\max_{\theta\in\Theta} \mathcal{Q}(\theta^\star, \theta)$, où $\mathcal{Q}$ est la fonction définie dans (54), par une procédure d'approximation stochastique de type Robbins-Monro. Pour tout état courant des paramètres $\hat{\theta}_n \in \Theta$ et toute observation Y_{n+1}, la version bruitée du paramètre $\hat{\theta}_{n+1} = \arg\max_{\theta\in\Theta} \mathcal{Q}(\hat{\theta}_n, \theta)$ serait le paramètre $\hat{\theta}' \in \Theta$ obtenu par plusieurs itérations des étapes (i) et (ii) ci-dessous tel qu'initialement $\hat{\theta}' = \hat{\theta}_n$:

(i) une étape de simulation d'un m_n-échantillon de données manquantes

$$(X_{n+1}^{(1)}, \ldots, X_{n+1}^{(m_n)}) \sim \pi_{\hat{\theta}'}(\,\cdot \,|\, Y_{n+1})\ ,$$

(ii) une étape d'optimisation

$$\hat{\theta}' = \arg\max_{\theta\in\Theta} \frac{1}{m_n} \sum_{j=1}^{m_n} \log f_\theta(X_{n+1}^{(j)}, Y_{n+1})\ .$$

Bien entendu, cette procédure est plus lourde à implémenter que le MCoEM car à chaque itération, les étapes de simulation (i) et d'optimisation (ii) doivent être répétées un certains nombre de fois, mais cette solution reste une première piste à explorer.

D Méthodes de Monte Carlo par chaînes de Markov

D .1 Éléments sur les chaînes de Markov

Soit $\{X_k,\ k \in \mathbb{N}\}$ une suite de variables aléatoires définies sur un espace mesurable $(\mathsf{X}, \mathcal{X})$. $\{X_k,\ k \in \mathbb{N}\}$ est une chaîne de Markov si pour tout $k \in \mathbb{N}^*$, la loi de X_k sachant $X_0, \ldots, X_{k-1}$ est indépendante de $X_0, \ldots, X_{k-2}$. En d'autres termes, la connaissance des k états précédents $(x_0, \ldots, x_{k-1})$ n'apporte pas plus d'information sur la loi de probabilité de X_k que la seule connaissance de l'état précédent x_{k-1}. Ainsi, pour tout $A \in \mathcal{X}$ et pour toute réalisation $(x_0, \ldots, x_{k-1}) \in \mathsf{X}^k$ des états $X_0, \ldots, X_{k-1}$,

$$\mathbb{P}[X_k \in A \,|\, x_{k-1}, \ldots, x_0] = \mathbb{P}[X_k \in A \,|\, x_{k-1}] \,. \tag{90}$$

Remarque D.1. Dans ce travail, nous ne travaillons que sur des chaînes de Markov définies sur un espace d'états général *i.e.* $\mathsf{X} = \mathbb{R}^d$, $d > 0$. D'autres domaines de recherche étudient les chaînes de Markov à valeurs discrètes et définies sur un espace d'état fini $\mathrm{Card}(\mathsf{X}) < \infty$ ou les processus de Markov à temps continu $\{X_t,\, t \in \mathbb{R}\}$.

Une chaîne de Markov $\{X_k,\ k \in \mathbb{N}\}$ est définie par :
- une distribution initiale ρ telle que $X_0 \sim \rho$,
- un noyau de transition $K : (\mathsf{X}, \mathcal{X}) \to [0, 1]$ qui définit la probabilité conditionnelle

$$\forall\, A \in \mathcal{X}, \quad \forall\, x \in \mathsf{X}, \quad K(x, A) = \mathbb{P}[X_k \in A \,|\, X_{k-1} = x] \,.$$

Définition D.2. Soit $\{Q_k,\ k \in \mathbb{N}\}$ une suite de noyaux de transition tels que pour $k \in \mathbb{N}$, $Q_k : (\mathsf{X}, \mathcal{X}) \to [0, 1]$. Toute chaîne de Markov évoluant suivant le schéma

$$X_0 \xrightarrow{Q_0} X_1 \xrightarrow{Q_1} X_2 \xrightarrow{Q_2} X_3 \xrightarrow{Q_3} \cdots \,,$$

est appelée chaîne inhomogène. Dans la majorité des cas, les chaînes étudiées sont homogènes : pour tout $k \in \mathbb{N}$, $Q_k = K$.

Nous définissons à présent deux propriétés essentielles relatives aux chaînes de Markov. Soit π une mesure de probabilité sur $(\mathsf{X}, \mathcal{X})$:

- **π-stationnarité** : $\{X_k,\ k \in \mathbb{N}\}$ admet π comme mesure stationnaire si pour tout $k \in \mathbb{N}$,

$$X_k \sim \pi \Longrightarrow X_{k+1} \sim \pi \,. \tag{91}$$

  Par transitivité, si $X_k \sim \pi$, alors pour tout $m \geq k$, $X_m \sim \pi$. On dit aussi que π est invariante pour le noyau de transition K ce qui s'écrit $\pi = \pi K$, *i.e.* pour tout $A \in \mathcal{X}$:

$$\pi(A) = \int_{\mathsf{X}} K(x, A)\pi(\mathrm{d}x) \,. \tag{92}$$

  Si π est une mesure stationnaire de la chaîne $\{X_k,\ k \in \mathbb{N}\}$ ayant une distribution initiale ρ différente de π, on dénomme par régime transitoire (par opposition au régime stationnaire), les états X_k tels que $X_k \nsim \pi$.

- **π-réversibilité** : $\{X_k,\ k \in \mathbb{N}\}$ est π-réversible si la mesure $(\mathrm{d}x, \mathrm{d}x') \to \pi(\mathrm{d}x)Q(x, \mathrm{d}x')$ définie sur l'espace produit $(\mathsf{X} \times \mathsf{X}, \mathcal{X} \otimes \mathcal{X})$ est symétrique *i.e.* pour tout $(A, B) \in \mathcal{X}^2$

$$\int_A \pi(\mathrm{d}x)K(x, B) = \int_B \pi(\mathrm{d}x)K(x, A) \,. \tag{93}$$

La π-réversibilité de $\{X_k,\ k \in \mathbb{N}\}$ est une propriété plus forte que la π-stationnarité et implique cette dernière. En effet, si $X_k \sim \pi$ alors pour tout $A \in \mathcal{X}$

$$\begin{aligned}\mathbb{P}[X_{k+1} \in A] = \int_{\mathsf{X}} \mathbb{P}[X_k \in \mathrm{d}x_k, X_{k+1} \in A] &= \int_{\mathsf{X}} \mathbb{P}[X_k \in \mathrm{d}x_k]\, \mathbb{P}[X_{k+1} \in A \,|\, x_k] \\ &= \int_{\mathsf{X}} \pi(\mathrm{d}x_k) K(x_k, A) = \int_A \pi(\mathrm{d}x_{k+1}) K(x_{k+1}, \mathsf{X}) = \pi(A)\ ,\end{aligned}$$

où l'avant dernière égalité découle de la π-réversibilité de la chaîne (93) et la dernière du fait que pour tout $x \in \mathsf{X}$, $K(x, \cdot)$ est une mesure de probabilité sur $(\mathsf{X}, \mathcal{X})$ et s'intègre donc à 1. De plus, si une chaîne π-réversible atteint son état stationnaire alors

$$\forall\, A \in \mathcal{X},\ \forall\, x \in \mathsf{X}, \quad \mathbb{P}[X_k \in A \,|\, X_{k-1} = x] = \mathbb{P}[X_k \in A \,|\, X_{k+1} = x]\ ,$$

i.e. le sens de parcours de la chaîne n'influence pas sa dynamique.

Pour d'autres propriétés sur les chaînes de Markov, on pourra se référer aux ouvrages [CMR05, Section 14] et [MT09].

D .2 Méthodes MCMC

Les méthodes de Monte Carlo regroupent l'ensemble des techniques permettant d'approcher numériquement des espérances (et plus généralement des intégrales) qui ne sont pas calculables analytiquement. Pour toute mesure de probabilité π sur $(\mathsf{X}, \mathcal{X})$ et toute fonction f π-intégrable, il s'agit d'approcher les quantités

$$\pi f = \int_{\mathsf{X}} f(x)\pi(\mathrm{d}x) \quad \text{par} \quad S_n(f) = \frac{1}{n}\sum_{k=1}^{n} f(X_k)\ , \tag{94}$$

où $X_1, \ldots, X_n$ sont des variables aléatoires indépendantes et identiquement distribuées (*i.i.d.*) de loi π. La Loi forte des Grands Nombres (LGN) garantie la convergence presque-sûrement de la moyenne empirique $S_n(f)$ vers πf. De plus, si $f \in \mathcal{L}^2(\pi)$, un Théorème Central Limite (TCL) précise la nature de la convergence :

$$\sqrt{n}\,(S_n(f) - \pi f) \xrightarrow{\mathbb{P}} \mathcal{N}(0, v(f)) \quad \text{où} \quad v(f) = \lim_{n\to\infty} \frac{1}{n}\mathbb{E}\left[\left\{\sum_{k=1}^{n}(f(X_k) - \pi f)\right\}^2\right]\ . \tag{95}$$

Cette propriété permet entre autre de déterminer la vitesse de convergence de l'estimateur $S_n(f)$ vers πf et d'établir des intervalles de confiance.

Toutefois, dans de nombreuses situations, simuler des échantillons *i.i.d.* de loi π n'est pas faisable : les méthodes de Monte Carlo par chaînes de Markov (MCMC) [Rob96, RC04, GRS96] sont une alternative aux méthodes de Monte Carlo classiques, qui ne sont plus utilisables dans ce contexte. L'idée est de substituer les échantillons *i.i.d.* de π dans l'estimateur $S_n(f)$ (94) par la réalisation d'une chaîne de Markov $\{X_k,\ k \in \mathbb{N}\}$ pour laquelle une Loi des Grands Nombres (94) et éventuellement un Théorème Central Limite (95) existent. Dans le contexte des MCMC, ces théorèmes sont souvent appelées théorèmes d'ergodicité et, par extension, une chaîne de Markov les vérifiant est appelée chaîne ergodique.

Cependant, les théorèmes d'ergodicité sont généralement difficiles à établir et nécessitent des développements théoriques conséquents. En effet, l'analyse des chaînes de Markov est complexe, notamment pour les raisons suivantes :

(i) par construction, les différents états de la chaîne ne sont pas indépendants (90),

(ii) en dehors du régime stationnaire, les différents états n'ont pas la même distribution.

En conséquence, la plupart des théorèmes d'ergodicité exigent des hypothèses sur les noyaux de transition qui ne sont pas toujours vérifiables en pratique. Nous n'énonçons qu'un théorème asymptotique fondamental, un TCL (95) prouvé par [KV86] justifiant la convergence de nombreux MCMC sans nécessiter un développement théorique supplémentaire.

Théorème D.3. Soit $\{X_k,\ k \in \mathbb{N}\}$ une chaîne de Markov et ϕ une mesure sur $(\mathsf{X}, \mathcal{X})$

- *apériodique* : Il n'y a pas de cycle dans la chaîne *i.e.* il n'existe pas de nombre entier $d > 1$ et une collection de sous ensembles de X $(A_1, \ldots, A_d) \in \mathcal{X}^d$ avec $\cup_{p=1}^{d} A_p = \mathsf{X}$ tels que
$$\forall\, (k,p) \in \mathbb{N} \times \{1, \ldots, d-1\}\,, \quad \begin{cases} \mathbb{P}[X_{k+1} \in A_1 \,|\, X_k \in A_d] = 1\,, \\ \mathbb{P}[X_{k+1} \in A_{p+1} \,|\, X_k \in A_p] = 1\,. \end{cases}$$
- *ϕ-irréductible* : Pour n'importe quel état initial, tous les ensembles $A \in \mathcal{X}$ tels que $\phi(A) > 0$ ont une probabilité strictement positive d'être visités au moins une fois par la chaîne.

Si de plus, $\{X_k,\ k \in \mathbb{N}\}$ est *π-réversible*, alors le Théorème Central Limite (95) s'applique dès lors que $v(f)$ existe et est fini.

Les hypothèses d'apériodicité et de ϕ-irréductibilité sont vérifiées par la plupart des noyaux de transition des chaînes de Markov utilisées dans le contexte des MCMC. Toutefois, l'hypothèse $f \in \mathcal{L}^2(\pi)$ n'est plus suffisante dans le cadre des MCMC pour avoir $v(f) < \infty$ et il faut en plus vérifier que :

$$\sum_{k=1}^{\infty} \mathrm{Cov}(\bar{f}(X_0), \bar{f}(X_k)) < \infty\,, \quad \text{où} \quad \bar{f} = f - \pi f\,. \tag{96}$$

Nous ferons l'hypothèse que les fonctions f que nous étudions vérifient (96). Tout l'enjeu des méthodes MCMC consiste donc à spécifier un noyau de transition sur $\mathsf{X} \times \mathcal{X}$ qui soit π-réversible.

L'algorithme de Metropolis-Hastings

L'échantillonneur de Metropolis-Hastings [MRR$^+$53, Has70] fait figure de référence parmi les méthodes MCMC. Pour toute distribution cible π sur $(\mathsf{X}, \mathcal{X})$, qui peut être connue à une constante près, cet algorithme permet de construire une chaîne de Markov $\{X_k,\ k \in \mathbb{N}\}$ π-réversible ; voir [CG95] pour une analyse détaillée de cette méthode. En particulier, dans les contextes bayésiens, les densités d'intérêt sont souvent connues à une constante de normalisation près et l'échantillonneur de Metropolis-Hastings est particulièrement bien adapté pour approcher des espérances conditionnelles dans ce type de modèle.

L'algorithme nécessite la spécification d'un noyau de proposition P défini sur $(\mathsf{X}, \mathcal{X})$ pour lequel pour tout $x \in \mathsf{X}$, il est possible d'obtenir des échantillons de loi $\tilde{X} \sim P(x, \cdot\,)$. Par simplicité, on suppose que P est absolument continu par rapport à une mesure de domination (typiquement la mesure de Lebesgue sur $\mathbb{R}^d$) *i.e.* il existe une densité p telle que pour tout $A \in \mathcal{X}$:

$$P(x, A) = \int_A p(x, \tilde{x}) \mathrm{d}\tilde{x}\,.$$

Aucune hypothèse supplémentaire n'est faite sur P : en particulier, il n'est pas nécessaire que P admette π comme mesure invariante (92) ou encore qu'il soit π-réversible (93). Dans

cette configuration, il n'y a aucune raison pour que la chaîne de Markov ayant P pour noyau de transition soit π-réversible.

L'idée de l'algorithme de Metropolis-Hastings est d'accepter un candidat proposé par P avec une certaine probabilité α choisie de façon à rendre la transition π-réversible. Dans le cas où le candidat est refusé, l'état suivant est identique à l'état courant. Le noyau de transition résultant K s'écrit alors pour tout état courant $x \in \mathsf{X}$:

$$K(x, \mathrm{d}x') = P(x, \mathrm{d}x')\alpha(x, x') + \delta_x(\mathrm{d}x')r(x) \,, \tag{97}$$

où $x \to r(x)$ est la probabilité que la proposition soit refusée. Soit $\tilde{X} \sim P(x, \cdot)$ la proposition et X' l'état suivant de la chaîne, il vient en intégrant (97)

$$K(x, \mathsf{X}) = 1 = \int P(x, dx')\alpha(x, x') + \int \delta_x(dx')r(x)$$

d'où l'on tire r.

Supposons que π soit absolument continue par rapport à une mesure domination (par exemple la mesure de Lebesgue) et écrivons, avec un léger abus de notation, pour tout $A \in \mathcal{X}$, $\pi(A) = \int_A \pi(x)\mathrm{d}x$. Dans ce cas, la condition de π-réversibilité (93) s'écrit pour tout $(x, x') \in A \times B$, $(A, B) \in \mathcal{X}^2$:

$$\pi(x)p(x, x')\alpha(x, x') = \pi(x')p(x', x)\alpha(x', x) \quad \text{presque-sûrement} \,. \tag{98}$$

En effet, le terme diagonal du noyau K est identique à droite et à gauche de l'égalité. Pour satisfaire (98), lorsque $\pi(x)p(x, x') \geq \pi(x')p(x', x)$ alors il suffit de poser

$$\begin{cases} \alpha(x', x) = 1 \,, \\ \alpha(x, x') = \frac{\pi(x')p(x', x)}{\pi(x)p(x, x')} \,, \end{cases} \quad \text{et dans le cas contraire} \quad \begin{cases} \alpha(x, x') = 1 \,, \\ \alpha(x', x) = \frac{\pi(x)p(x, x')}{\pi(x')p(x', x)} \,. \end{cases}$$

ce qui revient à choisir α comme étant la fonction de $\mathsf{X} \times \mathsf{X} \to [0, 1]$ définie par

$$\alpha(x, x') = 1 \wedge \frac{\pi(x')p(x', x)}{\pi(x)p(x, x')} \,. \tag{99}$$

Dans le cas où π ou P ne sont pas absolument continus par rapport à une mesure de domination commune, une formule plus générale de la probabilité d'acceptation est proposée dans [GRS96] et rappelée dans le Chapitre IV, Proposition IV.10.

L'algorithme de Metropolis-Hastings permet donc de simuler une chaîne de Markov $\{X_k, \, k \in \mathbb{N}\}$ π-réversible, dont la transition est décrite par la procédure suivante :

Procédure 1 Transition $X_k \to X_{k+1}$ de l'algorithme de Metropolis-Hastings

Entrées : La distribution cible π, le noyau de proposition P et l'état courant $X_k = x_k$.
1 - Simuler un candidat $\tilde{X} \sim P(x_k, \cdot)$, soit $\tilde{x}$ sa réalisation.
2 - Accepter le candidat *i.e.* poser $X_{k+1} = \tilde{x}$ avec probabilité $\alpha(x_k, \tilde{x})$ (99),
3 - Dans le cas contraire poser $X_{k+1} = x_k$.
4 - **Sortie :** L'état suivant X_{k+1}.

Remarque D.4. Lien avec la méthode d'Acceptation-Rejet
L'algorithme de Metropolis-Hastings présente des similitudes avec la méthode d'Acceptation-Rejet [Rob96, Section 2.3], dans laquelle une densité instrumentale g (équivalent au noyau de proposition P dans Metropolis-Hastings) doit être spécifiée telle que $\pi < Mg$ où $M \geq 1$ est une constante. On simule alors des candidats $\tilde{X} \sim g$ que l'on conserve avec une probabilité $\pi(\tilde{x})/Mg(\tilde{x})$. Les échantillons conservés de g sont des variables *i.i.d.* de loi π. Les deux inconvénients de la méthode d'Acceptation-Rejet par rapport à l'algorithme de Metropolis-Hastings sont

- Dans la plupart des cas, la densité π doit être connue explicitement tandis que l'algorithme de Metropolis-Hastings est implémentable lorsque la distribution cible est connue à une constante de normalisation près (π apparaît au numérateur et au dénominateur de α).
- Trouver une densité g simulable vérifiant la contrainte $\pi < Mg$ n'est pas toujours aisé et une « mauvaise » densité instrumentale (*i.e.* telle que $M \gg 1$) rend la méthode trop lente pour être utilisable ; les échantillons étant acceptés en moyenne avec une probabilité $1/M$.

En toute rigueur, la comparaison avec l'algorithme de Metropolis-Hastings sur ce dernier point est difficile à établir. En effet, même si à chaque itération de l'algorithme de Metropolis-Hastings un nouvel échantillon est obtenu, il peut s'agir, avec une probabilité r, de la copie de l'état précédent. Un « mauvais » choix de noyau de proposition P se traduit par une forte probabilité de rester dans le même état ce qui d'une part ralentit la progression de la chaîne et d'autre part, sous certaines hypothèses, entraîne une variance asymptotique importante [Tie95] (voir la Section D .4 pour plus de détails). Ainsi, dans certains cas, trouver un noyau de proposition permettant d'obtenir une variance asymptotique acceptable peut s'avérer aussi délicat que de trouver une densité instrumentale permettant d'obtenir des échantillons *i.i.d.* en un temps raisonnable par la méthode d'Acceptation-Rejet.

L'échantillonneur de Gibbs, un cas limite de l'algorithme de Metropolis-Hastings

L'échantillonneur de Gibbs [GG84] permet de simuler une chaîne de Markov π-réversible lorsque :

(i) $(\mathsf{X}, \mathcal{X})$ est un espace produit $\mathsf{X} = \mathsf{E}^d$ et $\mathcal{X} = \mathcal{E}^{\otimes d}$ ($d \geq 2$) où typiquement $\mathsf{E} \subseteq \mathbb{R}$.

(ii) Pour tout $\ell \in \{1, \ldots, d\}$, on peut simuler X_ℓ suivant la loi conditionnelle

$$X_\ell \sim \pi_\ell(\,\cdot \,|\, x_{-\ell}) \quad \text{où} \quad \forall\, u \in \mathsf{E}, \quad \pi_\ell(u \,|\, x_{-\ell}) = \frac{\pi(x_{1:\ell-1}, u, x_{\ell+1:d})}{\int_{\mathsf{E}} \pi(x_1, \ldots, x_d) \mathrm{d}x_\ell}.$$

L'algorithme de Gibbs simule une chaîne de Markov π-réversible dont la transition $X_k \to X_{k+1}$ s'écrit :

Procédure 2 Transition $X_k \to X_{k+1}$ de l'échantillonneur de Gibbs

Entrées : La distribution cible π, l'état courant $X_k = x_k$ et un vecteur de poids $(w_{k,1}, \ldots, w_{k,d})$.

1 - Sélectionner une coordonnée $\ell \in \{1, \ldots, d\}$ avec la probabilité $w_{k,\ell}$.

2 - Simuler un candidat $X_{k+1,\ell} \sim \pi_\ell(\,\cdot \,|\, x_{k,1:\ell-1}, x_{k,\ell+1:d})$.

3 - Pour tout $j \neq \ell$, poser $X_{k+1,j} = x_{k,j}$.

4 - **Sortie :** L'état suivant X_{k+1}.

Pour tout $k \in \mathbb{N}$, le vecteur de poids $(w_{k,1}, \ldots, w_{k,d})$ est un paramètre de l'échantillonneur de Gibbs tel que pour tout $p \in \{1, \ldots, d\}$, $0 \leq w_{k,p} \leq 1$ et et $\sum_{p=1}^d w_{k,p} = 1$. Lorsque les coordonnées sont successivement mises à jour *i.e.* $w_{1,1} = 1, w_{2,2} = 1, \ldots, w_{d+1,1} = 1, \ldots$, l'échantillonneur est appelé *Systematic Gibbs Sampler* et *Random Scan Gibbs Sampler* lorsque les coordonnées sont mises à jour aléatoirement [LWK94]. Dans ce dernier cas le vecteur de poids est indépendant de k.

Soit K_ℓ le noyau de transition de l'échantillonneur de Gibbs lorsque la composante ℓ est modifiée et que les autres restent inchangées. Pour tout $x \in \mathsf{X}$ et $A = (A_1 \times \ldots \times A_d) \in \mathcal{X}$, K_ℓ s'écrit :

$$K_\ell(x, A) = \pi_\ell(A_\ell \,|\, x_{-\ell}) \prod_{j \neq \ell} \delta_{x_j}(A_j) \; . \tag{100}$$

K_ℓ est en fait un noyau de transition de Metropolis-Hastings (97) dans le cas particulier où la proposition (étape 2 de la Procédure 2) est toujours acceptée. Considérons K_ℓ comme le noyau de proposition P du noyau de transition de Metropolis-Hastings et soit $\tilde{x}$ un candidat simulé par $K_\ell(x, \cdot)$. La probabilité d'acceptation $\alpha(x, \tilde{x})$ (99) s'écrit alors :

$$\alpha(x, \tilde{x}) = 1 \wedge \frac{\pi(\tilde{x})\pi_\ell(x_\ell \,|\, \tilde{x}_{-\ell})}{\pi(x)\pi_\ell(\tilde{x}_\ell \,|\, x_{-\ell})} = 1 \wedge \frac{\pi(\tilde{x})\pi_\ell(x_\ell \,|\, \tilde{x}_{-\ell})}{\pi_\ell(x_\ell \,|\, x_{-\ell})\pi(x_{1:\ell-1}, \tilde{x}_\ell, x_{\ell+1:d})} = 1 \; ,$$

où la dernière égalité est vérifiée car pour tout candidat $\tilde{x} \sim K_\ell(x, \cdot)$, $\tilde{x}_{-\ell} = x_{-\ell}$. En conséquence, la probabilité $r(x)$ de rester dans un état x est nulle pour tout $x \in \mathsf{X}$, ce qui prouve que K_ℓ est bien un cas particulier d'un noyau de transition de Metropolis-Hastings . En conséquence, K_ℓ est π-réversible. Notons ℓ_k la coordonnée mise à jour lors de la transition $X_k \to X_{k+1}$ du *Systematic Gibbs Sampler i.e.* $w_{k,\ell_k} = 1$, les deux noyaux $K^{\mathtt{s}}$ et $K^{\mathtt{r}}$ du *Systematic Gibbs Sampler* et du *Random Scan Gibbs Sampler* s'écrivent pour tout $x_k \in \mathsf{X}$ et $A \in \mathcal{X}$:

$$\begin{cases} K^{\mathtt{s}}(x_k, A) = K_{\ell_k}(x_k, A) \; , \\ K^{\mathtt{r}}(x_k, A) = \sum_{\ell=1}^{d} w_\ell K_\ell(x_k, A) \; . \end{cases}$$

Notons que :

- La chaîne de Markov $\{X_k,\ k \in \mathbb{N}\}$ produite par $K^{\mathtt{s}}$ alterne entre d noyaux π-réversibles et est donc π-réversible.
- La chaîne de Markov $\{X_k,\ k \in \mathbb{N}\}$ produite par $K^{\mathtt{r}}$ est π-réversible. En effet, étant un mélange de d noyaux π-réversibles, $K^{\mathtt{r}}$ est π-réversible (tout mélange fini de noyaux réversibles et un noyau réversible).

Il est possible d'envisager un échantillonneur de Gibbs pour lequel une transition $X_k \to X_{k+1}$ correspond à la mise à jour successive de toutes les composantes de l'espace d'état. Dans le cas du *Systematic Gibbs Sampler*, le noyau de transition $\tilde{K}^{\mathtt{s}}$ s'écrit pour tout $x \in \mathsf{X}$ et $A = (A_1 \times \ldots \times A_d) \in \mathcal{X}$:

$$\begin{aligned} \tilde{K}^{\mathtt{s}}(x, A) &= K_1 K_2 \cdots K_d(x, A) \; , \\ &= \int_{A_1} \cdots \int_{A_d} \pi_1(\tilde{x}_1 \,|\, x_{2:d})\pi_2(\tilde{x}_2 \,|\, \tilde{x}_1, x_{3:d}) \cdots \pi_d(\tilde{x}_d \,|\, \tilde{x}_{1:d-1}) \mathrm{d}\tilde{x} \; . \end{aligned}$$

Or d'après la Propriété D.5, $\tilde{K}^{\mathtt{s}}$ n'est π-réversible que si et seulement si $K_1 K_2 \cdots K_d = K_d K_{d-1} \cdots K_1$, ce qui représente une hypothèse forte sur π, rarement vérifiée.

Notons toutefois que $\tilde{K}^{\mathtt{s}}$ admet π comme mesure stationnaire. En effet, pour tout $A \in \mathcal{X}$:

$$\begin{aligned} \int_{\mathsf{X}} \pi(\mathrm{d}x)\tilde{K}^{\mathtt{s}}(x, A) \quad &= \int_A \int_{\mathsf{E}^d} \pi(\mathrm{d}x)\pi(\mathrm{d}\tilde{x}_1 \,|\, x_{2:d}) \cdots \pi(\mathrm{d}\tilde{x}_d \,|\, \tilde{x}_{1:d-1}) \; , \\ &= \int_A \int_{E^{d-1}} \pi(\mathrm{d}x_{2:d})\pi(\mathrm{d}\tilde{x}_1 \,|\, x_{2:d}) \cdots \pi(\mathrm{d}\tilde{x}_d \,|\, \tilde{x}_{1:d-1}) \; , \\ &= \int_A \int_{E^{d-2}} \pi(\mathrm{d}\tilde{x}_1, \mathrm{d}x_{3:d})\pi(\mathrm{d}\tilde{x}_2 \,|\, \tilde{x}_1, x_{3:d}) \cdots \pi(\mathrm{d}\tilde{x}_d \,|\, \tilde{x}_{1:d-1}) \; , \\ &= \cdots = \int_A \int_E \pi(\mathrm{d}\tilde{x}_{1:d-1}, \mathrm{d}x_d)\pi(\mathrm{d}\tilde{x}_d \,|\, \tilde{x}_{1:d-1}) = \pi(A) \; . \end{aligned}$$

Proposition D.5. Soit P et Q deux noyaux π-réversibles définis sur un espace mesurable $(\mathsf{X}, \mathcal{X})$. Le noyau produit PQ est π-réversible si P et Q commutent *i.e.* $PQ = QP$.

Démonstration. Pour tout $(A, B) \in \mathcal{X}^2$, la π-réversibilité des noyaux P et Q permet d'écrire :

$$\int_A \pi(\mathrm{d}x) PQ(x, B) = \int_A \int_{\mathsf{X}} \int_B \pi(\mathrm{d}x) P(x, \mathrm{d}\tilde{x}) Q(\tilde{x}, \mathrm{d}x') = \int_{\mathsf{X}} \int_B P(\tilde{x}, A) Q(x', \mathrm{d}\tilde{x}) \pi(\mathrm{d}x')$$
$$= \int_B \pi(\mathrm{d}x') QP(x', A) \; .$$

D'où

$$\int_A \pi(\mathrm{d}x) PQ(x, B) = \int_B \pi(\mathrm{d}x) QP(x, A) \quad \Longleftrightarrow \quad PQ = QP \quad \pi\text{-presque-sûrement} \; .$$

□

Ce résultat se généralise par récurrence au produit d'un nombre fini de noyaux.

Remarque D.6. ***Metropolis-within-Gibbs***
En pratique, la simulation de certaines lois conditionnelles $\{\pi_\ell,\, \ell \in I \subseteq \{1, \ldots, d\}\}$, peut s'avérer délicate. Pour les composantes $\ell \in I$ problématiques, il est possible de remplacer la simulation exacte suivant π_ℓ par une étape de Metropolis-Hastings : $X_{k+1,\ell}$ est alors simulé par un noyau de Metropolis-Hastings (97) ayant π_ℓ comme distribution cible. L'algorithme résultant est parfois appelé *Metropolis-within-Gibbs* et hérite la π-réversibilité des algorithmes de Metropolis-Hastings et de Gibbs.

Remarque D.7. ***Data Augmentation***
Bien que n'ayant pas le même objectif, la Data Augmentation [TW87] partage certaines caractéristiques communes avec l'algorithme EM. En effet, cette méthode est implémentable lorsque :

(i) il existe une densité $\tilde{\pi}$ définie sur (X, Y) telle que pour tout $A \in \mathcal{X}$

$$\pi(A) = \int_A \int_{\mathsf{Y}} \tilde{\pi}(x, y) \mathrm{d}x \mathrm{d}y \; ,$$

(ii) pour tout $(x, y) \in \mathsf{X} \times \mathsf{Y}$, les lois conditionnelles $\tilde{\pi}(\,\cdot \,|\, x)$ et $\tilde{\pi}(\,\cdot \,|\, y)$ sont simulables.

L'hypothèse (ii) permet, grâce à un échantillonneur de Gibbs, de simuler une chaîne de Markov $\{(X_k, Y_k),\ k \in \mathbb{N}\}$ $\tilde{\pi}$-réversible sur l'espace $(\mathsf{X} \times \mathsf{Y})$ suivant le schéma :

$$\begin{array}{ccccccccc} & & Y_1 \sim \tilde{\pi}(\,\cdot \,|\, x) & & X_2 \sim \tilde{\pi}(\,\cdot \,|\, y) & & Y_3 \sim \tilde{\pi}(\,\cdot \,|\, x') & & X_4 \sim \tilde{\pi}(\,\cdot \,|\, y') \\ \begin{pmatrix} X_0 = x \\ \end{pmatrix} & \longrightarrow & \begin{pmatrix} X_1 = x \\ Y_1 = y \end{pmatrix} & \longrightarrow & \begin{pmatrix} X_2 = x' \\ Y_2 = y \end{pmatrix} & \longrightarrow & \begin{pmatrix} X_3 = x' \\ Y_3 = y' \end{pmatrix} & \longrightarrow & \cdots \end{array}$$

Figure 30 – Data Augmentation

Par construction, la suite de variables $\{X_{2k},\, k \in \mathbb{N}\}$ est aussi une chaîne de Markov dont le noyau de transition est :

$$K(x, \mathrm{d}x') = \int_{\mathsf{Y}} \tilde{\pi}(\mathrm{d}y' \,|\, x) \tilde{\pi}(\mathrm{d}x' \,|\, y) \; .$$

Il s'avère que K est un noyau π-réversible et $\{X_{2k},\, k \in \mathbb{N}\}$ admet donc, marginalement, π comme distribution stationnaire. En effet, pour tout $(A, A') \in \mathcal{X}^2$:

$$\int_A \pi(\mathrm{d}x) K(x, A') = \int_A \int_{A'} \int_{\mathsf{Y}} \pi(\mathrm{d}x)\tilde{\pi}(\mathrm{d}y' \,|\, x)\tilde{\pi}(\mathrm{d}x' \,|\, y') = \cdots$$
$$= \int_A \int_{A'} \int_{\mathsf{Y}} \tilde{\pi}(\mathrm{d}x \,|\, y')\tilde{\pi}(\mathrm{d}y' \,|\, x')\pi(\mathrm{d}x') = \int_{A'} \pi(\mathrm{d}x') K(x', A) \; .$$

D .3 MCMC pour les modèles à prototype déformable

Nous considérons à nouveau le modèle à prototype déformable développé dans les exemples B.8 et C.10. Rappelons que :

(i) dans un modèle à prototype déformable, une observation Y s'écrit conditionnellement à une classe $j \in \{1, \ldots, C\}$ et à une déformation $\beta \in \mathsf{B}$

$$Y = \Phi_\beta \alpha_j + \sigma_j^2 \epsilon \; , \quad \epsilon \sim \mathcal{N}_{|\Omega|}(0, \mathrm{Id}_{|\Omega|}) \; , \tag{101}$$

(ii) des données $Y_1, \ldots, Y_n$ sont observées et pour tout $k \in \{1, \ldots, n\}$, les variables aléatoires (J_k, β_k) sont *cachées* dans Y_k et donc inconnues,

(iii) le modèle spécifie des lois *a priori* pour les variables manquantes $J \sim (\omega_1, \ldots, \omega_C)$, $\beta \,|\, j \sim g_\theta(\,\cdot \,|\, j)$ et on désigne par $p_\theta(\,\cdot \,|\, j, \beta)$ (68), la densité des observations sachant j et β,

(iv) le modèle est exponentiel (58) et on appelle f_θ la densité jointe des observations et des données manquantes $f_\theta(j, \beta, y) = p_\theta(y \,|\, j, \beta) g_\theta(\beta \,|\, j)\omega_j$,

(v) l'objectif est d'estimer les paramètres du modèle, formalisés par $\theta \in \Theta$ (70),

(vi) l'algorithme EM (56)-(57) n'est pas implémentable car l'espérance *a posteriori* n'est pas calculable explicitement (71).

Les solutions proposées à ce problème d'apprentissage font intervenir des algorithmes MCMC, que nous étudions à présent.

Modèle à prototype déformable à une seule classe

Dans le cas d'un modèle à prototype déformable à une seule composante, la seule variable cachée est β. La densité *a posteriori* de β sachant Y n'étant pas simulable (Cf. (71)), l'implémentation d'un algorithme MCEM ou SAEM n'est donc pas possible. Un algorithme hybride SAEM-MCMC , dont la convergence a été prouvée [KL04, Théorème 1], a été proposé par [AKT10a] pour résoudre le problème d'estimation des paramètres d'un modèle à prototype déformable. Dans cette approche, l'estimation des paramètres du modèle et la simulation des données cachées sont couplées. L'algorithme évolue de façon séquentielle partant d'un paramètre initial $\hat{\theta}_0 = \bar{\theta}(\hat{s}_0)$ (102), pour une valeur initial $\hat{s}_0 \in \mathsf{S}$ quelconque, en suivant la Procédure 3.

[AKT10a] suggère d'utiliser un noyau de Metropolis-within-Gibbs (Cf. Remarque IV.23) à l'étape 1 de la Procédure 3 pour simuler les variables de déformation cachées. Ainsi, pour tout $(\theta, y) \in \Theta \times \mathsf{Y}$, le noyau $K_{\theta,y}$ consiste à proposer successivement pour chaque composant β_q, $q \in \{1, \ldots, 2d\}$ un candidat $\tilde{\beta}_q$ et de l'accepter avec un ratio de Metropolis-Hastings 99, de sorte à ce que la transition soit $\pi_\theta(\,\cdot \,|\, \beta'_{1:q-1}, \beta_{q+1:2d}, y)$-réversible (on désigne par $\beta'_{1:q-1}$ les $q-1$ premières composantes de β qui ont été mises à jour). L'astuce

Procédure 3 Transition $\hat{\theta}_i \to \hat{\theta}_{i+1}$ par l'algorithme SAEM-MCMC

Entrées : L'estimateur courant $\hat{\theta}_i \in \Theta$, l'approximation stochastique $\hat{s}_i \in \mathsf{S}$, les observations $(y_1, \ldots, y_n) \in \mathsf{Y}^n$ et les données manquantes $(\beta_{1,i}, \ldots, \beta_{n,i}) \in \mathsf{B}^n$ simulées à l'itération précédente.

1 - **simulation :** Pour tout $k \in \{1, \ldots, n\}$, simuler une nouvelle donnée manquante

$$\beta_{k,i+1} \sim K_{\hat{\theta}_i, y_k}(\beta_{k,i}, \cdot) \ ,$$

où $K_{\hat{\theta}_i, y_k}$ est un noyau de transition sur $(\mathsf{B}, \mathcal{B})$.

2 - **approximation stochastique :** Mettre à jour l'approximation stochastique

$$\hat{s}_{i+1} = \hat{s}_i + \gamma_i (\sum_{k=1}^{n} S(\beta_{k,i+1}, y_k) - \hat{s}_i) \ ,$$

3 - **maximisation :** Calculer le nouvel estimateur $\hat{\theta}_{i+1}$

$$\hat{\theta}_{i+1} = \bar{\theta}(\hat{s}_{i+1}), \quad \text{où} \quad \bar{\theta} : \left\{ \begin{array}{l} \mathsf{S} \to \Theta \ , \\ s \mapsto \arg\max_{\theta \in \Theta} \ nt(\theta) + \langle s, r(\theta) \rangle \ . \end{array} \right. \tag{102}$$

4 - **Sorties :** L'état suivant $\hat{\theta}_{i+1}$, $\hat{s}_{i+1}$ et $\beta_{1,i+1}, \ldots, \beta_{n,i+1}$.

de [AKT10a] est de proposer $\tilde{\beta}_q \sim g_\theta(\cdot \,|\beta'_{1:q-1}, \beta_{q+1:2d})$ où g_θ est la loi *a priori* de β. Ce choix de proposition permet en effet d'exprimer simplement la probabilité d'acceptation :

$$\alpha_q(\beta_q, \tilde{\beta}_q) = 1 \wedge \frac{\pi_\theta(\beta'_{1:q-1}, \tilde{\beta}_q, \beta_{q+1:2d} \,|\, y)}{\pi_\theta(\beta'_{1:q-1}, \beta_{q:2d} \,|\, y)} \ .$$

Rappelons que $g_\theta = \mathcal{N}_{2d}(0, \Gamma)$ et que la loi de proposition est bien simulable car une loi gaussienne multivariée conditionnelle est une loi gaussienne dont la moyenne et la matrice de covariance s'expriment en fonction des coordonnées connues du vecteur, de la moyenne et de la matrice de covariance de la loi jointe.

Toutefois, l'application du Théorème 1 de [KL04], garantissant la convergence de la séquence $\{\hat{\theta}_i, \ i \in \mathbb{N}\}$ obtenue par la Procédure 3 vers un ensemble où la vraisemblance est stationnaire, nécessite que la séquence $\{\hat{s}_i, \ i \in \mathbb{N}\}$ reste dans un sous ensemble compact $\mathcal{K} \subset \mathsf{S}$. Comme $\beta \in \mathsf{B} \subseteq \mathbb{R}^{2d}$, cette hypothèse n'est pas vérifiée et il faut stabiliser la Procédure 3 par une méthode de troncature sur les bornes [CGG88, AMP05]. L'idée est de forcer $\{\hat{s}_i, \ i \in \mathbb{N}\}$ à rester dans une suite de compacts $\{\mathcal{K}_n, \ n \in \mathbb{N}\}$ telle que

$$\bigcup_n \mathcal{K}_n = \mathsf{S}, \quad \text{et} \quad \mathcal{K}_n \subset \text{int}(\mathcal{K}_{n+1}) \ ,$$

où $\text{int}(\mathcal{K}_{n+1})$ est l'intérieur de l'ensemble $\mathcal{K}_{n+1}$. Ainsi tant que $\hat{s}_i$ reste dans l'un de ces ensembles compacts et demeure dans un voisinage de $\hat{s}_{i-1}$, la mise à jour des paramètres s'effectue avec la Procédure 3 et dans le cas contraire, $\hat{s}_i$ et $\beta_{1,i}, \ldots, \beta_{n,i}$ sont réinitialisés pour que l'itération suivante se déroule dans un ensemble où la convergence est garantie. La convergence de l'algorithme d'apprentissage résultant, le SAEM-MCMC avec troncature sur les bornes, a été prouvée par le Théorème 3.1 dans [AKT10a].

D'un point de vue pratique, cette solution présente l'avantage d'être implémentable en un temps raisonnable. En effet, même dans le cas d'un nombre élevé de données, l'étape de simulation ne consiste qu'en un *rafraîchissement* des données manquantes par une étape

de Metropolis-Hastings . En pratique, pour que la chaîne visite l'espace d'état plus rapidement, il est recommandé de répéter plusieurs fois l'étape 1 de la procédure 3, en particulier au cours des premières itérations pour que la chaîne de Markov atteigne plus rapidement son état stationnaire [KL04]. La structure couplée de l'estimation des paramètres et de la simulation des données manquantes est de ce point de vue très intéressante : le besoin en ressource du SAEM-MCMC [KL04] n'excède pas celui du SAEM [DLM99] qui n'est implémentable que lorsque la loi *a posteriori* est simulable.

Remarque D.8. Remarquons que dans cet algorithme, pour tout $k \in \{1, \dots, n\}$, la séquence de variables aléatoires $\{\beta_{k,i},\, i \in \mathbb{N}\}$ n'est pas une chaîne de Markov. En effet, le noyau de transition $K_{\hat{\theta}_i, y_k}$ varie à chaque itération à travers $\hat{\theta}_i$ et par cet intermédiaire $\beta_{k,i}$ dépend aussi des variables $\beta_{k,1}, \dots, \beta_{k,i-2}$. Les méthodes MCMC dont des noyaux de transition évoluent au cours de l'algorithme en fonction des états précédents sont appelés MCMC adaptatifs (*Adaptive MCMC*) [AR05]. Les Théorèmes d'ergodicité tels que (D.3) justifiant la convergence des MCMC classiques ne sont plus applicables car un processus généré par un MCMC adaptatif n'est généralement pas une chaîne de Markov. Des propriétés justifiant l'ergodicité de ces méthodes ont été prouvées dans [AR05, AM06] et en particulier [ŁRR13] pour les algorithmes de Metropolis-within-Gibbs adaptatifs. Notons que la preuve de convergence du SAEM-MCMC proposée dans [KL04] ne fait pas intervenir ces outils.

Modèle de mélange à C composantes

Par rapport au modèle à prototype déformable simple, le modèle de mélange fait intervenir une nouvelle donnée manquante, la variable de classe $J_k \in \{1, \dots, C\}$ associée à chaque observation Y_k.

Une solution naturelle est d'utiliser la méthode SAEM-MCMC avec troncature sur les bornes en simulant des échantillons $(J_k, \beta_k) \sim \pi_\theta(\cdot \,|\, y_k)$ au lieu de $\beta_k \sim \pi_\theta(\cdot \,|\, y_k)$ comme dans la Procédure 3-1. Intuitivement, on peut implémenter un échantillonneur de Gibbs mettant à jour ces deux composantes à chaque itération $i \in \mathbb{N}$ et pour tout $k \in \{1, \dots, n\}$:

(i) $\beta_{k,i+1} \sim \pi_{\hat{\theta}_{i-1}}(\cdot \,|\, J_{k,i}, y_k)$,

(ii) $J_{k,i+1} \sim \pi_{\hat{\theta}_{i-1}}(\cdot \,|\, \beta_{k,i+1}, y_k)$,

où l'étape (i) est réalisée comme dans la Procédure 3-1. Le noyau de transition résultant admet bien $\pi_{\hat{\theta}_{i-1}}(\cdot \,|\, y_k)$ comme mesure invariante et le Théorème 3.1 de [AKT10a] garantie théoriquement la convergence de cette méthode. Toutefois comme remarqué dans [AKT10a], l'enchaînement des étapes (i) et (ii) ne favorise pas les changements de classes et la chaîne. En effet, si $J_{k,i} = j$ alors :

- la variable aléatoire $\beta_{k,i+1}$ simulée en (i) paramètre une déformation $G_{\beta_{i,k+1}}$ qui, appliquée au prototype $\mathcal{T}_{\alpha_j}$, permet de se rapprocher de l'observation : $\mathcal{T}_{\alpha_j} \circ G_{\beta_{i,k+1}} \approx y_k$.
- en revanche, il n'y pas de raison particulière pour « espérer » que cette déformation, appliquée aux autres prototypes, permette de retrouver l'observation : pour $j' \neq j$, $\mathcal{T}_{\alpha_{j'}} \circ G_{\beta_{k,i+1}} \not\approx y_k$.

En conséquence, les probabilités *a posteriori* $\pi_{\hat{\theta}_{i-1}}(j' \,|\, \beta_{i,k+1}, y_k)$ des classes $j' \neq j$ sont la plupart du temps trop faibles pour espérer tirer $J_{i,k+1} \neq j$ à l'étape (ii). Cette situation est illustrée dans l'exemple D.9 pour un mélange de gaussiennes.

Exemple D.9. On considère la distribution cible définie sur $\{1, \dots, 4\} \times \mathbb{R}^2$ par $\pi_\theta(j, x) = g_j(x)/4$ où g_j est la densité d'une loi normale de paramètre $(\mu_j, \sigma^2 \mathrm{Id}_2)$. Les 4 centres sont

$(1,1)$, $(-1,1)$, $(-1,-1)$, $(1,-1)$ et la variance est précisée dans les différents scénarios. Bien entendu, obtenir des réalisations *i.i.d.* de π_θ se fait sans difficulté : cet exemple a pour but d'illustrer les limites de l'échantillonneur de Gibbs dans un contexte de sélection de modèle sur un cas simple. L'échantillonneur de Gibbs alterne entre les étapes (i) $x \,|\, j \sim g_j$ et (ii) $j \,|\, x \propto g_j(x)$ pour simuler une chaîne $\{(J_k, X_k),\, k \in \mathbb{N}\}$ π_θ-réversible.

La Figure 31 illustre une réalisation de cette chaîne dans 4 scénarios ayant différentes variances. On remarque que plus la variance diminue, moins la loi des échantillons simulés correspond à π_θ : plus les modèles sont « éloignés » les uns des autres moins l'échantillonneur de Gibbs permet de changer de classe comme le montre la Figure 31(b). Le Tableau 5 indique la moyenne des probabilités de changement de classes.

σ^2	0.125	0.1	0.075	0.05
$\mathbb{P}[J_{k+1} \neq J_k]$	$2.3\,10^{-3}$	$8.6\,10^{-4}$	$8.5\,10^{-5}$	$1.2\,10^{-6}$

TABLEAU 5 – Moyenne empirique des probabilités de changement de classe en fonction de σ^2

Par analogie, l'échantillonneur de Gibbs implémentable pour la simulation de $\pi_{\hat{\theta}_i}(\,\cdot \,|\, y_k)$ dans le cas des modèles déformables peut se retrouver « piégé » dans certains modèles. Ce phénomène s'accentue lorsque la dimension du paramètre augmente : le passage d'un modèle à un autre est encore plus délicat.

Pour résoudre ce problème, un autre échantillonneur a été proposé dans [AK10]. Remarquant que la loi *a posteriori* peut s'écrire :

$$\pi_\theta(j, \beta \,|\, y) = \pi_\theta(\beta \,|\, j, y)\pi(j \,|\, y)\;, \quad \text{où} \quad \pi_\theta(j \,|\, y) = \int_{\mathsf{B}} \pi_\theta(j, \beta \,|\, y)\mathrm{d}\beta\;,$$

un échantillon *exacte* de la loi $\pi_\theta(\,\cdot \,|\, y)$ peut être obtenu en simulant (i) $j \propto \pi_\theta(j \,|\, y)$ et (ii) $\beta \sim \pi_\theta(\,\cdot \,|\, j, y)$. Malheureusement, aucune de ces deux lois n'est simulable exactement et il faut recourir à des approximations. Pour toute fonction de densité h, nous avons

$$\pi_\theta(j \,|\, y) \propto \left\{ \mathbb{E}_\theta \left[\frac{h(\beta)}{f_\theta(J, \beta, Y)} \,\middle|\, y, j \right] \right\}^{-1}$$

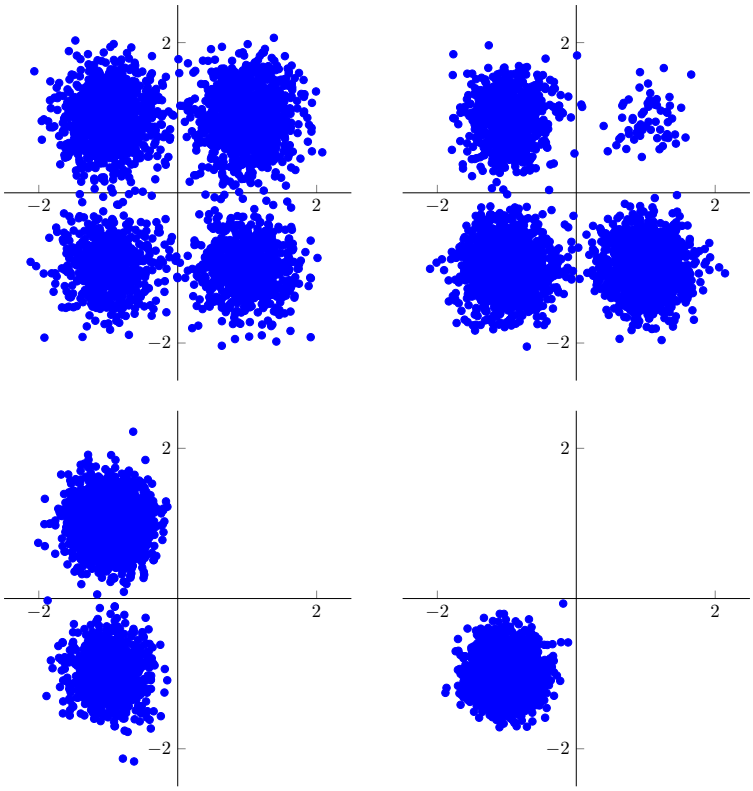
et un estimateur $\hat{\pi}^m_{\theta,j,y}$ de $\pi_\theta(j \,|\, y)$ peut être obtenu par un algorithme MCMC en simulant une chaîne de Markov *auxiliaire* $\{\tilde{\beta}^\ell,\, \ell \leq m\}$, $\pi_\theta(\,\cdot \,|\, j, y)$-réversible. Un noyau de Metropolis-within-Gibbs, similaire à celui utilisé dans la Procédure 3-1, permet d'après le TCL D.3, d'avoir un estimateur $\tilde{\pi}^m_{\theta,j,y}$ aussi proche que l'on veut de $\pi_\theta(j \,|\, y)$.

Cette étape préliminaire permet de simuler la classe de y, $J \sim (\tilde{\pi}^m_{\theta,1,y}, \ldots, \tilde{\pi}^m_{\theta,C,y})$ et l'étape (ii) est réalisée par m transitions d'un noyau de Markov du même type que précédemment à partir d'un paramètre de déformation $\beta_0 \in \mathsf{B}$ quelconque. L'algorithme d'apprentissage résultant [AK10] est résumé dans la Procédure 4.

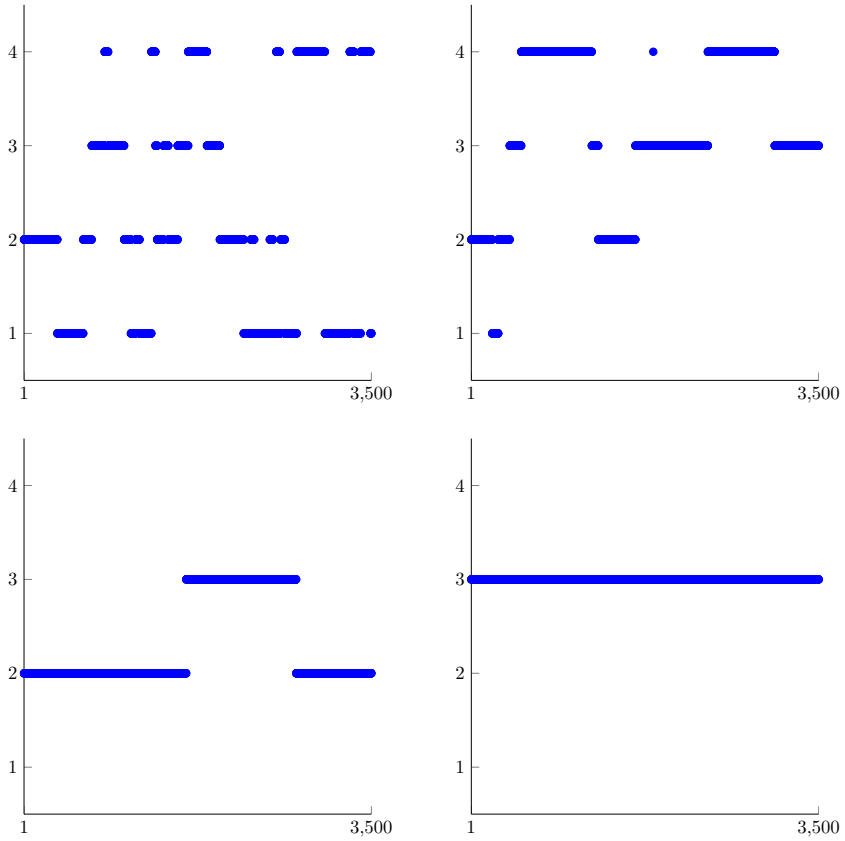
A la différence du SAEM-MCMC (détaillé dans la Procédure 3), cet algorithme ne couple pas l'estimation des paramètres avec les itérations du MCMC. A chaque itération, les variables manquantes sont simulées de façon indépendante suivant une loi approchée $\tilde{\pi}^m(\,\cdot \,|\, y)$ telle que pour tout $(I, A) \in \mathcal{P}(\{1, \ldots, C\}) \otimes \mathcal{B}$:

$$\sum_{j \in I} \int_A \tilde{\pi}^m_\theta(j, \mathrm{d}\beta \,|\, y) = \sum_{j \in I} \tilde{\pi}^m_{\theta,j,y} K^m_{\theta,j,y}(\beta_0, A) \underset{m \to \infty}{\longrightarrow} \sum_{j \in I} \int_A \pi_\theta(j, \mathrm{d}\beta \,|\, y) \quad \text{presque-sûrement .}$$

Échantillonner cette loi approchée représente une charge de calcul très importante puisqu'elle nécessite la simulation de $C+1$ chaînes de Markov de longueur m : chaque itération



(a) Réalisations de $\{X_k \in \mathbb{R}^2,\, k \leq 3500\}$



(b) Réalisations de $\{J_k \in \{1, \ldots, 4\},\, k \leq 3500\}$

FIGURE 31 – Simulation de la chaîne $\{(J_k, X_k),\, k \leq 3500\}$ par l'échantillonneur de Gibbs pour 4 variances : $\sigma^2 = 0.125$ en haut à gauche, $\sigma^2 = 0.1$ en haut à droite, $\sigma^2 = 0.075$ en bas à gauche et $\sigma^2 = 0.05$ en bas à droite

Procédure 4 Transition $\hat{\theta}_i \to \hat{\theta}_{i+1}$ par l'algorithme SAEM-multicomponent

Entrées : L'estimateur courant $\hat{\theta}_i \in \Theta$, l'approximation stochastique $\hat{s}_i \in \mathsf{S}$ et les observations $(y_1, \ldots, y_n) \in \mathsf{Y}^n$.

1 - **approximations des poids *a posteriori* :** Pour tout $k \in \{1, \ldots, n\}$ et pour tout $j \in \{1, \ldots, C\}$, simuler une chaîne de Markov *auxiliaire*

$$\tilde{\beta}_{j,k}^{\ell} \sim K_{\hat{\theta}_i, j, y_k}(\tilde{\beta}_{j,k}^{\ell-1}, \cdot) \ ,$$

et calculer l'approximation

$$\tilde{\pi}_{\theta,j,y}^{m} \propto \left\{ \frac{1}{m} \sum_{\ell=1}^{m} \frac{h(\tilde{\beta}_{j,k}^{\ell})}{f_\theta(j, \tilde{\beta}_{j,k}^{\ell}, y_k)} \right\}^{-1} \ .$$

2 - **simulation des données manquantes :** Pour tout $k \in \{1, \ldots, n\}$, simuler J_k puis β_k

$$j_k \sim (\tilde{\pi}_{\hat{\theta}_i,1,y_k}^{m}, \ldots, \tilde{\pi}_{\hat{\theta}_i,C,y_k}^{m}) \ , \quad \beta_k \sim K_{\hat{\theta}_i, j_k, y_k}^{m}(\beta_0, \cdot) \ ,$$

3 - **approximation stochastique :** Mettre à jour l'approximation stochastique

$$\hat{s}_{i+1} = \hat{s}_i + \gamma_i (\sum_{k=1}^{n} S(j_k, \beta_k, y_k) - \hat{s}_i) \ ,$$

4 - **maximisation :** Calculer le nouvel estimateur $\hat{\theta}_{i+1}$ à partir de $\hat{s}_{i+1}$ comme dans (102)

5 - **Sorties :** L'état suivant $\hat{\theta}_{i+1}$ et $\hat{s}_{i+1}$.

du SAEM-multicomponent nécessite donc la simulation de $n(C+1)$ chaîne de Markov, soit $2dn(C+1)$ étapes de Metropolis-Hastings ($\dim(\beta) = 2d$). Cette approche apparaît donc peu optimale puisqu'à la différence du SAEM-MCMC , ces échantillons ne sont pas réutilisées lors des itérations suivantes du SAEM-multicomponent , chaque chaîne étant initialisée par un paramètre $\beta_0 \in \mathsf{B}$ quelconque. Toutefois, le SAEM-multicomponent permet en pratique de traiter le cas des modèles de mélange à la différence du SAEM-MCMC .

Bien qu'impliquant la simulation de chaînes de Markov, le SAEM-multicomponent est plus proche du SAEM que du SAEM-MCMC : il peut être interprété comme une version bruitée du SAEM proposé par [DLM99]. Le bruit induit par la simulation des données manquantes suivant une loi approchée ne remet toutefois pas en cause la convergence de l'algorithme qui a été prouvée dans le contexte des mélanges à prototype déformable dans [AK10].

Sélection de modèles et MCMC

L'échantillonnage de la loi *a posteriori* $\pi_\theta(\cdot \,|\, y)$ soulève le problème de la simulation par un algorithme MCMC des lois mixtes dont un paramètre discret, le paramètre de classe, sélectionne un modèle dans une liste de modèles possibles, et l'autre (typiquement continu) est une variable explicative du modèle sélectionné.

Une telle loi π est définie sur l'espace X suivant

$$\mathsf{X} = \left\{ (j, \beta_j), \ j \in \mathsf{J}, \ \beta_j \in \mathsf{B}_j \right\} ,$$

où $\mathsf{J} = \{1, \ldots, C\}$ et pour tout $j \in \mathsf{J}$, $\mathsf{B}_j \subseteq \mathbb{R}^{d_j}$, $d_j > 0$. On définit de plus les tribus $\mathcal{J} = \mathcal{P}(\mathsf{J})$ et pour tout $j \in \mathsf{J}$, $\mathcal{B}_j = \mathcal{B}(\mathsf{B}_j)$ associées aux espaces J et $\mathsf{B}_1, \ldots, \mathsf{B}_C$. On munit X de la tribu $\mathcal{X}$

$$\mathcal{X} = \bigcup_{I \in \mathcal{J}} \mathcal{X}_I \quad \text{où} \quad \mathcal{X}_I = \bigcup_{j \in I} \{j, \mathcal{B}_j\} \, .$$

Ainsi, pour tout $A \in \mathcal{X}$, il existe $I \in \mathcal{J}$ et pour tout $j \in I$, $A_j \in \mathcal{B}_j$ tels que

$$A = \bigcup_{j \in I} \{j, A_j\} \, . \tag{103}$$

Le problème est donc de simuler des réalisations d'une variable aléatoire X définie sur l'espace de probabilité $(\mathsf{X}, \mathcal{X}, \pi)$.

Remarque D.10. Échantillonneur de Gibbs sur l'espace $(\mathsf{X}, \mathcal{X})$

- Dans le cas particulier où pour tout $j \in \mathsf{J}$, $\mathsf{B}_j = \mathsf{B}$, il est techniquement possible d'implémenter un échantillonneur de Gibbs pour simuler une chaîne de Markov π-réversible. Toutefois, pour des raisons remarquées dans le cas des mélanges de modèle à prototype déformable et dans l'exemple D.9, cette solution n'est pas envisageable.
- Dans le cas général, la structure de l'espace d'état X ne permet pas d'implementer un échantillonneur de Gibbs. En effet, l'étape de simulation de la variable de classe J s'écrirait alors conditionnellement à un paramètre $\beta_j \in \mathsf{B}_j$

  $$J' \sim \pi(1, \beta_j), \ldots, \pi(C, \beta_j) \, ,$$

  et ferait intervenir le calcul des poids $\pi(j', \beta_j)$ qui ne sont pas définis pour tout $j' \neq j$ car $(j', \beta_j) \notin \mathsf{X}$.

L'algorithme de Carlin et Chib [CC95] est une méthode partageant des similarités avec la Data-Augmentation (Cf. Remarque D.7) qui permet de simuler une chaîne de Markov $\{X_k = (J_k, \beta_k),\, k \in \mathbb{N}\}$ sur l'espace X admettant π comme distribution stationnaire. Considérons l'espace mesurable étendu $(\tilde{\mathsf{X}}, \tilde{\mathcal{X}})$

$$\begin{cases} \tilde{\mathsf{X}} = \mathsf{J} \times \mathsf{B}_1 \times \cdots \times \mathsf{B}_C \, , \\ \tilde{\mathcal{X}} = \mathcal{J} \otimes \mathcal{B}_1 \otimes \cdots \otimes \mathcal{B}_C \, , \end{cases}$$

et la distribution $\tilde{\pi}$ définie pour tout $\tilde{A} = (I, \tilde{A}_1, \ldots, \tilde{A}_C) \in \tilde{\mathcal{X}}$ par

$$\tilde{\pi}(\tilde{A}) = \sum_{j \in I} \pi(j, \tilde{A}_j) \prod_{\ell \neq j} \zeta_\ell(\tilde{A}_\ell) \, ,$$

où pour tout $j \in \mathsf{J}$, ζ_j est une densité simulable sur $(\mathsf{B}_j, \mathcal{B}_j)$, dénommée *pseudo-prior* ou *linking density* dans [CC95]. Les pseudo-priors ne sont pas déterminés par la distribution π et ces derniers peuvent être choisi comme n'importe quelle densité simulable sur $(\mathsf{B}_j, \mathcal{B}_j)$. Toutefois, en pratique comme en théorie, le choix des pseudo-priors s'avère être de première importance ; voir [CC95], Exemple D.11 et Chapitre IV.

Contrairement à l'espace X, $\tilde{\mathsf{X}}$ est un espace produit, sur lequel un échantillonneur de Gibbs est implémentable pourvu que les lois conditionnelles de densité

$$\begin{cases} 1. \quad \tilde{\pi}(j \,|\, \beta_1, \ldots, \beta_C) \propto \pi(j, \beta_j) \prod_{\ell \neq j} \zeta_\ell(\beta_\ell) \, , \\ 2. \quad \tilde{\pi}(\beta_j \,|\, j, \beta_{-j}) = \pi(\beta_j \,|\, j) \, , \\ 3. \quad \tilde{\pi}(\beta_\ell \,|\, j, \beta_{-\ell}) = \zeta_\ell(\beta_\ell) \, , \quad \forall\, \ell \neq j \, , \end{cases}$$

soient simulables.

Notons que 1. est simulable car c'est une loi discrète et que 3. est simulable par hypothèse sur les pseudo-priors. Ainsi, sous l'hypothèse que $\pi(\cdot \,|\, j)$ est simulable, il est possible de construire une chaîne de Markov *étendue* $\{\tilde{X}_k = (J^{(k)}, \beta_1^{(k)}, \ldots, \beta_C^{(k)}),\ k \in \mathbb{N}\}$ $\tilde{\pi}$-réversible, en utilisant un échantillonneur de Gibbs (pour des raisons de lisibilité, l'indice des itérations de la chaîne étendue est en exposant). Différentes implémentations de l'algorithme de Carlin et Chib sont possibles suivant l'ordre avec lequel les $C+1$ étapes élémentaires de l'échantillonneur de Gibbs sont regroupées.

Une première idée est d'alterner l'échantillonnage des données auxiliaires et des données d'intérêt à la manière d'un algorithme de Data Augmentation (Cf. Figure 30). Rappelons que pour tout $\tilde{X} = (j, \beta_1, \ldots, \beta_C) \in \tilde{\mathsf{X}}$, $(j, \beta_j) \in \mathsf{X}$ sont les données d'intérêts tandis que les paramètres $\beta_{-j} \in \mathsf{B}_{-j}$ sont considérées comme des variables auxiliaires. La figure 32 illustre cette approche : la chaîne alterne entre le noyau $K_1^{\mathtt{CC1}}$ qui regroupe la simulation des variables auxiliaires et le noyau $K_2^{\mathtt{CC1}}$ qui regroupe la simulation des paramètres d'intérêts.

$$\begin{pmatrix} J^{(0)} = j \\ \beta_j^{(0)} = b_j \end{pmatrix} \xrightarrow{K_1^{\mathtt{CC1}}} \begin{pmatrix} J^{(1)} = j \\ \beta_j^{(1)} = b_j \\ \\ \beta_{-j}^{(1)} = b_{-j} \end{pmatrix} \xrightarrow{K_2^{\mathtt{CC1}}} \begin{pmatrix} J^{(2)} = j' \\ \beta_{j'}^{(2)} = b'_{j'} \\ \\ \beta_{-j'}^{(2)} = b_{-j'} \end{pmatrix} \xrightarrow{K_1^{\mathtt{CC1}}} \begin{pmatrix} J^{(3)} = j' \\ \beta_{j'}^{(3)} = b'_{j'} \\ \\ \beta_{-j'}^{(3)} = b'_{-j'} \end{pmatrix} \longrightarrow \cdots$$

Figure 32 – Carlin et Chib : un premier schéma de transition possible

$$\begin{cases} K_1^{\mathtt{CC1}}(\tilde{x}, \mathrm{d}\tilde{x}') = \delta_{(j,\beta_j)}(j', \mathrm{d}\beta'_j) \prod_{\ell \neq j} \zeta_\ell(\mathrm{d}\beta'_\ell)\ , \\ \\ K_2^{\mathtt{CC1}}(\tilde{x}, \mathrm{d}\tilde{x}') = \tilde{\pi}(j' \,|\, \beta_1, \ldots, \beta_C) \pi(\mathrm{d}\beta'_{j'} \,|\, j') \prod_{\ell \neq j'} \delta_{\beta_\ell}(\mathrm{d}\beta'_\ell)\ . \end{cases}$$

$K_1^{\mathtt{CC1}}$ est un regroupement de $C-1$ étapes de Gibbs élémentaires et indépendantes (les densités 3. ci dessus) : ce noyau est donc $\tilde{\pi}$-réversible et en conséquent invariant par $\tilde{\pi}$. En revanche, $K_2^{\mathtt{CC1}}$ n'est pas $\tilde{\pi}$-réversible mais il peut être prouvé qu'il est invariant par $\tilde{\pi}$. Alternant entre deux noyaux invariants par $\tilde{\pi}$, la chaîne de Markov $\{J^{(k)}, \beta_1^{(k)}, \ldots, \beta_C^{(k)},\ k \in \mathbb{N}\}$ admet $\tilde{\pi}$ comme distribution stationnaire. De la même manière qu'un algorithme de Data Augmentation, la suite de variables aléatoires $\{J^{(2k)}, \beta_{J^{(2k)}}^{(2k)},\ k \in \mathbb{N}\}$ est une chaîne de Markov admettant π comme distribution stationnaire.

En effet, en régime stationnaire $(J^{(k)}, \beta_1^{(k)}, \ldots, \beta_C^{(k)}) \sim \tilde{\pi}$ et pour tout $A \in \mathcal{X}$ il vient avec (103),

$$\begin{aligned} \mathbb{P}[(J^{(k)}, \beta_{J^{(k)}}^{(k)}) \in A] \quad &= \sum_{j \in I} \int_{A_j} \mathbb{P}[J^{(k)} = j, \beta_j^{(k)} \in \mathrm{d}\beta_j] \\ &= \sum_{j \in I} \int_{\mathsf{B}_1} \cdots \int_{A_j} \cdots \int_{\mathsf{B}_C} \mathbb{P}[J^{(k)} = j, \beta_1^{(k)} \in \mathrm{d}\beta_1, \ldots, \beta_C^{(k)} \in \mathrm{d}\beta_C] \\ &= \sum_{j \in I} \int_{A_j} \pi(j, \mathrm{d}\beta_j) \prod_{\ell \neq j} \int_{\mathsf{B}_\ell} \zeta_\ell(\mathrm{d}\beta_\ell) = \pi(A)\ . \end{aligned}$$

Toutefois, la chaîne $\{\tilde{X}_k,\ k \in \mathbb{N}\}$ simulée suivant ce schéma n'est pas un exemple de Data Augmentation. En effet, le noyau de simulation des variables d'intérêt par un algorithme

de Data Augmentation serait

$$K_2^{\mathtt{DA}}(\tilde{x}, \mathrm{d}\tilde{x}') = \tilde{\pi}(j', \mathrm{d}\beta'_{j'} \,|\, \beta_{-j'}) \prod_{\ell \neq j'} \zeta_\ell(\mathrm{d}\beta_\ell) \;,$$

qui est différent de $K_2^{\mathtt{CC1}}$: $K_2^{\mathtt{CC1}}$ utilise tous les paramètres de déformations $(\beta_1, \ldots, \beta_C)$ pour simuler J' alors que $\beta_{j'}$ n'apparaît pas dans $K_2^{\mathtt{DA}}$.

Afin de simuler une chaîne de Markov étendue $\tilde{\pi}$-réversible, il est possible de décomposer les deux étapes regroupées dans $K_2^{\mathtt{CC1}}$: $\{\tilde{X}_k,\, k \in \mathbb{N}\}$ alterne alors entre 3 noyaux de transitions,

- $K_1^{\mathtt{CC2}}$ simule le paramètre du modèle courant suivant la loi conditionnelle

$$K_1^{\mathtt{CC2}}(\tilde{x}, \mathrm{d}\tilde{x}') = \tilde{\pi}(\mathrm{d}\beta'_j \,|\, j, \beta_{-j}) \prod_{\ell \neq j} \delta_{\beta_\ell}(\mathrm{d}\beta'_\ell)\delta_j(j') \;,$$

- $K_2^{\mathtt{CC2}} = K_1^{\mathtt{CC1}}$ simule les variables auxiliaires,
- $K_3^{\mathtt{CC2}}$ simule la variable de classe

$$K_3^{\mathtt{CC2}}(\tilde{x}, \mathrm{d}\tilde{x}') = \tilde{\pi}(j' \,|\, \beta_1, \ldots, \beta_C) \prod_{\ell \in \mathtt{J}} \delta_{\beta_\ell}(\mathrm{d}\beta'_\ell) \;.$$

$$\begin{pmatrix} J^{(0)} = j \\ \\ \\ \end{pmatrix} \xrightarrow{K_1^{\mathtt{CC2}}} \begin{pmatrix} J^{(1)} = j \\ \beta_j^{(1)} = b_j \\ \\ \end{pmatrix} \xrightarrow{K_2^{\mathtt{CC2}}} \begin{pmatrix} J^{(2)} = j \\ \beta_j^{(2)} = b_j \\ \\ \beta_{-j}^{(2)} = b_{-j} \end{pmatrix} \xrightarrow{K_3^{\mathtt{CC2}}} \begin{pmatrix} J^{(3)} = j' \\ \beta_{j'}^{(3)} = b_{j'} \\ \\ \beta_{-j'}^{(3)} = b_{-j'} \end{pmatrix} \xrightarrow{K_1^{\mathtt{CC2}}} \begin{pmatrix} J^{(4)} = j' \\ \beta_{j'}^{(4)} = b'_{j'} \\ \\ \beta_{-j'}^{(4)} = b_{-j'} \end{pmatrix} \longrightarrow \cdots$$

FIGURE 33 – Carlin et Chib : un second schéma de transition $\tilde{\pi}$-réversible

Les noyaux de transition $K_1^{\mathtt{CC2}}$, $K_2^{\mathtt{CC2}}$ et $K_3^{\mathtt{CC2}}$ sont $\tilde{\pi}$-réversibles car ils correspondent aux noyaux de Gibbs élémentaires simulant respectivement la variable explicative du modèle courant, les variables auxiliaires et la variable de classe ; la chaîne de Markov $\{\tilde{X}_k,\, k \in \mathbb{N}\}$ résultante admet donc $\tilde{\pi}$ comme distribution stationnaire. Par un raisonnement analogue à celui proposé pour la chaîne alternant entre $K_1^{\mathtt{CC1}}$ et $K_2^{\mathtt{CC1}}$, la suite de variables aléatoires $\{J^{(3k)}, \beta_{J^{(3k)}}^{(3k)},\, k \in \mathbb{N}\}$ est une chaîne de Markov admettant π comme distribution stationnaire.

Exemple D.11. Nous implémentons un échantillonneur de Carlin et Chib pour *résoudre* le problème de l'exemple jouet du mélange de 4 gaussiennes (Ex. D.9) dans le cas où $\sigma^2 = 0.05$. La figure 34 illustre une réalisation de la chaîne $\{X_k,\, k \in \mathbb{N}\}$ obtenue par l'algorithme de Carlin et Chib (Transition 33). Une loi gaussienne centrée en $(0, 0)$ et de variance 1 est utilisée pour les 4 pseudo-priors.

Le Tableau 6 compare les probabilités marginales empiriques des différentes classes obtenues par l'échantillonneur de Gibbs et par l'échantillonneur de Carlin et Chib avec différents choix de pseudo-priors. La figure 35 illustre les variances empiriques de l'estimateur de $\mathbb{P}[J = 1]$ obtenu à partir de 100 réalisations de $\{X_k,\, k \leq 5000\}$ des différents algorithmes MCMC comparés dans le Tableau 6.

Cet exemple montre l'influence des pseudo-priors sur la convergence de l'échantillonneur et confirme l'intérêt de la méthode de Carlin et Chib par rapport à l'échantillonneur de Gibbs quand celui-ci est implémentable.

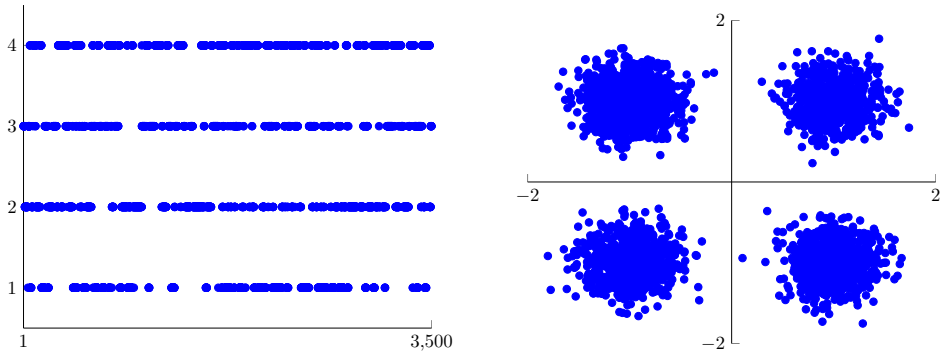


Figure 34 – Échantillons de la chaîne $\{X_k,\, k \in \mathbb{N}\}$ obtenue par la méthode de Carlin et Chib : variable de classe à gauche et paramètre explicatif à droite

classes/MCMC	1	2	3	4
Gibbs (Ex. D.9)	0	0	1	0
CC avec $\zeta_j = g_j$	0.26	0.24	0.25	0.25
CC avec $\zeta_j = \mathcal{N}(0,1)$	0.24	0.27	0.23	0.26
CC avec $\zeta_j = \mathcal{N}(0,0.2)$	0.44	0.17	0.25	0.14

Tableau 6 – Probabilité empirique des 4 modèles au bout de 3500 itérations de différents MCMC (Gibbs ou CC : Carlin et Chib)

Remarque D.12. Il est donc possible d'adapter l'algorithme SAEM-MCMC [KL04] au cas des modèles de mélanges, en substituant le noyau de Metropolis-within-Gibbs utilisé dans [AKT10a] par les noyaux $K_1^{\texttt{CC1}} K_2^{\texttt{CC1}}$ ou $K_1^{\texttt{CC2}} K_2^{\texttt{CC2}} K_3^{\texttt{CC2}}$.

D .4 Comparaison entre algorithmes MCMC

Lorsque différents algorithmes MCMC permettent d'approcher une même intégrale, la question du choix de la méthode la plus *adaptée* en fonction des contraintes de l'utilisateur (temps de convergence, précision de l'approximation, etc...) a une importance primordiale.

Soit K un noyau de transition sur $(\mathsf{X}, \mathcal{X})$ vérifiant un TCL (95). Dans le contexte des MCMC, la variance asymptotique dépend aussi du noyau de transition K et est notée $v(f, K, \pi)$

$$v(f, K, \pi) = \lim_{n\to\infty} \frac{1}{n} \mathbb{E}\left[\left\{\sum_{k=1}^{n} (f(X_k) - \pi f)\right\}^2\right] ,$$

où $\{X_k,\, k \in \mathbb{N}\}$, est une chaîne de Markov de noyau de transition K. La variance asymptotique est un critère de référence pour mesurer l'efficacité d'un algorithme MCMC [Pes73, Tie95, MG99, Mir01, ML09] Soit K_0 et K_1 deux noyaux de transitions sur $(\mathsf{X}, \mathcal{X})$ admettant π comme mesure de stationnarité. On dit que K_1 est au moins aussi efficace que K_0 et on note $K_1 \succeq_{\mathrm{E}} K_0$ si pour toutes fonctions $f \in \mathcal{L}^2(\pi)$ telles que pour $i \in \{0, 1\}$ $v(f, K_i, \pi) < \infty$

$$v(f, K_1, \pi) \leq v(f, K_0, \pi) \,. \tag{104}$$

Étant donné deux noyaux, il est généralement difficile de prouver analytiquement l'inégalité 104 de façon directe. En effet,

- l'inégalité (104) doit être vérifiée pour un large ensemble de fonctions,

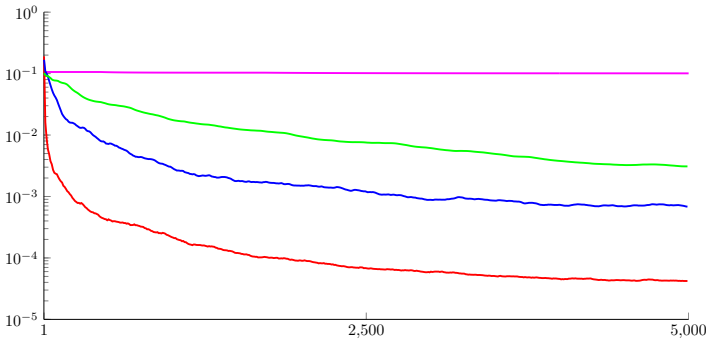


FIGURE 35 – Variance empirique de l'estimateur de $\mathbb{P}[J=1]$ obtenu par l'échantillonneur de Gibbs (magenta) et par la méthode de Carlin et Chib avec différents pseudo-priors : $\zeta_j = g_j$ (rouge), $\zeta_j = \mathcal{N}(0,1)$ (bleu) et $\zeta_j = \mathcal{N}(0,0.2)$ (vert) (100 simulations)

- la dépendance entre les différents états de la chaîne nécessite le calcul des covariances $\mathrm{Cov}(f(X_i), f(X_j))$ pour $j \neq i$, qui font généralement intervenir des séries dont une expression explicite peut être difficile à obtenir en pratique.

D'autres relations d'ordres partiels sur l'ensemble des noyaux de transition, plus faibles que (104), ont été proposées [Pes73, Tie95, MG99]. Elles ont comme intérêt d'être généralement plus simples à établir que (104) et impliquent, sous certaines conditions, l'ordre des variances asymptotiques.

L'ordre de Peskun [Pes73] (noté $\succeq_{\mathrm{P}}$), initialement défini pour des chaînes de Markov à espace d'état discret puis généralisé par Tierney [Tie95] pour les chaînes de Markov définies sur un espace d'état général, est un ordre partiel *hors-diagonal*. K_0 et K_1 étant deux noyaux admettant π comme distribution stationnaire, alors $K_1 \succeq_{\mathrm{P}} K_0$ si pour tout $A \in \mathcal{X}$

$$K_1(x, A\backslash\{x\}) \geq K_0(x, A\backslash\{x\}) \;, \quad \text{pour } \pi\text{-presque tous les } \; x \in \mathsf{X} \;. \tag{105}$$

Cet ordre compare les probabilités de K_0 et K_1 de se déplacer dans l'espace d'état : ainsi K_1 sera *meilleur* que K_0 au sens de Peskun si il explore plus efficacement l'espace d'état. L'ordre de Peskun a notamment permis de montrer le caractère optimal de l'échantillonneur de Metropolis-Hastings (97) par rapport à n'importe quel autre noyau de transition ayant la même structure mais une probabilité d'acceptation (98) différente [Pes73]. Le Théorème 4 de Tierney [Tie95] étend le Théorème 2.1.1 de Peskun [Pes73] à un cadre général et affirme que si K_0 et K_1 sont π-réversibles, alors

$$K_1 \succeq_{\mathrm{P}} K_0 \Longrightarrow K_1 \succeq_{\mathrm{E}} K_0 \;.$$

La démonstration du Théorème 4 de Tierney utilise un résultat sur la décomposition spectrale des opérateurs auto-adjoints, énonçant que pour une chaîne de Markov homogène de noyau π-reversible K

$$v(f, K, \pi) = \Big\langle f, (I-K)^{-1}(I+K)f \Big\rangle \;. \tag{106}$$

Rappelons que pour toutes fonctions $(f,g) \in \mathcal{L}^2(\pi)^2$, l'application

$$\langle f, g \rangle = \mathbb{E}_\pi[f(X)g(X)] = \int_{\mathsf{X}} f(x)g(x)\pi(\mathrm{d}x)$$

définit un produit scalaire sur $\mathcal{L}^2(\pi)$.

Considérant un noyau $K(\alpha)$ défini comme le mélange de deux noyaux π-réversibles K_0 et K_1 tels que $K_0 \succeq_{\mathrm{P}} K_1$ et pour tout $\alpha \in [0\,;1]$

$$K^{(\alpha)} = \alpha K_0 + (1-\alpha)K_1 \ ,$$

la preuve est achevée en montrant que la fonction $\alpha \to v(f, K^{(\alpha)}, \pi)$ est croissante sur $[0;1]$.

Toutefois, une large gamme de noyaux de transition sur $(\mathsf{X}, \mathcal{X})$ ne peuvent pas être comparés par l'ordre de Peskun. En particulier, les noyaux absolument continus par rapport à la mesure de Lebesgue ν

$$K(x, \mathrm{d}x') = k(x, x')\nu(\mathrm{d}x') \ ,$$

ont une probabilité nulle de rester sur la diagonale, *i.e.* dans le même état, et ne sont donc pas comparable par Peskun. Cette situation concerne notamment les échantillonneurs de Gibbs.

L'ordre de covariance de degré 1 (noté $\succeq_{\mathrm{C}}$) introduit par [MG99] permet de s'affranchir de cette contrainte. La covariance de degré 1 pour la chaîne de Markov $\{X_k,\, k \in \mathbb{N}\}$ de noyau de transition K s'écrit pour toute fonction $f \in \mathcal{L}^2(\pi)$

$$\mathrm{Cov}(f(X_k), f(X_{k+1})) = \iint_{\mathsf{X}} f(x)f(x')\pi(\mathrm{d}x)K(x, \mathrm{d}x') = \langle f, Kf\rangle \ .$$

Ainsi, K_1 domine K_0 au sens de l'ordre de covariance, $K_1 \succeq_{\mathrm{C}} K_0$, si

$$\forall f \in \mathcal{L}^2(\pi) \quad \langle f, K_1 f\rangle \leq \langle f, K_0 f\rangle \ ,$$

ou de façon équivalente si $K_0 - K_1$ est un opérateur positif sur $\mathcal{L}^2(\pi)$.

D'après le Lemme 3 de [Tie95], l'ordre de covariance est un ordre plus faible que celui de Peskun : si K_0 et K_1 admettent π comme distribution stationnaire, alors

$$K_1 \succeq_{\mathrm{P}} K_0 \Longrightarrow K_1 \succeq_{\mathrm{C}} K_0 \ .$$

Notons toutefois que dans le cas où les noyaux sont π-réversibles, le Théorème 6 de [ML09] prouve que les deux ordres sont équivalents

$$K_1 \succeq_{\mathrm{P}} K_0 \Longleftrightarrow K_1 \succeq_{\mathrm{C}} K_0 \ .$$

Ces outils permettent d'établir des comparaisons entre différents algorithmes MCMC de façon relativement simple sans recourir à des développements théoriques approfondis sur les chaînes de Markov. Cependant en pratique, de nombreux MCMC ne sont pas π-réversibles : typiquement plusieurs étapes de l'échantillonneur de Gibbs ou de Metropolis-within-Gibbs sont agrégés et le noyau composé perd sa réversibilité rendant inapplicable les précédents résultats.

Pour cette raison, l'algorithme de Carlin et Chib ne peut pas être comparé à d'autres méthodes MCMC. En effet, les résultats précédents ne s'appliquant qu'à des chaînes homogènes, le noyau de transition sur $(\mathsf{X}, \mathcal{X})$ à considérer s'écrit pour tout $x = (j, \beta_j)$

$$K(x, \mathrm{d}x') = \int_{\mathsf{B}_1} \cdots \int_{\mathsf{B}_{j-1}} \int_{\mathsf{B}_{j+1}} \cdots \int_{\mathsf{B}_C} \prod_{\ell \neq j} \zeta_\ell(\mathrm{d}\beta_\ell)\tilde{\pi}(j' \,|\, \beta_1, \ldots, \beta_C)\pi(\mathrm{d}\beta'_{j'} \,|\, j')$$

et n'est pas π-réversible.

Pourtant, il semble exister une corrélation entre la faculté de la chaîne à visiter équitablement les différentes classes (Tableau 6) et la variance asymptotique de la chaîne associée (Courbes 35), ce qui laisse penser qu'un résultat liant l'ordre de Peskun pour les noyaux de sélection de classes et l'efficacité du MCMC existe.

Résumé de la contribution

Notre contribution s'articule en trois points :

(i) Un résultat théorique général permettant de comparer des chaînes de Markov inhomogènes,

(ii) Une méthodologie utilisant ce résultat pour étudier l'efficacité de nombreux MCMC existants sans nécessiter de développement théorique conséquent,

(iii) La mise au point d'un algorithme MCMC de type Pseudo-Marginal *exacte* et plus efficace que certains algorithmes Pseudo-Marginaux existants.

Résultat théorique

Nous prouvons dans le Théorème IV.4, [M3], une extension du Théorème 4 de Tierney [Tie95] pour les chaînes de Markov inhomogènes. Plus précisément, pour deux chaînes de Markov $\{X_k^{(i)},\, k \in \mathbb{N}\}$, $i \in \{0,1\}$, alternant entre deux noyaux P_i et Q_i π-réversibles tels que :

$$P_1 \succeq_{\mathrm{P}} P_0 \quad \text{et} \quad Q_1 \succeq_{\mathrm{P}} Q_0 \tag{107}$$

alors pour les fonctions $f \in \mathcal{L}^2(\pi)$ vérifiant

$$\sum_{k=1}^{\infty} \left(|\mathrm{Cov}(f(X_0^{(i)}), f(X_k^{(i)}))| + |\mathrm{Cov}(f(X_1^{(i)}), f(X_{k+1}^{(i)}))| \right) < \infty \,, \tag{108}$$

nous prouvons que :

$$v_1(f) \leq v_0(f) \quad \text{où} \quad v_i(f) = \lim_{n\to\infty} \frac{1}{n} \mathrm{Var} \left(\sum_{k=0}^{n-1} f(X_k^{(i)}) \right) \quad (i \in \{0,1\}) \,.$$

Notons que le Théorème 4 de Tierney est valable pour l'ensemble des fonctions $\{f \in \mathcal{L}^2(\pi), v_i(f) < \infty\}$ tandis que notre Théorème IV.4 peut sembler plus restrictif par l'hypothèse (108). Toutefois, l'expression de $v_i(f)$ donné dans le Lemme IV.29 est proche de la condition (108), aux valeurs absolues près. Par conséquent, l'ensemble des fonctions vérifiant (108) est certes inclus mais très proche de l'ensemble $\{f \in \mathcal{L}^2(\pi), v_i(f) < \infty\}$: l'hypothèse (108) peut être considérée comme valable pour un très grand nombre de fonctions qui sont concernées par le Théorème 4 de Tierney.

Le résultat sur la décomposition spectrale des opérateurs auto-adjoints 106 utilisé par Tierney pour prouver le Théorème 4 n'est pas applicable dans notre contexte car la chaîne n'est pas homogène. Notre preuve est basée sur la théorie des opérateurs et montre que lorsque l'on considère la chaîne de Markov de noyau de transition

$$R_n^{(\alpha)} = (\alpha P_1 + (1-\alpha)P_2)\mathbb{1}_{\mathcal{E}}(n) + (\alpha Q_1 + (1-\alpha)Q_2)\mathbb{1}_{\mathcal{O}}(n) \,,$$

où P_1, P_2, Q_1 et Q_2 sont des noyaux vérifiant 107 et $\mathcal{O}$ et $\mathcal{E}$ désignent respectivement les sous ensembles de $\mathbb{N}$ des nombres pairs et impairs, alors la fonction $\alpha \to v(f, R_n^{(\alpha)}, \pi)$ est décroissante sur $[0;1]$; voir les Lemmes IV.29 et IV.30.

Méthodologie de comparaison

Un cas typique d'utilisation du Théorème IV.4 concerne la comparaison de méthodes MCMC, communément désignées sous le nom de *Data Augmentation* (voir la Remarque D.7), utilisant des chaînes de Markov à deux composantes $\{(Y_k, U_k),\, k \in \mathbb{N}\}$ où Y est une

$$\begin{array}{ccccc} & \begin{cases} U_{k+1} \sim P(u, y; \cdot) \\ Y_{k+1} \sim Q(u', y; \cdot) \end{cases} & & \begin{cases} U_{k+2} \sim P(u', y'; \cdot) \\ Y_{k+2} \sim Q(u'', y; \cdot) \end{cases} & \\ \begin{pmatrix} Y_k = y \\ U_k = u \end{pmatrix} & \longrightarrow & \begin{pmatrix} Y_{k+1} = y' \\ U_{k+1} = u' \end{pmatrix} & \longrightarrow & \begin{pmatrix} Y_{k+2} = y'' \\ U_{k+2} = u'' \end{pmatrix} \end{array}$$

Figure 36 – Chaîne de Markov de noyau $K = PQ$

variable d'intérêt et U est une variable auxiliaire. Une telle chaîne cible une loi pouvant s'écrire $\pi(\mathrm{d}y, \mathrm{d}u) = \pi^*(\mathrm{d}y)R(y, \mathrm{d}u)$ et évolue classiquement au moyen d'un noyau de transition $K = PQ$ comme le montre la Figure 36.

P et Q sont généralement des noyaux de transition de type Gibbs ou Metropolis-Hastings qui, pris individuellement sont π-réversibles, mais pour qui le produit $K = PQ$ n'est que rarement π-réversible (voir la Proposition D.5). Par conséquent, le Théorème 4 de Tierney ne peut s'appliquer pour comparer des chaînes de ce type et en particulier obtenir une inégalité sur les variances asymptotiques du type

$$\lim_{n\to\infty} \frac{1}{n} \mathrm{Var}\left(\sum_{k=0}^{n-1} f(Y_k^{(0)})\right) \leq \lim_{n\to\infty} \frac{1}{n} \mathrm{Var}\left(\sum_{k=0}^{n-1} f(Y_k^{(1)})\right) \tag{109}$$

où pour $i \in \{0, 1\}$, P_i et Q_i vérifient 107 et $\{(Y_k^{(i)}, U_k^{(i)}),\, k \in \mathbb{N}\}$ est une chaîne de Markov de noyau de transition $K_i = P_i Q_i$. Notre Théorème suggère de considérer la chaîne de Markov inhomogène $\{X_k^{(i)},\, k \in \mathbb{N}\}$ définie par :

$$\begin{array}{ccccc} & \begin{cases} \tilde{Y}_k^{(i)} \sim \delta_y(\,\cdot\,) \\ \tilde{U}_k^{(i)} \sim P_i(u, y; \cdot) \end{cases} & & \begin{cases} Y_{k+1}^{(i)} \sim Q_i(u', y; \cdot) \\ U_{k+1}^{(i)} \sim \delta_{u'}(\,\cdot\,) \end{cases} & \\ X_{2k}^{(i)} = \begin{pmatrix} Y_k^{(i)} = y \\ U_k^{(i)} = u \end{pmatrix} & \longrightarrow & X_{2k+1}^{(i)} = \begin{pmatrix} \tilde{Y}_{k+1}^{(i)} = y \\ \tilde{U}_{k+1}^{(i)} = u' \end{pmatrix} & \longrightarrow & X_{2k+2}^{(i)} = \begin{pmatrix} Y_{k+1}^{(i)} = y' \\ U_{k+1}^{(i)} = u' \end{pmatrix} \end{array}$$

Pour toute fonction $G : \mathsf{X} = \mathsf{Y} \times \mathsf{U} \to \mathbb{R}$ vérifiant 108, alors d'après le Théorème IV.4, $v_0(G) < v_1(G)$. Dans la majorité des cas, les fonctions d'intérêts G ne dépendent que de Y *i.e.* $G(Y, U) = f(Y)$ (la variable U a le plus souvent le seul rôle d'aider à la simulation de Y). Dans ce cas, pour $i \in \{0, 1\}$, pour n pair

$$\mathrm{Var}\left(\sum_{k=0}^{n-1} G(X_k^{(i)})\right) = \mathrm{Var}\left(\sum_{k=0}^{n/2-1} G(X_{2k}^{(i)}) + G(X_{2k+1}^{(i)})\right) = \mathrm{Var}\left(\sum_{k=0}^{n/2-1} f(Y_k^{(i)}) + f(\tilde{Y}_{k+1}^{(i)})\right)$$

et donc

$$v_i(G) = \lim_{n\to\infty} \frac{1}{n} \mathrm{Var}\left(\sum_{k=0}^{n-1} G(X_k^{(i)})\right) = 2 \lim_{n\to\infty} \frac{1}{n} \mathrm{Var}\left(\sum_{k=0}^{n/2-1} f(Y_k^{(i)})\right) = 2 v_i(f)$$

ce qui prouve l'inégalité 109 qui n'était pas démontrable sans le Théorème IV.4.

Cette approche nous permet entre autre d'établir une justification théorique prouvant l'efficacité de l'algorithme de Carlin et Chib comparé à l'algorithme de Gibbs dans les situations où ces deux algorithmes peuvent être implémentés ; voir Remarque D.10. Pour

pouvoir comparer ces deux algorithmes il est nécessaire d'écrire la chaîne de Carlin et Chib $\{Y_k^{(\mathtt{CC})}, k \in \mathbb{N}\}$ sur le même espace que celui de la chaîne de Gibbs $\{Y_k^{(\mathtt{G})}, k \in \mathbb{N}\}$. On désigne par $Y = (j, \beta_j)$ la variable d'intérêt, par π^* la loi cible de ces algorithmes et respectivement par P_i et Q_i ($i \in \{\mathtt{G}, \mathtt{CC}\}$) les noyaux de transition qui mettent à jour la variable de classe et la variable continue. Les algorithmes de Gibbs et de Carlin et Chib peuvent être décomposés en une chaîne inhomogène alternant entre les deux noyaux P_i et Q_i ($i \in \{\mathtt{G}, \mathtt{CC}\}$), comme le montre la figure ci-dessous

$$P_{\mathtt{G}} \begin{cases} \tilde{J}_k^{(\mathtt{G})} \sim \pi^*(\cdot \,|\, \beta_j) \\ \tilde{Z}_k^{(\mathtt{G})} \sim \delta_{\beta_j}(\cdot) \end{cases} \qquad Q_{\mathtt{G}} \begin{cases} J_{k+1}^{(\mathtt{G})} \sim \delta_{j'}(\cdot) \\ Z_{k+1}^{(\mathtt{G})} \sim \pi^*(\cdot \,|\, j') \end{cases}$$

$$\searrow \\ \nearrow$$

$$Y_{2k}^{(i)} = \begin{pmatrix} J_k^{(i)} = j \\ Z_k^{(i)} = \beta_j \end{pmatrix} \nearrow Y_{2k+1}^{(\mathtt{G})} = \begin{pmatrix} \tilde{J}_k^{(\mathtt{G})} = j' \\ \tilde{Z}_k^{(\mathtt{G})} = \beta_j \end{pmatrix} \searrow$$

$$\searrow Y_{2k+2}^{(i)} = \begin{pmatrix} J_{k+1}^{(i)} = j' \\ Z_{k+1}^{(i)} = \beta'_{j'} \end{pmatrix} \nearrow \\ \nearrow Y_{2k}^{(i)} = \begin{pmatrix} J_k^{(i)} = j \\ Z_k^{(i)} = \beta_j \end{pmatrix} \searrow Y_{2k+1}^{(\mathtt{CC})} = \begin{pmatrix} \tilde{J}_k^{(\mathtt{CC})} = j' \\ \tilde{Z}_k^{(\mathtt{CC})} = u_{j'} \end{pmatrix} \nearrow$$

$$P_{\mathtt{CC}} \begin{cases} \forall\, i \neq j \;\; U_i \sim \zeta_i(\cdot) \\ U_j = \beta_j \\ \tilde{J}_k^{(\mathtt{CC})} \sim \pi(\cdot \,|\, u_1, \ldots, u_C) \end{cases} \qquad Q_{\mathtt{CC}} \begin{cases} J_{k+1}^{(\mathtt{CC})} \sim \delta_{j'}(\cdot) \\ Z_{k+1}^{(\mathtt{CC})} \sim \pi^*(\cdot \,|\, j') \end{cases}$$

On remarque que l'étape de simulation de la variable continue β_j conditionnellement à la variable de classe est identique dans les deux algorithmes *i.e.* $Q_{\mathtt{G}} = Q_{\mathtt{CC}}$. Comme le montrent les Exemples D.9 et D.11, l'utilisation des variables auxiliaires dans l'algorithme de Carlin et Chib favorise la visite de divers modèles comparé à l'échantillonneur de Gibbs. En conséquent, l'hypothèse $P_{\mathtt{CC}} \succeq_{\mathrm{P}} P_{\mathtt{G}}$, bien qu'elle ne soit pas démontrable, est réaliste. Sous cette hypothèse, le Théorème IV.4 montre que la chaîne de Markov $\{Y_k^{(\mathtt{CC})}, k \in \mathbb{N}\}$ a une variance asymptotique plus faible que $\{Y_k^{(\mathtt{G})}, k \in \mathbb{N}\}$, ce que le Théorème 4 de Tierney ne permettait pas, ces deux chaînes étant inhomogènes.

Un nouvel algorithme Pseudo-Marginal

Les algorithmes *Pseudo-Marginaux* [AR09, AV12] construisent des chaînes de Markov qui visent une distribution cible π^* qui n'est pas connue explicitement. π^* est typiquement une loi marginale incalculable d'une loi jointe connue. Ce type d'algorithme utilise des variables auxiliaires pour calculer un estimateur $\hat{\pi}^*$ de π^*, par échantillonnage d'importance par exemple. Deux algorithmes Pseudo-Marginaux sont étudiés :

- le MCWM est une version *bruitée* de l'algorithme de Metropolis-Hastings dans lequel à chaque itération, un candidat est proposé et deux ensembles de variables auxiliaires sont simulées pour calculer les estimateurs de la densité de l'état courant et de la proposition,
- le GIMH est un algorithme de Metropolis-Hastings qui conserve d'une itération à l'autre les variables auxiliaires correspondant à l'état courant et qui simule à chaque itération un candidat et l'ensemble de variables auxiliaires qui lui est associé.

Le Théorème 4 de Tierney ne permet pas de comparer ces deux algorithmes car le MCWM n'est pas π^*-réversible (le MCWM n'admet même pas π^* comme loi stationnaire). En revanche, guidé par le Théorème IV.4, nous proposons un nouvel algorithme le *Random Refreshment Algorithm*, qui ré-échantillonne les variables auxiliaires avec une certaine probabilité. Par application de la méthodologie précédemment proposée et du Théorème IV.4,

nous montrons, dans la section 4 du chapitre IV, que le *Random Refreshment Algorithm* a une variance asymptotique plus faible que celle du GIMH.

D'autres classes de MCMC pourraient être explorées à l'appui du Théorème IV.4 pour comparer des algorithmes existants ou en proposer de nouveaux comme cela a été le cas dans le contexte des algorithmes Pseudo-Marginaux. Une généralisation du Théorème IV.4 pour le cas de chaînes inhomogènes alternant entre $n > 2$ noyaux permettrait par exemple de comparer d'une façon plus générale les algorithmes de Metropolis-within-Gibbs.

Les méthodes PMCMC (Particle Markov chain Monte Carlo) [ADH10] sont des algorithmes MCMC qui utilisent des méthodes de Monte Carlo séquentielles (SMC) pour construire une loi de proposition. L'algorithme *Particle Marginal Metropolis-Hastings* (PPMH) proposé dans [ADH10] est l'analogue de l'algorithme GIMH dans un contexte PMCMC pour des modèles espace-état ou de Markov caché. Il serait intéressant d'étudier si un théorème similaire au Théorème IV.4 existe pour comparer les méthodes PMCMC et, le cas échéant, de proposer un algorithme de type *Random Refreshment Algorithm* implémentable dans ce contexte.

Chapitre I

Détection de cibles dans l'infrarouge et sélection de bandes

Ce chapitre présente notre contribution à la détection de cibles dans des images infrarouge multispectrales et au problème associé de sélection de bandes spectrales. Bien qu'élaborée à partir de SIR d'avions, cette méthodologie ne s'y restreint pas et peut être appliquée à d'autres types de données. Une grande partie de ce chapitre est extraite de [M2].

1 Introduction

Progress made during the last fifty years in optics sensors enhanced the use of InfraRed (IR) detection for scientific, civil and military applications. IR sensors enable to detect targets that cannot be set apart from their surroundings in the visible spectral range, thanks to their emitted heat. This explains why knowledge of aircraft IR emission is compulsory to assess their detection probability and thus their susceptibility. In the last decade, the usefulness of multispectral or hyperspectral sensors for remote sensing assignments has been proven [HCYW96, CC02] and some studies [KR02, MS02] emphasize their potential for target detection. Multispectral sensors sample the incoming light from the scene in several, about 10 or less, wavelength bands, whereas hyperspectral sensors collect data in hundreds of narrow contiguous spectral bands. These sensors provide a powerful means to discriminate different materials on the basis of their unique spectral signatures.

However, few multispectral sensors are, for now, available in the IR field. In this paper, we focus on the specification of a low resolution multispectral InfraRed sensor for aircraft detection. For many reasons, the experimental approach is generally not feasible to evaluate the IRS (aircraft not available, safety reasons...), and computer programs are therefore extremely valuable tools. Yet, as existing computer simulations of aircraft IRS [JD06, NKSS91, Gau81] do not account for the dispersion induced by uncertainty on input data, such as aircraft aspect angles, meteorological conditions and optical properties, they must be coupled with uncertainty propagation methods to estimate the detection performance of IR optronic systems [LRVD10b]. In that case, the scenario encompasses a lot of possible situations that must indeed be addressed, but cannot be singly simulated and the uncertainties in the input data propagate through the simulation model to the output data. Therefore, the simulated result is no longer a single IRS value, but a set of possible IRS which should include the IRS measured at a given instant. ONERA has developed for thirty years a simulation of combat aircraft IRS, CRIRA, initiated by [Gau81] ; see the section Simulation for more details. Using CRIRA, we have defined a general metho-

dology for predicting simulated IRS dispersion of imperfectly known military aircraft and performing aircraft detection on the resulting set of low resolution spectrally integrated InfraRed images in [M6].

In this paper, we consider multispectral aircraft IRS : each pixel is a vector whose coordinates correspond to the irradiance of the optronic scene partially integrated over a specified set of bands of the IR spectrum. In order to be useful, the sensor should be able to detect an aircraft far ahead. This explains the coarse resolution of the data we consider : the IRS are 16×16 images and an aircraft rarely signs over 10 pixels. The sensor would indeed be too cumbersome otherwise.

Target detection in multi/hyperspectral images has given raised to a wealth of research efforts [MDC10, Man05, MS02]. The typical objective of these methods is to detect small and rare objects in a background clutter. Given a statistical model (possibly partially unknown), most of the target detection algorithms derive from the Neyman-Pearson Likelihood Ratio Test (LRT) or from the Generalized Likelihood Ratio Test (GLRT), when some parameters are unknown. A highly sought after feature for these detectors is the CFAR (Constant False Alarm Rate) property, implying that the probability of false alarm (PFA) does not depend on any unknown parameter. Therefore, it is theoretically possible to set the test rule so that the detector achieves a given false alarm rate.

In some applications, a characteristic spectral signature of the target is *a priori* known [MMS03, RFKN92, SRY90, SCE96]. The corresponding detection methods are referred to as *matched filter* algorithms in the literature ; see Example A.4. Although being CFAR [RFKN92], the detection performance of these algorithms rely upon the quality of the target reference spectrum, which may be difficult to obtain in practice. Moreover, it is not suitable to targets whose spectral signature cannot be described by a model.

Another approach referred to as *anomaly detection* does not request the target spectral signature knowledge [MDC10]. In this framework, it is assumed that most of the image is composed of a background clutter, whose first and second order statistics may be unknown. Anomaly detection algorithms aim at identifying areas or pixels of the image who significantly differs from the background. More precisely, in the multi/hyperspectral imagery context, this generally boils down to find pixels whose spectral properties stand out from those of the background. Reed and Yu proposed in [RY90] a CFAR anomaly detector, referred to as the RX (Reed Xiaoli) detector and considered as the benchmark among the anomaly detection algorithms designed for multi/hyperspectral images ; see Paragraph RX detector in Preamble A .2. In the RX detector, the background pixels are supposed to be independent and identically distributed (i.i.d.) with an unknown multivariate Gaussian distribution. Under this assumption, the Generalized Likelihood Ratio Test (GLRT) amounts to compare the squared Mahalanobis distance between the sampled background distribution and a pixel under test to a detection threshold. A consequent research effort based on the RX detector have thus emerged, mostly focused on the estimation of background distribution moments : as a matter of fact, the homogeneous multivariate Gaussian distribution assumption is generally not suitable for real backgrounds as a whole, and deviations from this model lead to high false alarm rates. Refer to page 37 and to [MDC10, BAR+11] for a review of these methods.

Although in their seminal paper [RY90], Reed and Yu assume the knowledge of the target optical pattern, as noted in Paragraph RX detector, Preamble A .2, most of the *evolved* RX detectors process each pixel separately and do not account for the target spatial pattern. However, a promising way of accounting for spatial contextual information was proposed in [CVGCMM+06]. To achieve hyperspectral classification, this method uses a support vector machine algorithm with composite kernels for the clustering step, in order to

add information about surrounding area of each pixel : a sum of two kernels is considered, one for the pixel signature, and the other one for the mean and/or the standard deviation of signatures of the surrounding pixels. Unfortunately, this is not suited to low resolution objects : in the case of aircraft multispectral IRS, nearly all aircraft pixels are adjacent to several background pixels.

Therefore, to the best of our knowledge, detector in multi/hyperspectral images for low resolution targets, whose spectral signature is unknown and taking into account the target sprawl, has yet to be proposed. The figure 14 gives a hint of the kind of images we are going to cope with and illustrates the two types of dispersion that should be handled by the proposed methodology :

- the spatial dispersion : the geometry of the aircraft is different in the three images,
- the spectral dispersion : the irradiance intensity varies in the different wavelength bands.

In this paper, we propose an innovative methodology for aircraft detection in a multispectral image : it takes simultaneously advantage of spectral and spatial discriminant features to reveal anomalies. It combines the Mahalanobis transform embedded in the RX algorithm with some level set technics proposed in [M6] ; see Preamble A .4.

In most cases, aircraft corresponds to hot temperatures at the sensor level. Hence it is natural to rely on a detection test that considers the hottest pixels in the sensed image, and therefore in its Mahalanobis transform. If these pixels are close, they are likely to come from a target ; otherwise they belong to the clutter. Instead of manually testing the neighborhood of each hot pixel, we propose to take advantage of a powerful tool in image analysis : the level sets [OP03, MG00, MM98]. The results emphasize that, in the context of aircraft detection, there is a great interest in using multispectral IRS rather than integrated IRS, as long as the IR bands are well chosen. As a matter of fact, the detection performances turn out to vary greatly according to the number and the wavelength bands location in the IR spectrum. We thus proposed to use a genetic algorithm [Gol89] to optimize the detection performance and to provide the set of 2, 3 or 4 optimal elementary band combinations for aircraft detection.

This paper is organized as follows : the multispectral detection algorithm is introduced in Section 2 and the wavelength bands selection strategy is detailed in Section 3 . Finally, experimental results are given in Section 4 .

2 A Level Set Approach to Anomaly Detection

2 .1 Statistical framework

A multispectral image with K spectral bands is a function $f : \Omega \to \mathbb{R}^K$, where Ω is a discrete and finite subset of $\mathbb{R}^2$. The set of multispectral images featuring K bands is denoted $\mathcal{F}_K$, and let for all $f \in \mathcal{F}_K$, $\Omega_B(f)$ and $\Omega_T(f)$ denote the subspaces of Ω related respectively to f background and target. We assume that $\Omega_T(f)$ and $\Omega_B(f)$ are complementary subsets of Ω, *ie* $\Omega = \Omega_B(f) \cup \Omega_T(f)$. In addition, let $|\Omega|$ denotes the number of pixels in the image f. Finally, for all $x \in \Omega$, the vector $y_x = f(x) \in \mathbb{R}^K$ is referred to as a spectral pixel, where for all $k \in \{1, \ldots, K\}$, $y_x^{(k)}$ denotes the pixel x irradiance integrated over the k-th spectral band.

On the basis of this observation, a decision of two hypotheses : the null hypothesis H_0 corresponding to sky background and the alternative hypothesis H_1, simply corresponding to everything but H_0, shall be made. We thus assume that the set $\mathcal{F}_K$ can be written as $\mathcal{F}_K = H_0 \cup H_1$. In our framework, an anomaly detection is a statistical test ϕ mapping

any multispectral image $f \in \mathcal{F}_K$ to $\{0,1\}$ such that :

$$\phi : f \to \begin{cases} 0 \Longrightarrow f \in H_0\,, \\ 1 \Longrightarrow f \in H_1\,. \end{cases}$$

In the following, Φ will denote the set of mapping from $\mathcal{F}_K$ to $\{0,1\}$.

The performance of a detection test $\phi \in \Phi$ is characterized by the two following statistics :

– the probability to detect true positive samples $\mathrm{P_D}$, defined for any $f \in \mathcal{F}_K$ as :

$$\mathrm{P_D}(\phi) = \mathbb{P}[\,\phi(f) = 1 \,|\, f \in H_1]\,,$$

– the probability to predict positive, samples that are actually negative, $\mathrm{P_{FA}}$, (also referred to as false alarm rate) and defined as :

$$\mathrm{P_{FA}}(\phi) = \mathbb{P}[\,\phi(f) = 1 \,|\, f \in H_0]\,.$$

When these probabilities are analytically intractable, one may use the following estimates :

$$\hat{\mathrm{P}}_\mathrm{D}(\phi) = \frac{1}{|H_1|} \sum_{f \in H_1} \phi(f)\,, \qquad \hat{\mathrm{P}}_\mathrm{FA}(\phi) = \frac{1}{|H_0|} \sum_{f \in H_0} \phi(f)\,,$$

where $|H_1|$ and $|H_0|$ respectively denote the number of positive and negative samples in the data set.

In addition to ϕ, anomaly detection methods such as the RX algorithm use a pixel-level test which we will denote ψ in the following. Given a background model B, possibly partially unknown, ψ writes similarly to ϕ as a mapping from $\mathcal{F}_K \times \Omega$ to $\{0,1\}$, such that :

$$\psi : (f,x) \to \begin{cases} 0 \Longrightarrow x \in \mathsf{B}\,, \\ 1 \Longrightarrow x \notin \mathsf{B}\,. \end{cases}$$

Defining ϕ and ψ are two important issues when implementing a detection algorithm. In particular, there is always a compromise between increasing the probability of detection $\mathrm{P_D}$, while keeping the probability of false alarm $\mathrm{P_{FA}}$ low. For any given detector, this trade-off may be described by the receiver operating characteristic (ROC) curve, which plots $\mathrm{P_D}$ versus $\mathrm{P_{FA}}$.

In this paper, we characterize the performance of a detection test $\phi \in \Phi$ with the two following statistics, which take into account a tradeoff between $\mathrm{P_D}$ and $\mathrm{P_{FA}}$:

– The area under the ROC curve denoted $S_1(\phi)$,
– The probability of detection given a fixed alarm rate ϵ, that is $S_2(\phi,\epsilon) = \mathrm{P_D}(\phi)$ given that $\mathrm{P_{FA}}(\phi) \leq \epsilon$.

2 .2 RX anomaly detection test

If we model the background as a K-dimensional multivariate Gaussian distribution $\mathsf{B} = \mathcal{N}(\mu, \Gamma)$, a proper anomaly detector is the well-known RX detector [RY90]. Under this assumption, the log-likelihood function of the background distribution is proportional to the Mahalanobis distance $d_M(\,\cdot\,, \mathsf{B})$. For any $f \in \mathcal{F}_K$ and $x \in \Omega$, $d_M(\,\cdot\,, \mathsf{B})$ provides a similarity measure between $f(x)$ and the background distribution B such that :

$$d_M(f(x), \mathsf{B}) = (f(x) - \mu)^{(T)} \Gamma^{-1} (f(x) - \mu)\,.$$

More precisely the pixel-level detection test $\psi_{RX} \in \Psi$ is defined for any $f \in \mathcal{F}_K$ and $x \in \Omega$ as :

$$\psi_{RX}(f, x) = \mathbb{1}_{]\alpha,\infty[}(d_M(f(x), \mathsf{B}))\,,$$

where $x \to \mathbb{1}_A(x)$ is the set A indicator function and $\alpha \in \mathbb{R}$ the test threshold.

Moreover for any $f \in \mathcal{F}_K$, the set of anomalous pixels may be defined as $A(f) = \{x \in \Omega, \psi_{RX}(f, x) = 1\}$. Finally, the resulting image-level detection test $\phi_{RX} \in \Phi$ may be expressed, for any $f \in \mathcal{F}_K$, as $\phi_{RX}(f) = 1$ if $A(f) \neq \{\emptyset\}$ and $\phi_{RX}(f) = 0$ otherwise.

As explained in Preamble A .1, we assume that the background distribution is known. We thus consider the base RX version before taking into account spatial information.

Under this assumption, a well known property of the Mahalanobis distance is that for some multispectral image $f \in \mathcal{F}_K$, the Mahalanobis transform of a pixel $x \in \Omega_B(f)$ is such that $d_M(f(x), \mathsf{B}) \sim \chi^2_K$, where χ^2_K is the chi-squared distribution with K degrees of freedom. For any $f \in H_0$, the probability of false alarm writes in this case :

$$\mathrm{P_{FA}}(\phi_{RX}) = \mathbb{P}[d_M(f(x), \mathsf{B}) > \alpha] = \int_\alpha^\infty \chi^2_K(u)\mathrm{d}u\,.$$

The threshold α can thus be set to achieve a specific (constant) false alarm rate (CFAR property).

For any multispectral image $f \in \mathcal{F}_K$, let $\hat{f}$ denote in the following the Mahalanobis transform of f defined for all $x \in \Omega$ as :

$$\hat{f} : x \to d_M(f(x), \mathsf{B})\,.$$

Although the spatial structure of f is preserved by the Mahalanobis transform, $\hat{f}$ does not convey any quantitative spectral information but provides a cartography where *high* gray levels are likely to be anomalies. Figure I-1 illustrates the Mahalanobis transform of a synthetic multispectral image f, with $K = 4$ spectral bands.

We define a first anomaly detection test $\phi^{(1)}_\alpha \in \Phi$, derived from the RX detector. Given that the Mahalanobis transform of the background pixels is known, we propose :

$$\phi^{(1)}_\alpha(f) = \mathbb{1}_{\{[\alpha;\infty[\}}\left(\max_{x\in\Omega} \hat{f}(x)\right)\,,$$

where $\alpha \in \mathbb{R}$ is the test threshold. A convenient property of this test is that the threshold α can be set analytically in order to achieve a given false alarm rate. Indeed, there exists a bijection $\Pi_1 : \mathbb{R} \to (0, 1)$ such that $\mathrm{P_{FA}}(\phi^{(1)}_\alpha) = \Pi_1(\alpha)$. A basic proposition on the distribution of an i.i.d. random field shows indeed that :

$$\mathrm{P_{FA}}(\phi^{(1)}_\alpha) = \mathbb{P}[\,\max(\hat{f}) \geq \alpha \,|\, f \in H_0\,] = 1 - F_{\chi^2_K}(\alpha)^{|\Omega|}\,,$$

where $F_{\chi^2_K}$ is the cumulative distribution function of the χ^2_K distribution. Therefore, for a given false alarm rate $p \in (0, 1)$

$$\alpha = \Pi_1^{-1}(p) = q_{\chi^2_K}\left((1-p)^{1/|\Omega|}\right)\,,$$

where for all $\lambda \in (0, 1)$, $q_{\chi^2_K}(\lambda)$ is the λ-quantile of a chi-squared cumulative distribution function. However, this test does not account for spatial information and can lead to low detection probabilities, especially when $|\Omega|$ is high.

A more general test $\phi^{(2)}_{\alpha,\beta}$ derived from $\phi^{(1)}_\alpha$ consists in detecting an anomaly in a multispectral image $f \in \mathcal{F}_K$ when at least one pixel is unlikely to be a realization of a

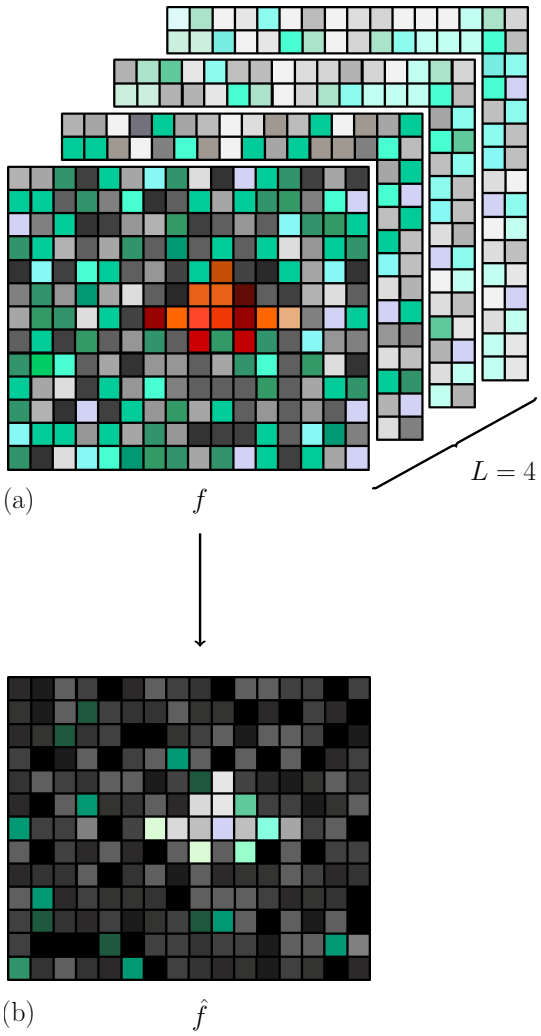


FIGURE I-1 – (a) simulated multispectral IRS, featuring $K = 4$ spectral bands and (b) its Mahalanobis transform

K-degree of freedom chi-squared distribution. That is, either one pixel exceeds (or is lower to) a high (or a low) quantile of a χ^2_K random field. From a physical standpoint, this test features the advantage of detecting anomalies with either a positive or a negative contrast from the background Infrared signature. $\phi^{(2)}_{\alpha,\beta}$ may be expressed as :

$$\phi^{(2)}_{\alpha,\beta}(f) = \max \left\{ \mathbb{1}_{[0;\beta]} \left(\min_{x\in\Omega} \hat{f}(x) \right), \mathbb{1}_{[\alpha;\infty[} \left(\max_{x\in\Omega} \hat{f}(x) \right) \right\} ,$$

where α and β are the two test thresholds. α is defined as previously, while using the same type of argument, β may be set as $\beta = \Pi_1^{-1}(1-p)$, where p is the test $\phi^{(1)}_\alpha$ desired false alarm rate.

2 .3 A detection test combining spectral and spatial information

Although in our context an aircraft is weakly resolved, its typical IRS spreads over a small set of adjacent pixels (Figure 14). As a consequence, the associated Mahalanobis transform also features relevant adjacent pixels. An alternative to $\phi^{(1)}_\alpha$, which only

considers the maximum of the Mahalanobis transform of some image $f \in \mathcal{F}_K$, is thus to study some well chosen level sets of $\hat{f}$. Level sets have long proved their usefulness in image processing [OP03],[MG00],[MM98]. Here they provide a handy tool for testing spatial proximity of hot pixels. Indeed, the assumption that the background pixels are not spatially correlated implies that it is very unlikely to have a level set of say, a high quantile of the χ^2_K distribution, which does not come from the target. As a result, compared to $\phi^{(1)}_\alpha$, for a given probability of detection, the false alarm rate should decrease considerably when taking into account the level sets.

Let us recall some basic definitions (refer, for instance, to [Mat75] for further details). In the sequel, let $\bar{\Omega}$ be the $\mathbb{R}^2$ extension of Ω and for any function $g : \Omega \to \mathbb{R}$, let $I(g) : \bar{\Omega} \to \mathbb{R}$ denote the bicubic interpolation of g on $\bar{\Omega}$ (many other interpolation schemes could also be used as well). Moreover, let for any $\alpha \in \mathbb{R}$, ${\mathcal{C}_\alpha}^+(g)$ and ${\mathcal{C}_\alpha}^-(g)$ be the two α-level sets defined by :

$$
\begin{aligned}
{\mathcal{C}_\alpha}^+(g) &= \{x \in \bar{\Omega},\, I \circ g(x) \geq \alpha\}\,, \\
{\mathcal{C}_\alpha}^-(g) &= \{x \in \bar{\Omega},\, I \circ g(x) \leq \alpha\}\,.
\end{aligned}
$$

The interpolation implies that $I(g)$ is continuous on $\bar{\Omega}$ and that regularity conditions hold. In particular, the α-level line set ${\mathcal{L}_\alpha}^+(g)$ (resp. ${\mathcal{L}_\alpha}^-(g)$) exists and is defined as ${\mathcal{L}_\alpha}^+(g) := \partial{\mathcal{C}_\alpha}^+(g)$ (resp. ${\mathcal{L}_\alpha}^-(g) := \partial{\mathcal{C}_\alpha}^-(g)$). Finally, let $C^+_\alpha(g)$ be the set of the closed elements of ${\mathcal{C}_\alpha}^+(g)$ and similarly define $C^-_\alpha(g)$, $L^+_\alpha(g)$ and $L^-_\alpha(g)$.

We propose a third detection test $\phi^{(3)}_{\alpha,\nu} \in \Phi$, making use of some level sets tools. First, let $A^{(3)}_{\alpha,\nu}$ be the set of anomalous regions of $\bar{\Omega}$ defined for any $f \in \mathcal{F}_K$ by :

$$
A^{(3)}_{\alpha,\nu}(f) = \{\, x \in \mathfrak{c},\, \mathfrak{c} \in C^+_\alpha(\hat{f}),\, \mathrm{Per}(\mathfrak{c}) > \nu \,\}\,,
$$

where for any level line $\mathfrak{l}$ or level set $\mathfrak{c}$, $\mathrm{Per}(\mathfrak{l})$ and $\mathrm{Per}(\mathfrak{c})$ both denote its perimeter.

The pixel-level and image-level detection tests may thus be expressed as follows :

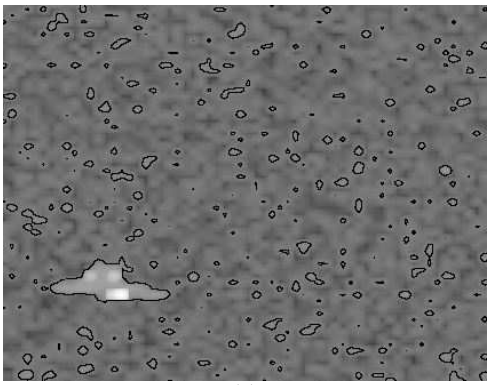
$$
\begin{aligned}
\psi^{(3)}_{\alpha,\nu}(f,x) &= \mathbb{1}_{A^{(3)}_{\alpha,\nu}(f)}(x)\,, \\
\phi^{(3)}_{\alpha,\nu}(f) &= \mathbb{1}_{[\nu;\infty[}\left(\max_{\mathfrak{l} \in L^+_\alpha(\hat{f})} \mathrm{Per}(\mathfrak{l})\right)\,.
\end{aligned}
$$

$\phi^{(3)}_{\alpha,\nu}$ exploits both spatial and spectral information in that it only retains sets of adjacent pixels of the Mahalanobis transform above the noise level. Moreover, compared to $\phi^{(1)}_\alpha$, it conveys a spatial information about the target location.

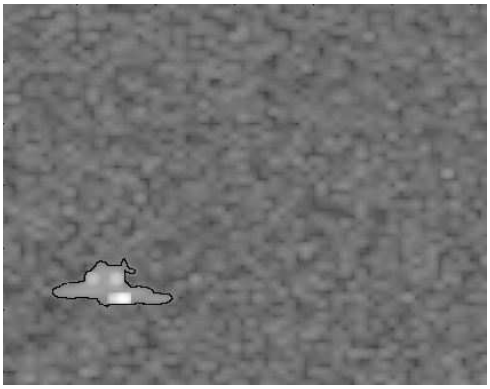
Figure I-2 illustrates the appropriateness of taking into account some well chosen level lines in the detection test $\phi^{(3)}_{\alpha,\nu}$ on some 64×64 multispectral images. The top panel (a) displays the level line set $L_\alpha(f)$. Even though $L_\alpha(f)$ contains the target, it also features level lines that belongs to the background. Conversely, the bottom panel (b), displays only the elements of $L_\alpha(f)$ whose perimeter exceeds a threshold ν : this set is actually restricted to the target.

To focus on the relevant level sets, α and ν should be set so that an α-level set with perimeter larger than ν is unlikely for a χ^2_K random field. However, the distribution of the maximum perimeter of an α-level set of a χ^2_K random field, denoted π_α, is intractable. Thus, contrarily to the test $\phi^{(1)}_\alpha$, the mapping $\mathrm{P_{FA}}(\phi^{(3)}_{\alpha,\nu}) = \Pi_3(\alpha,\nu)$ has no analytical expression. Nevertheless, note that

$$
\Pi_3(\alpha,\nu) = \int_\nu^\infty \pi_\alpha(\mathrm{d}u) = \mathbb{E}_{\pi_\alpha}\left[\mathbb{1}_{\{[\nu;\infty[\}}(U)\right]\,,
$$



(a)



(b)

FIGURE I-2 – Representation of $I(\hat{f})$, where $f \in H_1$ and of the level line set $L_\alpha^+(\hat{f})$ in (a) and $\{\mathfrak{l}, \mathfrak{l} \in L_\alpha^+(\hat{f}), \text{Per}(\mathfrak{l}) \geq \nu\}$ in (b)

yielding that a Monte Carlo (MC) approximation $\hat{\Pi}_3^N$ of Π_3 can be obtained for any data point $(\alpha, \nu) \in \mathbb{R}^2$, see Algorithm 5.

Remarks :

- Algorithm 5 allows a pointwise approximation of Π_3 but not directly of Π_3^{-1} which is the function of interest. Therefore, starting from any threshold value $(\alpha_0, \nu_0) \in \mathbb{R}^2$, a walk on the parameter space may be used to iteratively decrease $|\hat{\Pi}_3^N(\alpha_n, \nu_n) - p|$. In such a scheme, the number of MC draws can be adapted to start with some rough approximation of Π_3 in the early stage of the scheme and increasing N as $\hat{\Pi}_3^N(\alpha_n, \nu_n) \to p$.
- Note that compared to Π_1, Π_3 is not a bijection. Therefore, multiple parameter combinations (α, ν) may achieve the desired false alarm rate.
- Even though, one can argue that the test parameters calibration represents an obvious computation burden, note that this process is done only once before the anomaly detector $\phi_{\alpha,\nu}^{(3)}$ gets under way.
- Computing the test statistic $S_1(\phi_{\alpha,\nu}^{(3)})$ necessarily involves a similar walk through the parameter space. The parameters achieving a desired false alarm rate should thus be estimated during this process (see Section 4).

Procedure 5 Pointwise estimation $\hat{\Pi}_3^N$ of Π_3

Entrées : data point $(\alpha, \nu) \in \mathbb{R}^2$, MC simulation number N $S \leftarrow 0$
for $n = 1 : N$ **do**
(i) Draw a $|\Omega|$-dimensional χ^2_K random field

$$W_n \sim \otimes_{i=1}^{|\Omega|} \chi^2_K \,,$$

(ii) Determine the level lines $\mathfrak{l} \in L^+_\alpha(W_n)$
if $\max \mathrm{Per}(\mathfrak{l}) > \nu$ **then**
$S \leftarrow S + 1$
end if
end for
Sorties : $\hat{\Pi}_3^N(\alpha, \nu) = S/N$

Complementarily to $\phi^{(3)}_{\alpha,\nu}$ and similarly to $\phi^{(2)}_{\alpha,\beta}$, we define a last detection test $\phi^{(4)}_{\alpha,\beta,\nu} \in \Phi$ defined for any $f \in \mathcal{F}_K$ as :

$$\begin{aligned}
&A^{(4)}_{\alpha,\beta,\nu}(f) = \{\, x \in \mathfrak{c},\, \mathfrak{c} \in C^+_\alpha(\hat{f}) \cup C^-_\beta(\hat{f}),\, \mathrm{Per}(\mathfrak{c}) > \nu \,\}\,, \\
&\psi^{(4)}_{\alpha,\beta,\nu}(f, x) = \mathbb{1}_{A^{(4)}_{\alpha,\beta,\nu}(f)}(x)\,, \\
&\phi^{(4)}_{\alpha,\beta,\nu}(f) = \mathbb{1}_{[\nu;\infty[}\left(\max_{\mathfrak{l} \in L^+_\alpha(\hat{f}) \cup L^-_\beta(\hat{f})} \mathrm{Per}(\mathfrak{l}) \right)\,.
\end{aligned}$$

The possible anomalies are likely to belong either to a α-upper level set or to a β-lower level set of a $|\Omega|$-dimensional χ^2_K random field. α and β may be set respectively as high and low quantiles of the χ^2_K distribution. The perimeter threshold ν can be defined in the same way as for $\phi^{(3)}_{\alpha,\nu}$.

As mentioned in Section 1 , when the background distribution B is unknown, one can substitute the mean μ and the covariance matrix Σ with their Maximum Likelihood estimate $\hat{\mu}$ and $\hat{\Sigma}$. Moreover, it was demonstrated in [RY90] that the CFAR property remains : instead of a χ^2_K distribution, the false alarm rate follows a Beta distribution, whose parameters only depend upon K and $|\Omega|$.

As a consequence, the four detection tests we have proposed may also be used when the background distribution B is unknown. The only requirement is to have sufficiently background samples $f \in H_0$, to estimate properly the test thresholds via Monte-Carlo sampling.

3 Wavelength Bands Selection

Our database of simulated aircraft IRS consists of multispectral images featuring $K = 10$ elementary spectral bands $\{b_k\}_{k=1}^{10}$ evenly spaced across the 2000 - 3000 cm^{-1} spectral range (actually, 10 bands of spectral width 100 cm^{-1}). However, as already mentioned in [YRS93] and [SCE96], the number of bands K and their location in the 2000 - 3000 cm^{-1} spectrum both have a huge impact in the detection performance, regardless the detector $\phi \in \Phi$ choice.

First, the false alarm rate does not decrease when the number of spectral bands increases. Figure I-3 displays the distribution of the Mahalanobis Transform of pixels in a

multispectral image $f \in \mathcal{F}_K$ with $K = 1$, 2, 4 and 6 bands. On the one hand, the plain lines correspond to the analytic distribution of the background pixels *ie* the chi-squared distribution with 1, 2, 4 and 6 degrees of freedom. On the other hand, the dashed line refers to the empirical distribution of the target pixels denoted T, which, contrarily to the χ^2_K distribution, does not vary significantly with K. The hatched areas below the plain lines correspond to the respective false alarm rate achieved with the different values of K. This shows that, in the case of our distribution T, the higher the number of bands K involved in the multispectral image is, the higher the expected false alarm rate.

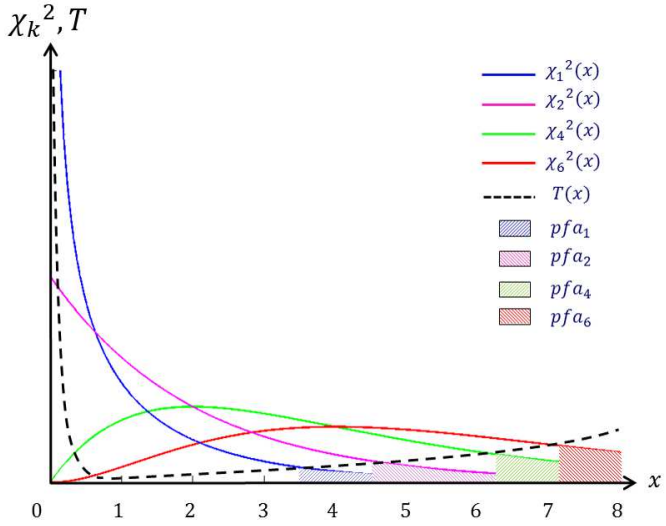


FIGURE I-3 – Distribution of the Mahalanobis transform of a background multispectral pixel featuring $K = 1, 2, 4, 6$ bands (plain lines) and the empiric Mahalanobis transform of a multispectral target pixel (dashed line).

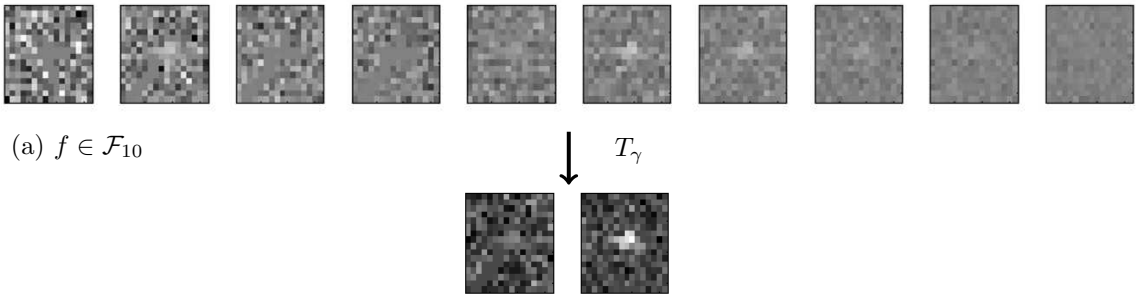
Second, a higher spectral resolution enables identification of narrow spectral features and thus leads to an easier discrimination between background and target pixels. Still compared with broader bands measurements, the narrow spectral bands may significantly reduce the signal-to-noise ratio. As a consequence, considering regroupments of consecutive elementary bands is a promising prospect.

Third, the location of some elementary bands is such that they do not provide any information about the targets. As a matter of fact, the atmospheric absorption phenomenon, due in particular to H_2O and CO_2 in the $2000 - 3000\,\mathrm{cm}^{-1}$ range, makes some elementary bands irrelevant.

For all these reasons, a trade off on the band number K and their respective bandwidth should be made, and a bands selection step is therefore mandatory. In the following, we consider multispectral images with K bands, such that :

- each band $\{r_k\}_{k=1}^K$, is a regroupment of some consecutive elementary bands,
- a spectral band r_k may not contain more than C consecutive elementary bands,
- two spectral bands r_k and r_{k+1} are separated by at least one elementary band.

These constraints induce a set $\mathcal{E}_K$ of the possible regroupments providing K band multispectral images. For all $\gamma \in \mathcal{E}_K$, define T_γ as the mapping of $\mathcal{F}_{10} \to \mathcal{F}_K$, such that for all $f \in \mathcal{F}_{10}$, $T_\gamma(f)$ is the multispectral image f corresponding to the regroupment γ (cf Figure I-4).



(a) $f \in \mathcal{F}_{10}$

(b) $T_\gamma(f) \in \mathcal{F}_2$

Figure I-4 – The multispectral images $f \in \mathcal{F}_{10}$ and $T_\gamma(f) \in \mathcal{F}_2$, where $\gamma = \{b_2 - b_{5:8}\}$.

For a given detector $\phi \in \Phi$, we propose in this section a method to find the element $\gamma^\star \in \mathcal{E}_K$ such that for all $\gamma \in \mathcal{E}_K$ either (i) or (ii) holds :

$$(i)\ {S_1}^\gamma(\phi) \leq {S_1}^{\gamma^\star}(\phi)\,, \qquad (ii)\ {S_2}^\gamma(\phi) \leq {S_2}^{\gamma^\star}(\phi)\,,$$

where ${S_1}^\gamma(\phi)$ and ${S_2}^\gamma(\phi)$ both refer to the detection statistics defined in Section 2 , but applied to the multispectral images $T_\gamma(f)$ instead of f.

Any regroupment $\gamma \in \mathcal{E}_K$ may be parameterized by some vector $\theta \in \Theta_K$, where Θ_K is the subset of $\mathbb{N}^{2K}$ defined as follows :

$$\theta \in \Theta_K\,, \qquad \theta_K = (t_1, \ldots, t_{2K})\,,$$

where :

$$\begin{cases} t_1 \geq 0 \quad \text{and} \quad t_2, \ldots, t_{2K} > 0\,, \\ \sum_{u=1}^{2K} t_u \leq K\,. \end{cases}$$

For all $k \in \{1, \ldots, K\}$, t_{2k} denotes the number of consecutive elementary band(s) in the regroupment r_k. Conversely, for all $k \in \{1, \ldots, K\}$, t_{2k-1} denotes the number of elementary band(s) left out between r_{k-1} and r_k. Thus, for each $\theta \in \Theta_K$, it corresponds an unique $\gamma \in \mathcal{E}_K$ and reciprocally. As a consequence, finding the regroupment $\gamma^\star \in \mathcal{E}_K$ satisfying the relations (i) or (ii) defined above is equivalent to finding the optimal parameter $\theta^\star \in \Theta_K$: each regroupment γ can be assimilated with its parametrization θ. Figure I-5 provides an example of such a parameterization.

Finding the optimal parameter $\theta^* \in \Theta_K$ is a discrete optimization problem with constraints. For this reason, we chose in this work to use a Genetic Algorithm (GA) [Gol89] to perform this task. By analogy with the evolutionary theory which predicts that in a random population only the individuals the most adapted to the environment will survive, a Genetic Algorithm looks for the gene that corresponds to the fittest individual. In our context, for a given number of band regroupments K, the genes are the parameters $\theta \in \Theta_K$, the population consists in the possible band combinations $\gamma \in \mathcal{E}_K$ and the fitting environment measure μ is either the function $\theta \to {S_1}^\theta(\phi)$ or $\theta \to {S_2}^\theta(\phi)$, where, with some abuse of notations, $S_i^\theta(\phi)$ denotes the statistic $S_i(\phi)$ evaluated on the images $T_\theta(f)$.

By deriving successive generations from an initial random population, the Genetic Algorithm will provide *in fine* individuals belonging to the last generation which are the fittest with respect to the environment. Algorithm 6 summarizes the different steps performed at each iteration of the Genetic Algorithm. Implementation issues are discussed in Section 4 .

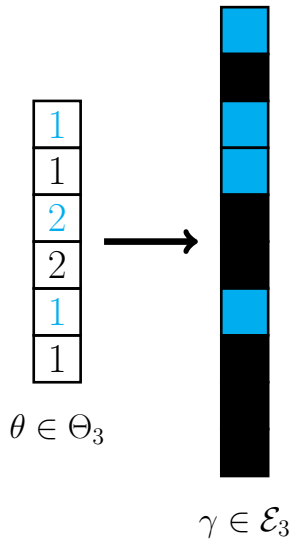


FIGURE I-5 – A band combination and the corresponding parameter for $K = 3$

Procedure 6 The Genetic Algorithm

Entrées : A detector $\phi \in \Phi$, a fitness function μ, the number of band regroupments K, the population size N and the number of generations M.

1 - Initialization : start with a random population $\{\theta^{(i)}\}_{i=1}^{N}$, $\theta^{(i)} \in \Theta_K$,

2 - Evaluation : measure the fitness of the population by computing $\mu(\theta^{(i)})$ and keeps the N fittest individual,

3 - Selection : draw N couples such that each individual has a probability corresponding to its fitness,

4 - Reproduction : each couple provides an individual such that :

(a) each gene shared by the parents is preserved,

(b) other genes are drawn randomly up to the limit that the new individual remains in Θ_K,

(c) random mutations also happen with low probability.

The resulting $2N$ population is then evaluated with step (2).

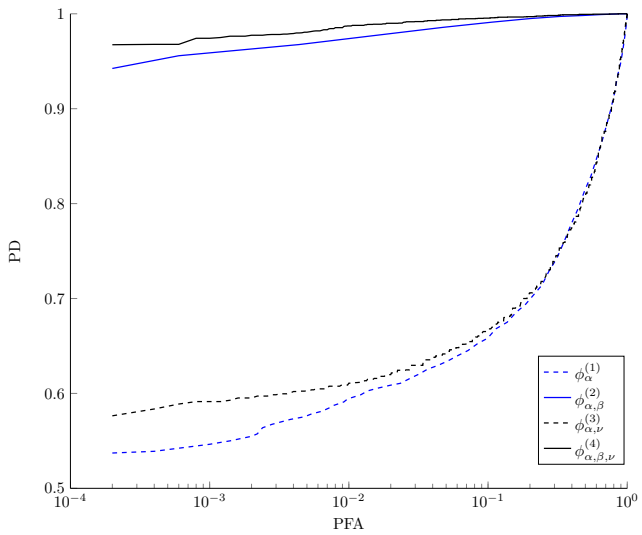


Figure I-6 – ROC curves of $\phi_{\alpha}^{(1)}$, $\phi_{\alpha,\beta}^{(2)}$, $\phi_{\alpha,\nu}^{(3)}$ and $\phi_{\alpha,\beta,\nu}^{(4)}$ applied to $f \in \mathcal{F}_{10}^{(2)}$

With a proper population size N and a generation number M specified, this constrained and discrete optimization problem may be achieved through the routine `ga` available in the `Matlab Global Optimization Toolbox`.

4 Results of the Methodology's Application

4 .1 Detectors comparison

In this section we compare the different detectors defined in Section 2 applied to the raw multispectral images $f \in \mathcal{F}_{10}$. The multispectral images database for the three combat aircraft will be respectively denoted $\mathcal{F}_{10}^{(1)}$, $\mathcal{F}_{10}^{(2)}$ and $\mathcal{F}_{10}^{(3)}$. Moreover $\mathcal{F}_{10}^{(\text{all})}$ will stand for the complete database gathering the three aircraft.

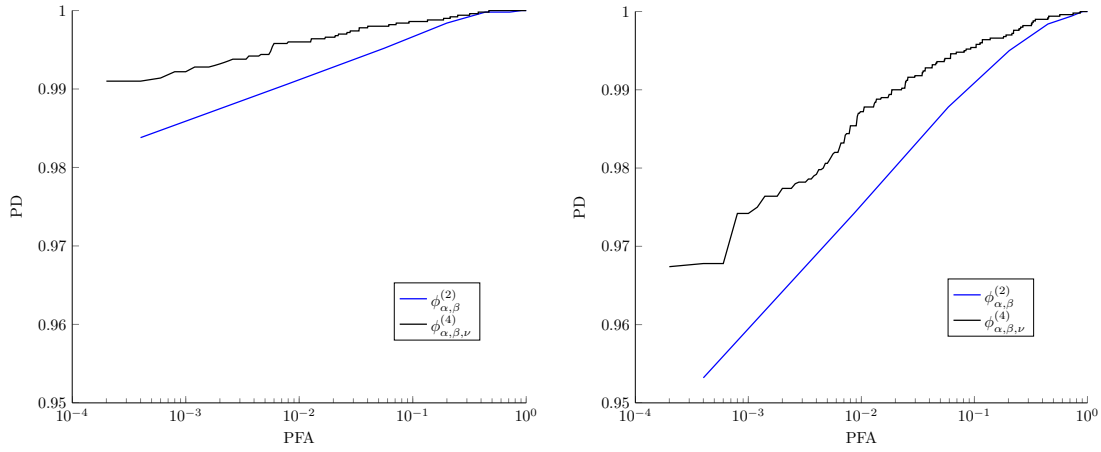
A first graphical evidence about the relevance of simultaneously taking into account high and low levels of the Mahalanobis transform in the detection test is given by Figure I-6. It displays the four detectors $\phi_{\alpha}^{(1)}$, $\phi_{\alpha,\beta}^{(2)}$, $\phi_{\alpha,\nu}^{(3)}$ and $\phi_{\alpha,\beta,\nu}^{(4)}$ ROC curves in log scale, obtained by applying these detectors using different parameters α, β, ν to $N = 10000$ multispectral images such that $\{f_n\}_{n=1}^{N/2} \in \mathcal{F}_{10}^{(2)}$ and $|H_0| = 5000$ background images. Clearly, the $\phi_{\alpha}^{(1)}$ and $\phi_{\alpha,\nu}^{(3)}$ ROC curves are below that of $\phi_{\alpha,\beta}^{(2)}$ and $\phi_{\alpha,\beta,\nu}^{(4)}$, implying that the latter two detectors outperform the two former. Quantitatively, Figure I-7 provides the statistics S_1 and S_2 characterizing the four detectors. First, this confirms the conjecture made with Figure I-6. Then, these statistics show that the level sets detection test $\phi_{\alpha,\nu}^{(3)}$ and $\phi_{\alpha,\beta,\nu}^{(4)}$ outperforms respectively the min/max tests $\phi_{\alpha}^{(1)}$ and $\phi_{\alpha,\beta}^{(2)}$, especially when low false alarm rate are required.

In the following, we compare the two best detectors $\phi_{\alpha,\beta}^{(2)}$ and $\phi_{\alpha,\beta,\nu}^{(4)}$ for each aircraft : Figure I-8 plots the ROC curves in log scale for the four different databases. In I.8(a), I.8(b) and I.8(c), the training set consisted of $N = 5000$ multispectral images f of respectively $\mathcal{F}_{10}^{(1)}$, $\mathcal{F}_{10}^{(2)}$ and $\mathcal{F}_{10}^{(3)}$ along with $|H_0| = 5000$ background images, while in I.8(d) $N = 15000$ multispectral images $f \in \mathcal{F}_{10}^{(\text{all})}$ were used along with $|H_0| = 5000$ background images.

Most of the type 1 aircraft feature a single hot pixel : as a consequence, it is no

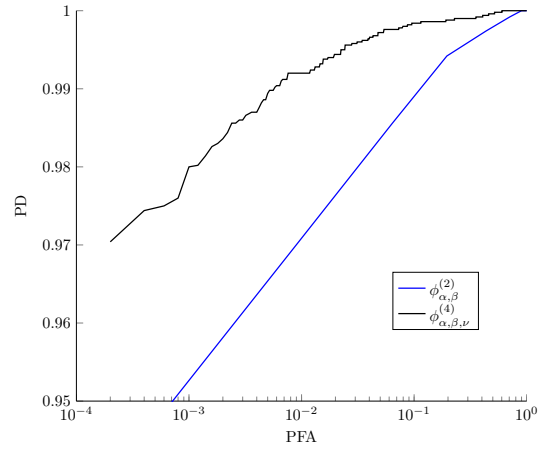
	$S_1(\phi)$	$S_2(\phi, \epsilon = 10^{-2})$	$S_2(\phi, \epsilon = 10^{-3})$
$\phi^{(1)}_{\alpha}$	0.805	0.59	0.561
$\phi^{(2)}_{\alpha,\beta}$	0.995	0.97	0.968
$\phi^{(3)}_{\lambda,\nu}$	0.810	0.61	0.591
$\phi^{(4)}_{\lambda,\rho,\nu}$	0.998	0.99	0.974

FIGURE I-7 – Statistics of the four detectors applied to $f \in \mathcal{F}^{(2)}_{10}$

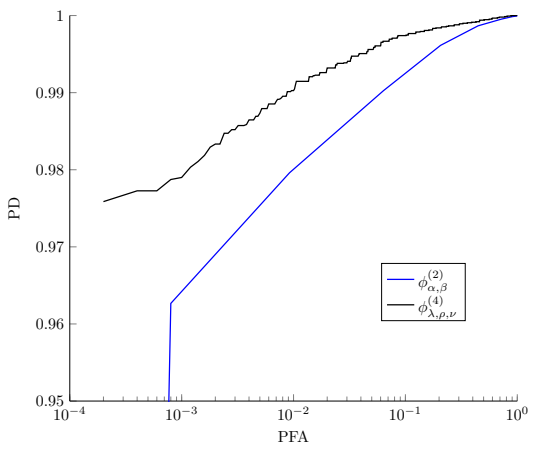


(a) $f \in \mathcal{F}^{(1)}_{10}$

(b) $f \in \mathcal{F}^{(2)}_{10}$



(c) $f \in \mathcal{F}^{(3)}_{10}$



(d) $f \in \mathcal{F}^{(\text{all})}_{10}$

FIGURE I-8 – ROC curves of the detectors $\phi^{(2)}_{\alpha,\beta}$ and $\phi^{(4)}_{\alpha,\beta,\nu}$ for different aircraft

surprise that the two ROC curves in I.8(a) are closer than in any other scenario. Indeed, the detector $\phi^{(4)}_{\alpha,\beta,\nu}$ ROC curve is achieved for level sets that encompass one or at most two pixels which is equivalent to the min/max test $\phi^{(2)}_{\alpha,\beta}$. The cases of aircraft 2 and 3 are interesting because the detection performance of $\phi^{(2)}_{\alpha,\beta}$ and $\phi^{(4)}_{\alpha,\beta,\nu}$ does not evolve similarly : $\phi^{(2)}_{\alpha,\beta}$ detects more easily type 2 aircraft than type 3 and conversely for $\phi^{(4)}_{\alpha,\beta,\nu}$. This actually meets the aircraft 2 and 3 characteristics : while most of the time the aircraft 2 has an higher Infrared signature than aircraft 3, with more localized hot spots, the min/max test $\phi^{(2)}_{\alpha,\beta}$ detects better aircraft 2 than aircraft 3. On the other hand, because aircraft 3 Infrared signature is characterized by a relatively low but constant level it is thus easier to detect for the level set test $\phi^{(4)}_{\alpha,\beta,\nu}$ than aircraft 2. In any case, the level set test outperforms the min/max detector.

In addition of being more efficient than $\phi^{(2)}_{\alpha,\beta}$, the detector $\phi^{(4)}_{\alpha,\beta,\nu}$ provides a spatial information about the aircraft location in the multispectral images.

The set of the estimated target pixels is defined as the inner pixels of the contour line $\mathfrak{l} \in L^+_\alpha(\hat{f}) \cup L^-_\beta(\hat{f})$ having the larger perimeter. If this perimeter is lower than the threshold ν, no target is detected and thus the estimated target pixels is the empty set. Figure I-9 illustrates the spatial information conveyed by $\phi^{(4)}_{\alpha,\beta,\nu}$: I.9(a) represents a (difficult) multispectral image f of a type 2 aircraft, I.9(b) shows the Mahalanobis transform $\hat{f}$ of f and the biggest contour lines respectively of the sets $L^+_\alpha(\hat{f})$ in yellow and $L^-_\beta(\hat{f})$ in cyan for two fixed thresholds α and ν. Finally I.9(c) displays the set of the estimated target pixels (the white pixels) and the true target pixels (inner pixels of the green line). To make the location test more challenging, a background was added to the original 16×16 images so that in Figure I-9 the detection algorithm is applied to a 48×48 sample.

4 .2 Optimal band regroupment

The raw database provides standard multispectral images $f \in \mathcal{F}_{10}$. We consider in the following multispectral images derived from this database, featuring $K = 2$, 3 or 4 regroupments $\{r_k\}_{k=1}^K$ of consecutive elementary subbands and for which the constraints imposed in Section 4 hold. In this implementation, we added a regroupment length restriction :

– For all $K \in \{2, 3, 4\}$ and for all $1 \leq k \leq K$, $t_{2k} \leq 4$.

Moreover, we used the following specification for the Genetic Algorithm 6 :

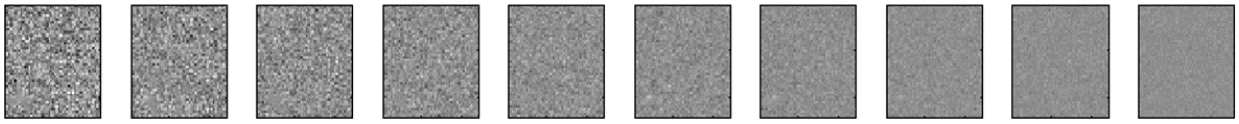
(i) the cross-over operator (Step 4-(b)) refreshes the genes of a children which are unshared by its parents. These genes $g \in G \subset \{1, \ldots, 2K\}$ are globally refreshed so that $\theta = (t_1, \ldots, t_{2K}) \in \Theta_K$ is replaced by $\theta' = (t'_1, \ldots, t'_{2K}) \in \Theta_K$ with a discrete uniform random draw on the set

$$\mathcal{S}_{\text{glob}} := \{(p_1, \ldots, p_{|G|}) \in \mathbb{N}^{|G|} \mid \forall i \in G\ t'_i = p_i,\ \forall i \notin G\ t'_i = t_i,\ \theta' \in \Theta_K\}\,.$$

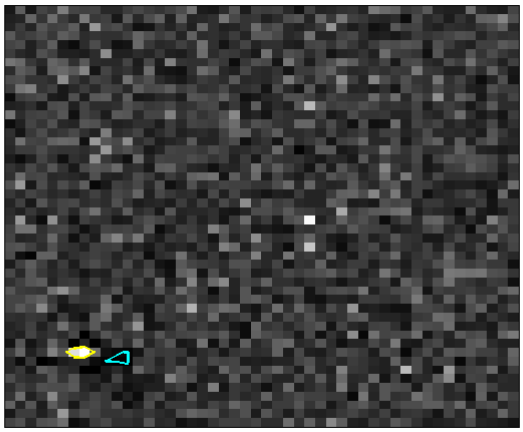
(ii) the mutation step (Step 4-(c)) allows the gene of any individual to randomly change at each iteration with a probability of 1%. If a mutation occurs locally on the gene t_k of some individual $\theta \in \Theta_K$, t_k is replaced with a discrete uniform random draw on the set

$$\mathcal{S}_{\text{loc}} := \{i \in \mathbb{N}, (t_1, \ldots, t_{k-1}, i, t_{k+1}, \ldots, t_{2K}) \in \Theta_K\}\,.$$

Figure I-10 shows some ROC curves obtained by applying the detector $\phi^{(4)}_{\alpha,\beta,\nu}$ to a database featuring $N = 5000$ multispectral images $\{f_n \in \mathcal{F}^{(1)}_3\}_{n=1}^N$ and $|H_0| = 5000$



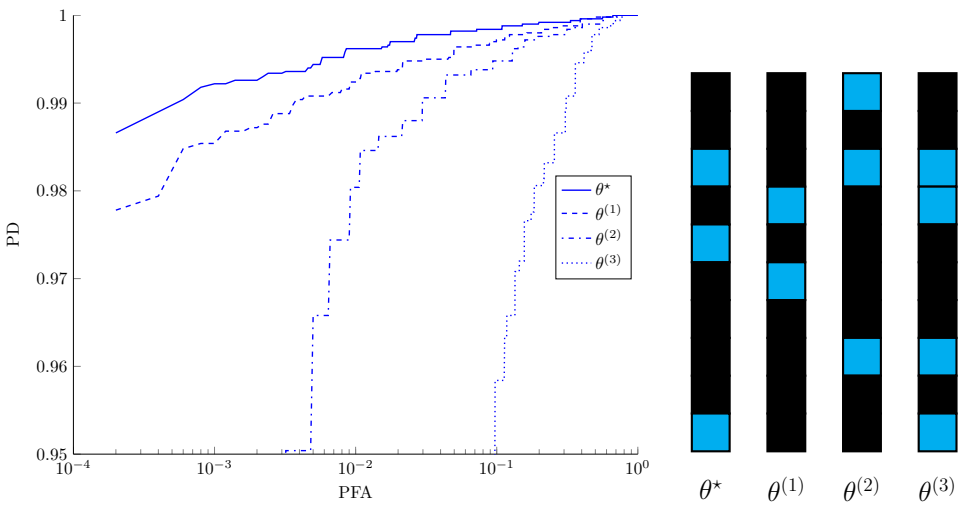
(a) $f \in \mathcal{F}_{10}^{(2)}$



(b) $\hat{f}$, $L_\alpha^+(\hat{f})$ and $L_\beta^-(\hat{f})$

(c) Part of the target detected and true target location

FIGURE I-9 – Location of a zone of interest in a multispectral image using the detector $\phi_{\alpha,\beta,\nu}^{(4)}$



(a) ROC curves (b) Parameters

Figure I-10 – ROC curves of $\phi^{(4)}_{\alpha,\beta,\nu}$ applied to $f \in \mathcal{F}^{(3)}_{10}$ for different parameters $\theta \in \Theta_3$

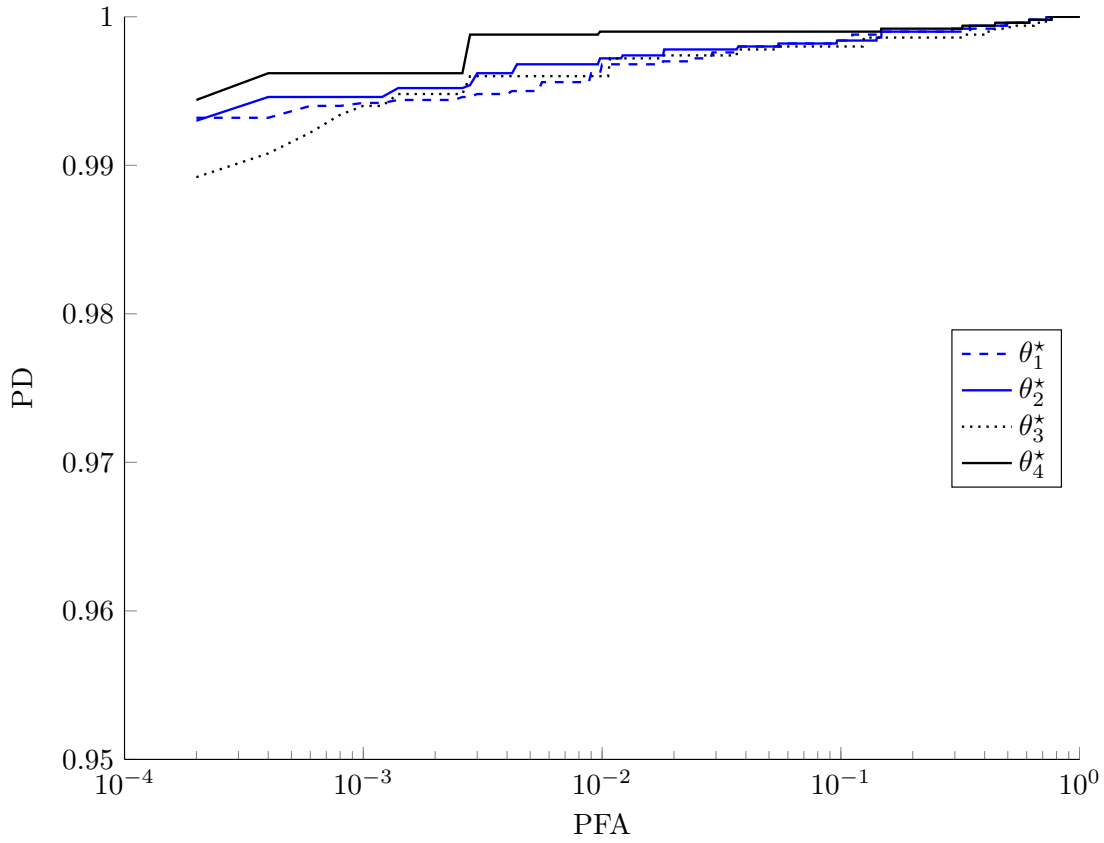
background images. The four curves correspond to different band regroupments $\theta \in \Theta_3$: the plain curve is achieved with the parameter $\theta^\star \in \Theta_3$ which maximizes the criterion $\theta \to S^\theta_1(\phi^{(4)}_{\alpha,\beta,\nu})$. The dashed curves are related to other parameters $\left\{\theta^{(i)} \in \Theta_3\right\}_{i=1}^{4}$ specified in I.10(b). Given similar band regroupments, the resulting AUC turns out to be significantly different : it is therefore crucial for the detection performance to know the optimal band combination. This optimal band combination will also help to discriminate decoys from real aircraft, provided that their emission spectrum does not perfectly match the radiation of the airplane. If not, following the methodology developed in [M6], an additional classification step making use of geometric features of the targets will be necessary.

Using the detector $\phi^{(4)}_{\alpha,\beta,\nu}$ and multispectral images of aircraft 1, Figure I-11 shows the Genetic Algorithm outcomes for $K = 2$ band regroupments and using different parameters. The blue curves are obtained for the fitness function $\mu = S_1(\phi^{(4)}_{\alpha,\beta,\nu})$ and the black ones for $\mu = S_2(\phi^{(4)}_{\alpha,\beta,\nu}, \epsilon = 5.10^{-2})$. The plain and dashed lines only differ according to the GA parameters : the population size and the generation number, respectively referred to as N and M in Algorithm 6.

For each scenario, the quantitative results of Figure I.11(b) confirm that the GA provides band regroupments which are coherent with the fitness function μ and that the optimization improves with M and N. Still, the computation time considerably increases with these two parameters. Therefore, given that the detection performances vary very little, the parameters $\mu = S_1$, $N = 10$ and $M = 50$ will be used in the following.

Figure I-12 displays the ROC curves of the detector $\phi^{(4)}$ applied to the multispectral images $f \in \mathcal{F}^{(1)}$ (I.12(a)), $f \in \mathcal{F}^{(2)}$ (I.12(b)), $f \in \mathcal{F}^{(3)}$ (I.12(c)) and $f \in \mathcal{F}^{(\text{all})}$ (I.12(d)). The GA was applied to these four data sets with the parameters μ, N and M specified above to obtain the optimal parameters $\theta^\star_2$, θ^*_3 and θ^*_4 corresponding respectively to the $K = 2$, $K = 3$ and $K = 4$ optimal band regroupments. For each data type, the detector $\phi^{(4)}$ is applied to the multispectral images featuring :

- the optimal band combinaison for $K = 2$ regroupments $\theta^\star_2$ (plain black curves)
- the optimal band combinaison for $K = 3$ regroupments $\theta^\star_3$ (dotted black curves)
- the optimal band combinaison for $K = 4$ regroupments $\theta^\star_4$ (dashed black curves)



(a) ROC curves

	μ	N	M	$S_1(\phi^4)$	$S_2(\phi^4, \epsilon = 5.10^{-2})$
$\theta^*_{2,1}$	S_2	10	50	0.991	0.996
$\theta^*_{2,2}$	S_2	25	100	0.994	0.998
$\theta^*_{2,3}$	S_1	10	50	0.993	0.995
$\theta^*_{2,4}$	S_1	25	100	0.995	0.996

(b) Optimized criterion

FIGURE I-11 – Optimization with different GA parameters

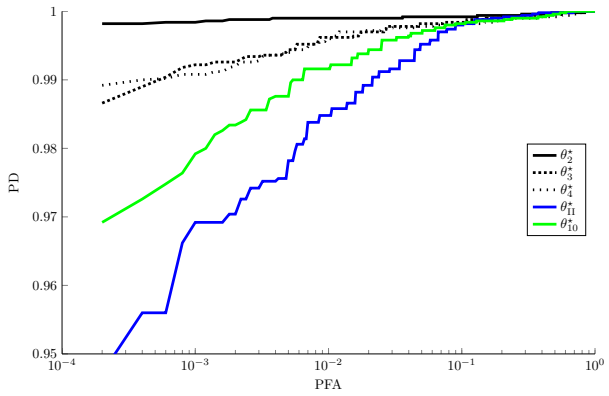
- the images integrated over the band II spectrum, *ie* monospectral $\theta^{\star}_{\text{II}}$ (plain green curves)
- the standard multispectral images, *ie* with $K = 10$ elementary subbands, $\theta^{\star}_{10}$ (plain blue curves)

For all data sets, the $K = 2$ band regroupments outperforms the other band regroupments with $K = 3$ and $K = 4$. However there is no such general ranking between the optimal regroupments of $K = 3$ and $K = 4$ bands. In comparison, the standard multispectral images with $K = 10$ subbands is always less efficient than the optimal $K = 2$ band regroupment but may achieve better performance than the optimal $K = 3$ and $K = 4$ band regroupments. For $K = 3$ and $K = 4$, the optimal band regroupments $\theta^{\star}_K \in \Theta_K$ are identical for the different aircraft. For $K = 2$, the optimal band regroupments are the same for two aircraft and only differ with one element for the third one. This simulation shows that provided the knowledge of the optimal band regroupments, there is a definite interest in using multispectral images to perform detection instead of monospectral images : indeed, the ROC curves obtained for the monospectral data set is always below the best multispectral ROC curve.

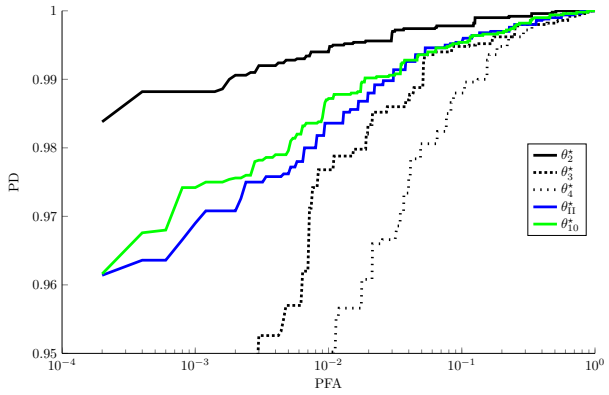
Regardless the data set, the optimal parameter $\theta^{\star}$ always presents the same spectral profile : a first short regroupment involving 2 elementary subbands early in the band II and a second regroupment of 3 or 4 elementary subbands further in the band II. The optimal nature of the $K = 2$ band regroupments may be explained with two arguments :

- Algorithm justification : For any standard multispectral image $f \in \mathcal{F}_{10}$, consider the multispectral image $T_{\theta^{\star}}(f) \in \mathcal{F}_2$, $(\theta^{\star} \in \Theta_2)$. If a target is present in the multispectral image $T_{\theta^{\star}}(f)$, then it may appear in inverted contrast in the first image and in positive contrast in the second one. Indeed, for the first regroupment, the noise level is high (the noise level is higher at the beginning of the band II) and the aircraft signature is low (only two bands are taken into account in the first regroupment of $\theta^{\star}$). Conversely the several bands of the second regroupment provide a stronger target signal while maintaining a low noise level. As a consequence, a target embedded in a multispectral image $T_{\theta^{\star}}(f) \in \mathcal{F}_2$ may thus be detected either with the level set $L^{+}_{\alpha}(\hat{f})$ or $L^{-}_{\beta}(\hat{f})$.
- Physical justification : The engine plume hot gases emissions, mainly constituted of CO_2 and H_2O, radiate at $2.7\,\mu m$ and may thus also sign at the beginning of the band II. Moreover, the CO_2 absorption band in the neighborhood of $3.5\,\mu m$ corresponds to the gap between the first and the second regroupment. Finally, the information conveyed by the second band regroupment coincides with the fuselage reflectance which signs at $4.3\,\mu m$.

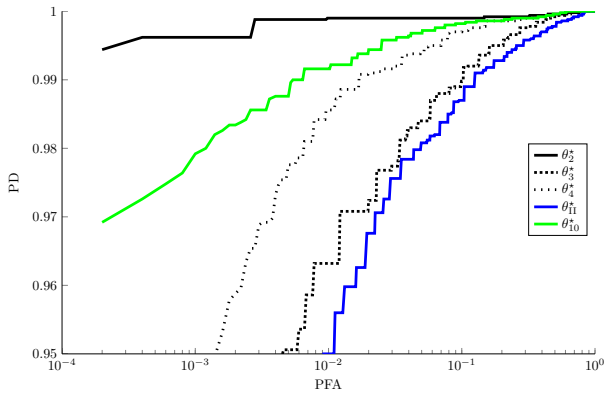
Note that the optimal band regroupments with $K = 3$ or $K = 4$ tend to adopt the same spectral profile as $\theta^{\star}_2$ by artificially adding one or two irrelevant band regroupment(s) in the end of the band II to reach their target number of regroupments K. As a consequence, in addition to providing an explanation of the optimality of $\theta^{\star}_2$, these two arguments provide a justification of the sub-optimality of the band regroupments $\theta^{\star}_3$ and $\theta^{\star}_4$. In this paper, we focus on the single sensor case, but if multiple sensors are used our methodology still apply, only the optimal band combination could differ.



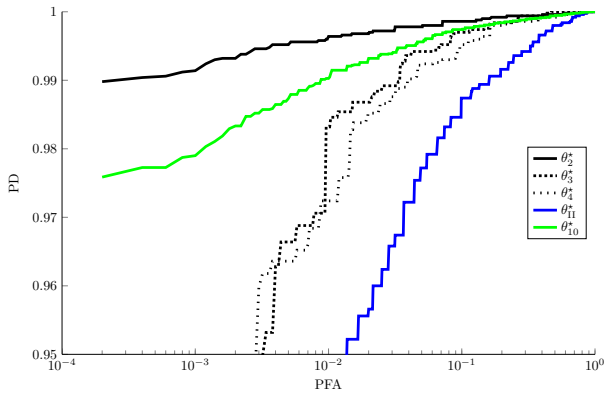
(a) $f \in \mathcal{F}^{(1)}$



(b) $f \in \mathcal{F}^{(2)}$



(c) $f \in \mathcal{F}^{(3)}$



(d) $f \in \mathcal{F}^{(\mathrm{all})}$

FIGURE I-12 – ROC curves for optimal band regroupments with $K = 2$, $K = 3$, $K = 4$

5 Conclusion

In this paper, we have introduced a novel method to perform anomaly detection in low resolution multispectral images. The detection task is challenging because the targets feature simultaneously spectral and spatial dispersion and limited prior knowledge are available. The proposed detector $\phi^{(4)}$, combining a Neyman-Pearson Likelihood Ratio test and a study of relevant level sets, is designed to handle these dispersions and is shown to outperform the standard RX detector, in our case.

This detector was then used to identify the optimal spectral band combination for aircraft detection, *ie* the band number and their location in the band II spectrum. For three different military aircraft, the same spectral profile featuring $K = 2$ band regroupments, provides the best detection performance. The optimization method and at large the proposed detection methodology can be extended to other problematics : other targets, different backgrounds, etc...

In particular, an interesting issue that remains to be addressed in the aircraft detection context is the anomaly detection in a cloudy sky background. In [M6], a Fractional Brownian motion was used to model this kind of textured background for monospectral images. Provided an extended model compatible with the multispectral approach, it would be interesting to study whether the detector $\phi^{(4)}$ can achieve good anomaly detection performance in multispectral images with such textured background.

Chapitre II

Online EM for Functional Data

Dans ce chapitre, nous présentons un modèle général d'observation présentant de fortes variations dans le temps, dans l'espace et en intensité ainsi que notre méthode d'estimation séquentielle de paramètres dans des modèles à données manquantes. Ce chapitre adopte une démarche méthodologique, dont l'implémentation est illustrée sur des bases de données classiques (courbes de croissance, chiffres manuscrits...). L'application de ces méthodes au cas des SIR multispectrales d'aéronefs fera l'objet du chapitre suivant. Une grande partie de ce chapitre est extraite de [M1].

1 Introduction

Functional data analysis is concerned with the analysis of curves and shapes, which often display common patterns but also variations (in amplitude, orientations, time-space warping, etc...). The problem of extracting common patterns (referred to as *templates*) from functional data, and the related problem of curves / images registration has given raised to a wealth of research efforts ; see [Ram06], [Zho08], [RL98] and the references therein.

Most of the proposed techniques used so far have been developed in a supervised classification context ; the method typically aims at finding a time / space warping transformation allowing to synchronize / register all the observations associated to a given class of curves / shapes and to estimate a template by computing a cross-sectional mean of the aligned patterns. In most case, the deformation is penalized, to favor "small" time / space shifts. Many different deformation models have been proposed for curves and for images. For curves, the warping function is often assumed to be monotone increasing ; in this context, the dynamic time warping algorithm is by far the most popular algorithm : it enables the alignment of curves by minimizing a cost function of the warping path, which can be solved by a dynamic programming algorithm [WG97]. Non parametric [KG92], [Sil85], [RL98] as well as Bayesian approaches [TI08], [LY09] have also been proposed, but they are still far less popular. The situation is more complex for shapes and images. Different deformation models have been proposed, involving rigid deformations, small deformations [CAM04] or deformation fields ruled by a differential equation ; see [Chr99].

In this paper, we introduce a common bayesian statistical framework for *unsupervised* clustering and template extraction, with applications to curve synchronization and shape registration. Following the seminal work by [AAT07] and [AK10], we consider a mixture of *deformable template models*, which models a curve / shape as a template (defined as a function of time or space), selected from a collection of templates, which undergoes a

random deformation and is observed in presence of an additive noise [BC11], [AAT07], [CRM96] and [M7] for a complete survey. Contrary to the classical time-warping / spatial registration algorithms which consists in synchronizing all the observations of a shape in a supervised framework, the mixture deformable template models is an unsupervised classifier : it estimates functional templates from a set of shapes / curves and consider the time warping / spatial deformations as a random nuisance parameter. It is important at this point to stress that the model allows to integrate the deformation conditionally on the observations while considering the templates as unknown deterministic functional parameters. In this context, the deformation might be seen as a *random effect* [PB00], which is similar to random effects in linear mixed models in longitudinal data analysis. Whereas this change in perspective might seem rather benign, it makes a huge difference both in theory and in practice.

In our model, the warping / deformation function and the cluster index is modeled as hidden data and the template is estimated by resorting to a Monte Carlo version of the online Expectation Maximization (MCoEM) algorithm. In this approach, the computation of the expectation step (E-step) turns out to be a major problem, because the conditional distribution of the cluster index and the deformation given the observations is in most cases not analytically available. To overcome this difficulty, several solutions have been considered. A rough approximation of the conditional expectation was proposed in [MMTY08], in which the posterior distribution is replaced by a point mass located at the posterior mode. Another elementary approach consists in linearizing the deformed template in the neighborhood of its nominal shape, under the assumption of small deformations ; this approach has been considered, among others by [LY09] and [FJ03], in which the transformed mixture of Gaussian models was used. Another way to handle the E-step, proposed by [GS04], consists in performing an approximate Bayesian integration, which amounts to replace the posterior distribution of the hidden data conditionally to the observation by a Gaussian distribution, obtained from a Laplace approximation. Here again, it is not always easy to justify such approximations. The expectation can also be approximated by Monte Carlo integration, an idea which was put forward by [AAT07] and [AK10] ; in these works, a Monte Carlo Markov Chain (MCMC) algorithm is used to sample the posterior distribution ; these samples are used to approximate the E-step by using a stochastic approximation version of the EM algorithm, referred to as the SAEM [DLM99]. The resulting algorithm has been shown to perform satisfactorily, but it is a time-consuming solution, especially when the dimension of the missing data is large ; the extension to multiple classes is even more computationally involved.

In this paper, we propose an online algorithm (in which the curves / shapes are processed one at a time) to estimate the unknown parameters of the mixture of deformable templates model. We adapt the online EM algorithm proposed in [CM07] to allow inference in our model. Our model is too general to allow the linearization or the use of Gaussian approximation of the complete data log-likelihood, as it was done in [LY09] and [GS04]. We thus propose to approximate the conditional expectation thanks to a MCMC algorithm adapted from [CC95]. Indeed, working online implies processing the data on the fly without storing them afterwards and this conveys open problems, such as the posterior distribution estimation. Our sampler is designed to sample the joint distribution of the cluster index and the deformation parameters in a sequential framework. Our algorithm happens to reduce significantly the computational burden of the method proposed in [AK10].

This paper is organized as follows : in section 2 , the mixture of the dense deformable template model is introduced, then the online EM is adapted to our context in section 3 . The sampling method of the joint posterior distribution is proposed in section 4 . Finally, illustrations of templates obtained by applying the learning algorithm to three sets of curves, shapes and images are proposed in section 6 . Some practical issues are discussed and potential perspectives are given.

2 A mixture of deformable template model

2 .1 A basic deformable model

In this section, we introduce a basic model for curves and images. A *template* is a function defined on a space $\mathbb{U}$ and taking for simplicity real values. Typically, for curves $\mathbb{U} = \mathbb{R}$ and for shapes $\mathbb{U} = \mathbb{R}^2$. We denote by T the set of templates.

The observations are modeled as the stochastic process $\mathcal{Y}$ indexed by $u \in \mathbb{U}$ and given by :

$$\mathcal{Y}(u) = \lambda \mathcal{T} \circ G_\beta(u) + \sigma \mathcal{W}(u) \ , \tag{II.1}$$

where, $\mathcal{T} \in \mathsf{T}$ is a template function, $\lambda \in \mathbb{R}^{+*}$ is a scaling factor, $\sigma^2 \in \mathbb{R}^{+*}$ is the noise variance and $\mathcal{W}$ a Gaussian process with zero-mean, unit variance and known covariance function. G_β is a function, belonging to G, the set of mappings from $\mathbb{U}$ to itself parameterized by a vector $\beta \in \mathsf{B}$, where B is an open subset of some euclidean space of dimension d_β. For curves, G can be chosen as the homotheties and translations mappings and more generally as the set of monotone functions (with appropriate smoothness conditions). For shapes, G can be taken as the set of rigid transformations of the plane, such as rotations, homotheties or translations and a local deformation field. The models for the set of deformations G are problem dependent ; see section 6 .

In this setting, β and λ are random variables, *i.e.* each realization of $\mathcal{Y}$ corresponds to different β and λ. The quantity of interest is the template $\mathcal{T}$ (a deterministic functional parameter), while the deformation G_β and the global scaling λ are regarded as nuisance parameters, that should be integrated out.

Finally, we assume that the set of templates T is the linear subspace spanned by the basis vectors $\{\phi_\ell\}_{1 \leq \ell \leq m}$. Hence, a template $\mathcal{T}_\alpha \in \mathsf{T}$ may be expressed as :

$$\mathcal{T}_\alpha = \sum_{\ell=1}^{m} \alpha_\ell \phi_\ell \ , \quad \text{where } \alpha = (\alpha_1, \ldots, \alpha_m)^T \in \mathcal{A}, \tag{II.2}$$

where for all $\ell \in \{1, \ldots, m\}$, $\phi_\ell : \mathbb{U} \to \mathbb{R}$ and $\mathcal{A}$ is a subset of $\mathbb{R}^m$. The pattern is observed at some design points denoted $\Omega = \{u_1, \ldots, u_{|\Omega|}\}$, where $|\Omega|$ is the dimension of the observations such that for all $s \in \{1, \ldots, |\Omega|\}$, $u_s \in \mathbb{U}$. Let Φ_β be the $|\Omega| \times m$ matrix defined such that for all $(s, \ell) \in \{1, \ldots, |\Omega|\} \times \{1, \ldots, m\}$,

$$[\Phi_\beta]_{s,\ell} = \phi_\ell \circ G_\beta(u_s) \ . \tag{II.3}$$

In the sequel, let Y and W be the vector defined by $Y = (\mathcal{Y}(u_1), \ldots, \mathcal{Y}(u_{|\Omega|}))^T$ and $W = (\mathcal{W}(u_1), \ldots, \mathcal{W}(u_{|\Omega|}))^T$. Using (II.1), the observation model may be expressed in a matrix-vector form as :

$$Y = \lambda \Phi_\beta \alpha + \sigma W \ . \tag{II.4}$$

2 .2 A mixture of deformable templates

We extend the model to include multiple templates corresponding to the different "typical" shapes that we are willing to cluster and then recognize. To that purpose, we construct a mixture of the template model introduced in the previous section. Denote by C the number of classes $(\mathcal{C}_1, \ldots, \mathcal{C}_C)$. We associate to each observation Y an (hidden) class index $J \in \mathsf{J}$, where $\mathsf{J} = \{1, \ldots, C\}$. To each class $\{\mathcal{C}_j,\, j \in \mathsf{J}\}$ is attached a template function $\{\mathcal{T}_j \in \mathsf{T},\, j \in \mathsf{J}\}$, which is parameterized by $\{\alpha_j \in \mathbb{R}^m,\, j \in \mathsf{J}\}$. Moreover, it is assumed that each class $j \in \mathsf{J}$ has a prior weight ω_j and we denote by $\omega = (\omega_1, \ldots, \omega_C)$ the set of prior weights. To sum up, we consider the following hierarchical model :

$$Y \in \mathcal{C}_j, \qquad Y = \lambda \Phi_\beta \alpha_j + \sigma W \; . \tag{II.5}$$

It is assumed that the observations $\{Y_n,\, n \in \mathbb{N}\}$ are independent random variables, generated as follows :

$$\begin{cases} J_n \sim \text{Multi}(1, \omega) \, , \\ \lambda_n \sim \text{Gamma}(a, b) \, , \\ \beta_n \,|\, J_n = j \sim \mathcal{N}_{d_\beta}(0_{d_\beta}, \Gamma_j) \; , \end{cases} \tag{II.6}$$

where Multi denotes the multinomial distribution, (a, b) the parameters of the Gamma distribution (assumed known), 0_{d_β} the d_β-dimensional null vector and Γ_j the deformation covariance matrix associated to the $\mathcal{C}_j$. In section 6 , different covariance models are used in function of the deformation model adopted. We stress that the distribution of the scaling parameter is independent of the class index, while the deformation prior distribution is class-dependent.

In the sequel we assume that $\{W_n,\, n \in \mathbb{N}\}$ is a vector-valued white noise with zero-mean and identity covariance matrix. The extension to more general covariance is straightforward.

Hence, conditionally on the class index J_n, the global scale λ_n and local deformation β_n, the observation Y_n has a Gaussian distribution :

$$Y_n \,|\, J_n = j, \lambda_n, \beta_n, \sim \mathcal{N}_{|\Omega|}(\lambda_n \Phi_{\beta_n} \alpha_j, \sigma^2 \mathrm{Id}_{|\Omega|}) \; . \tag{II.7}$$

Denote by Θ the set of parameters

$$\Theta = \bigcup_{j=1}^{C} \left\{ \, (\alpha_j, \Gamma_j, \omega_j, \sigma) \,|\, \alpha_j \in \mathcal{A},\, \Gamma_j \in \mathcal{M}^+(\mathbb{R}),\, \omega_j \in (0,1),\, \sigma > 0 \, \right\} \cap \left\{ \sum_{j=1}^{C} \omega_j = 1 \right\} \; . \tag{II.8}$$

where $\mathcal{M}^+(\mathbb{R})$ is the set of $d_\beta \times d_\beta$ positive definite matrices.

Let X_n be the random vector $X_n = (\beta_n, \lambda_n)$ taking its values in $\mathsf{X} = \mathsf{B} \times \mathbb{R}^{+*}$ with dimension $d = d_\beta + 1$. In the sequel, we will use the formalism and the terminology of the incomplete data model ; see [MK07]. In this formalism, the observation Y_n stands for the incomplete data, (J_n, X_n) are the missing data and (J_n, X_n, Y_n) are the the complete data. For a given value of the parameter $\theta \in \Theta$, the complete data likelihood f_θ writes :

$$f_\theta(j, x, y) = p_\theta(y \,|\, j, x) g_\theta(x \,|\, j) \omega_j \; , \tag{II.9}$$

where, for a given value of the parameter $\theta \in \Theta$, p_θ is the conditional density of the observations given the missing data and g_θ is the conditional density of the scaling factor and the local deformation parameter given the class index. Using (II.7) and (II.6), these

densities write

$$p_\theta(y \,|\, j, x) \propto \exp\left(-(1/2\sigma^2)\|y - \lambda\Phi_\beta\alpha_j\|^2\right) , \tag{II.10}$$

$$g_\theta(x \,|\, j) \propto \exp\left(-(1/2)\beta^T\Gamma_j^{-1}\beta\right)\lambda^{a-1}\exp(-b\lambda) . \tag{II.11}$$

The incomplete data likelihood is obtained by marginalizing the complete data likelihood with respect to the missing data (the scaling parameter, the deformation and the class index).

3 Sequential parameter estimation with the Monte-Carlo Online EM

In its original version [DLR77], the Expectation-Maximization (EM) is a batch algorithm to perform maximum likelihood estimation in incomplete data models. It produces a sequence of parameters, in such a way that the observed likelihood is increased at each iteration. Each iteration is decomposed into two steps. In the E-step, the conditional expectation of the complete data log-likelihood function given the observations and the current fit of the parameters is computed ; in the M-step, the parameters are updated by maximizing the conditional expectation computed in the E-step.

In this paper, we focus on a learning setup in which the observations are obtained sequentially from an unknown probability distribution $\mathbb{P}_\star$ and the parameters are updated as soon as a new observation is available. Among several sequential learning algorithms designed to estimate parameters in missing data models, the Online EM algorithm proposed in [CM07] sticks closely to the original EM methodology [DLR77]. It does not require to compute the gradient of the incomplete data likelihood nor the inverse of the complete data Fisher information matrix. Under some mild assumptions, it is shown in [CM07] that, even when the model is misspecified, the algorithm converges to the set of stationary points of the Kullback-Leibler divergence between the observation likelihood $\mathbb{P}_\star$ (which does not necessarily belongs to the statistical model) and the incomplete data likelihood $\mathbb{P}_\theta$.

For a given value of the parameter $\theta \in \Theta$, denote by $\pi_\theta(\,\cdot \,|\, Y_n)$ the posterior density of the missing data (J_n, X_n), given the observation Y_n.

Starting from an initial guess $\hat{\theta}_0 \in \Theta$, the E-step of the n-th iteration consists in computing the function $\hat{Q}_n : \Theta \to \mathbb{R}$ initialized with $\hat{Q}_1(\theta) = \mathbb{E}_{\hat{\theta}_0}\left[\log f_\theta(J_1, X_1, Y_1) \,|\, Y_1\right]$ and defined recursively for all $n > 1$ by :

$$\hat{Q}_n(\theta) = \hat{Q}_{n-1}(\theta) + \rho_n\left(\mathbb{E}_{\hat{\theta}_{n-1}}\left[\log f_\theta(J_n, X_n, Y_n) \,|\, Y_n\right] - \hat{Q}_{n-1}(\theta)\right) , \quad Y_n \sim \mathbb{P}_\star \tag{II.12}$$

where $\{\rho_n\}_{n>0}$ is a decreasing sequence of positive step sizes. In the M-step, the next estimator $\hat{\theta}_n$ is obtained by maximizing

$$\hat{\theta}_n = \arg\max_{\theta\in\Theta} \hat{Q}_n(\theta) . \tag{II.13}$$

Under our model specification, the complete data log-likelihood belongs to a curved exponential family *i.e.* for a given parameter $\theta \in \Theta$, $\log f_\theta$ writes

$$\log f_\theta(j, x, y) = t(\theta) + \langle r(\theta), S(j, x, y)\rangle , \tag{II.14}$$

where the function t is given by

$$t(\theta) = \log\frac{b^a}{\mathcal{G}(a)} - \frac{|\Omega|}{2}\log 2\pi\sigma^2 - d_\beta \log 2\pi$$

and the functions $r(\theta) = (r_1(\theta), \ldots, r_C(\theta))$ and $S(j,x,y) = (S_1(j,x,y), \ldots, S_C(j,x,y))$, are defined for all $i \in \{1, \ldots, C\}$ $r_i : \Theta \to \mathsf{S}$ and $S_i : \mathsf{J} \times \mathsf{X} \times \mathsf{Y} \to \mathsf{S}$ as :

$$r_i(\theta) = (1/2)\left(2\log(\omega_i) - \log\det\Gamma_i, 2\sigma^{-2}\alpha_i, -\sigma^{-2}(\alpha_i\alpha_i^T), -{\Gamma_i^{-1}}^T, -\sigma^{-2}, -2b, 2(a-1)\right) ,$$
$$S_i(j,x,y) = \delta_i(j)\left(1, \lambda\phi_\beta^T y, \lambda^2\phi_\beta^T\phi_\beta, \beta\beta^T, \|y\|^2, \lambda, \log\lambda\right) .$$

As a consequence, the two steps of the online EM consist in computing for all $i \in \{1, \ldots, C\}$ the sequence $\{\hat{s}_{n,i},\, n \in \mathbb{N}\}$ starting from some $\hat{s}_{0,i} \in \mathsf{S}$ with the recursion

$$\hat{s}_{n,i} = \hat{s}_{n-1,i} + \rho_n\left(\bar{s}_{n,i}(Y_n; \hat{\theta}_{n-1}) - \hat{s}_{n-1,i}\right), \quad Y_n \sim \mathbb{P}_\star , \tag{II.15}$$

where $\bar{s}_{n,i}(Y_n; \hat{\theta}_{n-1}) = \mathbb{E}_{\hat{\theta}_{n-1}}\left[\, S_i(J_n, X_n, Y_n) \,|\, Y_n\right]$ and then updating the parameters according to

$$\hat{\theta}_n = \bar{\theta}(\hat{s}_n) , \qquad \hat{s}_n = (\hat{s}_{n,1}, \ldots, \hat{s}_{n,C}) , \tag{II.16}$$

where for all $s = (s_1, \ldots, s_C) \in \mathsf{S}^C$

$$\bar{\theta}(s) = \arg\max_{\theta\in\Theta}\; t(\theta) + \sum_{i=1}^{C} \langle r_i(\theta), s_i\rangle . \tag{II.17}$$

However, this algorithm is mainly of theoretical interest because the conditional expectation $\bar{s}_{n,i}(Y_n; \hat{\theta}_{n-1})$ cannot be computed in closed form. Intractable E-steps have already been addressed for batch EM algorithms. In [DLM99], the authors proved the convergence of the Stochastic Approximation EM (SAEM) algorithm in which the E-step is replaced by a stochastic approximation making use of realizations of the missing data generated according to the posterior distribution. Still, extending the SAEM algorithm to the online setup is not feasible in our case. Indeed, in our model, the distribution of $\pi_{\hat{\theta}_{n-1}}(\,\cdot\,|\,Y_n)$ is difficult to sample directly. An alternative to the SAEM algorithm was proposed in [KL04] : the authors suggested to use Markov chain Monte Carlo (MCMC) methods (see [ADFDJ03] for an introduction) to obtain samples from the posterior distribution.

In this paper, we adapt this approach to the sequential setting outlined above leading to the MCoEM (Monte Carlo online EM) algorithm. It is a 3-step iterative algorithm. Given the current fit of parameter $\hat{\theta}_{n-1}$ and an observation $Y_n \in \mathbb{P}_\star$, the algorithm proceeds as follows :

(1) *simulation step* : sample a reversible Markov chain $\{J_n^{(k)}, X_n^{(k)},\, k > 0\}$ on $(\mathsf{J} \times \mathsf{X})$, having $\pi_{\hat{\theta}_{n-1}}(\,\cdot\,|\,Y_n)$ as its unique invariant distribution,

(2) *stochastic approximation step* : update for each class $i \in \{1, \ldots, C\}$, the complete data sufficient statistics using the following recursion

$$\hat{s}_{n,i} = \hat{s}_{n-1,i} + \rho_n\left(\frac{1}{v_n}\sum_{k=1}^{v_n} S_i(J_n^{(k)}, X_n^{(k)}, Y_n) - \hat{s}_{n-1,i}\right), \tag{II.18}$$

where v_n is the number of MCMC iterations performed at the n-th iteration of the MCoEM algorithm,

(3) *maximization step* : update the parameter $\hat{\theta}_n$ by maximizing the function :

$$\hat{\theta}_n = \bar{\theta}(\hat{s}_n) . \tag{II.19}$$

The maximization is in closed form. We denote by $(\hat{s}_{n,i}^{(1)}, \ldots, \hat{s}_{n,i}^{(7)})$ the vector $\hat{s}_{n,i}$ components. The parameters $\{\hat{\omega}_{n,i}, \hat{\Gamma}_{n,i}, \hat{\alpha}_{n,i}\}_{1 \leq i \leq C}$ and $\hat{\sigma}_n^2$ are updated as follows :

$$\hat{\omega}_{n,i} = \frac{\hat{s}_{n,i}^{(1)}}{\sum_{i=1}^{C} \hat{s}_{n,i}^{(1)}}, \qquad \hat{\Gamma}_{n,i} = \frac{\hat{s}_{n,i}^{(4)}}{d_\beta \, \hat{s}_{n,i}^{(1)}}, \tag{II.20}$$

$$\hat{\alpha}_{n,i} = (\hat{s}_{n,i}^{(3)})^{-1} \hat{s}_{n,i}^{(2)}, \qquad \hat{\sigma}_n^2 = \frac{1}{|\Omega|} \left(\sum_{i=1}^{C} \hat{s}_{n,i}^{(5)} - 2\hat{\alpha}_{n,i}^T \hat{s}_{n,i}^{(2)} + \hat{\alpha}_{n,i}^T \hat{s}_{n,i}^{(3)} \hat{\alpha}_{n,i} \right) . \tag{II.21}$$

4 Approximating the hidden data joint posterior distribution

In this section, we construct a transition kernel K to sample the target distribution $\pi_\theta(\cdot \,|\, Y)$ (for simplicity, the iteration index n of the EM algorithm is omitted in this section). Following the classical Markov Chain Monte Carlo (MCMC) principles, we construct a kernel K which is reversible with respect to $\pi_\theta(\cdot \,|\, Y)$; see [CR99]. The basic idea, stemming from [CC95], is to specify a joint distribution over the class index J and auxiliary variables $X_1, \ldots, X_C$, where for all $j \in \mathsf{J}$, $X_j \in \mathsf{X}$ is a deformation parameter associated to the class $\mathcal{C}_j$. We stress that, in this approach, we sample at each iteration deformation parameters for each class. To specify the joint distribution, we introduce the *pseudo-priors* or *linking densities* which are denoted $\{\zeta_j,\, j \in \mathsf{J}\}$. Note that whereas the knowledge of the normalizing constant is not required for an MCMC algorithm, the normalizing constant of the pseudo-priors are assumed to be known, *i.e.* the pseudo-priors $\{\zeta_j,\, j \in \mathsf{J}\}$ should integrate to 1. Also, it is assumed that exactly sampling from the pseudo-priors is doable (and is computationally inexpensive). Given an observation $Y = y$, the joint posterior density $\tilde{\pi}_\theta(\cdot \,|\, y)$ on the product space $\mathsf{J} \times \mathsf{X} \times \cdots \times \mathsf{X}$ is defined as :

$$\begin{aligned} \tilde{\pi}_\theta(j, x_1, \ldots, x_C \,|\, y) &= \pi_\theta(j, x_j \,|\, y) \prod_{i \neq j} \zeta_i(x_i) , \\ &\propto p_\theta(y \,|\, j, x_j) g_\theta(x_j \,|\, j) \omega_j \prod_{i \neq j} \zeta_i(x_i) , \end{aligned} \tag{II.22}$$

where ω_j, g_θ and p_θ are defined in (II.6), (II.10) and (II.11) respectively. The choice of pseudo-priors does not influence the target posterior distribution, which is the marginal of $\tilde{\pi}_\theta(\cdot \,|\, y)$ with respect to to the missing deformation parameters :

$$\pi_\theta(j, x \,|\, y) = \int \cdots \int \tilde{\pi}_\theta(j, x_{1:j-1}, x, x_{j+1:C} \,|\, y) \mathrm{d}x_{-j} , \tag{II.23}$$

where for all $(i, j) \in \mathsf{J}^2$, such that $i < j$, $a_{i:j} = (a_i, a_{i+1}, \ldots, a_j)$ and for all $i \in \mathsf{J}$, $a_{-i} = \{a_j\}_{j=1, j \neq i}^{C}$.

A Metropolis-within-Gibbs sampler is used to simulate a Markov chain

$$\{J^{(k)}, X_1^{(k)}, \ldots, X_C^{(k)},\, k \in \mathbb{N}\}$$

on the product space $(\mathsf{J} \times \mathsf{X} \times \ldots \times \mathsf{X})$, which targets the density $\tilde{\pi}_\theta(\cdot \,|\, Y)$. Suppose the state $(j, x_1, \ldots, x_C)$, the conditional posterior distributions required for the Gibbs sampler are :

$$\tilde{\pi}_\theta(j \,|\, x_{1:C}, y) \propto p_\theta(y \,|\, j, x_j) g_\theta(x_j \,|\, j) \omega_j \prod_{i \neq j} \zeta_i(x_i) , \tag{II.24}$$

$$\tilde{\pi}_\theta(x_i \,|\, j, x_{-i}, y) \propto \begin{cases} p_\theta(y \,|\, j, x_j) g_\theta(x_j \,|\, j) , & i = j \\ \zeta_i(x_i) . & i \neq j \end{cases} \tag{II.25}$$

To sample X_j, a Metropolis-Hastings kernel $P_\theta(x_j, j, y\,;\,\cdot)$ having $\pi_\theta(\,\cdot\,|\,j, y)$ as its unnormalized stationary distribution is used. For $i \neq j$, the parameters $\{X_i,\, i \neq j\}$ are sampled from the pseudo-priors. The model indicator is sampled from the discrete distribution specified by (II.24). The Carlin and Chib kernel $\tilde{K}^{\mathsf{CC}}$ may therefore be expressed as a product of two kernels $\tilde{K}_1^{\mathsf{CC}}$ and $\tilde{K}_2^{\mathsf{CC}}$, the first updates the auxiliary variables and the later the class index and the deformation parameter consistent with the class index :

$$\begin{cases} \tilde{K}_1^{\mathsf{CC}}(j, x_{1:C}; j', \mathrm{d}x'_{1:C}) = \delta_{j,x_j}(j', \mathrm{d}x'_j) \prod_{i\neq j} \zeta_i(\mathrm{d}x'_i)\ , \\ \tilde{K}_2^{\mathsf{CC}}(j, x_{1:C}; j', \mathrm{d}x'_{1:C}) = \tilde{\pi}_\theta(j'\,|\,x_{1:C}) P_\theta(x_{j'}, j', y\,;\,\mathrm{d}x'_{j'}) \prod_{i\neq j'} \delta_{x_i}(\mathrm{d}x'_i)\ . \end{cases} \tag{II.26}$$

Note that these two kernels are stationary with respect to $\tilde{\pi}_\theta(\,\cdot\,|\,y)$ and therefore so his the product $\tilde{K}^{\mathsf{CC}} = \tilde{K}_1^{\mathsf{CC}} \tilde{K}_2^{\mathsf{CC}}$. The Algorithm 7 explains how the marginal chain K is obtained from $\tilde{K}^{\mathsf{CC}}$.

Algorithm 7 Transition $(J^{(k)}, X^{(k)}) \to (J^{(k+1)}, X^{(k+1)})$ of the Carlin and Chib sampler

Input : the current state $(J^{(k)}, X^{(k)}) = (j, x)$, the parameter $\theta \in \Theta$, the observation y

1 - **Simulation of the auxiliary variables :** for all $i \in \{1, \ldots, C\}$ draw the auxiliary variables $(X_1, \ldots, X_C)$ as follow

$$\begin{cases} X_j \sim \delta_x(\,\cdot\,)\ , \\ X_i \sim \zeta_i, \quad \forall i \neq j\ , \end{cases}$$

2 - **Simulation of the class index :** Draw

$$\mathbb{P}[J^{(k+1)} = j'] \propto p_\theta(y\,|\,j', x_{j'}) g_\theta(x_{j'}\,|\,j') \omega_{j'} \prod_{i\neq j'} \zeta_i(x_i)\ ,$$

3 - **Simulation of the parameter :** Draw

$$X^{(k+1)} \sim P_\theta(x_{j'}, j', y\,;\,\cdot)\ .$$

4 - **Output :** the next state $(J^{(k+1)}, X^{(k+1)})$.

The specification of the linking densities is essential for sampling efficiency. Ideally, these densities should be close to the marginal posterior, that is, for all $j \in \{1, \ldots, C\}$, the density $x \to \zeta_j(\,x\,)$ should be chosen as a proxy to $x \to \pi_\theta(x\,|\,j, y)$. An idea is for instance to set the pseudo-prior density as a Gaussian approximation of the target density. Such an approximation can be obtained using the Laplace method [Wol93] or other approximate Bayesian sampling method. Under the (weak) assumption that the function $x \to \pi_\theta(x\,|\,j, y)$ admits a maximum,

$$x_j^\star = \arg\max_{x\in\mathsf{X}} \pi_\theta(x\,|\,j, y)\,, \tag{II.27}$$

the Taylor-expansion of the logarithm of $\pi_\theta(x\,|\,j, y)$ writes :

$$\log \pi_\theta(x\,|\,j, y) = \log \pi_\theta(x_j^\star\,|\,j, y) + \frac{1}{2}(x - x_j^\star)^T H_j (x - x_j^\star) + o(\|x - x_j^\star\|^2)\,, \tag{II.28}$$

where for all $j \in \{1, \ldots, C\}$, H_j is the Hessian matrix, whose coefficients are given for all $(q, r) \in \{1, \ldots, d\}^2$ by :

$$[H_j]_{q,r} = \left.\frac{\partial^2}{\partial x_q \partial x_r} \log \pi_\theta(x\,|\,j, y)\right|_{x=x_j^\star}\ . \tag{II.29}$$

Note that for better readability, for all $j \in \{1, \dots, C\}$, the dependence of the linking densities ζ_j, and the parameters $X_j^\star$, H_j on Y and θ is not made explicit in these notations, but may exist.

The previous discussion suggests that $\mathcal{N}_d(x_j^\star, -H_j^{-1})$ is a sensible candidate for ζ_j. The pseudo-priors parameters $x_j^\star$ may obtained using standard nonlinear optimization methods. Since $X^\star$ is only used in the pseudo-prior specification, the precision of the optimizer does not matter much and simple heuristics can be used.

Our proposed kernel shares some similarities with that proposed in [AK10], which also makes use of auxiliary variable $\{X_1, \dots, X_C\}$: they propose to first sample the class index $J \sim \pi_\theta(\,\cdot\,|\,Y)$ and then to draw $X \sim \pi_\theta(\,\cdot\,|\,J, Y)$. However, since sampling the class index from the posterior distribution is not doable (indeed the weights are proportional to $\{\pi_\theta(j, Y),\, j \in \mathsf{J}\}$ which are not analytically tractable), they sample auxiliary variables $\{X_1^{(k)}, \dots, X_C^{(k)},\, k > 0\}$ from C independent Markov chains each targeting $\pi_\theta(\,\cdot\,|\,j, Y)$, $j \in \mathsf{J}$ in order to approximate the posterior weights. These approximate weights allow to sample J and then a parameter samples X is drawn using a Markov chain targeting $\pi_\theta(\,\cdot\,|\,J, Y)$. However, this scheme is computationally intensive all the more that [AK10] uses a batch learning setup which implies that at each iteration, as many latent variables $\{J_n^{(k)}, X_n^{(k)},\, k > 0\}$ as there are observations need to be sampled. Moreover, the samples provided by our Algorithm 7 are *exact* whereas, because it involves approximate weights, those from the algorithm proposed in [AK10] are *noisy*. Note that this does not call into question the merits of the learning setup proposed in [AK10] which has been theoretically proven using the *noisy* samples from the posterior distribution.

5 A mixture of Gaussian regression

We first apply the MCoEM algorithm to estimate the parameters of a Mixture of Gaussian Regressions, which was already studied in [CMR05]. This model is actually analogous to a mixture of deformable templates model, where the observations are $|\Omega| = 1$ dimensional data *i.e.* $Y \in \mathbb{R}$, the regression parameters (similar to the template parameters) are $m = 3$ dimensional vectors such that $\alpha \in \mathcal{A} = \mathbb{R}^3$ and the missing data are a scalar $\beta \in \mathbb{R}$. In this setting, compared to the models (II.5) and (II.6), we do not take into account the scaling factor λ and therefore $d_\beta = d = 1$. We consider a mixture model with $C = 2$ components, where the observations are simulated as follows :

$$\begin{cases} J_n \sim \text{Multi}(\omega_1, \omega_2)\,, \\ \beta_n \,|\, J_n = j \sim \mathcal{N}(\mu_j, \gamma^2)\,, \\ Y_n \,|\, J_n = j, \beta_n \sim \mathcal{N}(\Phi_{\beta_n}\alpha_j, \sigma_j^2)\,. \end{cases} \tag{II.30}$$

The vector of regressors $\Phi_{\beta_n} \in \mathcal{M}_{1,3}(\mathbb{R})$ is defined as in [CMR05] with $\Phi_{\beta_n} = (1, \beta_n, {\beta_n}^2/10)$. The model is exponential and the sufficient statistics vector writes

$$S(j, \beta, Y) = (S_1(j, \beta, Y), S_2(j, \beta, Y))\,,$$

where for $i \in \{1, 2\}$,

$$S_i(j, \beta, Y) = \delta_{i=j}(1, \Phi_\beta^T Y, \Phi_\beta^T \Phi_\beta, Y^2, \beta, \beta^2)\,.$$

The regression parameters (α_1, α_2) are unknown and we wish to estimate them using an EM algorithm in an online setting. The E and M steps respectively write with the notations introduced in the Section 3 for $i \in \{1, 2\}$

$$\text{(i) } \hat{s}_i = \mathbb{E}[S_i(j, \beta, Y)\,|\,Y] \quad \text{and} \quad \text{(ii) } \hat{\alpha}_i = (\hat{s}_i^{(3)})^{-1}\hat{s}_i^{(2)}\,. \tag{II.31}$$

We consider two learning setups :

(a) First, we assume that for each data Y_n, the missing data β_n is known. This situation corresponds exactly to the example proposed in [CM07] to illustrate the Online EM algorithm. Indeed, since in this context the missing data reduce to the class index, the conditional expectation of the complete data log-likelihood is tractable :

$$\begin{cases} \hat{s}_i^{(2)} = \mathbb{E}_{\hat{\theta}}[S_i^{(2)}(J_n, \beta_n, Y_n) \,|\, \beta_n, Y_n] = \Phi_{\beta_n} Y_n \, \mathbb{P}_{\hat{\theta}}[J_n = j \,|\, \beta_n, Y_n] \,, \\ \hat{s}_i^{(3)} = \mathbb{E}_{\hat{\theta}}[S_i^{(3)}(J_n, \beta_n, Y_n) \,|\, \beta_n, Y_n] = \Phi_{\beta_n}^T \Phi_{\beta_n} \, \mathbb{P}_{\hat{\theta}}[J_n = j \,|\, \beta_n, Y_n] \,, \end{cases}$$

where

$$\pi_{\hat{\theta}}(J_n = j \,|\, \beta_n, Y_n) \propto \frac{\omega_j}{\sigma_j \gamma_j} \exp\left[-(\beta_n - \mu_j)^2/(2\gamma_j^2) - (Y_n - \Phi_{\beta_n}\alpha_j)^2/(2\sigma_j^2)\right] \,. \quad \text{(II.32)}$$

As a consequence, the Online EM can be implemented.

(b) Then, we assume that the missing data consist in the vector $\{(J_n, \beta_n),\, n \in \mathbb{N}\}$ *i.e.* the parameter β_n is unknown for each observation Y_n. This is a typical situation where the Online EM cannot be implemented. Indeed, due to the intractable conditional expectation

$$\hat{s}_i = \sum_{j=1}^{C} \int S_i(j, \beta, Y) \exp\left[-(\beta - \mu_j)^2/(2\gamma_j^2) - (Y - \Phi_\beta \alpha_j)^2/(2\sigma_j^2)\right] \mathrm{d}\beta \,,$$

the E-step of the Online EM cannot be performed. As a consequence, one may turn to the MCoEM to estimate the parameters. We study the convergence of the MCoEM using two different algorithms to sample the missing data : the Carlin and Chib algorithm detailed in Algorithm 7 and the Metropolis-within-Gibbs sampler which alternatively updates

(i) the class index J_n given the current observation Y_n and the current parameter β_n according to (II.32)

(ii) the parameter β_n given the current observation Y_n and the current class index J_n with a Metropolis kernel which targets the following distribution defined for all $A \in \mathcal{B}(\mathbb{R})$ by

$$\pi_{\hat{\theta}}(\beta_n \in A \,|\, J_n = j, Y_n) \propto \int_A \frac{\omega_j}{\sigma_j \gamma_j} e^{(\beta - \mu_j)^2/(2\gamma_j^2) - (Y_n - \Phi_\beta \alpha_j)^2/(2\sigma_j^2)} \mathrm{d}\beta \,.$$

As suggested previously, the pseudoprior distributions ζ_1, ζ_2 involved in the Carlin and Chib algorithm are set as two Gaussian distributions with mean given by (II.27) and covariance matrix given by (II.29). In this model, these two parameters can be computed in closed form. Indeed, for $j \in \{1, 2\}$, $\beta_j^\star$ is the solution of a cubic equation whose parameters are all tractable. Moreover, since $d = 1$, the Hessian matrix computation (II.29) boils down to the evaluation of the second derivative of the function $\beta \to \pi_{\hat{\theta}}(\beta \,|\, Y_n, j)$ in $\beta_j^\star$, which does not present a major difficulty.

We run $n = 10000$ iterations of the Online EM and the MCoEM algorithms starting from the same initial guess. At each iteration, a new observation is simulated according to (II.30) and is processed only once. The parameters $(\omega_1, \omega_2, \mu_1, \mu_2, \gamma, \sigma_1, \sigma_2)$ are set constant to their *exact* value throughout the process while the estimate $\{\hat{\alpha}_{n,j},\, n \leq 10000\}_{j=1}^C$ are updated at each iteration. Moreover, as recommended in [CMR05] :

- The parameters are not updated for the 20 first iterations. This time lag allows to obtain a sufficient statistics estimate $\{\hat{s}_1, \hat{s}_2\}$ which lives in the sufficient statistics space. In particular, $\hat{s}_1^{(3)}$ and $\hat{s}_2^{(3)}$ should be invertible matrices in this model.
- The sequence of positive step-size involved in the stochastic approximation step II.18 is set as $\rho_n = n^{-0.6}$.
- A Polyak-Ruppert averaging technique [Pol90, Rup88] is used as a post-processing step : instead of considering the estimate $\hat{\alpha}_{n,j}$ provided by (II.31), we use

$$\tilde{\alpha}_{n,j} = \begin{cases} \hat{\alpha}_{n,j} & n \leq n_0 \\ (n - n_0)^{-1} \sum_{k=n_0}^{n} \hat{\alpha}_{k,j} & n > n_0 \end{cases}$$

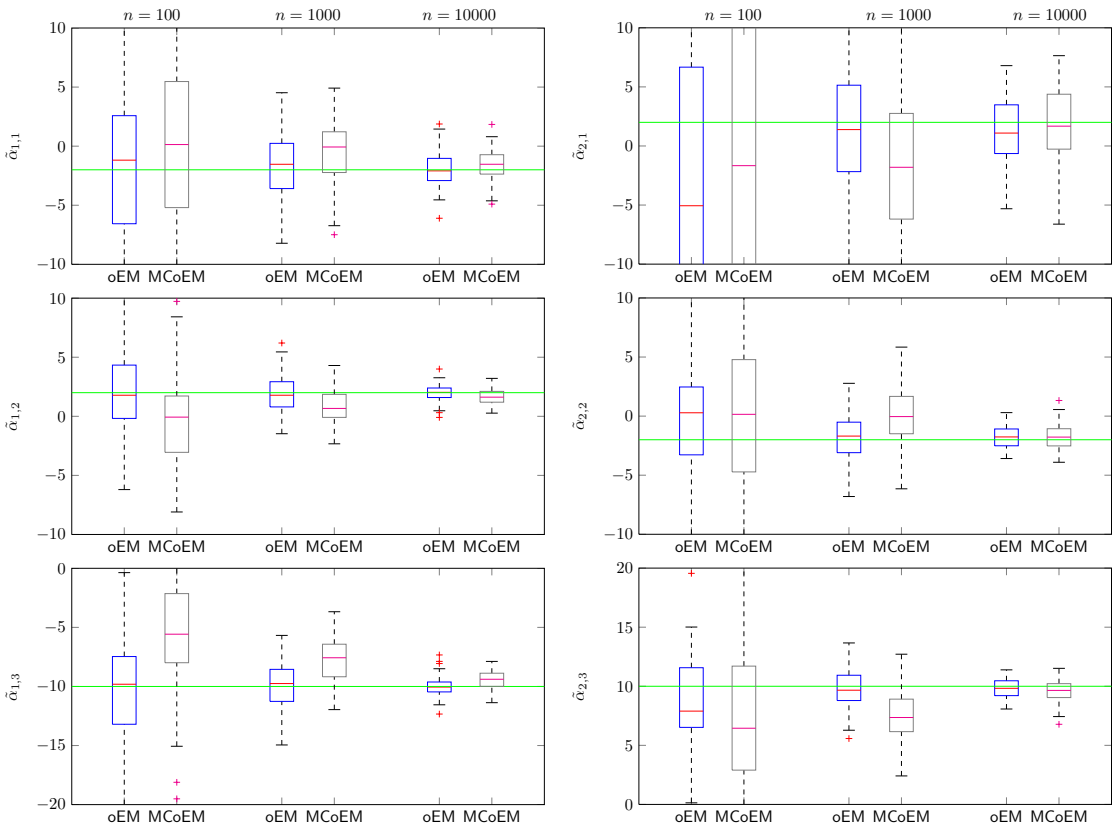
  For a stochastic approximation involving a sequence $\{\rho_n = n^{-r}, n \in \mathbb{N}, 0.5 \leq r \leq 1\}$, it was shown in [Pol90, PJ92] that $\tilde{\alpha}_{n,j}$ converges at the rate $1/\sqrt{n}$ for all $0.5 \leq r \leq 1$.

The Figure II-1 compares the learning setups (a), where the inference is performed with the Online EM and (b), where the inference is performed with the MCoEM coupled with the Carlin and Chib algorithm. In both scenario, the Polyak-Ruppert averaging procedure begins at the iteration $n_0 = 7500$. Starting from the same initial guess for α, 100 runs of each learning algorithms were performed and the Figure II.1(a) summarizes the results with the box-and-whisker plots for $n = 100$, $n = 1000$ and $n = 10000$. The true value of the regression parameter α is displayed with the green lines, the estimation of the three components of the class $j = 1$ and $j = 2$ regression parameter are respectively positioned on the left column and on the right column of Figure II.1(a). This experiment shows that even though in the early iterations, the MCoEM estimate features a much more significant bias and a larger variability than the Online EM estimate, after $n = 10000$ iterations, the estimates are very similar. The Figures II.1(b) and II.1(c) display a sample path of $n = 10000$ iterations of the Online EM and the MCoEM respectively. This highlights the quantitative similarity of the two approaches whereas the Online EM requires the parameters β_n to be known while the MCoEM can still be implemented when it is missing.

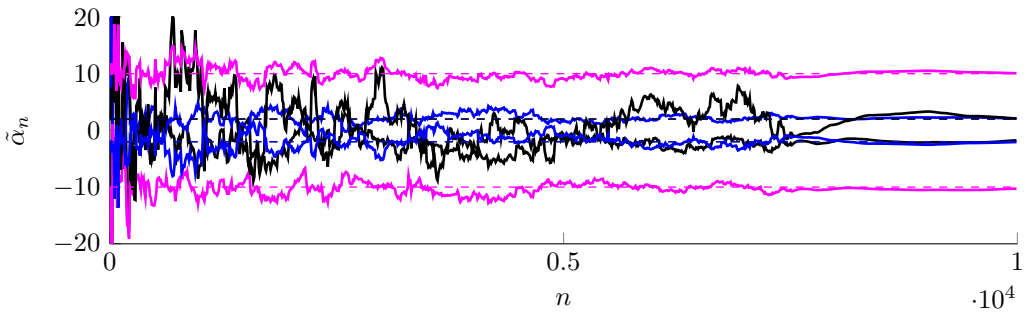
We also implement the MCoEM algorithm with a Metropolis-within-Gibbs sampler to simulate the missing data. As already stated in the Preamble D .3, the Metropolis-within-Gibbs sampler (and more generally the Gibbs samplers) does not allow an efficient sampling of a distribution involving a selection model random variable. As a result, the estimates $\{\tilde{\alpha}_{n,j}, n \in \mathbb{N}\}$ does not converge to the true value of the parameter as illustrated by the Figure II-2. As in the Figure II-1, the box-and-whisker plot proposed in the Figure II.2(a) summarizes 100 runs of the MCoEM with a Gibbs sampler and the Figure II.2(b) provides an example of sample path of the 10000 iterations of the MCoEM . The significant bias is the result of the inappropriate parameters $\{(J_n, \beta_n), n \in \mathbb{N}\}$ sampled by the Gibbs sampler : instead of switching from one class to another when needed, the Gibbs sampler tends to produce a parameter β_n which will fit the possibly *wrong* current class. As a consequence, *huge deformations* are expected such that the matrix $\hat{s}_i^{(3)}$ might become barely non-inversible in some cases. The online setup increases significantly the vulnerability of the parameter estimation to this problem since only one *missampled* missing data sampling may definitely affect the estimate. This highlights the important role played by the missing data sampler and justifies the use of the Carlin and Chib algorithm instead of the Gibbs sampler.

This toy-example shows that :

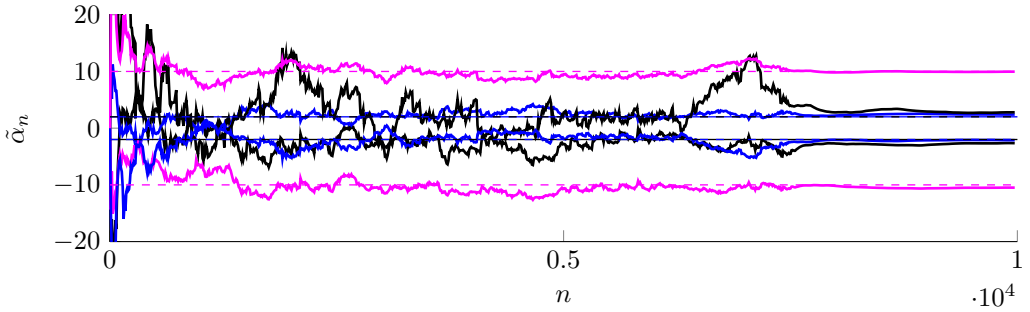
(i) the MCoEM is an alternative to the Online EM when the conditional expectation cannot be analytically computed, typically when some data are censored,



(a) Estimation of the regression parameters for 100 runs of the Online EM and MCoEM algorithms



(b) Sample path of the Online EM



(c) Sample path of the MCoEM implemented with the Carlin and Chib sampler

FIGURE II-1 – Estimation of the regression parameters with the Online EM and the MCoEM algorithms

(ii) the estimates are similar even though the MCoEM convergence is slower than the Online EM ,

(iii) the combination of the MCoEM with the Carlin and Chib sampler provides a robust and unbiased estimate of unknown parameters in mixtures of missing data exponential models.

6 Numerical results

We evaluate the performance of our online learning algorithm by inferring two types of data : growth velocity curves and handwritten digits. These three examples illustrate the flexibility and the effectiveness of the proposed online learning algorithm.

6 .1 Growth velocity curve study

The growth velocity curve example is a classical benchmark in curve registration [Ram06, Zho08] ; it is used here for illustrative purposes, because the rationale of the model is easy to grasp. The growth curves are obtained from the Berkeley Growth Study data [TS54] and display the evolution of the growth velocity between 2 and 18 years, for 39 boys and 54 girls ; see Figure II-3. The objective of the algorithm is to retrieve a standard growth profile for boys and girls from the unlabeled growth velocity curves. The growth velocity curves, plot the growth velocity of individuals observed at $|\Omega| = 31$ landmarks $\Omega = \{u_1, \ldots, u_{|\Omega|}\}$, irregularly spaced, such that for all $s \in \{1, \ldots, |\Omega|\}$, $2 \leq u_s \leq 18$.

Growth profiles may vary from an individual to another, both as a function of the time and in amplitude. The algorithm aims to extract templates for the growth velocity curves : it associates to each observation Y_n a monotonically increasing time warping function $u \mapsto D_{\delta_n}(u)$, $\delta \in \mathsf{D} \subset \mathbb{R}^{d_\mathsf{D}}$, as well as a global scaling parameter λ_n. In this model, note that the global deformation model $\{G_\beta,\, \beta \in \mathsf{B}\}$ boils down to the time warping function and therefore $\mathsf{B} = \mathsf{D}$ and $d = d_\mathsf{D}$. We consider a mixture model with $C = 2$, implying that we aim at retrieving templates for boys and girls growth velocity separately *i.e.* the class index $J_n \in \{1, 2\}$ models the boys and girls clusters. In this illustration, the template is a function $\mathcal{T}_\alpha$ defined on an open segment $\mathbb{U} = (u_i, u_f) = (2, 18)$ parameterized as :

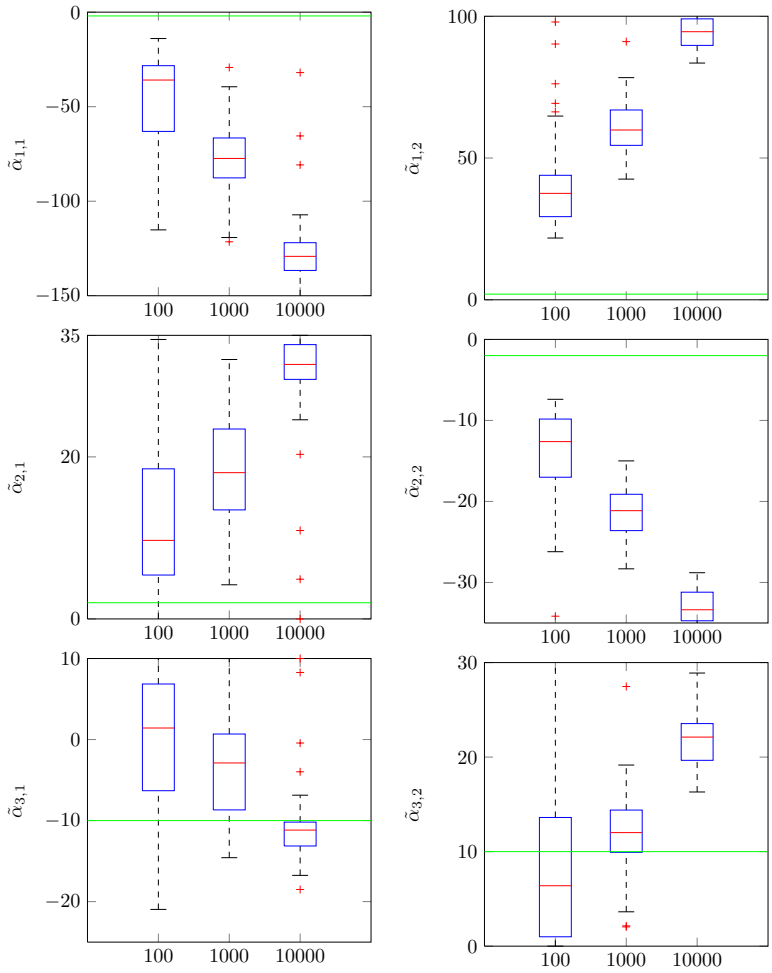
$$\mathcal{T}_\alpha(u) = \sum_{\ell=1}^{m} \alpha_\ell \phi_\ell(u) \;, \quad (\alpha_1, \ldots, \alpha_m) \in \mathcal{A} = \mathbb{R}^{+m} \tag{II.33}$$

where $\{\phi_\ell\}_{\ell=1}^m$ is set as $u \mapsto \phi_\ell(u) = \exp\left(\nu_\ell^{-2}(u - r_\ell)^2\right)$, where $\{r_\ell\}_{\ell=1}^m$ are regularly spaced landmark points in $\mathbb{U}$. The choice of $\{\phi_\ell\}_{\ell=1}^m$ and $\mathcal{A}$ ensures that the template function $u \mapsto f_\alpha(u)$ is a positive function, which is a natural constraint for growth velocity curves. For all $\ell \in \{1, \ldots, m\}$, the bandwidth of ϕ_ℓ is set as $\nu_\ell^2 = -\frac{\min_{u \in \Omega} \|r_\ell - u\|^2}{\log \varepsilon}$, where $\varepsilon \in (0, 1)$ is the value of ϕ_ℓ in the nearest design point of r_ℓ. This choice of bandwidth enables to take into account the irregularly spaced measurement points in Ω. The deformable template model (II.1) simply writes for all $u \in \mathbb{U}$:

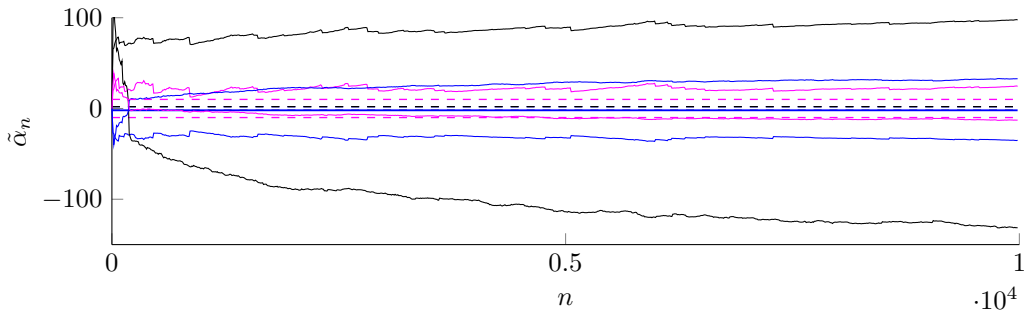
$$\mathcal{Y}_n \in \mathcal{C}_j, \qquad \mathcal{Y}_n(u) = \lambda_n \mathcal{T}_{\alpha_j} \circ D_{\delta_n}(u) + \sigma \mathcal{W}_n(u) \,.$$

In this setting, the time warping function $u \mapsto D_\delta(u)$ is monotonically increasing and should satisfy $D_\delta(u_i) \geq u_i$ and $D_\delta(u_f) \leq u_f$ (indeed, outside (u_i, u_f), the template vanishes (II.33)). In order to satisfy these constraints, we write $D_\delta(\,\cdot\,)$ as :

$$D_\delta(u) = u_i + (u_f - u_i) V_\delta(u) \;, \tag{II.34}$$



(a) Sample path of the MCoEM implemented with the Gibbs sampler



(b) Estimation of the regression parameters for 100 runs of the MCoEM algorithm implemented with a Gibbs sampler

FIGURE II-2 – Estimation of the regression parameters with the MCoEM algorithm coupled with a Gibbs sampler

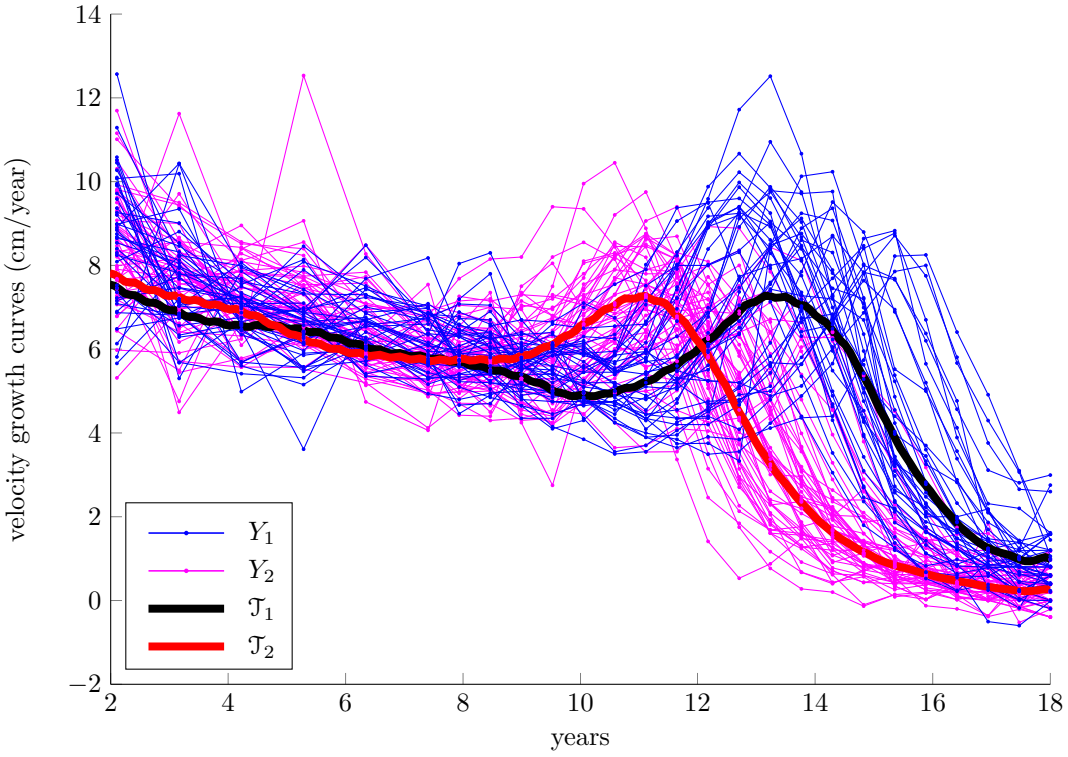


Figure II-3 – Growth velocity samples and templates extraction

where $D_\delta(\,\cdot\,)$ is modeled as proposed in [RL98] with :

$$V_\delta(u) = \frac{\int_{u_i'}^{u} \exp\left[\sum_{k=1}^{d} \delta_k \psi_k(v)\right] \mathrm{d}v}{\int_{u_i'}^{u_f'} \exp\left[\sum_{k=1}^{d} \delta_k \psi_k(v)\right] \mathrm{d}v}\,, \tag{II.35}$$

where $u_i' \leq u_i$ and $u_f' \geq u_f$ allow to satisfy the constraints stated above. For all k in $\{1,\dots,d\}$, $\delta_k \in \mathbb{R}$ and $\{\psi_k\}_{k=1}^d$ is a dictionary of Gaussian kernels centered on the landmark points $\{q_k\}_{k=1}^d$ with the same bandwidth. In this implementation, we set $u_i' = 0$, $u_f' = 20$ and use $d = 20$ regularly spaced landmark points such that $q_1 = u_i'$ and $q_d = u_f'$; the kernel variance is set to 1. Moreover, the prior distribution (II.10) of δ is set with a mean equals to $(1,\dots,1)^T$ and for all $j \in \{1,2\}$ a covariance matrix Γ_j parameterized by the variance γ_j, such that $\Gamma_j = \gamma_j^2 \mathrm{Id}_d$. The estimate $\hat{\gamma}_{j,n}^2$ of γ_j^2 after $n = 1000$ iterations is $\hat{\gamma}_{1,1000}^2 = 0.08$ and $\hat{\gamma}_{2,1000}^2 = 0.07$. A Gamma prior with parameters $a = b = 10$ is assumed for λ.

The Figures II-4-II-5 illustrate the sampling scheme proposed in Section 4 . For $j \in \{1,2\}$, the auxiliary variable X_j introduced in (II.22) consist in $X_j = (\lambda_j, \delta_j)$. Green dots represent an observation along with the templates in plain curves (boys on the left panel and girls on the right panel) and the templates distorted by some deformation parameters $X_1^{(k)} = (\delta_1^{(k)}, \lambda_1^{(k)})$ and $X_2^{(k)} = (\delta_2^{(k)}, \lambda_2^{(k)})$ sampled using the kernel $\tilde{K}^{\mathtt{CC}}$ (dashed curves). For each observation Y, we use 200 iterations of the sampling scheme detailed in Algorithm 7. The pseudo-priors ζ_1 and ζ_2 are set as Gaussian distributions, as specified in 4 . For $j \in \{1,2\}$, the mean λ_j^*, δ_j^* is given with a quasi-Newton optimization method (with an early stopping rule, because the precision of the fit does not matter much) ; in addition, for computational efficiency, the covariance matrix is set as $\hat{\Gamma}_{j,n} = \hat{\gamma}_{j,n}^2 \mathrm{Id}_{d_\beta}$

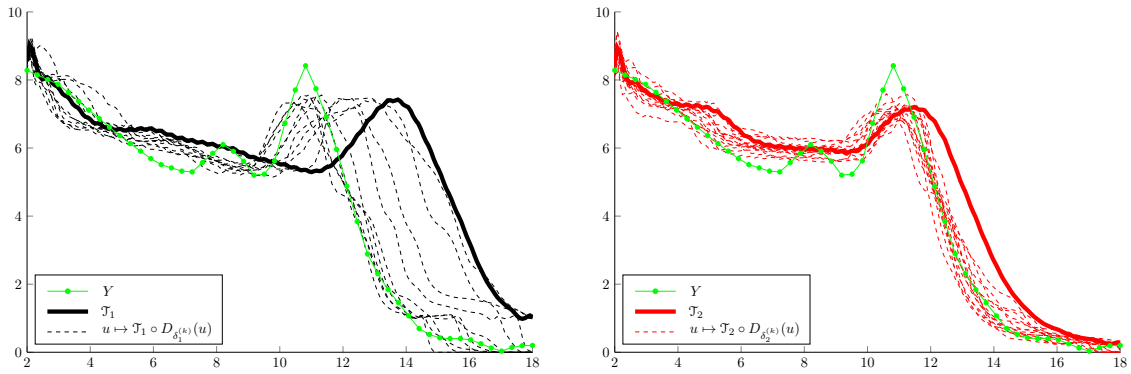


FIGURE II-4 – Sampling of the hidden data posterior distribution

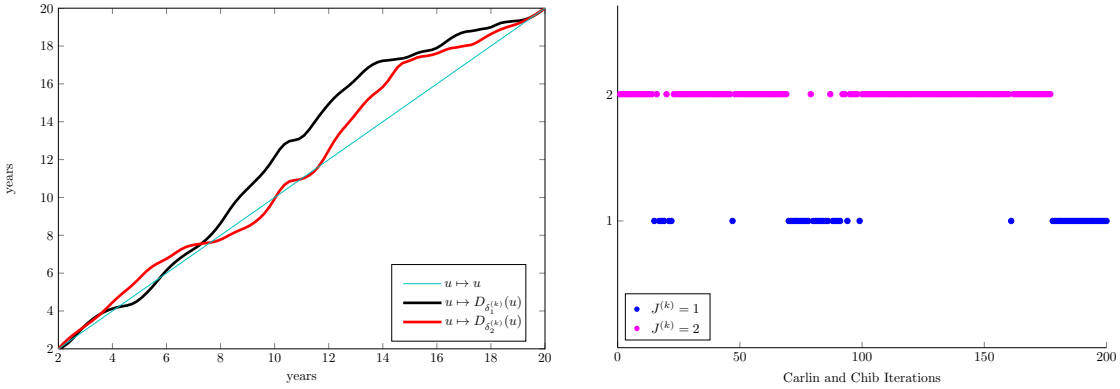


FIGURE II-5 – Time warping function and class sampling

(which is the j-th class prior covariance matrix estimate). Even though, the pseudo-priors distribution provide rough deformation parameters (see some samples from the random function $u \to D_{\delta_1^{(k)}}(u)$ on the left side of II-4), they allow to change models as illustrated in the right panel of II-5. On the other hand, the deformations sampled from the "true" posterior distribution are consistent with Y : the distorted templates tend to match the observation.

The left panel of Figure II-5 shows the warping functions $u \to D_{\delta_1}(u)$ and $u \to D_{\delta_2}(u)$ corresponding to the last samples $\delta_1^{(k)}$ and $\delta_2^{(k)}$ from the kernel $\tilde{K}^{\mathtt{CC}}$ *i.e.* $k = 200$. The sampling of the cluster index (II.24) makes use of the complete data log-likelihood and promotes models involving small deformations. Therefore, the class $J = 2$ is more likely as confirmed by the right panel of II-5 representing the class sampling scheme throughout 200 MCMC iterations.

Starting with two $m = 40$ dimensional random vectors $\hat{\alpha}_{0,1}$ and $\hat{\alpha}_{1,1}$, the two templates $\mathcal{T}_{\hat{\alpha}_{1,1000}}$ and $\mathcal{T}_{\hat{\alpha}_{2,1000}}$, displayed in II-3 were obtained after $n = 1000$ iterations of the MCoEM algorithm. Since a limited number of observations were available, each observation is processed several times, drawn at random throughout the iterations. The templates show that the girls reach the pubertal growth spurt earlier (between 11 and 12 years) than boys (between 13 and 14 years). Moreover, we notice that the boys growth velocity profile features a pre-pubertal dip more pronounced than for the girls.

6 .2 Handwritten digits template extraction

The algorithm is then applied to a collection of handwritten digits, the US postal database, featuring $N = 1000$ samples for each handwritten digit from 0 to 9, each of which consists of a 16×16 pixel image. The USPS digits data were gathered at the Center of Excellence in Document Analysis and Recognition (CEDAR) at SUNY Buffalo, as part of a project sponsored by the US Postal Service ; see [Hul94]. The main difficulty with these data stems from the geometric dispersion within each class. The sources of dispersions are numerous. First, a digit may be associated to several prototype shapes : for instance the digits two may be written with or without a loop in the lower left-hand corner and the digits seven feature an horizontal line on the vertical bar. These prototypes may suffer from small deformations (modeling the variability of the different realizations). Global deformations such as a rotations, homotheties and translations may also affect the templates.

An observation Y_n is a 16×16 matrix, regarded as a $|\Omega| = 256$ dimensional vector, whose coordinates correspond to the photometry of a fixed set of pixels, $(u_1, \ldots, u_{|\Omega|})$, such that for all s in $\{1, \ldots, |\Omega|\}$, $u_s \in (-1, 1) \times (-1, 1)$. The raw database consists of noise-free observations, such that for all $s \in \{1, \ldots, |\Omega|\}$, $Y_{n,s} \in (0, 1)$. To make the problem more challenging, an additive Gaussian noise $W_s = \sigma\epsilon$, where $\sigma = 0.2$ and $\epsilon \sim \mathcal{N}(0, 1)$, is added to $Y_{n,s}$ for all s in $\{1, \ldots, |\Omega|\}$ (see Figure II-7 (a)).

A template $\mathcal{T}$ is a function defined on $\mathbb{U} = \mathbb{R}^2$. The dictionary of functions $\{\phi_\ell\}_{\ell=1}^m$ is set as Gaussian kernels with $m = 256$. The landmark points $\{r_\ell\}_{\ell=1}^m$ are regularly spaced in the square $(-1, 1) \times (-1, 1)$ and similarly to the growth velocity curve illustration, the variances $\{\nu_\ell^2\}_{\ell=1}^m$ are set to $\nu_\ell^2 = -\frac{\min_{x \in \Omega} \|r_\ell - x\|^2}{\log \varepsilon}$, where $\varepsilon \in (0, 1)$ is the value of ϕ_ℓ in the nearest design point of r_ℓ.

Contrary to the growth velocity curve case, where the profile features a scaling factor between individuals, in this database, the scale dispersion in the measurement space is limited. As a consequence, using a scaling factor λ_n is not relevant. Each image Y_n is associated to a class $J_n \in \{1, \ldots, C\}$ and a deformation parameter β_n, such that a template can be geometrically deformed under the action of a function $u \mapsto G_{\beta_n}(u)$. We consider two complementary types of deformation :

- A rigid deformation which allows rotations, homotheties and translations. Indeed, the templates need to be allowed to rotate and to be translated in space, in order to match the observations and in particular those which are partially censored by the observation window. Homotheties allow to zoom in or to zoom out the templates. The homothety $H_\eta : u \mapsto \eta u$ is parameterized by $\eta > 0$, the translation $T_\tau : u \mapsto u + \tau$ by $\tau \in \mathbb{R}^2$ and the rotation $R_\rho : u \mapsto \mathcal{R}_\varphi(u - \vartheta) + \vartheta$ by $\rho = (\varphi, \vartheta) \in [0; 2\pi] \times \mathbb{U}$, where $\mathcal{R}_\varphi$ denotes the rotation matrix with angle φ. A Gaussian prior is set assumed, with zero mean for the components (τ, ρ) and a mean one for η. The covariance matrix is diagonal with unknown class-dependent variances.
- A smooth small deformation field is used to register locally a template with the observation. It is parameterized by a d_{D}-dimensional vector $\delta = (\delta_1, \ldots, \delta_{d_{\mathsf{D}}}) \in \mathsf{D}$ and writes for all $u \in \mathbb{U}$ as

$$D_\delta(u) = \sum_{k=1}^{d_{\mathsf{D}}} \delta_k \psi_k(u) \ ,$$

  where for all k in $\{1, \ldots, d_{\mathsf{D}}\}$, $\delta_k \in \mathbb{R}^2$, in order to allow small displacements in the two directions. The smoothness of the deformation is enforced by the choice of functions $\{\psi_k\}_{k=1}^{d_{\mathsf{D}}}$ which belong to a dictionary of Gaussian kernels defined on $\mathbb{R}^2$ and centered on

the landmark points $\{q_k\}_{k=1}^{d_{\mathsf{D}}}$ with identical variance σ_{D}^2, such that for all $k \in \{1, \ldots, d_{\mathsf{D}}\}$, $\psi_k(u) = \exp \sigma_{\mathsf{D}}^{-2} \|u - q_k\|^2$. In this implementation, we used $d_{\mathsf{D}} = 36$ landmark points at the vertices of a regular grid on the square $(-0.5, 0.5) \times (-0.5, 0.5)$ and a bandwidth $\sigma_{\mathsf{D}}^2 = 0.16$. As a consequence the local deformation parameter δ is a 72 dimensional vector. Similarly to τ, conditionally on $J_n = j$, a Gaussian distribution with zero mean and covariance matrix Γ_j is assumed for the parameter δ_n. In this implementation, for all $j \in \{1, \ldots, C\}$, Γ_j writes $\Gamma_j = \gamma_j^2 M$ where M is a fixed matrix with ones on the diagonal and 0.2 on the lower and upper diagonals.

Hence, the parameter β is a $d = 78$-dimensional vector which writes $\beta = (\eta, \tau, \rho, \delta)$ and deformation model, illustrated with the Figure II-6, writes in this setting for all $u \in \mathbb{U}$:

$$G_\beta(u) = \mathcal{R}_\varphi(\eta u + \tau - \vartheta) + \vartheta + \sum_{k=1}^{d_{\mathsf{D}}} \delta_k \psi_k(u) \, . \tag{II.36}$$

Therefore, the observation model writes :

$$\mathcal{Y}_n \in \mathcal{C}_j, \qquad \mathcal{Y}_n(u) = \mathcal{T}_{\alpha_j} \circ G_{\beta_n}(u) + \sigma_j \mathcal{W}_n(u) \, . \tag{II.37}$$

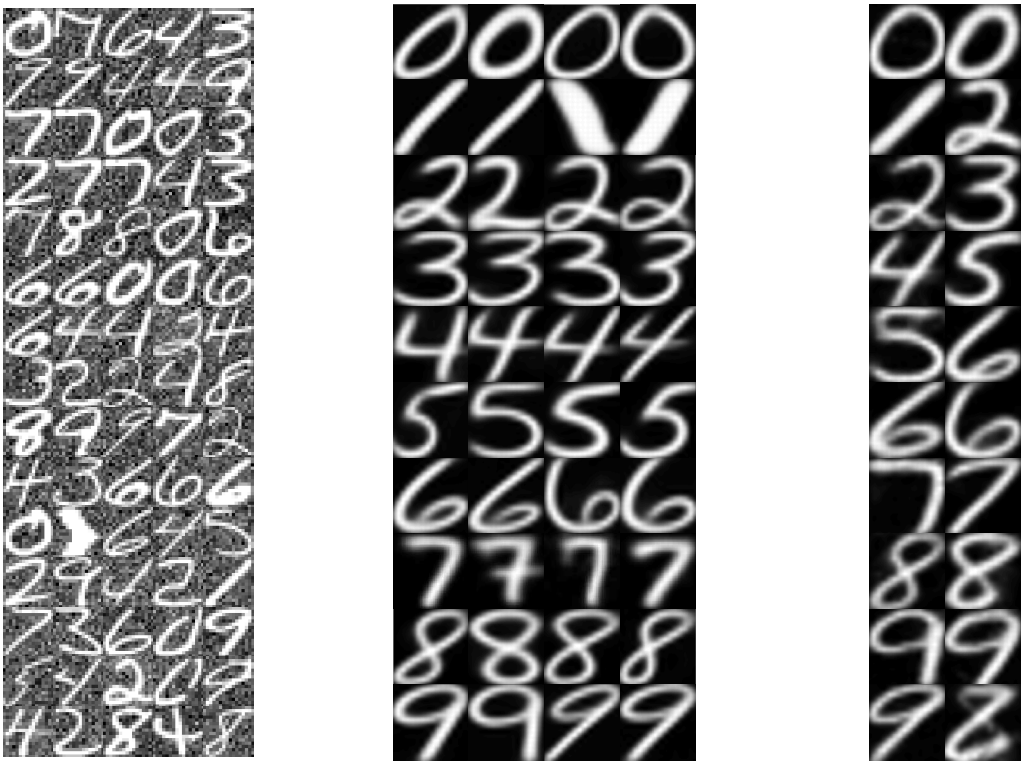
template $u \mapsto \mathcal{T}_\alpha(u)$ | rotation of the template: $u \mapsto \mathcal{T}_\alpha \circ R_\rho(u)$ | rotation and zoom out of the template: $u \mapsto \mathcal{T}_\alpha \circ R_\rho(\eta u)$ | rotation, zoom out and translation of the template: $u \mapsto \mathcal{T}_\alpha \circ \{R_\rho(\eta u) + \tau\}$ | rotation, zoom out, translation and small deformations of the template: $u \mapsto \mathcal{T}_\alpha \circ G_\beta(u)$

FIGURE II-6 – Global and local deformations of a given template

We consider two learning setups. In the first one, see Figure II-7-(b), the templates are learnt for each digit separately with $C_1 = 4$ classes and $N_1 = 1000$ iterations of the MCoEM. Then, we compare with a fully unsupervised scheme, with $C_2 = 20$ classes and $N_2 = 5000$, see Figure II-7-(c).

The templates obtained in the two settings are similar, even though in Figure II-7-(c), the algorithm makes use of a class for digits that can hardly be classified in one of the existing mixture component. Moreover, in the scheme (c) the number of classes describing a digit is ruled by the learning algorithm and may not be optimal : for instance a digit two could be described with more than two clusters, whereas three classes for a digit nine are a bit excessive.

The hidden data $\beta_n = (\eta_n, \tau_n, \rho_n, \delta_n)$, and J_n are simulated with $v_n = 200$ iterations if $n \leq 100$ and $v_n = 500$ iterations otherwise of the sampling scheme detailed in Algorithm 7. This choice of v_n is driven by the fact that when the templates are not well resolved a rough approximation of the conditional expectation is sufficient. Moreover, a burn in period of 50 iterations of the MCMC algorithm is used. Finally, given the high dimension of β_n, the quasi-Newton optimization methods to estimate $\{\beta_j^*\}_{j=1}^C$ in (II.27) is time-consuming. Therefore, the pseudo-priors parameters are set as the sample mean and covariance matrix derived from 100 iterations of a random walk targeting the posterior distribution and taking place before the first MCMC iteration.



(a) Samples of handwritten digits

(b) Partially supervised scheme

(c) Fully unsupervised scheme

Figure II-7 – Templates extraction

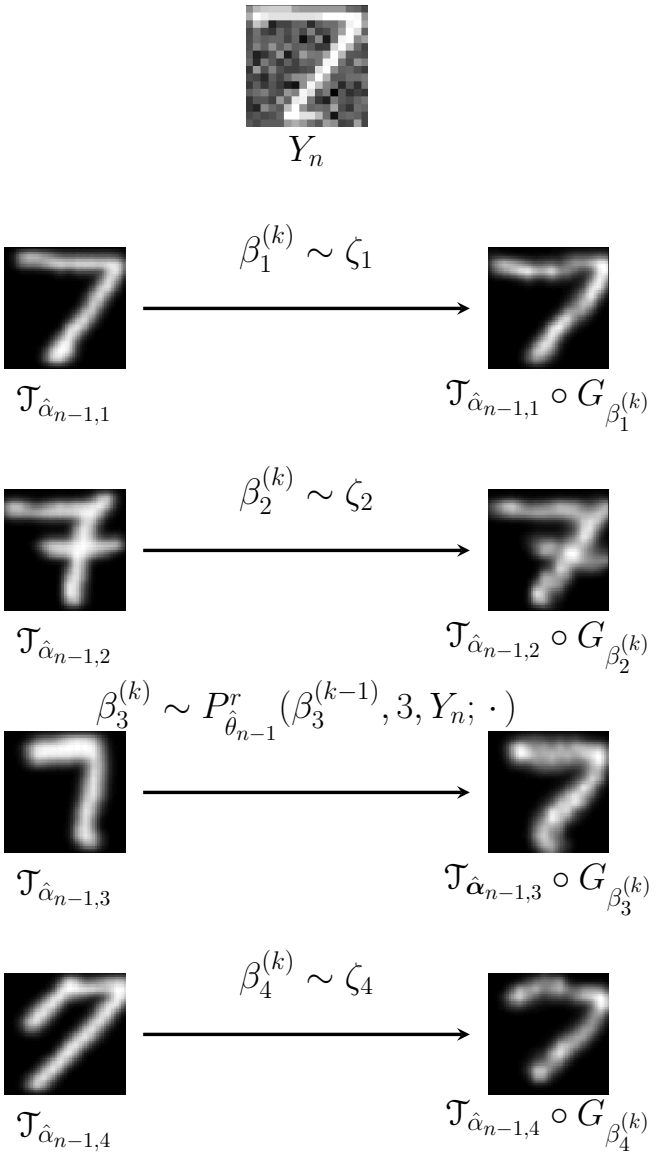
The Figure II-8-(a) shows a realization of the $k = 450$-th iteration of the Markov chain $\tilde{K}_n$ occurring in the $n = 600$-th iteration of the MCoEM algorithm. In this scenario, we aim at extracting $C = 4$ templates of the digit 7 in a supervised setting (see Figure II-7-(b)). Given $J_n^{(k-1)} = 3$, the auxiliary variables $\{\beta_j^{(k)}\}_{j\neq 3}$ are sampled from the linking densities $\{\zeta_j\}_{j=1,j\neq 3}^{C}$, while $\beta_3^{(k)}$ is simulated with $r = 20$ iterations of a Gaussian increment random walk Metropolis algorithm, whose variance is adjusted to obtain an overall acceptance rate of 40% (see [ADFDJ03]). Despite the rough approximation on the pseudo-priors parameters, Figure II-8(a) shows that the simulated deformations $\beta_j^{(k)}$ are consistent with the observation Y_n for each model $j \in \{1, \ldots, C\}$. As a consequence, the Markov chain $\{J_n^{(k)}, \beta_n^{(k)}, k > 0\}$ mixes well ; see Figure II-8(b) which displays the class index samples $\{J_n^{(k)}, k > 0\}$ throughout the $v_n = 500$ MCMC iterations.

Following the prescription of [CM07], the sufficient statistics (and consequently the parameters) should be updated for the first time once several observations have been gathered. Indeed, the parameters update (II.20) - (II.21) requires that the sufficient statistics vector check some constraints. In particular $\{\tilde{s}_{n,j,1}\}_{j=1}^{C}$ should be nonzero scalars and $\{\tilde{s}_{n,j,3}\}_{j=1}^{C}$ should be invertible matrices. In practice, these assumptions hold, when the first update happens after $n = 50$ MCoEM iterations, the second after $n = 75$ and as soon as a new observation is available from $n = 100$.

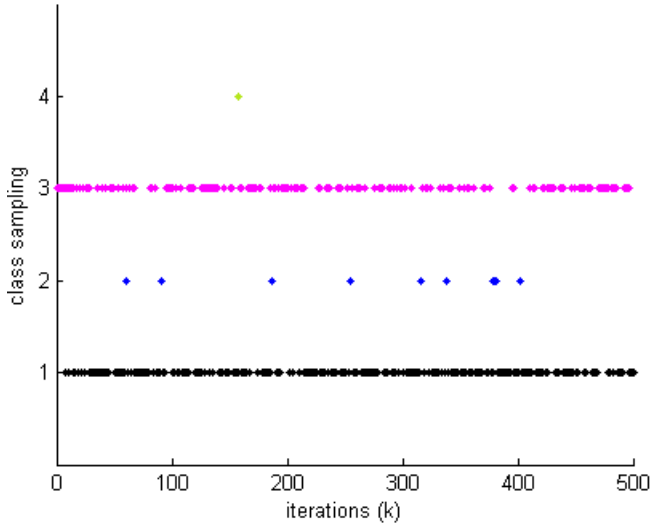
The Figure II-9 shows the parameters estimate $\{\hat{\theta}_n\}_{n=1}^{1000}$ throughout the Monte Carlo online EM algorithm for the digit two learnt separately with $C = 4$ classes. The templates are initialized using the four first observations, such that for all $j \in \{1, \ldots, 4\}$, $\hat{\alpha}_{0,j} = (\Phi_{O_d}^T \Phi_{O_d})^{-1} \Phi_{O_d}^T Y_j$, where Φ_β is defined in (II.3). The functions $\{\mathcal{T}_{\boldsymbol{\alpha}_{n,j}}\}_{1 \leq j \leq C}$ tends progressively to usual reference shapes and each new observation available enhances the templates estimate (Figure II-9-(a)).

7 Conclusion

We have proposed a statistical framework to perform sequential and unsupervised inference in a deformable template model, with application to curve synchronization and shape extraction and registration. It makes use of a Monte Carlo online EM (MCoEM), derived from [CM07] and a novel MCMC sampling method, allowing to simulate the untractable joint distribution of the cluster index and deformation parameters. The method has been used successfully to extract reference templates from several data sets featuring high time / geometric dispersion.

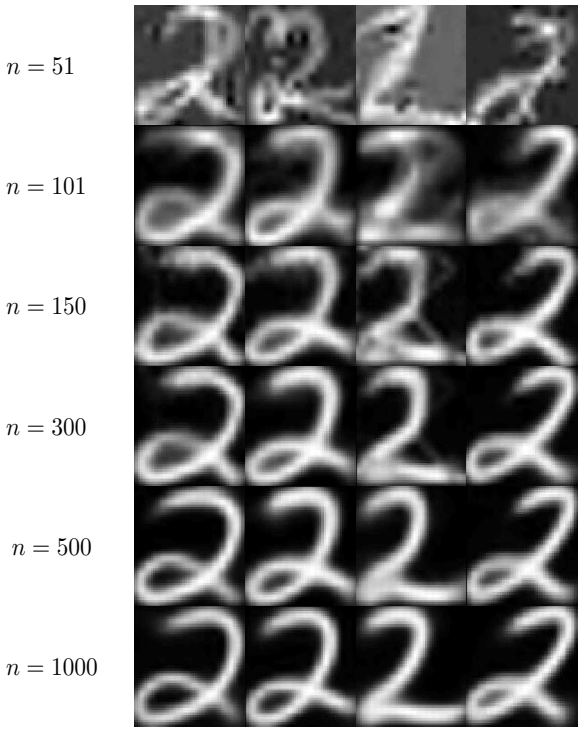


(a) Simulation of the auxiliary variables $\{\beta_j^{(k)}\}_{j=1}^C$

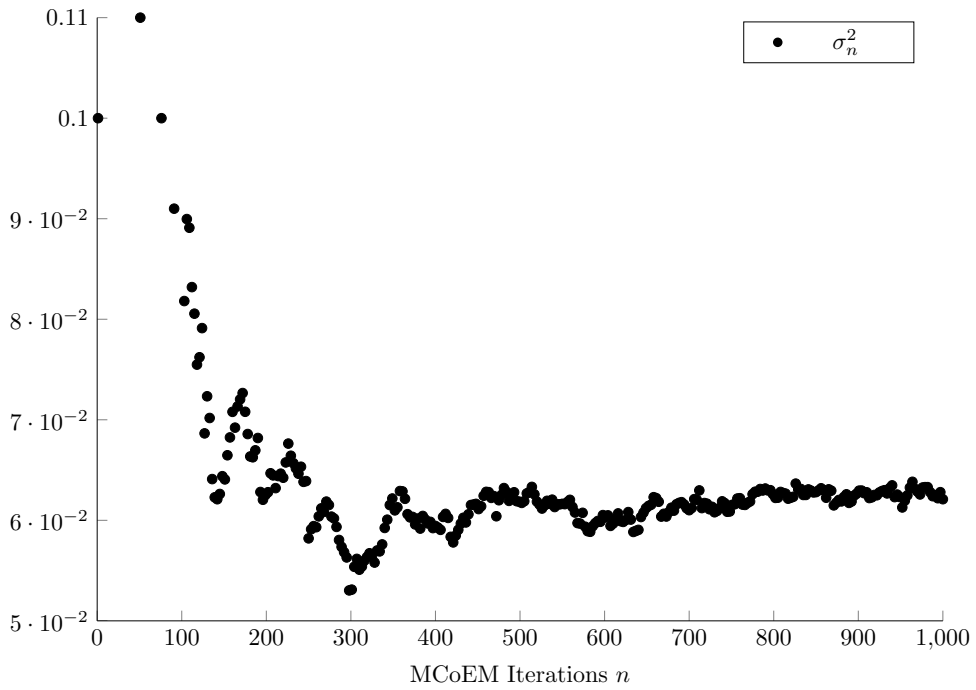


(b) Class index samples $\{J_n^{(k)}, k > 0\}$ using K_n

Figure II-8 – Sampling posterior distribution



(a) Evolution of the templates $\{\mathcal{T}_{\hat{\alpha}_j}\}_{1\leq j\leq C}$



(b) Evolution of the noise variance estimate $\hat{\sigma}_n^2$

FIGURE II-9 – Templates extraction and inference

Chapitre III

Modélisation et classification des SIR d'aéronefs

1 Introduction

Dans ce chapitre, nous appliquons la méthodologie d'estimation des paramètres de modèles à prototype déformable présentée dans le Chapitre II, au cas des Signatures Infrarouge d'aéronefs, obtenues par la méthode détaillée dans la Partie Simulation de SIR. L'objectif est de déterminer les caractéristiques géométriques, photométriques et spectrales de deux aéronefs différents puis de proposer une méthodologie de classification *a posteriori* .

D'un point de vue pratique, ces deux étapes s'articulent de façon séquentielle :

- Un premier ensemble d'images, simulées par la méthode de la Partie Simulation, est utilisé comme base d'apprentissage pour estimer les paramètres du modèle d'observation. En pratique, dans le cas d'un système de surveillance sur le terrain, la base d'apprentissage serait constituée d'images acquises par un capteur infrarouge et pour lesquelles une cible a été détectée par la méthode proposée dans le Chapitre I,
- Une fois que les estimateurs des paramètres du modèle ont convergé, de nouvelles données simulées par la méthode de la Partie Simulation (en pratique, acquises par un capteur infrarouge et pour lesquelles une cible a été détectée par la méthode proposée dans le Chapitre I), peuvent être classées à partir du modèle d'observation ainsi établi.

Nous présentons tout d'abord un modèle d'observation pour les SIR monospectrales qui diffère du modèle utilisé pour les chiffres manuscrits proposé dans le Chapitre II par la prise en compte d'un facteur scalaire d'ajustement de l'intensité photométrique. Comme pour les chiffres manuscrits, deux scénarios d'apprentissage sont envisagés. Le premier cas consiste en un apprentissage *semi-supervisé* dans lequel les deux avions sont appris de façon séparée et où plusieurs classes sont apprises pour un même avion. Cette approche ne correspond pas véritablement au cadre de notre étude car elle suppose l'existence d'une base d'apprentissage pour chaque avion. Le second scénario, plus réaliste par rapport à notre contexte, est *non-supervisé* car la base d'apprentissage contient des avions des deux classes mélangées : l'objectif est d'apprendre une classe par avion.

Le passage aux SIR multispectrales soulève la question de la corrélation spectrale et spatiale des images dans les différentes bandes. Deux modèles de corrélation sont envisagés, chacun menant à un modèle de SIR multispectrales. Comme dans le cas des SIR monospectrales, l'apprentissage des paramètres du modèle est effectué au moyen des deux scénario *semi-supervisé* et *non-supervisé* et les résultats sont présentés. Par rapport au cas

monospectral, l'utilisation des SIR multispectrales nécessite la spécification du regroupement de bandes spectrales pris en compte dans l'apprentissage : différentes combinaisons de bandes spectrales sont étudiées.

Enfin, nous proposons une méthode de classification basée sur le calcul de la classe qui maximise la vraisemblance *a posteriori* . L'algorithme de classification diffère dans le cas *semi-supervisé* et dans le cas *non-supervisé*. Les résultats de classification montrent que, comme pour la détection, la classification d'aéronefs donne de meilleurs résultats lorsque des SIR multispectrales plutôt que monospectrales sont utilisées.

Dans tout ce chapitre, nous considérons un modèle de fond Gaussien (Cf. Section Simulation de fonds) avec les deux écarts types correspondant au cas de ciel clair et au cas de ciel nuageux. Le modèle de fond de ciel texturé multispectral étant encore en cours d'élaboration, il n'a pas été possible de tester les performances de notre méthode d'apprentissage dans le cas des fonds nuageux texturés.

2 Cas monospectral

2 .1 Modèle d'observation

Bien que les SIR monospectrales soient des données de même dimension que les images de chiffres manuscrits (Cf. Illustration de la Section 6 .2), il n'est pas possible d'utiliser directement le même modèle d'observation que (II.37). En effet :

- Bien qu'assimilés à des fonctions $\mathcal{Y}_n : \mathbb{R}^2 \to \mathbb{R}$, les chiffres manuscrits ne présentent en réalité que deux degrés de liberté car il est possible de passer d'un chiffre à un autre simplement par une déformation du plan. Comme le montrent les Figures III-1 (a) et (b) les valeurs des pixels appartenant à un chiffre sont presque toutes égales à 1, si bien que l'on peut considérer que la fonction $\mathcal{Y}_n$ prends ses valeurs dans $\{0, 1\}$. Les Figures III-1 (c) et (d) montrent que les SIR monospectrales ont un degré de liberté supplémentaire car les pixels appartenant à la cible ont des valeurs variables, et qu'il n'est donc pas possible de restreindre l'espace image de $\mathcal{Y}_n$ à $\{0, 1\}$ dans ce cas.
- L'échelle de niveau de gris est la même pour toutes les images de chiffres manuscrits. Par conséquent, l'information photométrique, que l'on peut définir comme étant la valeur numérique associée à chaque pixel, est confondue avec l'information géométrique, que l'on peut définir comme étant la répartition spatiale des pixels de chiffre. La situation est différente dans le cas des SIR monospectrales, comme le montre la Figure III-2. La première ligne de la Figure III-2 représente cinq observations d'un même aéronef dans le même scénario par un niveau de gris indépendant pour chaque image : pour chaque image le pixel le plus *chaud* a une valeur de 1 et le pixel le plus *froid* 0. La seconde ligne représente les mêmes observations mais par un niveau de gris commun aux cinq images : le pixel le plus *chaud* des cinq observations a une valeur de 1 et le pixel le plus *froid* des cinq observations a une valeur de 0. Cette représentation met en évidence l'importance de la photométrie dans le cas des SIR : les observations 1 et 3 (en partant de la gauche) ont une géométrie similaire comme le montre la première ligne, mais ont des photométries différentes, comme le montre la seconde. Il est donc important de prendre en compte un facteur d'échelle permettant d'ajuster les niveaux de photométries entre plusieurs observations. En effet, la méthodologie d'extraction de formes caractéristiques présentée dans le Chapitre II fait intervenir une étape de recalage, basée sur l'intensité des observations, qui n'est donc pas réalisable sans la prise en compte des variations photométriques.

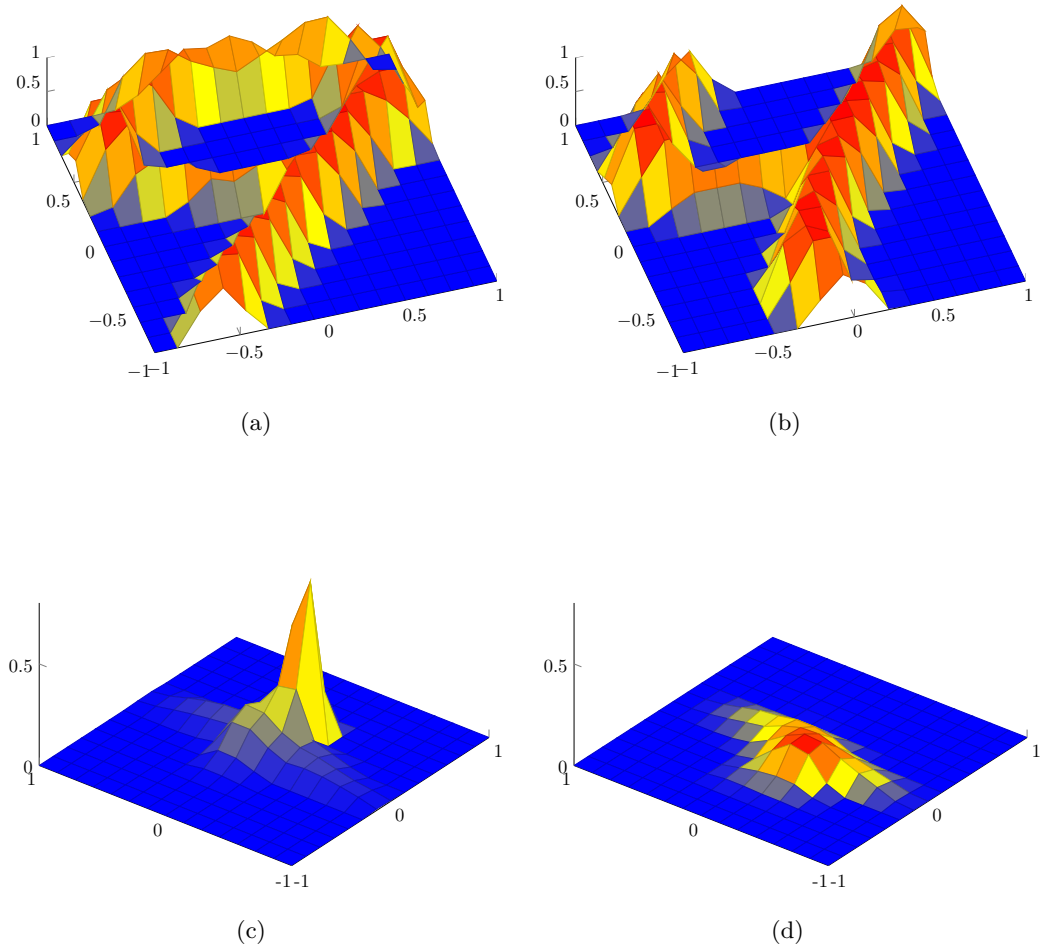


Figure III-1 – Représentation en trois dimensions de deux chiffres manuscrits et de deux SIR monospectrales

- Les cibles étant situées à une même distance du capteur, elles ont toutes approximativement la même taille. Par conséquent, il n'est pas nécessaire de prendre en compte les homothéties du plan dans le modèle de déformation.

D'après ces remarques, il existe pour toute observation $\mathcal{Y}_n : \mathbb{R}^2 \to \mathbb{R}$ des données non observées : une classe $j_n \in \{1, \ldots, C\}$, un facteur d'échelle $\lambda_n > 0$ et une déformation du plan G_{β_n}, $\beta_n \in \mathsf{B}$. Conditionnellement au vecteur des données manquantes $(j_n, \beta_n, \lambda_n)$, la fonction $\mathcal{Y}_n$ s'écrit pour tout $u \in \mathbb{R}^2$

$$\mathcal{Y}_n(u) = \lambda_n \, \mathcal{T}_{j_n} \circ G_{\beta_n}(u) + \sigma \mathcal{W}_n(u) \; , \tag{III.1}$$

où, pour tout $j \in \{1, \ldots, C\}$, $\mathcal{T}_j : \mathbb{U} \to \mathbb{R}$ est la fonction prototype de la classe j et $\mathcal{W}_n : \mathbb{U} \to \mathbb{R}$ est un processus de bruit additif, considéré dans tout ce Chapitre comme étant un processus Gaussien indépendant centré et réduit.

Nous considérons le modèle de déformation $\{G_\beta, \, \beta \in \mathsf{B}\}$ défini pour tout $u \in \mathbb{R}^2$ par

$$G_\beta(u) = \mathcal{R}_\varphi(u - \vartheta) + \tau + \sum_{k=1}^{d_{\mathsf{D}}} \delta_k \psi_k(u) \; , \tag{III.2}$$

où τ est le paramètre de translation, (φ, ϑ) les paramètres de rotation avec $\mathcal{R}_\varphi$ la matrice de rotation du plan d'angle φ et $\delta = (\delta_1, \ldots, \delta_{d_{\mathsf{D}}}) \in \mathsf{D}$ le paramètre de déformation locale.

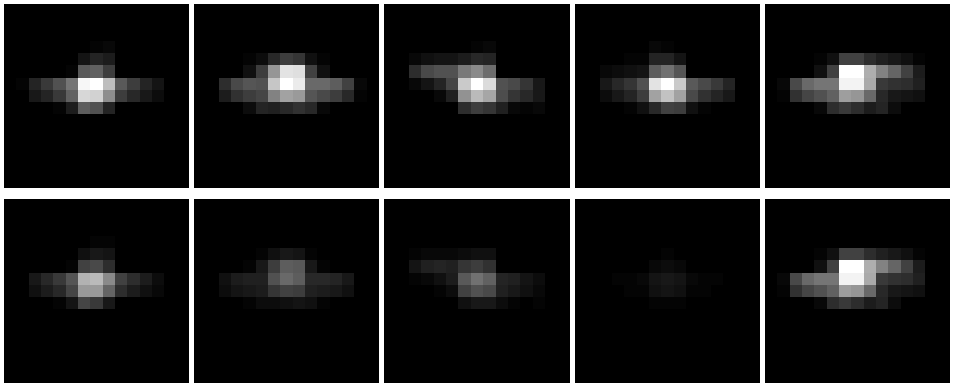


FIGURE III-2 – Cinq SIR monospectrales du même aéronef dans le même scénario - sur la 1ère ligne, les observations sont représentées par des niveaux de gris indépendants ; sur la 2ème ligne, les observations sont représentées par une échelle commune de niveau de gris

Dans cette application, pour tout $k \in \{1, \ldots, d_{\mathsf{D}}\}$, nous avons $\delta_k \in \mathbb{R}^2$ et par conséquent $\mathsf{D} = \mathbb{R}^{2d_{\mathsf{D}}}$. Le paramètre de déformation s'écrit donc $\beta = (\varphi, \vartheta, \tau, \delta)$ et appartient à $\mathsf{B} = [0, 2\pi] \times \mathbb{R}^{2(d_{\mathsf{D}}+2)}$.

Dans notre modèle d'observation, les données cachées sont des variables aléatoires (J, β, λ) admettant une loi *a priori* g_θ, par rapport à la mesure de domination de Lebesgue, définie par :

$$g_\theta(j, \beta, \lambda) \propto \omega_j \exp\left(-1/2(\beta^T {\Gamma_{\gamma_j}}^{-1} \beta)\right) \lambda^{a-1} \exp\left(-b\lambda\right) . \tag{III.3}$$

La matrice de covariance de β est supposée diagonale par blocs : elle est paramétrée par un scalaire $\{\gamma_j > 0\}_{j=1}^{C}$ tel que, conditionnellement à la classe j, $\gamma_j^2 M$ est la matrice de covariance des paramètres de déformation locale δ avec $M \in \mathcal{M}^+(\mathbb{R})$. Les variances des paramètres φ, ϑ et τ sont supposées connues et fixées à 0.3. La loi *a priori* de λ est une loi Gamma de paramètre $(a, b) = (10, 10)$, ce qui permet d'assurer la positivité du facteur d'échelle.

Nous supposons que le prototype $\mathcal{T}_j$ de chaque classe $j \in \{1, \ldots, C\}$ est une fonction $\mathcal{T}_{\alpha_j} : \mathbb{R}^2 \to \mathbb{R}$, définie comme dans (II.2), et paramétrée par un vecteur $\alpha_j \in \mathbb{R}^m$. Enfin, à la différence du modèle d'observation des chiffres manuscrits, nous supposons que l'écart type du bruit additif $\sigma > 0$ ne dépend pas de la classe j. En effet, dans cette application, le bruit additif modélise le fond de ciel et les perturbations dues à l'instrument de mesure et ne dépend donc pas du type d'aéronef. Dans tout ce chapitre, nous considérons le cas d'un bruit blanc Gaussien indépendant et identiquement distribué.

Conditionnellement à J_n, β_n, λ_n, une SIR monospectrale Y_n est un vecteur correspondant à une discrétisation sur une grille de pixel $\Omega = (u_1, \ldots, u_{|\Omega|}) \subseteq [-1, 1] \times [-1, 1]^{|\Omega|}$ de la fonction $\mathcal{Y}_n$. Y_n peut donc s'écrire sous forme vectorielle comme

$$Y_n = \lambda_n \Phi_{\beta_n} \alpha_{J_n} + \sigma W_n , \tag{III.4}$$

où $W_n = (\mathcal{W}_n(u_1), \ldots, \mathcal{W}_n(u_{|\Omega|}))^T$ et Φ_{β_n} est la matrice définie comme dans (II.3). Ainsi, dans cette approche, Y_n est une variable aléatoire admettant pour densité, par rapport à la mesure de Lebesgue et conditionnellement à β_n, λ_n et J_n, la fonction p_θ définie par :

$$p_\theta(y \,|\, \beta, \lambda, j) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(y - \lambda \Phi_\beta \alpha_j)^2\right) . \tag{III.5}$$

L'apprentissage du modèle d'observation consiste donc en l'estimation des paramètres

$$\Theta = \bigcup_{j=1}^{C} \left\{ (\alpha_j, \gamma_j, \omega_j, \sigma) \,|\, \alpha_j \in \mathbb{R}^m,\, \gamma_j > 0,\, \omega_j \in (0,1),\, \sigma > 0 \right\} \cap \left\{ \sum_{j=1}^{C} \omega_j = 1 \right\} ,$$

à partir de SIR $\{Y_n,\, n > 0\}$. L'algorithme MCoEM introduit au Chapitre II permet d'obtenir une séquence d'estimateurs $\{\hat{\theta}_n \in \Theta,\, n \in \mathbb{N}\}$ qui converge vers un estimateur $\theta^\star$ décrivant ces observations.

2 .2 Estimation des paramètres

Pour évaluer les performances de l'algorithme d'apprentissage, nous disposons :

- d'indicateurs qualitatifs tels que les prototypes $\{\mathcal{T}_{\hat{\alpha}_n},\, n \in \mathbb{N}\}$ ou l'allure d'observations $Y \sim \mathbb{P}_{\hat{\theta}_n}$ simulées par le modèle génératif avec les paramètres estimés,
- d'indicateurs quantitatifs tels que les estimateurs $\{\hat{\theta}_n,\, n \in \mathbb{N}\}$ des paramètres θ du modèle.

Nous exposons à présent les résultats d'estimation des paramètres pour deux avions distincts, l'*Avion 1* et l'*Alphajet* et sous les deux niveaux de bruits σ_1 et σ_2. Ces simulations ont été effectuées avec les paramètres d'apprentissage suivant :

- Nombre total d'observations $n = 1000$,
- Dimension des observations : $|\Omega| = 225$,
- Coefficient d'approximation stochastique $\rho = 0.6$,
- Initialisation des paramètres : $\hat{\omega}_{j,1} = 1/C$, $\hat{\gamma}_{j,1} = 0.1$, $\hat{\sigma}_1 = 0.1$ et

$$\hat{\alpha}_{j,1} = (\Phi_{0_d}^T \Phi_{0_d})^{-1} \Phi_{0_d}^T Y_j ,$$

- Les paramètres ne sont mis à jour qu'à partir de la 50-ième itération,
- Fonction prototype : $m = 15$, $\nu_p = 0.065$ et pour tout $\ell \in \{1, \ldots, m\}$

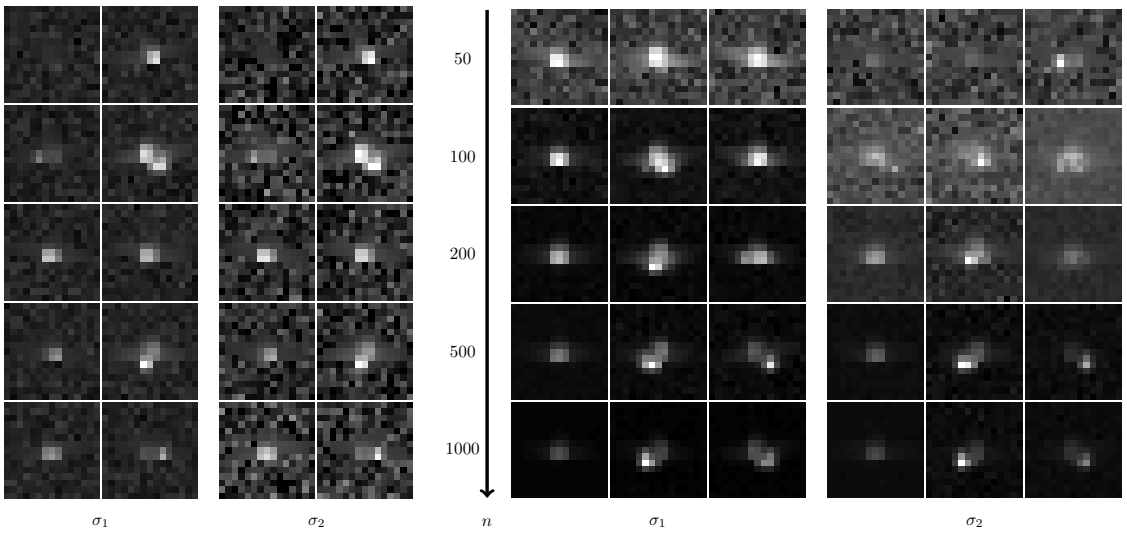
$$\phi_\ell(u) = (\sqrt{2\pi}\nu_p)^{-1} \exp -\nu_p^{-2}(u - r_\ell)^2 ,$$

  où $\{r_\ell\}_{\ell=1}^m$ est un ensemble de points régulièrement réparti dans $[-1; 1]^2$,
- Déformation locale : $d_{\mathsf{D}} = 6$, $\nu_{\mathsf{D}} = 0.1$ et pour tout $k \in \{1, \ldots, d_{\mathsf{D}}\}$

$$\psi_k(u) = (\sqrt{2\pi}\nu_p)^{-1} \exp -\nu_g^{-2}(u - q_k)^2 ,$$

  où $\{q_k\}_{k=1}^{d_{\mathsf{D}}}$ est un ensemble de points régulièrement réparti dans $[-0.8; 0.8]^2$,
- Matrice de covariance de la déformation locale $\Gamma_{\gamma_j} = \gamma_j^2 \mathrm{Id}_{2d_{\mathsf{D}}^2}$,
- Simulation des données manquantes : $v_n = 100$ itérations de l'algorithme de Carlin et Chib pour $n \leq 50$ et $v_n = 200$ pour $n > 50$,
- Post-traitement : Moyenne de Polyak-Ruppert à partir de $n_0 = 750$,

$$\tilde{\theta}_n = \begin{cases} \hat{\theta}_n & n \leq n_0 \\ (n - n_0)^{-1} \sum_{k=n_0}^{n} \hat{\theta}_k & n > n_0 \end{cases}$$



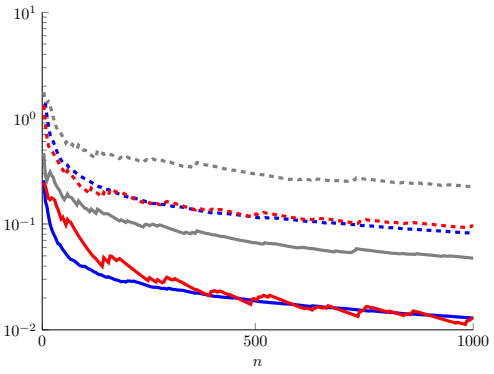
(a) Y_n (b) $\mathcal{T}_{\hat{\alpha}_{j,n}}$, $j \in \{1, 3\}$

FIGURE III-3 – Exemple d'observations (a) et estimation des fonctions prototypes (b) de l'Avion 1 pour deux niveaux de bruit

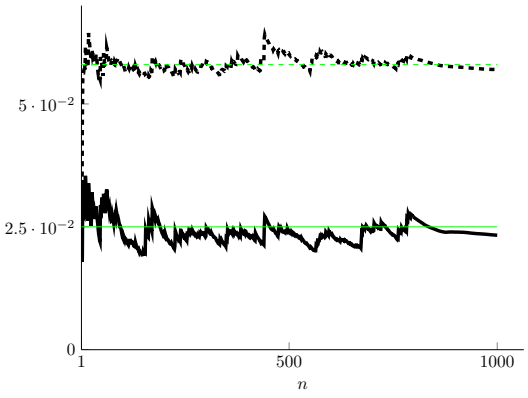
Apprentissage *semi-supervisé*

Avion 1 La Figure III-3 illustre l'estimation des fonctions prototypes pour un mélange de $C = 3$ modèles déformables sous les niveaux de bruits σ_1 et σ_2 envisagés, à partir d'une base d'observations ne contenant que des aéronefs de type Avion 1. Qualitativement, le niveau de bruit ne perturbe pas l'estimation des templates qui sont, au bout de 1000 itérations, fortement similaires. Notons que l'ordre dans lequel sont traitées les observations est aléatoire et ne modifie pas l'estimation des paramètres à convergence. Les observations peuvent donc être décrites comme des variations autour de 3 modes : un mode majoritaire (classe $j = 1$) où le fuselage de l'avion est centré et symétrique et deux classes minoritaires équiprobables pour lesquelles sur le côté du fuselage de l'avion se trouve un point chaud (à gauche pour la classe $j = 2$ et à droite pour la classe $j = 3$).

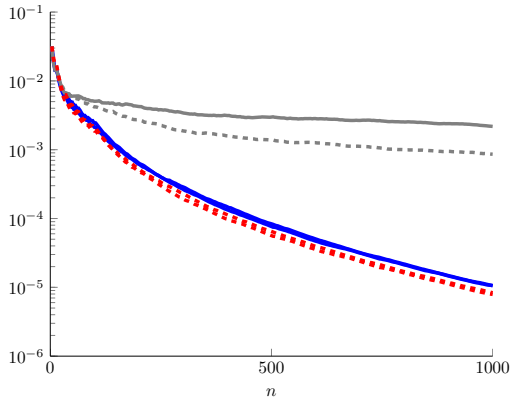
La Figure III-4 présente l'apprentissage des autres paramètres du modèle : les traits pleins et les courbes en pointillés représentent respectivement l'estimation des paramètres pour le niveau de bruit σ_1 et pour le niveau de bruit σ_2. Les courbes grises, rouges et bleues représentent respectivement les classes $j = 1, 2$ et 3 et les courbes vertes dans la Figure III.4(c) donnent les vrais valeurs des écarts types de bruit σ_1 (trait plein) et σ_2 (pointillés). On constate que pour les deux niveaux de bruits, l'estimation des paramètres des lois *a priori* des données manquantes du modèle varie peu. La seule exception provient de la variance des déformations qui est plus faible pour la classe $j = 1$ que pour les classes $j = 2$ et 3 ; Cf. Figure III.4(c). En effet, les prototypes des classes 2 et 3 étant caractérisés par un point très chaud, la simulation des déformations pour ces observations s'apparente généralement à un simple recalage local de cette zone de l'image pour les observations concernées. La classe 1 décrit des observations plus diversifiées et des déformations plus importantes sont nécessaires pour recaler le prototype et ces observations. Ceci explique la plus grande variabilité des déformations pour cette classe.



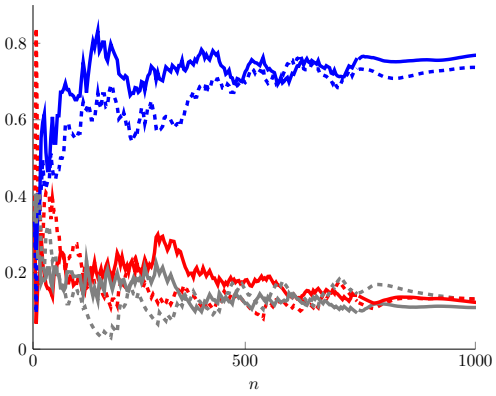
(a) Norme en variation totale de $\{\mathcal{T}_{\hat{\alpha}_{n,j}}\}$



(b) Evolution du paramètre de bruit $\{\hat{\sigma}_n\}$



(c) Evolution du paramètre de covariance $\{\hat{\gamma}_{j,n}\}_{j=1}^{3}$



(d) Evolution du paramètre de mélange $\{\hat{\omega}_{j,n}\}_{j=1}^{3}$

Figure III-4 – Estimation des paramètres du modèle pour l'Avion 1 : les courbes en traits pleins correspondent au cas σ_1 et les courbes en pointillées au cas σ_2. Les classes $j = 1, 2$ et 3 sont respectivement représentées en gris, bleu et rouge.

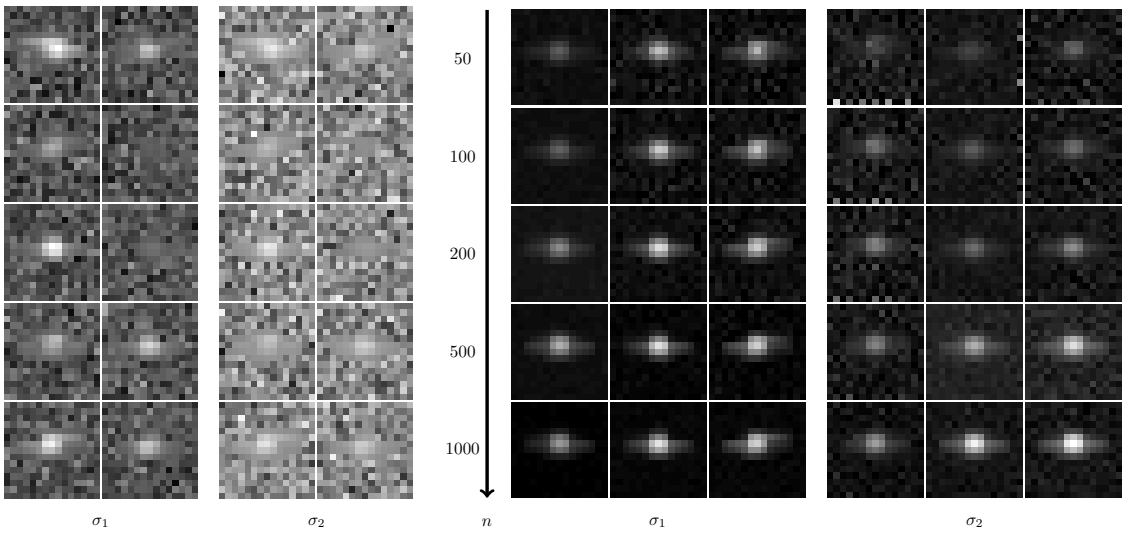
Alphajet Les Figures III-5 et III-6 illustrent l'apprentissage des paramètres du modèle à prototype déformable lorsque la base de données ne contient que des observations d'Alphajet. L'Alphajet présente deux différences majeures par rapport à l'Avion 1 :

(i) L'Alphajet possède une signature infrarouge en moyenne plus faible que l'Avion 1 comme le montre la comparaison entre les Figures III.5(a) et III.3(a).

(ii) La valeur des luminances est plus varié pour l'Alphajet que pour l'Avion 1.

Les prototypes font apparaître 3 classes : la classe $j = 2$ symétrique et majoritaire et deux autres, moins importantes en nombre, qui semblent être des rotations de la classe 2 d'un angle négatif pour la classe $j = 1$ et d'un angle positif pour la classe $j = 3$.

D'après la remarque (i), la cible a un contraste plus faible avec le fond lorsqu'il s'agit d'un Alphajet que d'un Avion 1 et l'étape de simulation des données manquantes de déformation est par conséquent plus délicate pour l'Alphajet car il y a moins de caractéristiques remarquables facilitant le recalage entre prototypes et observations. Ceci se traduit au niveau des prototypes, en particulier dans le cas du bruit σ_2, par des prototypes moins variés : les classes $j = 2$ et $j = 3$ sont en effet plus proches avec σ_2 qu'avec σ_1. Pour cette même raison, l'estimateur du paramètre de covariance de la classe $j = 2$ est plus faible sous le niveau de bruit σ_2 que sous le niveau σ_1. A cette exception près et comme pour



(a) Y_n (b) $\mathcal{T}_{\hat{\alpha}_{j,n}}$, $j \in \{1, 3\}$

FIGURE III-5 – Exemple d'observations (a) et estimation des fonctions prototypes (b) de l'Alphajet pour deux niveaux de bruit

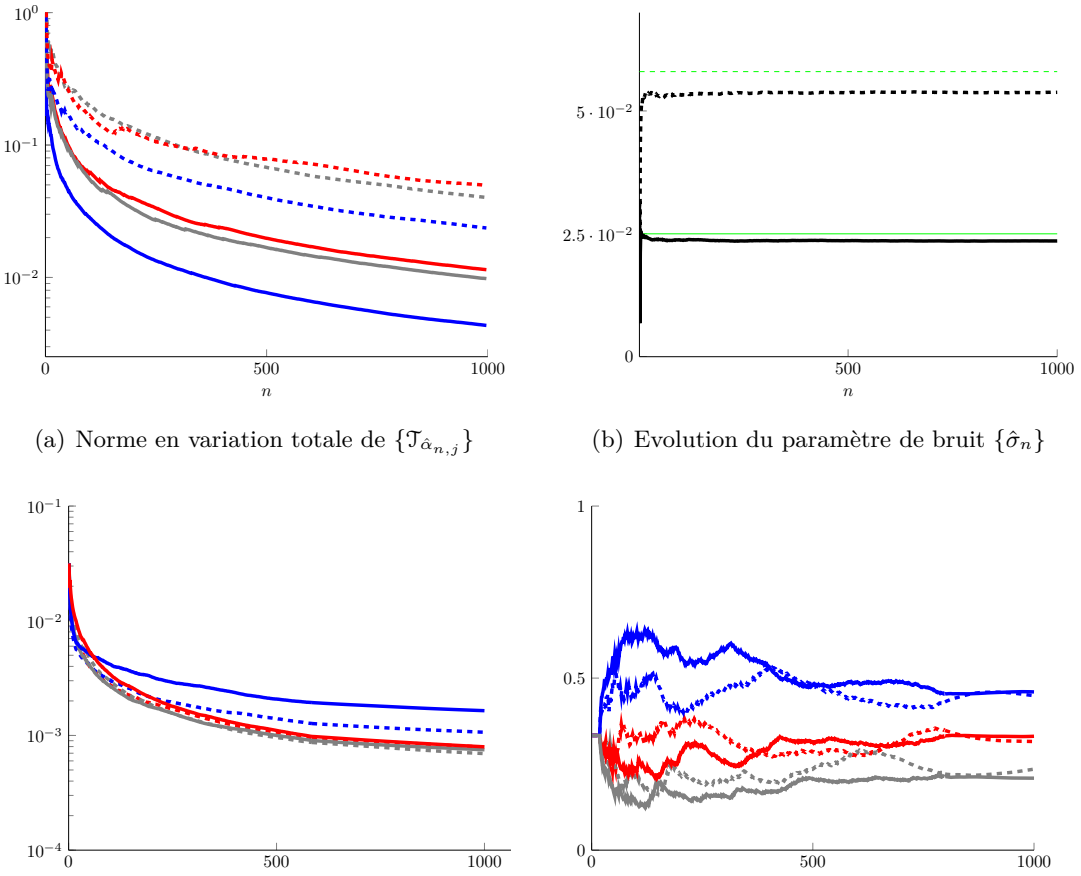
l'Avion 1, l'estimation des paramètres des lois *a priori* des données manquantes du modèle est semblable pour les deux niveaux de bruits.

La remarque (ii) explique quant à elle que le niveau d'intensité estimé pour la classe $j = 1$ de l'Alphajet, soit plus faible que celui estimé pour les classes $j = 2$ et $j = 3$, et ce pour les deux niveaux de bruits.

Remarque sur l'estimation des variances de la photométrie Sur les deux Figures III.4(b) et III.6(b) illustrant l'estimation des variances de la photométrie, un léger biais est observé : la variance estimée est inférieure à la variance réelle pour les deux avions et pour les deux niveaux de bruit. Ceci s'explique par la façon dont sont bruitées les images. Dans ce contexte, le bruit modélise le fond de ciel situé à une distance infinie du capteur et qui est donc partiellement masqué lorsqu'un avion est présent. Par conséquent, comme le montrent les Figures III.3(a) et III.5(a), la zone de l'image bruitée correspond aux pixels où la SIR a une valeur négligeable ; voir la Remarque 3 dans la Partie Simulation pour plus de précisions. Le modèle (III.4), qui fait l'hypothèse d'un bruit identique dans toute l'image, ne correspond donc pas parfaitement aux observations traitées d'où le léger biais. Enfin, comme l'Alphajet a moins de pixels ayant une valeur négligeable que l'Avion 1, il y a plus de pixels non bruités dans les images d'Alphajet que dans les images d'Avion 1 : il est donc logique d'observer un biais plus important dans le cas de l'Alphajet que dans le cas de l'Avion 1.

Apprentissage *non-supervisé*

Dans le mode d'apprentissage non-supervisé, nous ne considérons que le cas d'un mélange de $C = 2$ classes. En effet, lorsque $C > 2$, les classes sont librement organisées par l'algorithme et ni le nombre ni les indices des classes décrivant chaque aéronef ne sont spécifiés *a priori* . Par conséquent, un observateur extérieur doit nécessairement attribuer à chaque classe un label correspondant au type d'avion qu'elle décrit, pour pouvoir classer



(a) Norme en variation totale de $\{\mathcal{T}_{\hat{\alpha}_{n,j}}\}$

(b) Evolution du paramètre de bruit $\{\hat{\sigma}_n\}$

(c) Evolution du paramètre de covariance $\{\hat{\gamma}_{j,n}\}_{j=1}^3$

(d) Evolution du paramètre de mélange $\{\hat{\omega}_{j,n}\}_{j=1}^3$

Figure III-6 – Estimation des paramètres des lois *a priori* pour l'Alphajet : les courbes en traits pleins correspondent au cas $\sigma = 0.025$ et les courbes en pointillées au cas $\sigma = 0.058$. Les classes $j = 1, 2$ et 3 sont respectivement représentées en gris, bleu et rouge.

des observations inconnues *a posteriori* . Pour cette raison, l'algorithme ne peut plus être considéré comme totalement non supervisé.

La Figure III-7 présente les $C = 2$ prototypes estimés dans le cas où les deux types d'avions sont appris simultanément. L'Avion 1 est représenté en marron et l'Alphajet en orange et comme précédemment les traits pleins et les pointillés correspondent respectivement aux cas σ_1 et σ_2. Tandis que dans l'approche semi-supervisée, la géométrie joue un rôle discriminant entre les différentes classes du modèle, la photométrie prend plus d'importance lorsque le nombre de classes diminue. La géométrie du prototype des deux classes est semblable et correspond approximativement au prototype majoritaire obtenu pour chaque type d'avion dans le cas *semi-supervisé*. En revanche, l'information photométrique est bien distincte. Comme le montre la Figure III-8, l'information géométrique n'est pas apprise dans les prototypes mais intégrée dans la déformation géométrique. La variance des déformations est en effet plus de 20 fois supérieure dans le cas *non-supervisé* que dans le cas *semi-supervisé*. Pour l'Avion 1 (classe $j = 1$), les déformations sont particulièrement variables car le recalage entre les observations décrites par les classes $j = 2$ et $j = 3$ dans le cas *semi-supervisé* nécessite d'importantes déformations. L'estimation sous σ_2 est davantage délicate comme le montre la Figure III-8 : un certain nombre d'Avion

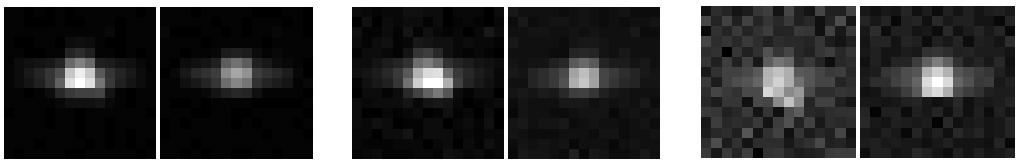
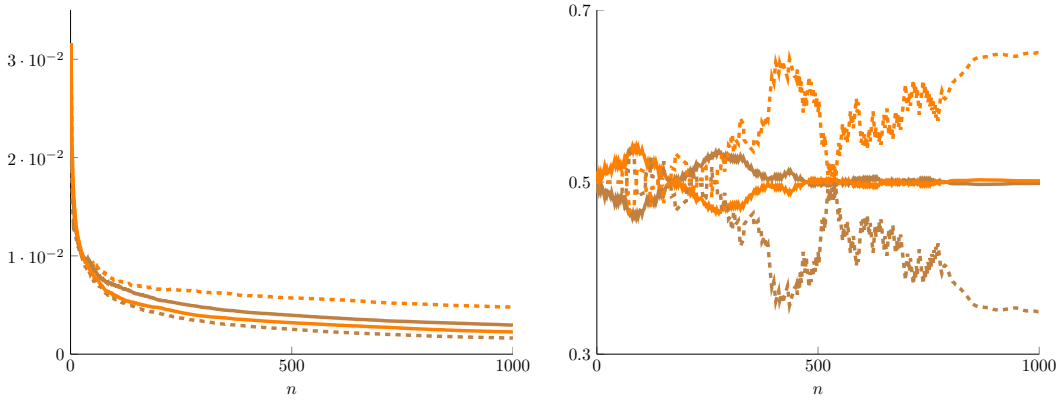


FIGURE III-7 – Estimation des prototypes dans le cas non-supervisé



(a) $\hat{\gamma}_{j,n}$ (b) $\hat{\omega}_{j,n}$

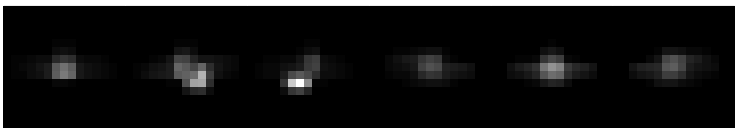
FIGURE III-8 – Estimation des variances de la géométrie (a) et des poids (b) dans le cas non-supervisé avec $C = 2$

1 sont appris en tant qu'Alphajet et pour cette raison la variance des déformations de l'Alphajet est plus importante.

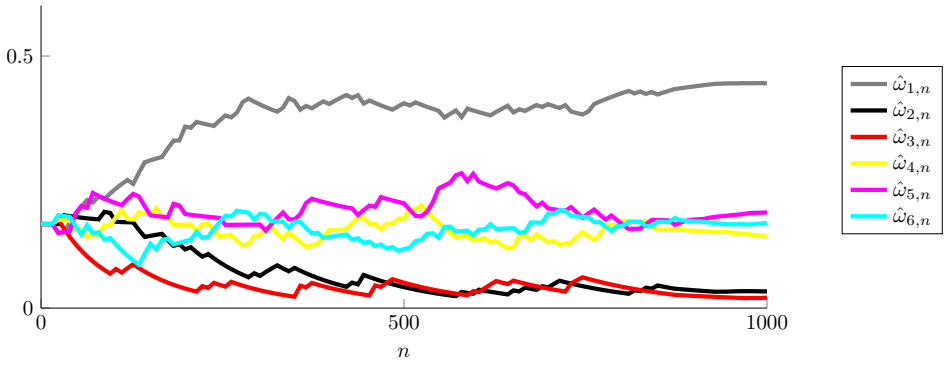
Notons que la prise en compte du facteur d'échelle $\lambda > 0$ est d'une importance primordiale dans le cas d'un apprentissage *non-supervisé* comme le montre la 3-ème colonne de la Figure III-8 illustrant l'apprentissage des deux prototypes sous σ_2 lorsque λ est fixé à 1. La perte de ce degré de liberté empêche l'algorithme de recaler les observations ayant une plus forte intensité, comme c'est le cas de certaines images d'Avion 1. L'estimation d'une classe, ici la classe $j = 1$, décrivant ces données singulières conduit à la disparition de la classe décrivant l'avion ayant la photométrie la plus faible, ici l'Alphajet. Sans prise en compte de λ, les deux classes de cette simulation décrivent l'Avion 1.

Même si pour les raisons évoquées ci-dessus ce cas ne correspond pas à une situation réaliste en pratique, nous avons tout de même effectué un apprentissage *non-supervisé* avec $C = 6$ classes pour le niveau de bruit σ_1. Les résultats illustrés par la Figure III-9 montrent que les classes obtenues coïncident avec les résultats de l'apprentissage *semi-supervisé* : les trois premières classes correspondent aux classes de l'Avion 1 et les trois dernières aux classes de l'Alphajet avec des proportions similaires à l'apprentissage *semi-supervisé* (Cf. Figures III.4(d) III.6(d)).

L'analyse de ces résultats montre que l'algorithme MCoEM permet d'extraire des fonctions prototypes à partir d'images faiblement résolues. Bien que ne correspondant pas au scénario de défense envisagé dans cette thèse, nous avons également testé notre algorithme sur des observations ayant une meilleure résolution. Le simulateur présenté dans la Partie Simulation de SIR permet en effet d'obtenir des images de différentes résolutions. Celles que nous utilisons à présent sont des images de taille 64×64 dont nous ne gardons que



(a) $\mathcal{T}_{\hat{\alpha}_{j,n}}$



(b) $\hat{\omega}_{j,n}$

Figure III-9 – Estimation des prototypes (a) et des poids (b) pour un mélange de $C = 6$ classes dans le cas non-supervisé et sous le niveau de bruit σ_1

la partie centrale pour former des images 20×20. Les prototypes que nous extrayons de ces images confirment l'estimation effectuée sur les images de basse résolution. Nous exposons deux résultats d'apprentissage : l'Alphajet en *semi-supervisé* avec $C = 3$ classes et l'Alphajet et l'Avion 1 en *non-supervisé*.

Pour les simulations avec ces images plus résolues, nous choisissons comme matrice de covariance des déformations locales la matrice structurée définie par :

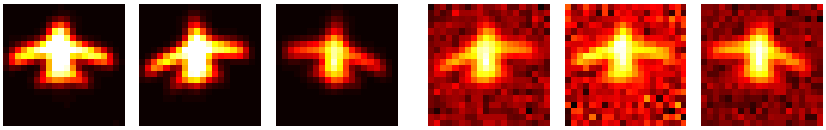
$$\Gamma_{\gamma_j} = \gamma_j^2 M \quad \text{où} \quad \forall (k,k') \in \{1,\dots,d_{\mathsf{D}}\},\ M_{k,k'} = \psi_k(k')$$

et $m = 20$ noyaux Gaussiens d'écart type $\nu_p = 0.005$ comme base des fonctions prototypes $\mathcal{T}$. La Figure III.10(c) illustre la simulation d'observations à partir du modèle génératif (III.3) et (III.4) utilisant les paramètres estimés. Qualitativement, la ressemblance entre ces observations simulées et les observations *réelles* de la Figure III.10(b) montre qu'à l'exception du paramètre de bruit (Cf. la remarque ci-dessus) les paramètres du modèle sont estimés de façon satisfaisante.

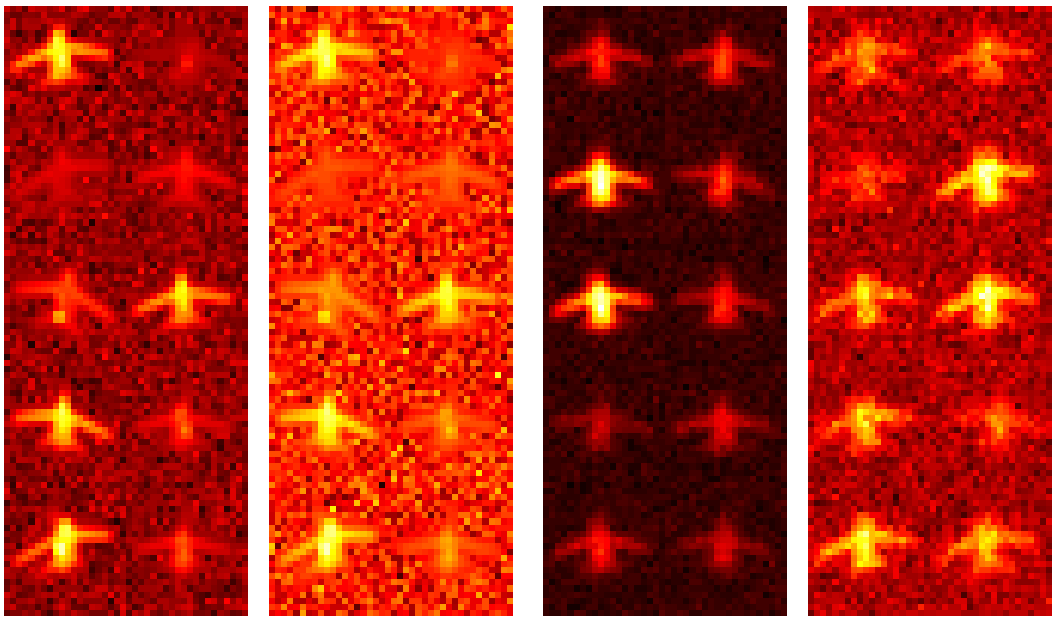
2 .3 Influence de l'échantillonneur MCMC

L'échantillonneur MCMC joue un rôle primordial dans l'estimation des paramètres. Nous illustrons à présent la simulation des données manquantes (J, β, λ) par la méthode de Carlin et Chib que nous comparons à un échantillonneur moins performant, l'algorithme de Gibbs. Les résultats montrent que dans ce contexte, l'utilisation de la méthode de Carlin et Chib est indispensable pour que le MCoEM converge. Pour des raisons d'illustrations, nous considérons dans ce paragraphe l'apprentissage *semi-supervisé* de SIR d'Alphajet fortement résolues.

Nous nous plaçons lors de la 500-ième itération de l'algorithme MCoEM qui a permis d'obtenir les prototypes présentés dans la Figure III.10(b). Une nouvelle observation est disponible et il convient de simuler les données manquantes à savoir l'indice de classe J, les



(a) $\mathcal{T}_{\hat{\alpha}_{j,n}}$



(b) Y_n (c) $Y_n \sim \mathbb{P}_{\hat{\theta}}$

FIGURE III-10 – Prototypes (a), exemple d'observations (b) et simulation de nouvelles observations à partir des paramètres estimés (c), pour l'Alphajet et sous les deux niveaux de bruits σ_1 (gauche) et σ_2 (droite)

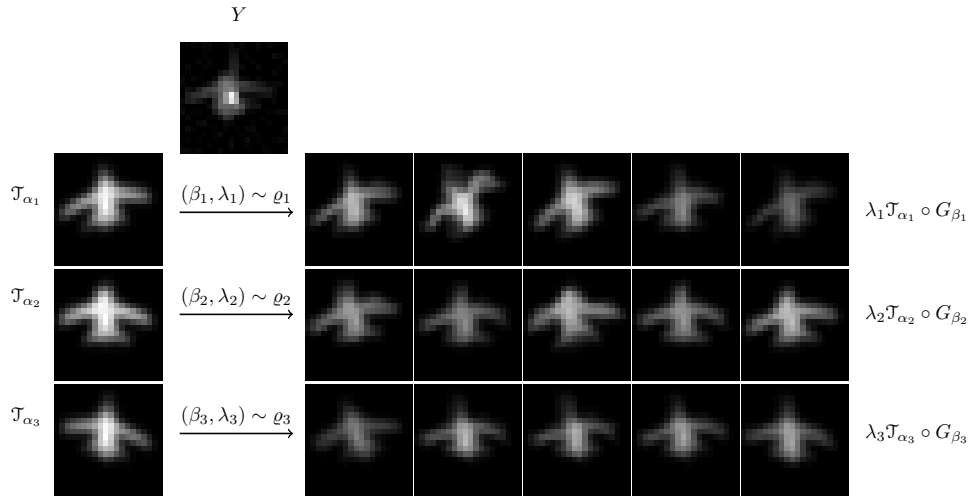
paramètres de déformation du plan β, incluant une rotation, une translation et un champ de déformation local et le paramètre d'échelle λ. Ces variables admettent comme densité *a posteriori* par rapport à la mesure de Lebesgue, la fonction π_θ définie sur $\{1,\dots,C\} \times \mathsf{B} \times \mathbb{R}^+$ par

$$\pi_\theta(j,\beta,\lambda \,|\, Y) \propto p_\theta(Y \,|\, j,\beta,\lambda) g_\theta(j,\beta,\lambda) \; . \tag{III.6}$$

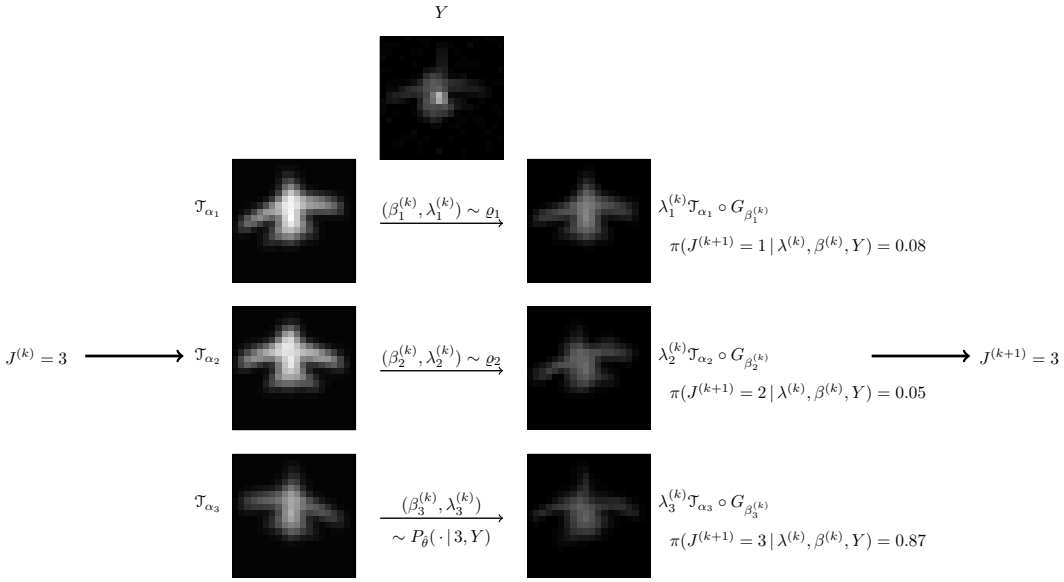
Lorsque l'on utilise l'échantillonneur de Carlin et Chib (voir Algorithme 7), la première étape concerne le calcul des paramètres des pseudo-priors $\{\zeta_j,\, j \leq C\}$ que nous choisissons Gaussiens. A la différence de l'exemple de la regression gaussienne (voir Chapitre II, Section 5), l'optimisation de la fonction $(\beta,\lambda) \to \pi_\theta(\,\cdot \,|\, j, Y)$ n'est pas réalisable en un temps *raisonnable*, car le vecteur (β,λ) possède une centaine de coordonnées. Nous simulons pour chaque classe $j \in \{1,\dots,C\}$ une chaîne de Markov $\{(\beta_j^{(k)}, \lambda_j^{(k)}), k \in \mathbb{N}\}$ au moyen d'un algorithme de Metropolis-within-Gibbs (voir Remarque IV.23) admettant comme distribution stationnaire $\pi_\theta(\,\cdot \,|\, j, Y)$. La moyenne et la matrice de covariance $\{(\mu_j, \Sigma_j),\, j \leq C\}$ des pseudo-priors sont fixés comme la moyenne et la covariance empirique des réalisations de la chaîne. Rappelons que dans ce contexte, l'objectif des pseudo-priors est de permettre d'effectuer un recalage grossier pour chaque classe, de sorte à ce que tous les modèles soient en compétition. Il n'y a donc pas de contrainte d'optimalité sur ces paramètres et en pratique 1000 étapes de Metropolis sont suffisantes pour obtenir des déformations acceptables comme le montre la Figure III.11(a). Les 200 premières itérations de la chaîne de Markov, correspondant au régime transitoire, ne sont pas prises en compte pour le calcul de ces paramètres. La Figure III.11(b) illustre quant à elle une étape de transition de la chaîne de Carlin et Chib sur l'espace étendu : à l'itération k, la classe $J^{(k)} = 3$ indique que les données manquantes pour les classes 1 et 2 sont simulées respectivement par ζ_1 et ζ_2. Les données manquantes relatives à la classe $j = 3$ sont simulées par un noyau de Metropolis-within-Gibbs identique à celui employé pour estimer les paramètres de ζ_3. Bien que les classes 1 et 3 soient majoritairement tirées *a posteriori* , la Figure III.11(c) montre que la chaîne a visité au moins une fois tous les modèles au cours de ses 100 itérations.

La Figure III-12 illustre en détail le recalage du prototype de la classe 1 sur une observation Y. Les paramètres (λ,β) permettant ces déformations sont obtenus lorsque la chaîne de Carlin et Chib a atteint sa distribution stationnaire $\pi_{\hat\theta}(\,\cdot \,|\, 1, Y)$. La géométrie du prototype est parfaitement déformée tandis que le seul paramètre d'échelle λ ne permet pas d'appréhender la complexité de Y dans l'espace de mesure : ce paramètre permet simplement de recaler le niveau moyen d'intensité. A ce stade, les représentations en trois dimensions (dernière ligne), nous invitent à considérer en plus du paramètre λ un champ de déformation local similaire à $\sum_{k=1}^{d_{\mathsf{D}}} \delta_k \psi_k$ pour ajuster les niveaux d'intensité localement. Toutefois, dans cette optique, le modèle ne serait plus exponentiel et le MCoEM ne peut être utilisé pour estimer les paramètres du modèle sans nécessiter une adaptation particulière.

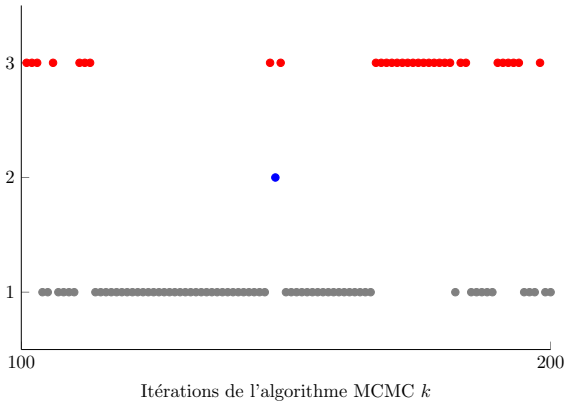
La Figure III-13 montre l'influence du nombre d'itérations de la chaîne de Carlin et Chib sur l'estimation des prototypes. Le scénario est le suivant : nous considérons toujours l'itération $n = 500$ de l'algorithme MCoEM et l'estimateur $\hat\theta_n$ associé mais nous prenons en compte uniquement les classes 1 et 2. Une nouvelle observation Y_{n+1} qui n'est décrite *a priori* par aucun prototype est disponible et permet de calculer l'estimateur suivant $\hat\theta_{n+1}$. Rappelons que la qualité de l'estimateur $\hat\theta_{n+1}$ dépend de la qualité de l'approximation de l'espérance des statistiques suffisantes du modèle $\mathbb{E}_{\hat\theta_n}[S(J,\beta,\lambda) \,|\, Y_{n+1}]$ qui est approchée dans le MCoEM par une méthode MCMC. Les deux colonnes du tableau de la Figure III-13 donnent respectivement le nombre des premières simulations (*burn*) de la chaîne de Carlin et Chib qui ne sont pas conservées pour l'approximation de l'espérance et le nombre total d'itérations effectuées. Pour chaque ligne, les prototypes estimés sont représentés dans la colonne centrale.



(a) Simulation des données manquantes par les pseudopriors



(b) Simulation des variables auxiliaires lors d'une itération de la chaîne de Carlin et Chib sur l'espace étendu



(c) Échantillonnage de la variable de classe par la chaîne de Carlin et Chib

FIGURE III-11 – Illustration de la chaîne de Markov de Carlin et Chib

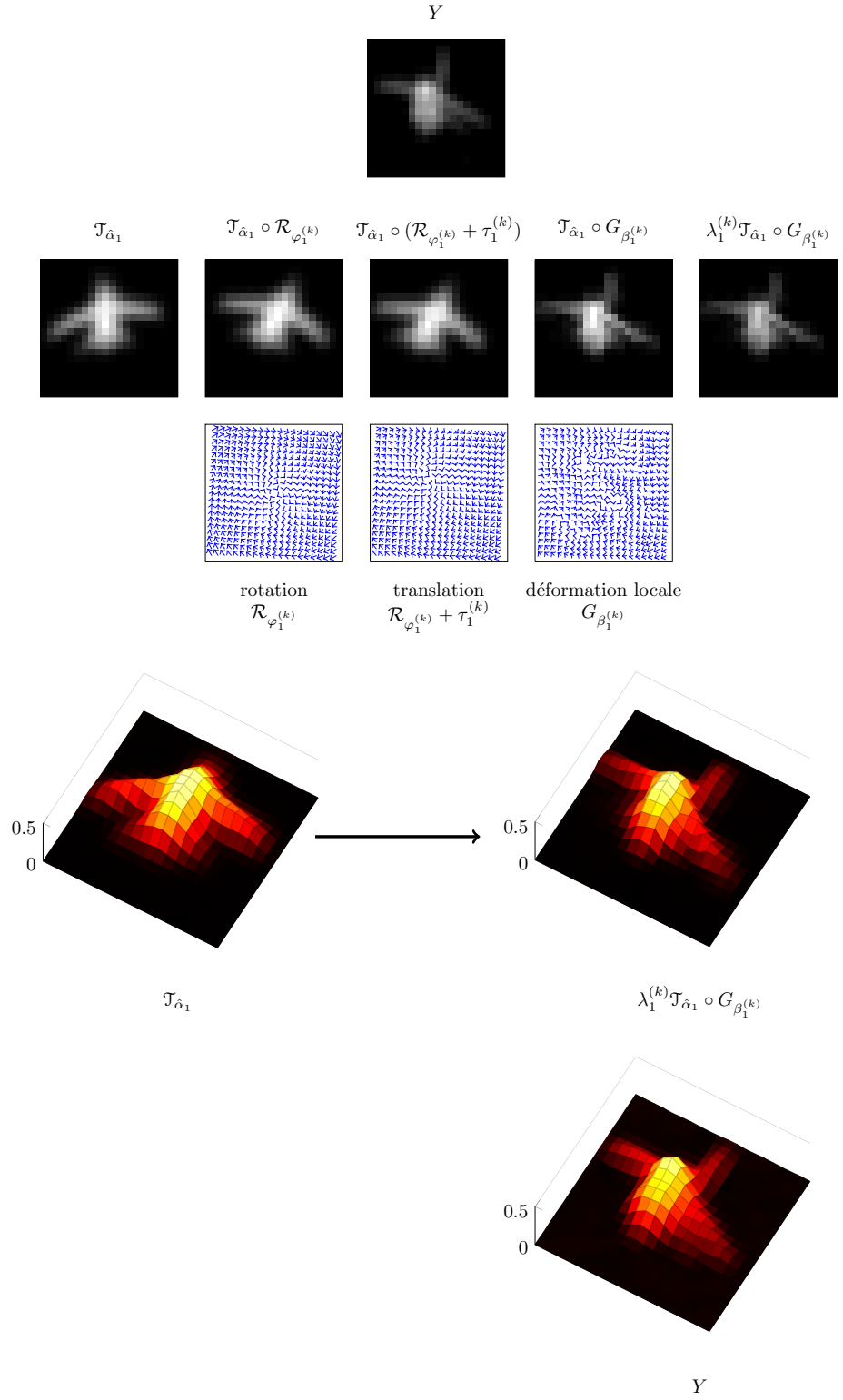
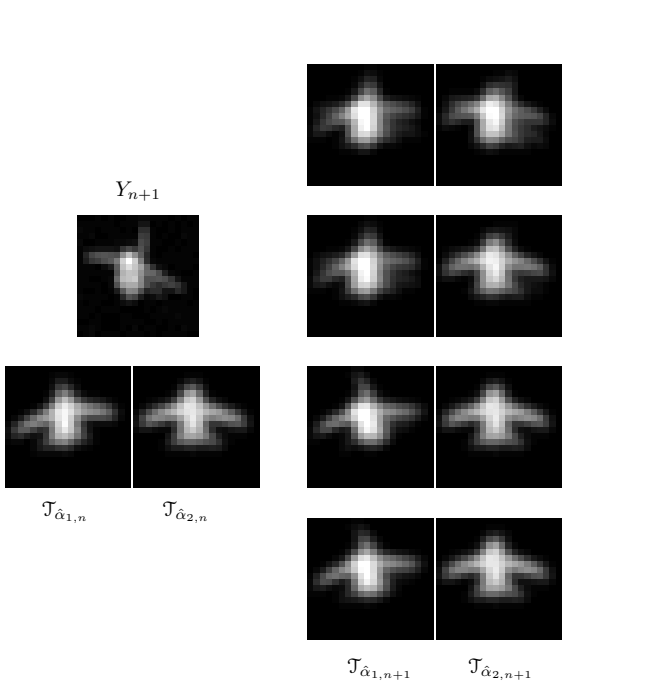


Figure III-12 – Simulation des déformations d'un prototype conditionnellement à une observation



Itérations MCMC	Burn MCMC
20	0
100	50
200	100
Optimisation	

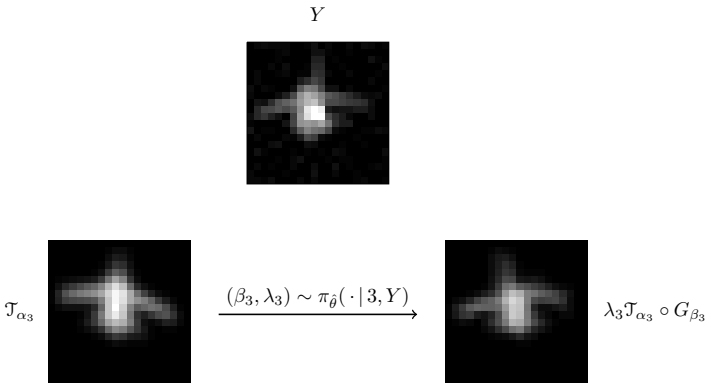
FIGURE III-13 – Influence du nombre d'itérations de la chaîne de Markov sur l'estimation des prototypes

La mise à jour des paramètres peut être interprétée de la façon suivante : les éléments géométriques et photométriques de la nouvelle observation qui n'ont pas pu être reconstruits à partir des prototypes et du modèle de déformation sont intégrés par les nouveaux paramètres. Cette situation est présente dans le premier cas où le faible nombre d'itérations de la chaîne de Markov ne permet pas de recaler l'observation sur les prototypes. La géométrie des nouveaux prototypes est par conséquent affectée brutalement par cette observation. A l'inverse, les nouveaux prototypes ne varient pas par rapport aux anciens dans les zones où les déformations simulées ont permis de reconstruire l'observation, comme c'est le cas lorsque la loi *a posteriori* est correctement échantillonnée. Dans cet exemple, les déformations simulées par la chaîne de Markov et intervenant dans l'approximation de l'espérance conditionnelle ont permis de retrouver la géométrie du fuselage et des ailes et le nouveau prototype de la classe 1 ne change donc pas à ce niveau. En revanche la présence de l'aileron sur l'observation n'a pas été expliqué par le modèle et se retrouve intégré dans le prototype. La dernière ligne correspond au cas où la loi *a posteriori* est assimilée à la distribution de Dirac au point

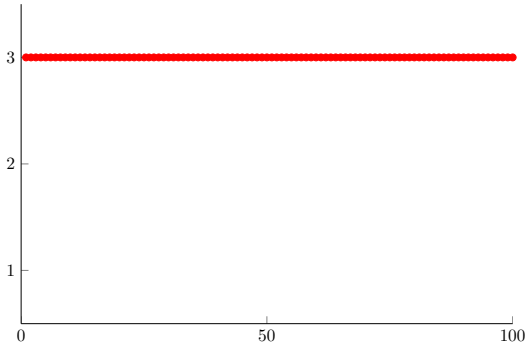
$$(J^{\star}_{n+1}, \beta^{\star}_{n+1}, \lambda^{\star}_{n+1}) = \arg\max_{j,\beta,\lambda} \pi_{\hat{\theta}_n}(j, \beta, \lambda \,|\, Y_{n+1}) \; .$$

Comme le montre la Figure III-12, il existe des paramètres (j, β, λ) permettant de reconstruire presque parfaitement l'observation et par conséquent les prototypes ne présentent pas de variation d'une itération à l'autre. La solution consistant à remplacer le calcul de l'espérance par l'approximation des modes [MMTY08, AAT07] ne semble pour cette raison pas appropriée pour extraire des formes typiques dans ce contexte.

Nous proposons dans la Figure III-14 une illustration de la simulation des données manquantes par l'échantillonneur de Gibbs. Le contexte est le même que celui de la Figure III-11 *i.e.* la simulation se déroule conditionnellement à la même observation Y et au



(a) Simulation des données manquantes par les pseudopriors

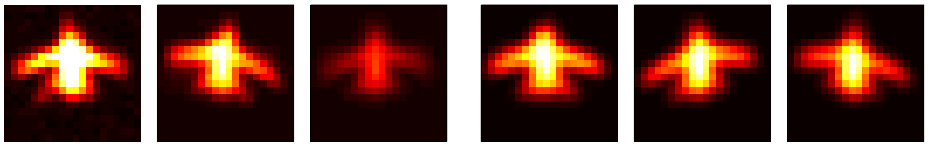


(b) Simulation des variables auxiliaires lors d'une itération de la chaîne de Carlin et Chib sur l'espace étendu

Figure III-14 – Illustration de la chaîne de Markov obtenue par l'échantillonneur de Gibbs

même estimateur de paramètres $\hat{\theta}$. La chaîne de Markov simulée par l'échantillonneur de Gibbs reste durant toute la simulation dans la même classe tandis que, comme le montre l'algorithme de Carlin et Chib, la classe $j = 1$ a un poids *a posteriori* majoritaire. Ce scenario est une nouvelle illustration du problème des classes absorbantes déjà rencontré dans les Sections D .3 et 5 .

Enfin, nous comparons les prototypes estimés après $n = 1000$ itérations de l'algorithme MCoEM couplé avec l'échantillonneur de Gibbs (III.15(b)) et l'algorithme de Carlin et Chib (III.15(a)) pour simuler les données manquantes. Les mauvaises performances de l'échantillonneur de Gibbs à-propos de la simulation de la variable de classe impactent les prototypes qui ne sont pas aussi résolus (par exemple la classe $j = 1$) que ceux obtenus en utilisant l'algorithme de Carlin et Chib. Dans cette situation, les prototypes peuvent rencontrer des difficultés à se stabiliser à cause des mises à jour parfois inappropriées, causées par l'algorithme de Gibbs. Dans certains cas, des classes peuvent même devenir marginales en terme de représentativité des données comme la classe $j = 3$ dans la Figure III.15(b).



(a) Estimation des prototypes par l'algorithme MCoEM - Gibbs

(b) Estimation des prototypes par l'algorithme MCoEM - Carlin et Chib

FIGURE III-15 – MCoEM avec deux échantillonneurs différents

3 Cas multispectral

Nous nous intéressons dans cette section à l'extraction de caractéristiques photométriques, géométriques et spectrales de SIR multispectrales. Rappelons qu'une SIR multispectrale à K bandes est une observation $Y = (Y_1, \ldots, Y_K)$ telle que pour tout $k \in \{1, \ldots, K\}$, $Y_k \in \mathsf{Y} = \mathbb{R}^{|\Omega|}$.

3 .1 Adaptation du modèle d'observation

Grâce à une étude préalable sur les SIR multispectrales, nous remarquons que :

- la structure géométrique d'un aéronef varie peu en fonction de la bande spectrale considérée, comme le montrent les Figures 2, 5 et 10, de la Partie Simulation de SIR,
- comme les bandes d'absorption de certaines molécules contenues dans l'atmosphère (CO_2 et H_2O) se situent entre 2000 et 3000 cm^{-1}, le profil spectral des différentes SIR multispectrales présente certaines similitudes, voir par exemple la Figure 3.

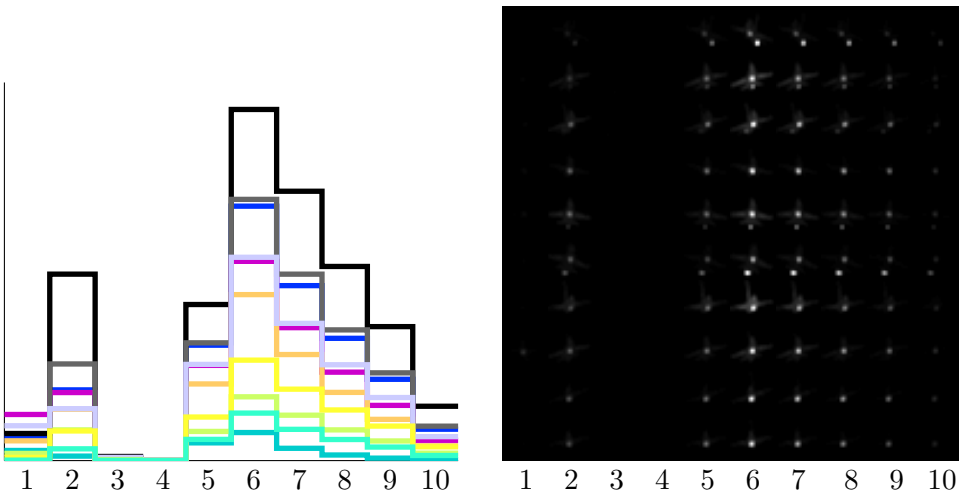
Ces deux remarques sont illustrées par la Figure III-16 qui représente pour 10 observations, la valeur *moyenne* des pixels de la cible dans chaque bande $k \in \{1, \ldots, 10\}$ (a) et les SIR multispectrales associées (b). Chaque observation étant représentée par une couleur différente dans la Figure III-16 (a), l'existence d'un coefficient permettant de passer d'un profil spectral à un autre semble vraisemblable. Toutefois, nous insistons sur le fait que les profils spectraux représentés ici correspondent à la *moyenne* des pixels de la cible dans chaque bande et ne tiennent par conséquent pas compte de la géométrie. Des disparités locales en terme d'intensité, qui n'apparaissent pas dans cette figure, sont susceptibles d'intervenir.

A partir de ces observations, nous considérons dans un premier temps que deux SIR multispectrales $Y_i = (Y_{1,i}, \ldots, Y_{K,i})$ et $Y_j = (Y_{1,j}, \ldots, Y_{K,j})$ sont reliées par deux données cachées : un facteur d'échelle $\lambda_{i,j} > 0$ et une déformation du plan $G_{\beta_{i,j}} \in \mathsf{G}$, $\beta_{i,j} \in \mathsf{B}$, tels que pour tout $k \in \{1, \ldots, K\}$

$$\mathcal{Y}_{k,i} = \lambda_{i,j} \mathcal{Y}_{k,j} \circ G_{\beta_{i,j}} \ , \tag{III.7}$$

où $\mathcal{Y}_i = (\mathcal{Y}_{1,i}, \ldots, \mathcal{Y}_{K,i})$ est une interpolation de Y_i.

Le mélange de modèles à prototype déformable ayant permis de modéliser avec succès les SIR monospectrales, nous conservons cette approche et nous généralisons le modèle d'observation proposé dans la section précédente. S'inspirant de l'équation (III.7), une



(a) (b)

Figure III-16 – Exemples de 10 observations ne variant approximativement qu'à un facteur de proportionnalité près et dont la géométrie est similaire dans les bandes spectrales

SIR multispectrale à K bandes $Y = (Y_1, \dots, Y_K)$ suit le modèle hiérarchique suivant :

$$\begin{cases} J \sim (\omega_1, \dots, \omega_C) \ , \\ \lambda \sim \text{Gamma}(a, b) \ , \\ \beta \,|\, J = j \sim g_\theta(\cdot \,|\, j) \ , \\ \forall k \in \{1, \dots, K\}, \text{ tirer indépendamment} \\ Y_k \,|\, J = j, \beta, \lambda \sim \mathcal{N}_{|\Omega|}(\lambda \Phi_\beta \alpha_{k,j}, \sigma_k \text{Id}_{|\Omega|}) \ . \end{cases} \tag{III.8}$$

Ce modèle fait l'hypothèse qu'il existe pour chaque classe $j \in \{1, \dots, C\}$, un prototype spectral $\mathcal{T}_j = (\mathcal{T}_{1,j}, \dots, \mathcal{T}_{K,j})$ tel que pour chaque bande $k \in \{1, \dots, K\}$, $\mathcal{T}_{k,j}$ est paramétré par un vecteur $\alpha_{k,j} \in \mathbb{R}^m$. Les commentaires de la Partie Simulation relatifs au modèle de fond de ciel non texturé en multispectral nous conduisent à considérer un niveau de bruit σ_k différent pour chaque bande $k \in \{1, \dots, K\}$.

La corrélation entre les différentes bandes d'une observation est modélisée

(i) au niveau géométrique - conditionnellement à la classe $j \in \{1, \dots, C\}$, les prototypes de chaque bande $\{\mathcal{T}_{\alpha_{k,j}}\}_{k=1}^K$ subissent la même déformation $G_\beta \in \mathsf{G}$,

(ii) au niveau photométrique - le facteur d'échelle $\lambda > 0$ est identique quelque soit la bande spectrale considérée.

La prise en compte de différentes classes de prototypes spectraux modélisent différentes géométries et profils spectraux caractéristiques.

Sous ces hypothèses, le modèle III.8 reste exponentiel et l'ensemble des paramètres inconnus du modèle à estimer est

$$\Theta = \bigcup_{j=1}^{C} \left\{ (\Gamma_j, \omega_j), \bigcup_{k=1}^{K} (\alpha_{k,j}, \sigma_k) \,|\, \alpha_{k,j} \in \mathbb{R}^m, \Gamma_j \in \mathcal{M}^+(\mathbb{R}), \omega_j \in (0,1), \sigma_k > 0 \right\} \cap \left\{ \sum_{j=1}^{C} \omega_j = 1 \right\} .$$

Toutefois, des différences par rapport à l'apprentissage dans le cas monospectral existent et se situent principalement à trois niveaux :

(i) Les deuxième et cinquième composantes du vecteur des statistiques exhaustives de la classe $i \in \{1, \dots, C\}$ ne sont plus les fonctions $S_i^{(2)}(j, \lambda, \beta, y) = \lambda \Phi_\beta^T y \delta_i(j)$ et $S_i^{(5)}(j, \lambda, \beta, y) = \|y_k\|^2 \delta_i(j)$ comme dans (3) mais les vecteurs de fonctions définis pour $\ell \in \{2, 5\}$ par

$$S_i^{(\ell)}(j, \lambda, \beta, y) = (S_{i,1}^{(\ell)}(j, \lambda, \beta, y), \dots, S_{i,K}^{(\ell)}(j, \lambda, \beta, y)) \ ,$$

où pour tout $k \in \{1, \dots, K\}$, $S_{i,k}^{(\ell)}(j, \lambda, \beta, y) = S_i^\ell(j, \lambda, \beta, y_k)$.

(ii) La loi des données manquantes *a posteriori* , cible de l'échantillonneur MCMC, admet pour densité la fonction

$$\pi_\theta(j, \beta, \lambda \,|\, y) \propto \omega_j g_\theta(\beta \,|\, j) \prod_{k=1}^{K} p_\theta(y_k \,|\, j, \beta, \lambda) \ , \tag{III.9}$$

où g_θ et p_θ sont les densités respectivement définies en (III.3) et (III.4).

(iii) La mise à jour des paramètres des prototypes et des niveaux de bruit s'écrivent pour tout $(i, k) \in \{1, \dots, C\} \times \{1, \dots, K\}$

$$\begin{cases} \hat{\alpha}_{k,i,n} = (\hat{s}_{k,i,n}^{(3)})^{-1} \hat{s}_{i,n}^{(2)} \ , \\ \hat{\sigma}_{k,n}^2 = \frac{1}{|\Omega|} \left(\sum_{i=1}^{C} \hat{s}_{k,i,n}^{(5)} - 2\hat{\alpha}_{k,i,n}^T \hat{s}_{k,i,n}^{(2)} + \hat{\alpha}_{k,i,n}^T \hat{s}_{i,n}^{(3)} \hat{\alpha}_{k,i,n} \right) \ . \end{cases}$$

où pour $\ell \in \{2, 5\}$, $\hat{s}_{k,i,n}^{(\ell)}$ est l'approximation stochastique de $\mathbb{E}_{\hat{\theta}_{n-1}}[S_{i,k}^\ell(J, \lambda, \beta, Y) \,|\, Y]$ obtenue à la n-ième itération du MCoEM .

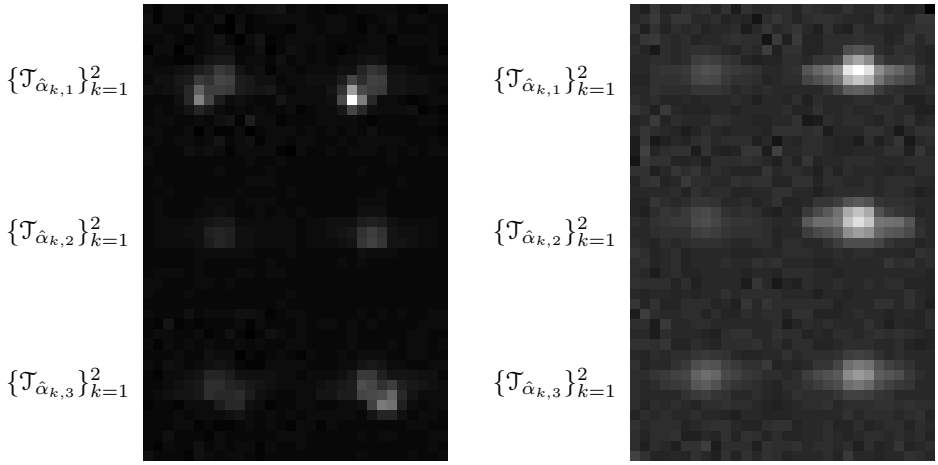
Ces changements mineurs ne remettent pas en question l'utilisation du MCoEM pour estimer les paramètres du modèle. De plus, le passage aux données multispectrales ne ralentit pas l'apprentissage : en effet, l'étape *coûteuse* en ressource est la simulation des données manquantes qui ne changent pas sous le modèle (III.8). La seule différence concerne la loi *a posteriori* des données manquantes (III.9) qui n'est pas plus compliquée à échantillonner que la loi équivalente dans le cas monospectral (III.6).

L'utilisation des SIR multispectrales nécessite de spécifier, en amont de l'apprentissage, les bandes (ou regroupements de bandes) prises en compte dans l'estimation des paramètres. Nous reprenons le système d'identification des regroupements de bandes spectrales proposé dans le Chapitre I, voir Figure I-5. Nous désignerons dans la suite de cette partie par η_K le vecteur paramétrant un regroupement de K bandes et par H_K l'espace associé. Avec cette notation, les images monospectrales intégrées en bande large traitées dans la section précédent correspondent au paramètre $\eta_1 = (0, 10)$.

Comme le montrent les résultats d'apprentissage des paramètres du modèle dans le cas des SIR monospectrales, le rapport signal à bruit des données d'entrée est un paramètre fortement influent sur les performances du MCoEM , comparer par exemple l'estimation des prototypes de l'Alphajet sous les niveaux de bruit σ_1 et σ_2 (Figure III.10(a)). Ainsi, pour un mauvais choix de bandes (par exemple $\eta_2 = (2, 2, 5, 1)$), le modèle n'apprendra aucun paramètre caractéristique des cibles, les aéronefs ayant une signature infrarouge très faible (quasiment nulle) dans ces bandes, voir la Figure III-16 (b). Rappelons que les luminances spectrales des cibles et les variances du bruit dans les différentes bandes sont des grandeurs additives. Il semble donc intéressant de regrouper des bandes contiguës associées à de fortes luminances et un faible niveau de bruit. Observant les SIR multispectrales présentées dans

ce document, comme par exemple celles de la Figure III-16 (b), un premier choix naturel de bandes correspond au regroupement des deux modes d'intensité, soit $\eta_2 = (1, 1, 2, 4)$. Nous considérons pour chaque bande spectrale $k \in \{1, \ldots, K\}$ deux niveaux de bruits $\sigma_{k,1}$ et $\sigma_{k,2}$ équivalents aux niveaux de bruit σ_1 et σ_2 en monospectral utilisés dans la section précédente ; voir (11).

Apprentissage semi-supervisé La Figure III-17 illustre l'apprentissage semi-supervisé des fonctions prototypes pour les deux avions sous le niveau de bruit équivalent à σ_1. Pour l'Avion 1, on retrouve les trois classes obtenues en monospectral (Figure III.3(b)), aussi bien en terme de prototypes, de variances des déformations, que de poids *a priori* : la prise en compte de plusieurs bandes spectrales ne remet pas en cause le caractère discriminant des trois types de géométries majoritaires. Pour l'Alphajet en revanche, la géométrie semble avoir moins d'importance que dans le cas de l'Avion 1. En effet, les prototypes des trois classes ont une géométrie plus proche que dans le cas monospectral (Figure III.5(b)) et les variances des paramètres de déformations sont plus grandes, indiquant que pour chaque classe, la loi *a posteriori* autorise l'échantillonnage de déformations plus variées. Les caractéristiques spectrales de l'Alphajet semblent être le critère discriminant pour l'apprentissage : les trois classes se différencient en effet dans l'intensité relative des prototypes des deux regroupements.



(a) Prototypes de l'Avion 1 (b) Prototypes de l'Alphajet

Figure III-17 – Prototype des deux avions pour $C = 3$ classes, $\eta_2 = (1, 1, 2, 4)$ et sous le niveau de bruit équivalent à σ_1

Apprentissage non-supervisé La Figure III-18 (a) présente les prototypes et les poids *a priori* obtenus après $n = 1000$ itérations du MCoEM sous le niveau de bruit équivalent à σ_2, à partir d'images paramétrées par $\eta_2 = (1, 1, 2, 4)$. Comme dans le cadre d'apprentissage semi-supervisé, les deux classes de l'Alphajet sont caractérisées par des géométries similaires et par des rapports de photométries différents entre les deux regroupements de bandes. Cette information n'est pas accessible dans le cas des images monospectrales en bande large et, comme le montre la Figure III-18 (b), une seule classe pour l'Alphajet est estimée dans ce cas : une composante du modèle de mélange avec $C = 4$ a un poids *a priori* effectivement nul. En conséquence, la prise en compte d'images multispectrales

permet un apprentissage plus précis des caractéristiques géométriques et spectrales des aéronefs.

Autres intérêts offerts par l'utilisation d'images multispectrales En plus d'améliorer les résultats d'apprentissage du MCoEM , l'utilisation de SIR multispectrales possède d'autres avantages.

- A partir de SIR multispectrales, il est possible d'obtenir une base d'apprentissage de SIR monospectrales lorsqu'un paramètre $\eta_1 \in \mathsf{H}_1$ est spécifié. Dans ce cas, l'algorithme d'apprentissage en monospectral est implémentable sur des images ayant un rapport signal à bruit supérieur à celui des SIR monospectrales intégrées en bande large. Nous comparons l'apprentissage dans un cadre *semi-supervisé* de l'Alphajet sous un niveau de bruit équivalent à σ_2, en monospectral avec $\eta_1 = (5, 3)$ (Figure III-19 (a)) et $\eta_1 = (0, 10)$ (Figure III-19 (b)). L'apprentissage en bande étroite permet d'estimer trois classes distinctes en terme de photométrie et de géométrie ce qui n'est pas le cas en bande large.
- Dans certains cas, la base d'apprentissage peut contenir des aberrations, parfois appelées *outliers*. Nous illustrons sur la Figure III-19 (a) le scenario d'apprentissage de l'Avion 1 en bande large sous un bruit σ_1. L'observation Y_{n_0} est un outlier : 4 pixels appartenant à la cible ont des valeurs en moyenne 100 fois plus importantes que des pixels de cibles *normales*. Dans ce cas, le modèle ne permet pas de décrire cette observation pathologique car les déformations nécessaires pour recaler les prototypes existants sur Y_{n_0} ont une probabilité nulle sous la loi *a posteriori* des données manquantes. Par conséquent, les données manquantes associées à cette observation sont échantillonnées de façon quasi-aléatoire et les prototypes des classes qui auront été simulées par la chaîne de Markov deviennent eux aussi pathologiques. Dans ce scénario, seule la classe $j = 2$ a été simulée par la loi *a posteriori* et cette composante du modèle de mélange devient immédiatement inapte à décrire d'autres observations, comme le montre l'estimation des poids *a priori* . Un apprentissage identique mais effectué sur un regroupement de 3 bandes $\eta_3 = (0, 2, 2, 1, 1, 3)$ des mêmes SIR multispectrales (Figure III-19 (b)) montre que seul le troisième regroupement semble affecté par cette aberration. En effectuant l'apprentissage sur les deux premiers regroupement *i.e.* $\eta_2 = (0, 2, 2, 1)$, la Figure III-19 (c) montre que l'aberration est bien gérée par l'algorithme : à convergence, les estimateurs sont identiques à ceux obtenus en l'absence d'anomalie (voir Figure III-17).

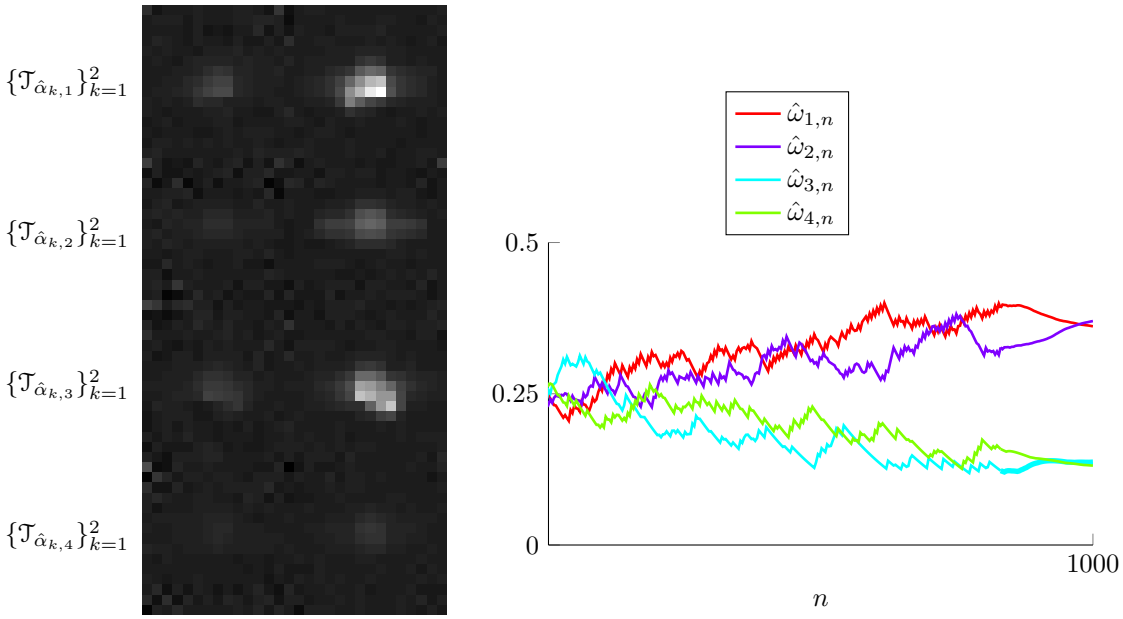
3 .2 Vers un modèle plus général de déformation spectro-spatial

Un modèle plus réaliste pour des SIR plus résolues spatialement et spectralement consiste à considérer un prototype variant simultanément dans l'espace et en fonction du nombre d'onde. Comme précédemment, nous considérons une bande spectrale (ν^-, ν^+) inclue dans le domaine de l'infrarouge et une observation multispectrale $\mathcal{Y} = (\mathcal{Y}_1, \ldots, \mathcal{Y}_K)$ telle que où pour tout $k \in \{1, \ldots, K\}$, $\mathcal{Y}_k : \mathbb{R}^2 \to \mathbb{R}$ est l'image intégrée dans la bande (ν_k^-, ν_k^+) avec $\nu^- \leq \nu_1^-$ et $\nu_K^+ \leq \nu^+$. Dans ce modèle, conditionnellement à une déformation du plan et à une déformation du spectre formalisées par l'application

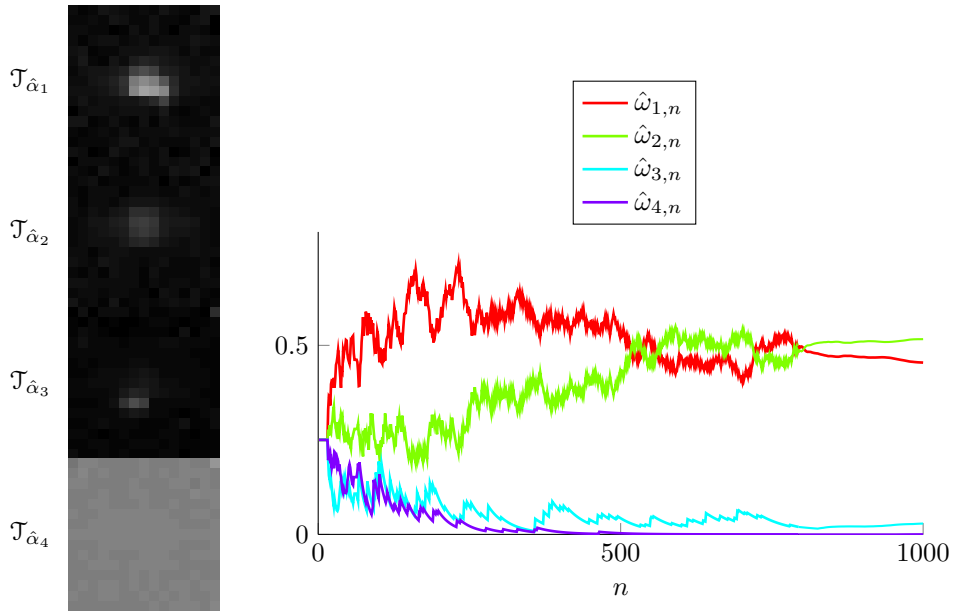
$$T_\varphi : \mathbb{R}^2 \times (\nu^-, \nu^+) \to \mathbb{R}^2 \times (\nu^-, \nu^+)$$

où $\varphi \in \mathsf{F} \subseteq \mathbb{R}^n$ et à un facteur d'échelle $\lambda > 0$, pour tout $k \in \{1, \ldots, K\}$, $\mathcal{Y}_k$ s'écrit pour tout $u \in \mathbb{R}^2$

$$\mathcal{Y}_k(u) = \lambda \int_{\nu_k^-}^{\nu_k^+} \mathcal{F} \circ T_\varphi(u, \nu)\, \mathrm{d}\nu + \sigma_k \mathcal{W}(u)\ , \tag{III.10}$$

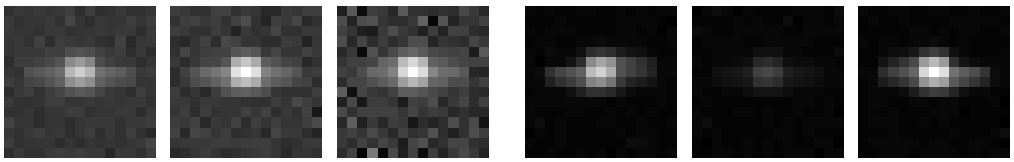


(a) Multispectral



(b) Bande Large

Figure III-18 – Apprentissage non-supervisé des deux avions pour $C = 4$ classes, sous le niveau de bruit équivalent à σ_2, en multispectral $\eta_K = (1, 1, 2, 4)$ (a) et en monospectral en bande large $\eta_1 = (0, 10)$ (b)



(a) Estimation en bande large (b) Estimation en bande étroite

FIGURE III-19 – Comparaison d'apprentissage en monospectral en bande large $\eta_1 = (0, 10)$ (a) et en bande étroite $\eta_1 = (5, 3)$ (b)

où $\mathcal{W}$ est un processus de bruit additif de moyenne nulle et de variance 1.

Les prototypes obtenus par le modèle (III.8) montrent que la géométrie varie peu suivant la bande spectrale considérée. En conséquence, nous considérons les deux hypothèses suivantes :

(i) il existe deux fonctions $\mathcal{T} : \mathbb{R}^2 \to \mathbb{R}$ et $\mathcal{S} : (\nu^-, \nu^+) \to \mathbb{R}^+$ telles que

$$\mathcal{F}(u, \nu) = \mathcal{S}(\nu)\mathcal{T}(u),$$

(ii) il existe deux applications $G_\beta : \mathbb{R}^2 \to \mathbb{R}^2$ et $H_\chi : (\nu^-, \nu^+) \to (\nu^-, \nu^+)$ telles que

$$T_\varphi(u, \nu) = (G_\beta(u), H_\chi(\nu)) \ ,$$

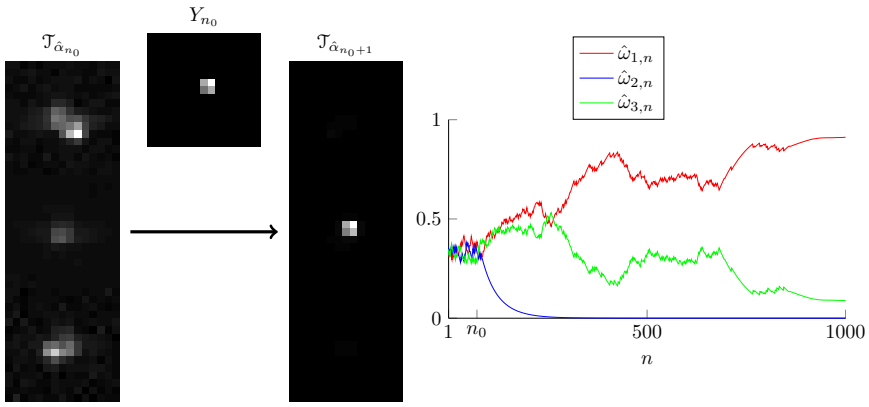
où $\varphi = (\beta, \chi)$.

Sous ces hypothèses, le modèle III.10 s'écrit :

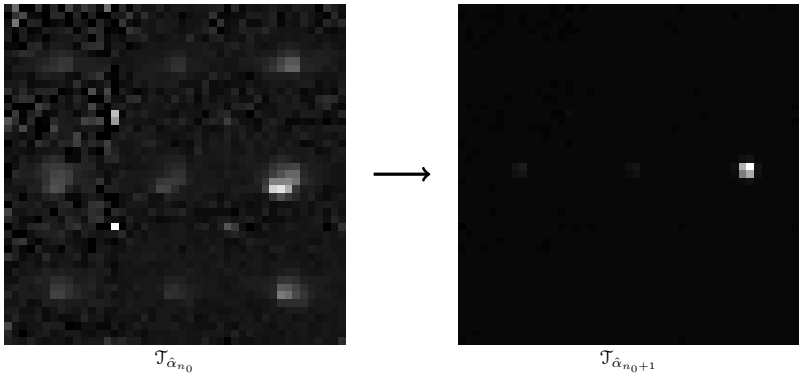
$$\mathcal{Y}_k(u) = \lambda \left(\int_{\nu_k^-}^{\nu_k^+} \mathcal{S} \circ H_\chi(\nu) \, \mathrm{d}\nu \right) \mathcal{T} \circ G_\beta(u) + \sigma_k \mathcal{W}(u) \ . \tag{III.11}$$

Notons que les fonctions $\mathcal{T}$ et G_β jouent un rôle similaire pour la géométrie aux fonctions $\mathcal{S}$ et H_χ pour la photométrie. Les fonctions $\mathcal{T}$ et $\mathcal{S}$ sont des paramètres *a priori* inconnus décrivant respectivement une géométrie caractéristique et un profil spectral caractéristique tandis que les déformations G_β et H_χ permettent des variations autour de ces prototypes. Comme dans les modèles déformables précédents, nous considérons que les observations varient autour de plusieurs prototypes conduisant ainsi à un modèle de mélange. En raison de la faible corrélation entre le profil spectral des observations et leur géométrie, les deux ensembles d'indices $I \in \{1, \ldots, C_s\}$ et $J \in \{1, \ldots, C_g\}$ indiquant respectivement la classe du profil spectral et de la géométrie, sont supposés indépendants. Les mêmes modèles de prototype (II.33) et de déformation II.34 que dans l'exemple des courbes de croissances (Cf. Section II-6) peuvent être choisis pour $\mathcal{S}$ et H_χ. Pour corréler les déformations de la géométrie et du spectre, il est possible de spécifier une loi *a priori* jointe pour le couple de variables aléatoires (β, χ), par exemple en prenant comme densité une fonction dépendant simultanément de ces deux paramètres.

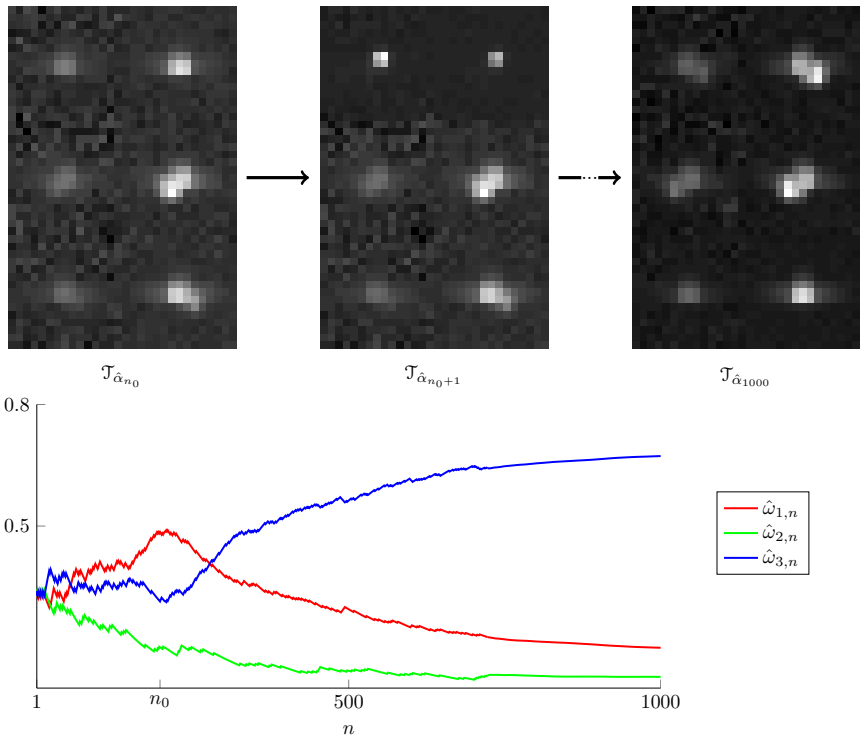
Toutefois, l'algorithme MCoEM ne permet pas d'estimer simultanément les paramètres des prototypes du profil spectral et de la géométrie. En effet, lorsque (III.11) est mise sous forme vectorielle, la fonction de vraisemblance complète ne peut plus se mettre sous forme exponentielle. Une solution envisageable pour se ramener à un modèle exponentiel est de considérer que les profils spectraux sont connus. Comme le montrent les images multispectrales de la Figure 5(b), nous pouvons faire l'hypothèse qu'il existe trois profils spectraux typiques différents ($C_s = 3$) possédant deux modes autour de 2100 cm^{-1} et 2500 cm^{-1}



(a) Apprentissage en bande large



(b) Apprentissage sur un regroupement de 3 bandes $\eta_3 = (0, 2, 2, 1, 1, 3)$



(c) Apprentissage sur un regroupement de 2 bandes $\eta_2 = (0, 2, 2, 1)$

Figure III-20 – Utilité du multispectral lors de la présence d'une aberration dans la base d'apprentissage

dont les amplitudes relatives varient. La Figure III-21 illustre cette approche dans le cadre d'un apprentissage semi-supervisé avec $C_g = 4$ classes à partir d'images multispectrales à $K = 10$ bandes d'Avion 1 sous un bruit équivalent à σ_1. Une observation Y est représentée sur la première ligne, les $C_g = 4$ prototypes géométriques sur la seconde et les $C_s = 3$ prototypes de profil spectral sur la troisième. Les données manquantes *i.e.* $(I, J, \beta, \chi, \lambda)$ sont simulées conditionnellement à Y comme précédemment par l'échantillonneur de Carlin et Chib. Nous illustrons sur la quatrième ligne une déformation du spectre obtenue par la simulation de (J, χ) et sur la cinquième le recalage des prototypes sur Y obtenu par un échantillon de la chaîne de Carlin et Chib.

Comparé au modèle (III.8), l'ensemble des paramètres à estimer est moins important. Rappelons en effet que le prototypes sont des fonctions $\mathcal{T}_j : \mathbb{R}^2 \to \mathbb{R}^K$ dans (III.8) et $\mathcal{T}_j : \mathbb{R}^2 \to \mathbb{R}$ dans le modèle III.10. Par conséquent, un prototype est caractérisé par m^K paramètres dans (III.8) contre m dans (III.10). De plus, ce modèle permet d'accéder à une connaissance plus précise des variations typiques des profils spectraux par l'intermédiaire de l'estimation des lois *a priori* .

L'estimation des caractéristiques géométriques obtenues par ce modèle est sensiblement la même que lors de l'apprentissage effectué en utilisant le modèle (III.8). Bien que proposant un recalage spectral et géométrique satisfaisant, l'étape de simulation des données manquantes par la chaîne de Carlin et Chib ralentit considérablement la procédure d'apprentissage. En effet,

- à la simulation du paramètre manquant de géométrie $\beta \in \mathsf{B}$ nécessaire dans (III.8), s'ajoute la simulation du paramètre manquant $\chi \in \mathsf{H}$ relatif à la déformation du spectre,
- deux variables de classe devant être échantillonnées, l'étape intermédiaire de l'algorithme de Carlin et Chib nécessite la simulation de $C_s \times C_g - 1$ variables auxiliaires contre $C_g - 1$ dans le modèle (III.8).

Ce modèle serait davantage adapté à des données plus résolues spectralement et présentant des profils spectraux plus variés que les SIR multispectrales d'aéronefs. Dans notre cas, comme le montre le diagramme en boîte des luminances des deux cibles différentes pour les 10 bandes spectrales élémentaires (Figure III-22), les prototypes $\mathcal{S}_1$ et $\mathcal{S}_3$ ont un poids marginal. Les deux aéronefs que nous cherchons à classifier ont des profils spectraux trop similaires pour que ce paramètre soit utilisé en tant que critère discriminant. La modèle de variation du profil spectral par le seul intermédiaire du facteur d'échelle $\lambda > 0$ semble être suffisant dans ce contexte.

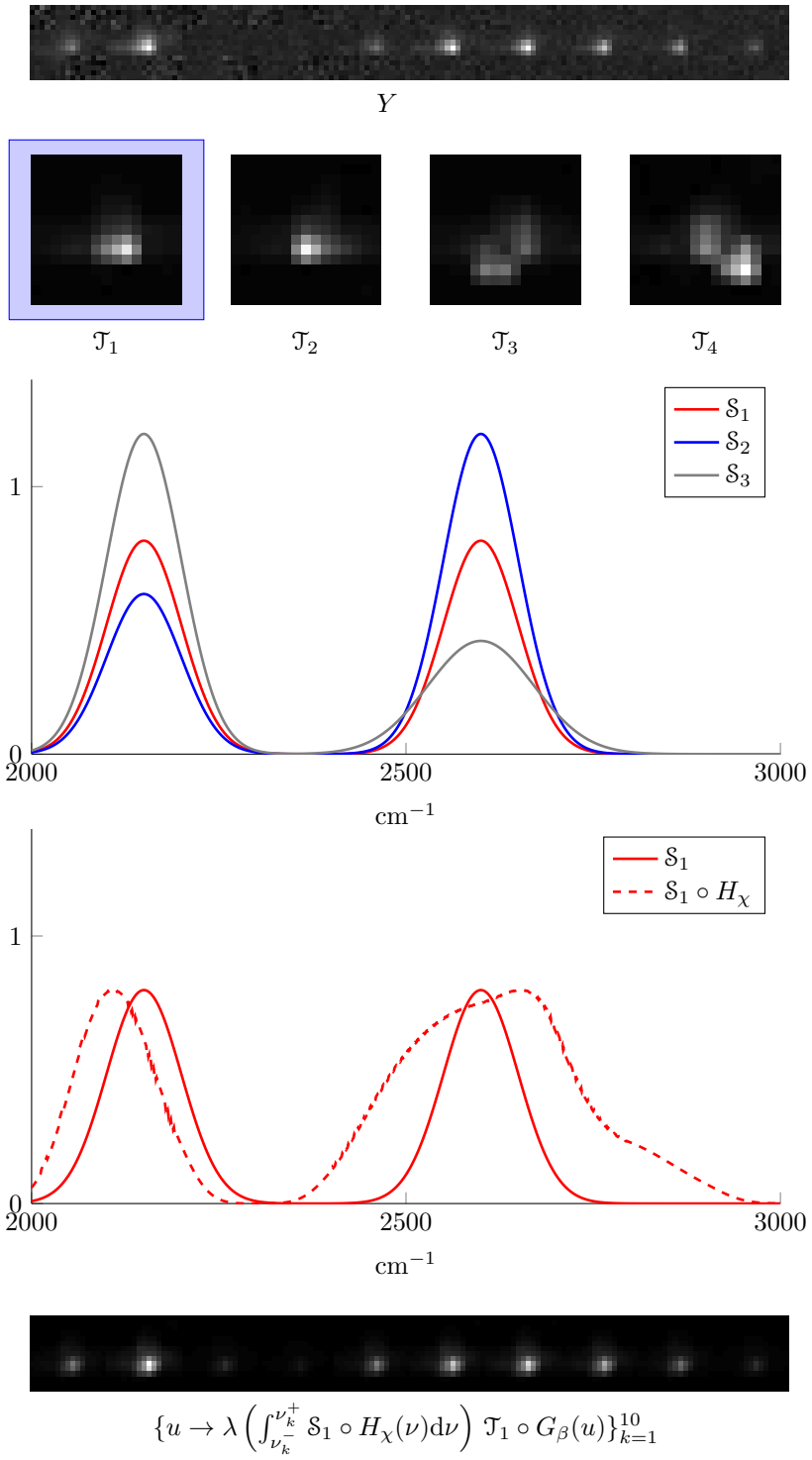


Figure III-21 – Illustration du modèle III.11

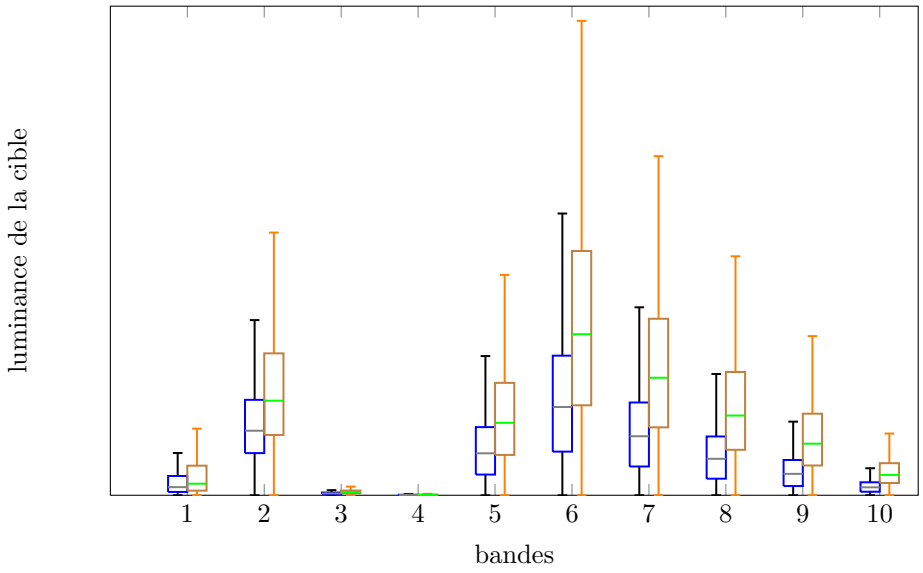


FIGURE III-22 – Statistique des pixels moyens de chaque bande spectrale pour les deux avions

4 Méthode de classification

L'estimation des paramètres des modèles de SIR monospectrales III.4 et multispectrales III.8 réalisée par le MCoEM dans les deux sections précédentes nous permettent de proposer une méthode de classification *a posteriori* . Nous désignons par $\hat{\theta} \in \Theta$ l'estimateur à convergence obtenu par le MCoEM . Pour tout $n \in \mathbb{N}$, soit $V_n \in \{1, 2\}$ la variable indiquant le type d'aéronef (Alphajet ou Avion 1) présent dans l'observation Y_n. Une première méthode de classification, induite par les modèles d'observation, consiste à calculer la classe $\hat{V}_n$ qui maximise la probabilité suivante

$$\hat{V}_n = \arg \max_{v \in \{1,2\}} \mathbb{P}_{\hat{\theta}} \left[V_n = v \,|\, Y_n \right] \ . \tag{III.12}$$

Dans ce contexte, V_n est une variable aléatoire liée à la variable de classe J_n du modèle d'observation d'une façon différente suivant que le cadre d'apprentissage est *semi-supervisé* ou *non-supervisé* :

- *semi-supervisé* : pour $v \in \{1, 2\}$, on désigne par $\hat{\theta}_v$ l'estimateur obtenu en appliquant le MCoEM sur une base d'apprentissage ne contenant que des aéronefs de type v. Dans ce cas, l'événement $\{V_n = v\}$ s'écrit

$$\{V_n = v\} = \{J_n \in \{1, \ldots, C\}\} \cap \{\theta = \hat{\theta}_v\} \ .$$

  Ainsi, pour tout $v \in \{1, 2\}$, nous avons

$$\mathbb{P}_{\hat{\theta}}[V_n = v \,|\, Y_n] = \sum_{j=1}^{C} \mathbb{P}_{\hat{\theta}_v}[J_n = j \,|\, Y_n] \propto \sum_{j=1}^{C} \hat{\omega}_{j,v} \iint p_{\hat{\theta}_v}(Y_n \,|\, \beta_n, \lambda_n, j) g_{\hat{\theta}_v}(\beta_n, \lambda_n \,|\, j) \mathrm{d}\lambda_n \mathrm{d}\beta_n \ . \tag{III.13}$$

- *non-supervisé* :
  – soit $C = 2$ auquel cas $\{V_n = v\} = \{J_n = v\}$ et $\mathbb{P}_{\hat{\theta}}[V_n = v \,|\, Y_n] = \mathbb{P}_{\hat{\theta}}[J_n = v \,|\, Y_n]$,

– soit $C > 2$ et il existe une subdivision de l'ensemble $\{1, \ldots, C\} = \mathsf{J}_1 \cup \mathsf{J}_2$, avec $\mathsf{J}_1 \cap \mathsf{J}_2 = \{\emptyset\}$, spécifiée par un agent extérieur après l'apprentissage, telle que

$$\{V_n = v\} = \{J_n \in \mathsf{J}_v\} .$$

Dans cette configuration, la classe *a posteriori* s'écrit

$$\mathbb{P}_{\hat{\theta}}[V_n = v \,|\, Y_n] \propto \sum_{j \in \mathsf{J}_v} \hat{\omega}_j \iint p_{\hat{\theta}}(Y_n \,|\, \beta_n, \lambda_n, j) g_{\hat{\theta}}(\beta_n, \lambda_n \,|\, j) \mathrm{d}\lambda_n \mathrm{d}\beta_n . \tag{III.14}$$

La double intégrale sur les paramètres β_n et λ_n intervenant dans le calcul des probabilités $\{\mathbb{P}_{\hat{\theta}}[V_n = v \,|\, Y_n]\}_{v=1}^2$ ne peut s'effectuer de façon exacte dans ces deux situations et nous avons recours à une approximation. Remarquant que

$$\iint p_{\hat{\theta}}(Y_n \,|\, \beta_n, \lambda_n, j) g_{\hat{\theta}}(\beta_n, \lambda_n \,|\, j) \mathrm{d}\beta_n \mathrm{d}\lambda_n = \mathbb{E}_{\hat{\theta}} \left[p_{\hat{\theta}}(Y_n \,|\, \beta_n, \lambda_n, J_n) \,|\, J_n = j \right] , \tag{III.15}$$

une façon d'approcher la double intégrale est de calculer un estimateur de Monte Carlo à partir d'échantillons *i.i.d.* des données manquantes simulées sous la loi *a priori* spécifiée par le modèle d'observation $(\beta_n, \lambda_n) \sim g_{\hat{\theta}}(\,\cdot \,|\, j)$. Toutefois les résultats de classification obtenus par cette méthode ne sont pas satisfaisants : pour une base de test contenant 1000 SIR monospectrales de type Avion 1 et autant de type Alphajet avec un bruit σ_1, un aéronef n'est reconnu en moyenne que dans 81% des cas. Ce faible résultat s'explique par le fait que la loi *a priori* est faiblement informative : nous avons considéré dans notre modèle les variables aléatoires (β_n, λ_n) comme des nuisances dont l'objectif était de recaler les prototypes sur les observations et non comme des paramètres caractéristiques de chaque classe. Par conséquent, les données manquantes simulées suivant $(\beta_n, \lambda_n) \sim g_{\hat{\theta}}(\,\cdot \,|\, j)$ ne permettent pas de recaler correctement les observations.

Une méthode de classification plus appropriée à un contexte où les observations doivent, dans un premier temps, être recalées consiste à calculer les poids *a posteriori* π_j définis pour tout $j \in \{1, \ldots, C\}$ par :

$$\pi_j(Y_n) \propto \mathbb{E}_{\hat{\theta}} \left[p_{\hat{\theta}}(Y_n \,|\, \beta_n, \lambda_n, J_n) \,|\, J_n = j, Y_n \right] . \tag{III.16}$$

La classe associée à un aéronef est calculée de la même façon que dans la méthode précédente à la différence que la double intégrale (III.15) est remplacée par $\pi_j(Y_n)$. La différence entre ces deux méthodes réside donc dans le conditionnement de l'espérance conditionnelle par l'observation à classer. Cette différence, mineure en apparence, permet de recaler les prototypes estimés sur Y_n. En effet, comme dans (III.15), l'espérance conditionnelle (III.16) n'est pas calculable exactement et une méthode MCMC de type Metropolis-within-Gibbs (voir Remarque IV.23) est utilisée pour simuler les données manquantes conditionnellement à l'observation Y_n et à la classe j. De la même façon que pour la simulation des données manquantes du MCoEM , cette étape s'apparente à un recalage.

Les résultats de classification obtenus par cette méthode sur une base de test composée de 1000 aéronefs de chaque type et pour les deux niveaux de bruit σ_1 et σ_2 sont présentés dans les Figures III-23, III-24,III-25, III-26 et III-27. Ils montrent notamment :

- l'importance du recalage dans la procédure de classification : le modèle de classification (III.16) est à privilégier par rapport au modèle (III.15), le taux de bonne classification passant de 77.1% à 95.5%, dans le cas σ_1 et en monospectrale (Figure III-23),
- l'importance de l'apprentissage : les performances de classification pour des SIR monospectrales sont moins bonnes sous le niveau de bruit σ_2 que sous σ_1 et pour une approche non-supervisée que pour une approche semi-supervisée, ces configurations correspondant à des scénarios d'apprentissages de plus en plus difficiles (Figure III-24).

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.82	0.278
Alphajet	0.18	0.722

(a) Classification sans recalage (III.15)

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.966	0.056
Alphajet	0.034	0.944

(b) Classification avec recalage (III.16)

FIGURE III-23 – Comparaison des deux modèles de classification dans le cas de SIR monospectrales sous le niveau de bruit σ_1 et en apprentissage semi-supervisé

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.966	0.056
Alphajet	0.034	0.944

(a) σ_1 et semi-supervisé

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.912	0.060
Alphajet	0.088	0.940

(b) σ_2 et semi-supervisé

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.938	0.07
Alphajet	0.062	0.93

(c) σ_1 et non-supervisé

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.718	0.092
Alphajet	0.282	0.908

(d) σ_2 et non-supervisé

FIGURE III-24 – Classification des deux aéronefs à partir de SIR monospectrales en bande II

- en monospectral, il y a un intérêt à travailler sur une bande étroite présentant un meilleur rapport signal à bruit *e.g.* $\eta_1 = (4, 4)$ plutôt qu'en bande large $\eta_1 = (0, 10)$ (Figure III-25), ce qui était prévisible au vu des prototypes estimés dans ces deux situations (Figure III-19),
- en multispectral, il y a des disparités importantes dans les taux de bonne classification suivant les regroupements envisagés (Figure III-26) et certaines combinaisons vont favoriser la reconnaissance de l'un ou l'autre des aéronefs (comparer par exemple III.26(c) et III.26(d)).
- en multispectral, il semble préférable de regrouper plusieurs bandes dans le domaine spectrale correspondant au second mode de luminance (comparer III.26(a) avec III.26(b) et III.26(c) avec III.26(d)).
- d'une façon générale, pour des regroupements de bandes bien choisis, l'utilisation de SIR multispectrales permet d'obtenir des meilleurs taux de bonne classification qu'en monospectral et en particulier dans les cas d'apprentissage défavorables (niveau de bruit élevé, apprentissage non-supervisé) qui correspondent à notre scénario de référence (Figure III-27).

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.97	0.01
Alphajet	0.03	0.99

(a) σ_1, semi-supervisé et bande étroite $\eta_1 = (4, 4)$

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.936	0.048
Alphajet	0.064	0.952

(b) σ_2, semi-supervisé et bande étroite $\eta_1 = (4, 4)$

Figure III-25 – Classification des deux aéronefs à partir de SIR monospectrales en bande étroite

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.936	0.068
Alphajet	0.064	0.932

(a) σ_1, semi-supervisé et $\eta_2 = (0, 2, 3, 1)$

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.974	0.026
Alphajet	0.064	0.986

(b) σ_1, semi-supervisé et $\eta_2 = (0, 2, 2, 4)$

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.751	0.047
Alphajet	0.249	0.953

(c) σ_2, semi-supervisé et $\eta_2 = (0, 2, 2, 2)$

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.97	0.057
Alphajet	0.03	0.943

(d) σ_2, semi-supervisé et $\eta_2 = (1, 1, 2, 5)$

Figure III-26 – Comparaison de la classification des deux aéronefs à partir de SIR multispectrales pour deux regroupements de bandes différents

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.718	0.092
Alphajet	0.282	0.908

(a) σ_2, non-supervisé, monospectral en bande large

Prédiction \ Réalité	Avion 1	Alphajet
Avion 1	0.89	0.07
Alphajet	0.11	0.93

(b) σ_2, non-supervisé, multispectral $\eta_2 = (1, 1, 2, 5)$

Figure III-27 – Comparaison de la classification non-supervisé sous σ_2 en monospectral et en multispectral

5 Conclusion

Nous avons appliqué avec succès l'algorithme MCoEM développé dans le Chapitre II, au contexte des signatures infrarouge d'aéronefs monospectrales et multispectrales. Les modèles d'observations proposés permettent de recaler prototypes et observations de façon satisfaisante comme le montrent les Figures III-12 et III-21. Un équilibre entre la description photométrique et géométrique des aéronefs s'installe naturellement lors de l'apprentissage : les géométries de l'Alphajet étant relativement similaires, les différents prototypes de cet avion se différencient principalement par la photométrie, signifiant que la pénalisation des déformations géométriques nécessaires au recalage des données est tolérée par le modèle. A l'inverse, les différents prototypes décrivant l'Avion 1 varient principalement par leur géométrie ce qui indique que la pénalisation du terme d'ajustement de la photométrie est plus faible que celui relatif à la déformation de la géométrie.

L'utilisation de SIR multispectrales offre deux principaux avantages :

- il est possible d'apprendre des variations dans le profil spectral qui ne sont pas visibles en monospectral et d'avoir donc une connaissance plus précise des aéronefs ; comparer les Figures III.5(b) et III.17(b),
- la nécessaire spécification des bandes spectrales prises en compte lors de l'apprentissage permet de réduire le rapport signal à bruit par rapport au cas de la bande large, ce qui facilite le recalage et donc l'apprentissage ; voir la Figure III-19.

Ces éléments justifient également que le taux de bonne classification *a posteriori* entre les deux aéronefs est meilleur lorsque la base d'apprentissage est constituée de SIR multispectrales. Toutefois, contrairement à la méthode de détection d'aéronefs proposée dans le Chapitre I, l'implémentation d'un algorithme génétique optimisant le paramètre $\eta_K \in \mathsf{H}_K$ en fonction du taux de bonne classification n'est pas envisageable ici dans la mesure où l'algorithme d'apprentissage et de classification sont une complexes et nécessitent un temps d'exécution conséquent. Dans ces simulations, nous avons travaillé avec l'heuristique qui consiste à utiliser les bandes optimales de la détection pour la classification. Il semble en effet logique de considérer que si un regroupement de bandes ne permet pas de détecter un aéronef, il ne permettra pas de les différencier. En revanche, rien ne dit que le regroupement optimal pour la détection l'est également pour la classification. Dans le cas où les cibles que nous cherchons à classifier ont des profils spectraux différents, le regroupement optimal est celui qui restituera au mieux les particularités de chaque type de cible, ce qui ne correspond pas nécessairement avec celui qui permettra de les détecter au mieux. La méthode actuelle permet de comparer les performances en terme de classification de différents regroupements de bandes, mais pas de trouver le regroupement optimal. Des méthodes basées sur le boosting [Sch03] combinant les classifieurs obtenus en utilisant plusieurs regroupements de bandes jugés *prometteurs* par notre méthode sont actuellement en cours d'étude.

L'utilisation du modèle proposé dans la Section 3 .2 et de l'algorithme MCoEM s'annonce prometteur dans le cas où les images multispectrales contiennent des objets ayant des profils de variations spectrales plus variés comme des avions de combat, des avions de lignes, des drones, d'autres dispositifs de vol (ULM, parachutes, parapentes) ou d'autres leurres. Le recalage géométrique et spectral simultané illustré par la Figure III-21 prendrait alors toute son importance. Enfin, le modèle de déformation de la géométrie (III.2) pourrait intégrer un facteur scalaire d'homothétie permettant ainsi de zoomer ou de dézoomer les prototypes, afin d'apprendre les caractéristiques et de classifier des observations se situant à différentes distances.

Chapitre IV

Comparison of Asymptotic Variances of Inhomogeneous Markov chains with Application to Markov chain Monte Carlo Methods

Abstract

In this paper we study the asymptotic variance of sample path averages for inhomogeneous Markov chains that evolve alternatingly according to two different π-reversible Markov transition kernels P and Q. More specifically, our main result allows us to compare directly the asymptotic variances of two inhomogeneous Markov chains associated with different kernels P_i and Q_i, $i \in \{0, 1\}$, as soon as the kernels of each pair (P_0, P_1) and (Q_0, Q_1) can be ordered in the sense of lag-one autocovariance. As an important application we use this result for comparing different data-augmentation-type Metropolis-Hastings algorithms. In particular, we compare some pseudo-marginal algorithms and propose a novel exact algorithm, referred to as the *random refreshment* algorithm, which is more efficient, in terms of asymptotic variance, than the Grouped Independence Metropolis Hastings algorithm and has a computational complexity that does not exceed that of the Monte Carlo Within Metropolis algorithm.

1 Introduction

Markov chain Monte Carlo (MCMC) *methods* allow samples from virtually any target distribution π, known up to a normalizing constant, to be generated. In particular, the celebrated *Metropolis-Hastings algorithm* (introduced in [MRR$^+$53] and [Has70]) simulates a Markov chain evolving according to a π-reversible Markov transition kernel by first generating, using some instrumental kernel, a candidate and then accepting or rejecting the same with a probability adjusted to satisfy the detailed balance condition [Tie95]. When choosing between several Metropolis-Hastings algorithms, it is desirable to be able to compare the efficiencies, in terms of the asymptotic variance of sample path averages, of different π-reversible Markov chains. Despite the practical importance of this question, only a few results in this direction exist the literature. P. H. Peskun [Pes73] defined a

partial ordering for finite state space Markov chains, where one transition kernel has a higher order than another if the former dominates the latter on the off-diagonal (see Definition IV.1). This ordering was extended later by L. Tierney [Tie95] to general state space Markov chains and another even more general ordering, the covariance ordering, was proposed in [MG99]. In general it holds that if a homogeneous π-reversible Markov transition kernel is greater than another according to one of these orderings, then the asymptotic variance of sample path averages for a Markov chain evolving according to the former is smaller for all square integrable (with respect to π) target functions.

We provide an extension of this result to inhomogeneous Markov chains that evolve alternatingly according to two different π-reversible Markov transition kernels. To the best of our knowledge, this is the first work dealing with systematic comparison of asymptotic variances of inhomogeneous Markov chains. The approach is linked with the operator theory for Markov chains but does not make use of any spectral representation. After some preliminaries (Section 2), our main result, Section IV.4, is stated in Section 5 . In Section 4 we apply Theorem IV.4 in the context of MCMC algorithms by comparing the efficiency, in terms of asymptotic variance, of some existing data-augmentation-type algorithms. Moreover, we propose a novel pseudo-marginal algorithm (in the sense of [AR09]), referred to as the *random refreshment* algorithm, which—on the contrary to the pseudo-marginal version of the *Monte Carlo Within Metropolis* (MCWM) algorithm—turns out to be exact and more efficient than the pseudo-marginal version of the *Grouped Independence Metropolis Hastings* (GIMH) algorithm. Here the analysis is again driven by Theorem IV.4. The proof of Theorem IV.4 is given in Section 5 and some technical lemmas are postponed to Section 7 . Finally, Section 8 relates some existing MCMC algorithms to the framework considered in this paper.

2 Preliminaries

We denote by $\mathbb{N} := \{0, 1, 2, \ldots\}$ and $\mathbb{N}^* := \{1, 2, \ldots\}$ the sets of non-negative and positive integers, respectively. In the following all random variables are assumed to be defined on a common probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Let $(\mathsf{X}, \mathcal{X})$ be a measurable space ; then we denote by $\mathcal{M}(\mathcal{X})$ and $\mathcal{F}(\mathcal{X})$ the spaces of positive measures and measurable functions on $(\mathsf{X}, \mathcal{X})$, respectively. The Lebesgue integral of $f \in \mathcal{F}(\mathcal{X})$ over X with respect to the measure $\mu \in \mathcal{M}(\mathcal{X})$ is, when well-defined, denoted by $\mu f := \int f(x)\, \mu(\mathrm{d}x)$. Recall that a *Markov transition kernel* P on $(\mathsf{X}, \mathcal{X})$ is a mapping $P : \mathsf{X} \times \mathcal{X} \to [0, 1]$ such that

– for all $\mathsf{A} \in \mathcal{X}$, $\mathsf{X} \ni x \mapsto P(x, \mathsf{A})$ is a measurable function,
– for all $x \in \mathsf{X}$, $\mathcal{X} \ni \mathsf{A} \mapsto P(x, \mathsf{A})$ is a probability measure.

A kernel P induces two integral operators, one acting on $\mathcal{M}(\mathcal{X})$ and the other on $\mathcal{F}(\mathcal{X})$; more specifically, for $\mu \in \mathcal{M}(\mathcal{X})$ and $f \in \mathcal{F}(\mathsf{X})$, we define the measure

$$\mu P : \mathcal{X} \ni \mathsf{A} \mapsto \int P(x, \mathsf{A}) \mu(\mathrm{d}x)$$

and the measurable function

$$Pf : \mathsf{X} \ni x \mapsto \int f(x') P(x, \mathrm{d}x').$$

Moreover, the *composition* (or *product*) of two kernels P and Q on $(\mathsf{X}, \mathcal{X})$ is the kernel defined by

$$PQ : \mathsf{X} \times \mathcal{X} \ni (x, \mathsf{A}) \mapsto \int Q(x', \mathsf{A}) P(x, \mathrm{d}x').$$

We will from now on fix a distinguished probability measure π on $(\mathsf{X}, \mathcal{X})$. Given π, we denote by $\mathsf{L}^2(\pi) := \{f \in \mathcal{F}(\mathcal{X}) : \pi f^2 < \infty\}$ the space of square integrable functions with respect to π and furnish the same with the scalar product

$$\langle f, g \rangle := \int f(x)g(x)\pi(\mathrm{d}x) \quad (f \in \mathsf{L}^2(\pi), g \in \mathsf{L}^2(\pi))$$

and the associated norm

$$\|f\|_{\mathsf{L}^2} := \left(\pi f^2\right)^{1/2} \quad (f \in \mathsf{L}^2(\pi)).$$

Here we have expunged the measure π from the notation for brevity. If P is a Markov kernel on $(\mathsf{X}, \mathcal{X})$ admitting π as an invariant distribution, then the mapping $f \mapsto Pf$ defines an operator on $\mathsf{L}^2(\pi)$, and by Jensen's inequality it holds that

$$\|P\| := \sup_{f \in \mathsf{L}^2(\pi) : \|f\|_{\mathsf{L}^2} \leq 1} \|Pf\|_{\mathsf{L}^2} \leq 1. \tag{IV.1}$$

Recall that a kernel P is π-*reversible* if and only if the detailed balance relation

$$\pi(\mathrm{d}x)P(x, \mathrm{d}x') = \pi(\mathrm{d}x')P(x', \mathrm{d}x)$$

holds. If the Markov kernel P is π-reversible, then $f \mapsto Pf$ defines a self-adjoint operator on $\mathsf{L}^2(\pi)$, i.e. for all f and g belonging to $\mathsf{L}^2(\pi)$,

$$\langle f, Pg \rangle = \langle Pf, g \rangle . \tag{IV.2}$$

The following off-diagonal ordering of Markov transition kernels on a common state space was, in the case of Markov chains in a finite state space, proposed in [Pes73]. The ordering was extended later in [Tie95] to the case of general state space Markov chains.

Definition IV.1. Let P_0 and P_1 be Markov transition kernels on $(\mathsf{X}, \mathcal{X})$ with invariant distribution π. We say that P_1 *dominates* P_0 *on the off-diagonal*, denoted $P_1 \succeq P_0$, if for all $\mathsf{A} \in \mathcal{X}$ and π-a.s. all $x \in \mathsf{X}$,

$$P_1(x, \mathsf{A} \setminus \{x\}) \geq P_0(x, \mathsf{A} \setminus \{x\}).$$

The previous ordering allows for comparing the asymptotic efficiency of different reversible kernels. More specifically, the following seminal result was established in [Pes73, Theorem 2.1.1] for Markov chains in discrete state space and extended later in [Tie95, Theorem 4] to Markov chains in general state space.

Theorem IV.2. Let P_0 and P_1 be two π-reversible kernels on $(\mathsf{X}, \mathcal{X})$. If $P_1 \succeq P_0$, then for all $f \in \mathsf{L}^2(\pi)$,

$$v(f, P_1) \leq v(f, P_0),$$

where we have defined, for a Markov chain $\{X_k; k \in \mathbb{N}\}$ with π-reversible transition kernel P and initial distribution π,

$$v(f, P) := \lim_{n \to \infty} \frac{1}{n} \mathrm{Var} \left(\sum_{k=0}^{n-1} f(X_k) \right). \tag{IV.3}$$

Note that according to [Tie95], if $\{X_k; k \in \mathbb{N}\}$ is a π-reversible Markov chain and $f \in \mathsf{L}^2(\pi)$, then $\lim_{n\to\infty} n^{-1}\mathrm{Var}(\sum_{k=0}^{n-1} f(X_k))$ is guaranteed to exist (but may be infinite). Nevertheless, the ordering in question does not allow Markov kernels lacking probability mass on the diagonal, i.e. kernels P satisfying $P(x,\{x\}) = 0$ for all $x \in \mathsf{X}$, to be compared. This is in particular the case for Gibbs samplers in general state space. To overcome this limitation, one may consider instead the following covariance ordering based on lag-one autocovariances.

Definition IV.3. Let P_0 and P_1 be Markov transition kernels on $(\mathsf{X}, \mathcal{X})$ with invariant distribution π. We say that P_1 *dominates* P_0 *in the covariance ordering*, denoted $P_1 \succcurlyeq P_0$, if for all $f \in \mathsf{L}^2(\pi)$,

$$\langle f, P_1 f\rangle \leq \langle f, P_0 f\rangle\,.$$

The covariance ordering, which was introduced implicitly in [Tie95, p. 5] and formalized in [MG99], is an extension of the off-diagonal ordering since according to [Tie95, Lemma 3], $P_1 \succeq P_0$ implies $P_1 \succcurlyeq P_0$. Moreover, it turns out that for reversible kernels, $P_1 \succcurlyeq P_0$ implies $v(f, P_0) \geq v(f, P_1)$ (see the proof of [Tie95, Theorem 4]).

All these results concern homogeneous Markov chains, whereas many MCMC algorithms such as the Gibbs or the Metropolis-within-Gibbs samplers use several kernels, e.g. P and Q in the case of two kernels [RC04]. A natural idea would then be to apply Theorem IV.2 to the homogeneous Markov chain having the block kernel PQ as transition kernel; however, even when the kernels P and Q are both π-reversible, the product PQ of the same is usually not π-reversible, except in the particular case when P and Q commute, i.e. $PQ = QP$. Thus, Theorem IV.2 cannot in general be applied directly in this case.

3 Main assumptions and results

In the following, let P_i and Q_i, $i \in \{0,1\}$, be Markov transition kernels on $(\mathsf{X}, \mathcal{X})$. Define $\{X_k^{(0)}; k \in \mathbb{N}\}$ and $\{X_k^{(1)}; k \in \mathbb{N}\}$ as the Markov chains evolving as follows :

$$X_0^{(i)} \xrightarrow{P_i} X_1^{(i)} \xrightarrow{Q_i} X_2^{(i)} \xrightarrow{P_i} X_3^{(i)} \xrightarrow{Q_i} \cdots \tag{IV.4}$$

This means that for all $k \in \mathbb{N}$, $i \in \{0,1\}$ and $\mathsf{A} \in \mathcal{X}$,

- $\mathbb{P}\left(X_{2k+1}^{(i)} \in \mathsf{A} \mid \mathcal{F}_{2k}^{(i)}\right) = P_i(X_{2k}^{(i)}, \mathsf{A})$,
- $\mathbb{P}\left(X_{2k+2}^{(i)} \in \mathsf{A} \mid \mathcal{F}_{2k+1}^{(i)}\right) = Q_i(X_{2k+1}^{(i)}, \mathsf{A})$,

where $\mathcal{F}_n^{(i)} := \sigma(X_0^{(i)}, \ldots, X_n^{(i)})$, $n \in \mathbb{N}$.

We impose the following assumption :

(A1) (*i*) P_i and Q_i, $i \in \{0,1\}$, are π-reversible,
(*ii*) $P_1 \succcurlyeq P_0$ and $Q_1 \succcurlyeq Q_0$.

As mentioned above, $P_1 \succeq P_0$ implies $P_1 \succcurlyeq P_0$; thus, in practice, a sufficient condition for **(A1)**(*ii*) is that $P_1 \succeq P_0$ and $Q_1 \succeq Q_0$.

Theorem IV.4. Assume that P_i and Q_i, $i \in \{0,1\}$, satisfy (**A1**) and let $\{X_k^{(i)}; k \in \mathbb{N}\}$, $i \in \{0,1\}$, be Markov chains evolving as in (IV.4) with initial distribution π. Then for all $f \in \mathsf{L}^2(\pi)$ such that for $i \in \{0,1\}$,

$$\sum_{k=1}^{\infty} \Big(|\mathrm{Cov}(f(X_0^{(i)}), f(X_k^{(i)}))| + |\mathrm{Cov}(f(X_1^{(i)}), f(X_{k+1}^{(i)}))| \Big) < \infty, \tag{IV.5}$$

it holds that

$$v_1(f) \le v_0(f), \tag{IV.6}$$

where

$$v_i(f) := \lim_{n\to\infty} \frac{1}{n} \mathrm{Var} \left(\sum_{k=0}^{n-1} f(X_k^{(i)}) \right) \quad (i \in \{0,1\}). \tag{IV.7}$$

Remark IV.5. At present, we have not been able to extend the arguments of our current proof of Theorem IV.4 (see Section 5) to inhomogeneous Markov chains evolving alternatingly according to *more* than two different kernels. On the other hand, we have not been able to find a counterexample rejecting the hypothesis that a similar result would hold true also in that case. We leave this as an open problem.

Remark IV.6. The condition (IV.5) is *not* a necessary condition for (IV.6) ; indeed, letting $\mathsf{X} = \{-1, 1\}$, $\pi(\mathrm{d}x') = P_0(x, \mathrm{d}x') = (\delta_1(\mathrm{d}x') + \delta_{-1}(\mathrm{d}x'))/2$, $Q_1 = Q_0 = P_1$, where, as in [HR07, Example 5], $P_1(x, \mathrm{d}x') = \delta_{-x}(\mathrm{d}x')$, provides a straightforward counterexample.

When verifying if a given f satisfies the condition (IV.5) it may be convenient to consider the homogeneous Markov chains $\{X_{2k}; k \in \mathbb{N}\}$ or $\{X_{2k+1}; k \in \mathbb{N}\}$ or even $\{(X_{2k}, X_{2k+1}); k \in \mathbb{N}\}$. Typically, none of these chains are π-reversible. Nevertheless, π-reversibility is not needed for checking conditions of type (IV.5), which can be established using upper bounds on the *V-norm* between the distribution given by the n^{th} iterate of a homogeneous kernel and its stationary distribution. This will be developed in the following section.

3 .1 Sufficient conditions for the absolute summability assumption (IV.5)

For any measurable real-valued function f on $(\mathsf{X}, \mathcal{X})$, define the *$V$-norm* of the *function* f by

$$|f|_V := \sup_{x \in \mathsf{X}} \frac{|f(x)|}{V(x)}.$$

Moreover, let ξ be a finite signed measure on $(\mathsf{X}, \mathcal{X})$. Then by the Jordan decomposition theorem there exists a unique pair of positive, finite, and singular measures ξ_+ and ξ_- on $(\mathsf{X}, \mathcal{X})$ such that $\xi = \xi_+ - \xi_-$. The pair $\xi_\pm$ is referred to as the *Jordan decomposition* of the signed measure ξ. The finite measure $|\xi| := \xi_+ + \xi_-$ is called the *total variation* of ξ. Let V be a nonnegative function taking values in $[1, \infty)$; then the *V-norm* of the *signed measure* ξ is defined by

$$\|\xi\|_V := |\xi|(V) = \sup_{f : |f|_V \le 1} \xi f.$$

Definition IV.7. A Markov kernel P on $(\mathsf{X}, \mathcal{X})$ is *V-geometrically ergodic* if it admits a unique invariant distribution π and there exists a measurable function $V : \mathsf{X} \to [1, \infty)$ satisfying $\pi V < \infty$ and such that the following holds.

(a) There exist constants $(C, \rho) \in \mathbb{R}^+ \times (0, 1)$ such that for all $x \in \mathsf{X}$ and all $n \in \mathbb{N}$,

$$\|P^n(x, \cdot) - \pi\|_V \le C \rho^n V(x). \tag{IV.8}$$

(b) There exist constants $(b, \lambda) \in \mathbb{R}^+ \times (0,1)$ such that $PV \leq \lambda V + b$.

Remark IV.8. [HM11, Theorem 1.2] provides sufficient conditions, in terms of drift towards a *small set*, for (a) in Definition IV.7 to hold ; see also [RR04, Fact 10] for necessary and sufficient conditions under the assumption of aperiodicity and irreducibility. Moreover, the coming developments require only the bound (IV.8) to hold π-a.s.

We have now all necessary tools for giving sufficient conditions that imply the absolute summability assumption (IV.5). Let the chain $\{X_k; k \in \mathbb{N}\}$ evolve according to

$$X_0 \xrightarrow{P} X_1 \xrightarrow{Q} X_2 \xrightarrow{P} X_3 \xrightarrow{Q} \cdots \tag{IV.9}$$

with $X_0 \sim \pi$, for some Markov kernels P and Q.

Proposition IV.9. If the Markov kernel PQ is V-geometrically ergodic, then for all functions f such that $|f|_{V^{1/2}} < \infty$ and $|Pf|_{V^{1/2}} < \infty$,

$$\sum_{k=1}^{\infty} \left(|\mathrm{Cov}(f(X_0), f(X_k))| + |\mathrm{Cov}(f(X_1), f(X_{k+1}))| \right) < \infty,$$

where $\{X_k; k \in \mathbb{N}\}$ evolves as in (IV.9).

The proof of Proposition IV.9 is found in Section 7 .1.

4 Applications

Before considering some applications of Theorem IV.4 we recall the following proposition, describing how to obtain a π-reversible Markov chain using some instrumental kernel K. Although this result is fundamental in the Metropolis-Hastings literature (see for example [RR04], [RC04], [GRS96], and the references therein), it is restated here as it will be used in various situations in the sequel (especially when there is no fixed reference measure dominating all the distributions $\{K(x, \cdot);\, x \in \mathsf{X}\}$).

Proposition IV.10. Let K be a Markov transition kernel on $\mathsf{X} \times \mathcal{X}$ and π a probability measure on $(\mathsf{X}, \mathcal{X})$. Define the probability measures $\mu(\mathrm{d}x \times \mathrm{d}x') := \pi(\mathrm{d}x)K(x, \mathrm{d}x')$ and $\nu(\mathrm{d}x \times \mathrm{d}x') := \pi(\mathrm{d}x')K(x', \mathrm{d}x)$. Assume that the measures ν and μ are equivalent and such that for μ-a.s. all $(x, x') \in \mathsf{X}^2$,

$$0 < \frac{\mathrm{d}\nu}{\mathrm{d}\mu}(x, x') < \infty, \tag{IV.10}$$

where $\frac{\mathrm{d}\nu}{\mathrm{d}\mu}$ denotes the Radon-Nikodym derivative. Then the Markov kernel $P(x, \mathrm{d}x') := K(x, \mathrm{d}x')\alpha(x, x') + \delta_x(\mathrm{d}x')\beta(x)$, where

$$\alpha(x, x') := 1 \wedge \frac{\mathrm{d}\nu}{\mathrm{d}\mu}(x, x') \quad \textit{and} \quad \beta(x) := 1 - \int K(x, \mathrm{d}x')\alpha(x, x'),$$

is π-reversible.

A natural application of Theorem IV.4 consists in using the result for comparing different data-augmentation-type algorithms. In the following we wish to target a probability distribution π^* defined on $(\mathsf{Y}, \mathcal{Y})$ using a sequence $\{Y_k; k \in \mathbb{N}\}$ of Y-valued random variables. To this aim, M. A. Tanner and W. H. Wong [TW87] suggest writing π^* as the

marginal of some distribution π defined on the product space $(\mathsf{Y} \times \mathsf{U}, \mathcal{Y} \otimes \mathcal{U})$ in the sense that $\pi(\mathrm{d}y \times \mathrm{d}u) = \pi^*(\mathrm{d}y)\, R(y, \mathrm{d}u)$, where R is some Markov transition kernel on $\mathsf{Y} \times \mathcal{U}$. In most cases the marginal π^* is of sole interest, while the component u is introduced for convenience as a means of coping with analytic intractability of the marginal. (It could also be the case that the marginal π^* is too computationally expensive to evaluate.) A first solution consists in letting $\{Y_k; k \in \mathbb{N}\}$ be the first-component process $\{Y_k^{(1)}; k \in \mathbb{N}\}$ of the π-reversible Markov chain $\{(Y_k^{(1)}, U_k^{(1)}); k \in \mathbb{N}\}$ defined as follows. Let S and T be instrumental Markov transition kernels on $\mathsf{Y} \times \mathsf{U} \times \mathcal{Y}$ and $\mathsf{Y} \times \mathsf{U} \times \mathsf{Y} \times \mathcal{U}$, respectively, and define a transition of the chain $\{(Y_k^{(1)}, U_k^{(1)}); k \in \mathbb{N}\}$ by Algorithm 8.

Algorithm 8 The *freeze* algorithm

Given $(Y_k^{(1)}, U_k^{(1)}) = (y, u)$,

(i) draw $\hat{Y} \sim S(y, u; \cdot)$ and call the outcome $\hat{y}$ (abbr. $\rightsquigarrow \hat{y}$),

(ii) draw $\hat{U} \sim T(y, u, \hat{y}; \cdot) \rightsquigarrow \hat{u}$,

(iii) set

$$(Y_{k+1}^{(1)}, U_{k+1}^{(1)}) \leftarrow \begin{cases} (\hat{y}, \hat{u}) & \text{with probability } \alpha(y, u, \hat{y}, \hat{u}) \\ & \quad := 1 \wedge \dfrac{\pi^*(\hat{y}) r(\hat{y}, \hat{u}) s(\hat{y}, \hat{u}; y) t(\hat{y}, \hat{u}, y; u)}{\pi^*(y) r(y, u) s(y, u; \hat{y}) t(y, u, \hat{y}; \hat{u})}, \\ (y, u) & \text{otherwise.} \end{cases} \tag{IV.11}$$

Remark IV.11. In the expression (IV.11) of α we assume implicitly that the families $\{S(y, u; \cdot); (y, u) \in \mathsf{Y} \times \mathsf{U}\}$ and $\{T(y, u, \hat{y}; \cdot); (y, u, \hat{y}) \in \mathsf{Y} \times \mathsf{U} \times \mathsf{Y}\}$ of probability measures are dominated by a fixed nonnegative measure and we denote by s and t the corresponding transition kernel densities, respectively. In some cases (see e.g. [NFW12]) it may however happen (typically when some Dirac mass is involved) that these kernels are not dominated by a nonnegative measure ; nevertheless, Algorithm 8 as well as Algorithm 9 defined below remain valid provided that the ratio in α is replaced by the corresponding Radon-Nikodym derivative $\frac{\mathrm{d}\nu}{\mathrm{d}\mu}(y, u, \hat{y}, \hat{u})$, where, in this case,

$$\begin{aligned} \mu(\mathrm{d}y \times \mathrm{d}u \times \mathrm{d}\hat{y} \times \mathrm{d}\hat{u}) &:= \pi(\mathrm{d}y) R(y, \mathrm{d}u) S(y, u; \mathrm{d}\hat{y}) T(y, u, \hat{y}; \mathrm{d}\hat{u}), \\ \nu(\mathrm{d}y \times \mathrm{d}u \times \mathrm{d}\hat{y} \times \mathrm{d}\hat{u}) &:= \pi(\mathrm{d}\hat{y}) R(\hat{y}, \mathrm{d}\hat{u}) S(\hat{y}, \hat{u}; \mathrm{d}y) T(\hat{y}, \hat{u}, y; \mathrm{d}u). \end{aligned}$$

By applying Proposition IV.10 we deduce that the output $\{(Y_k^{(1)}, U_k^{(1)}); k \in \mathbb{N}\}$ is a π-reversible Markov chain. As a consequence, the sequence $\{Y_k^{(1)}; k \in \mathbb{N}\}$ targets, although it is not itself a Markov chain, the marginal distribution π^*. Note that the method requires the product $\pi^*(y) r(y, u) s(y, u; \hat{y}) t(y, u, \hat{y}; \hat{u})$ to be known at least up to a multiplicative constant to guarantee the computability of the acceptance probability α in (IV.11).

Example 12 (Grouped-independence Metropolis-Hastings)**.** The Grouped-independence Metropolis-Hastings (GIMH) algorithm (see [Bea03, AR09]) is used in situations where π^* is analytically intractable. In this algorithm, the quantity $\pi^*(y)$ is in the acceptance probability replaced by an importance sampling estimate

$$\pi_N^*(y) := \frac{1}{N} \sum_{\ell=1}^{N} \frac{\bar{\pi}(y, v_\ell)}{q_y(v_\ell)}, \tag{IV.12}$$

where $\bar{\pi}(y, v)$ is the density of some augmented target distribution $\bar{\pi}(\mathrm{d}y \times \mathrm{d}v)$ defined on the product space $(\mathsf{Y} \times \mathsf{V}, \mathcal{Y} \otimes \mathcal{V})$, known up to a normalizing constant and allowing π^* as marginal distribution, and $\{v_1, \ldots, v_N\}$ are i.i.d. draws from the proposal q_y. Denoting by $s(y, \cdot)$ the density used for proposing new candidates $\hat{y}$, one obtains the acceptance probability ratio

$$\frac{\pi_N^*(\hat{y})s(\hat{y}, y)}{\pi_N^*(y)s(y, \hat{y})} = \frac{\pi^*(\hat{y})r(\hat{y}, \hat{u})s(\hat{y}, y)t(y, u)}{\pi^*(y)r(y, u)s(y, \hat{y})t(\hat{y}, \hat{u})},$$

where $u := (v_1, \ldots, v_N)$ and

$$\pi^*(y)r(y, u) = \frac{1}{N}\sum_{\ell=1}^{N}\left(\bar{\pi}(y, v_\ell)\prod_{m\neq\ell} q_y(v_m)\right),$$

$$t(y, u) = \prod_{\ell=1}^{N} q_y(v_\ell).$$

Consequently, the GIMH algorithm can be perfectly cast into the framework of the freeze algorithm, with the auxiliary variable U playing the role of the N-dimensional Monte Carlo sample and $\mathsf{U} = \mathsf{V}^n$.

In the following we use Theorem IV.4 for comparing the performance of Algorithm 8 to that of different modifications of the same obtained in the cases where

(I) simulating R-transitions is feasible,

(II) simulating R-transitions is infeasible.

Case I : simulating R-transitions is feasible

In this case, an alternative to Algorithm 8 consists in letting $\{Y_k; k \in \mathbb{N}\}$ be the sequence $\{Y_k^{(2)}; k \in \mathbb{N}\}$ generated through Algorithm 9. Note that Algorithm 9 "refreshes",

Algorithm 9 The *systematic refreshment* algorithm

Given $Y_k^{(2)} = y$,

(i) draw $U \sim R(y, \cdot) \rightsquigarrow u$,

(ii) draw $\hat{Y} \sim S(y, u; \cdot) \rightsquigarrow \hat{y}$,

(iii) draw $\hat{U} \sim T(y, u, \hat{y}; \cdot) \rightsquigarrow \hat{u}$,

(iv) set $Y_{k+1}^{(2)} \leftarrow \begin{cases} \hat{y} & \text{with probability } \alpha(y, u, \hat{y}, \hat{u}) \quad \text{(defined in (IV.11))}, \\ y & \text{otherwise.} \end{cases}$

in Step (i), systematically the second component of the Markov chain, which advocates Algorithm 9 to have better mixing properties than Algorithm 8. The main task of the present section is to establish rigorously this heuristics. The output $\{Y_k^{(2)}; k \in \mathbb{N}\}$ of Algorithm 9 is, on the contrary to $\{Y_k^{(1)}; k \in \mathbb{N}\}$, a Markov chain. It is not a classical Metropolis-Hastings Markov chain due to the auxiliary variables U and $\hat{U}$ that appear explicitly in the acceptance probability. However, as established in the following proposition, whose proof is found in Section 7 .2, the π-reversibility of $\{(Y_k^{(1)}, U_k^{(1)}); k \in \mathbb{N}\}$ implies π^*-reversibility of $\{Y_k^{(2)}; k \in \mathbb{N}\}$.

Proposition IV.13. The sequence $\{Y_k^{(2)}; k \in \mathbb{N}\}$ generated in Algorithm 9 is a π^*-reversible Markov chain.

Example 14 (Randomized MCMC [NFW12])**.** In [NFW12], the authors use the terminology *Randomized MCMC* (r-MCMC) for a π^*-reversible Metropolis-Hastings chain $\{Y_k; k \in \mathbb{N}\}$ generated using a set of auxiliary variables $\{U_k; k \in \mathbb{N}\}$ with a particular expression of the acceptance probability. Although only one of these auxiliary variables is sampled at each time step, one may actually cast this approach into the framework of Algorithm 9 by creating artificially another auxiliary variable according to the deterministic kernel

$$T(y, u, \hat{y}; \mathrm{d}\hat{u}) = \delta_{f(u)}(\mathrm{d}\hat{u}),$$

where f is any continuously differentiable involution on U. Even though T is not dominated, it is possible to verify (IV.10) using that f is an involution. We prove in Section 8 .1 that the r-MCMC algorithm is a special case of Algorithm 9 with this particular choice of T and with the general form of the acceptance probability described in Remark IV.11.

Example 15 (Generalized Multiple-try Metropolis [PBF10])**.** The *Generalized Multiple-try Metropolis* (GMTM) *algorithm* [PBF10] is an extension of the *Multiple-try Metropolis Hastings algorithm* proposed in [LLW00]. Given $Y_k = y$, one draws n *i.i.d.* possible moves $V_1, \ldots, V_n$ according to $\check{R}(y, \cdot)$. After this, a random index J taking the value $j \in \{1, \ldots, n\}$ with probability proportional to $\omega(y, V_j)$ is generated, whereupon a candidate is constructed as $\hat{Y} = V_J$. The candidate is then accepted with some probability that is computed using n additional random variables $\hat{V}_1, \ldots, \hat{V}_n$, where $\hat{V}_1, \ldots, \hat{V}_{n-1}$ are i.i.d. draws from $\check{R}(\hat{y}, \cdot)$, and $\hat{V}_n$ is set deterministically to $\hat{V}_n = y$ (see Section 8 .2 for more details concerning the acceptance probability). In Section 8 .2, Lemma IV.33, it is shown that the GMTM algorithm is in fact a special case of Algorithm 9 with $U = (V_1, \ldots, V_{J-1}, V_{J+1}, \ldots, V_n)$ and $\hat{U} = (\hat{V}_1, \ldots, \hat{V}_{n-1})$.

When the function $k : (y, \hat{y}) \mapsto \int R(y, \mathrm{d}u) s(y, u; \hat{y})$ is known explicitly, one may obtain another π^*-reversible Markov chain by means of the classical Metropolis-Hastings ratio, i.e. we use again Algorithm 9 but replace the acceptance probability $\alpha(y, u, \hat{y}, \hat{u})$ by

$$\hat{\alpha}(y, \hat{y}) := 1 \wedge \frac{\pi^*(\hat{y}) k(\hat{y}, y)}{\pi^*(y) k(y, \hat{y})}. \tag{IV.13}$$

The following proposition, which generalizes a similar result obtained in [NFW12, Section 2.3] for the r-MCMC algorithm, shows, when combined with [Tie95, Theorem 4], that the asymptotic variance of the classical Metropolis-Hastings estimator is smaller than that of the estimator based on Algorithm 9.

Proposition IV.16. The Metropolis-Hastings kernel associated with the acceptance probability (IV.13) is larger, in the sense of Definition IV.1, than the transition kernel associated with Algorithm 9.

Proof. Set

$$\mu(\mathrm{d}u \times \mathrm{d}\hat{u}) := \frac{R(y, \mathrm{d}u) s(y, u; \hat{y}) T(y, u, \hat{y}; \mathrm{d}\hat{u})}{k(y, \hat{y})}$$

and note that μ is a probability measure. Hence, as the mapping $\mathbb{R} \ni v \mapsto 1 \wedge v$ is concave,

Jensen's inequality implies that

$$\begin{aligned}
&\frac{\iint R(y,\mathrm{d}u)s(y,u;\hat{y})T(y,u,\hat{y};\mathrm{d}\hat{u})\alpha(y,u,\hat{y},\hat{u})}{k(y,\hat{y})\hat{\alpha}(y,\hat{y})} \\
&\quad= \frac{\iint \mu(\mathrm{d}u\times\mathrm{d}\hat{u})\alpha(y,u,\hat{y},\hat{u})}{\hat{\alpha}(y,\hat{y})} \\
&\quad\leq \left(1\wedge\iint\mu(\mathrm{d}u\times\mathrm{d}\hat{u})\frac{\pi(\hat{y},\hat{u})s(\hat{y},\hat{u};y)t(\hat{y},\hat{u},y;u)}{\pi(y,u)s(y,u;\hat{y})t(y,u,\hat{y};\hat{u})}\right)\Big/\hat{\alpha}(y,\hat{y}) \\
&\quad= 1\wedge\frac{\pi^*(\hat{y})k(\hat{y},y)}{\pi^*(y)k(y,\hat{y})}\Big/\hat{\alpha}(y,\hat{y}) = 1
\end{aligned}$$

(a similar technique was used in the proof of [AV12, Lemma 1]). The previous computation shows that the off-diagonal transition density function of the Metropolis-Hastings Markov chain associated with the acceptance probability (IV.13) is larger than that of the chain in Algorithm 9. This completes the proof. □

However, in practice a closed-form expression of k is rarely available, which prevents the classical Metropolis-Hastings algorithm from being implemented. Thus, if the transition density r is known explicitly and can be sampled we have to choose between Algorithm 8 and Algorithm 9 for approximating π^*. The classical tools (such as the ordering in Definition IV.1) for comparing $\{Y_k^{(1)}; k\in\mathbb{N}\}$ and $\{Y_k^{(2)}; k\in\mathbb{N}\}$ cannot be applied here, since $\{Y_k^{(1)}; k\in\mathbb{N}\}$ is not even a Markov chain. Nevertheless, Theorem IV.4 allows the theoretical comparison between these two algorithms by embedding $\{Y_k^{(1)}; k\in\mathbb{N}\}$ and $\{Y_k^{(2)}; k\in\mathbb{N}\}$ into inhomogeneous π-reversible Markov chains. The construction, which will be carried through in full detail below, leads to the following result.

Theorem IV.17. Let $\{Y_k^{(1)}; k\in\mathbb{N}\}$ and $\{Y_k^{(2)}; k\in\mathbb{N}\}$ be sequences of random variables generated by Algorithm 8 and Algorithm 9, respectively, where $(Y_0^{(1)}, U_0^{(1)}) \sim \pi$ and $Y_0^{(2)}\sim\pi^*$. Then for all $h\in\mathsf{L}^2(\pi^*)$ satisfying

$$\sum_{k=1}^{\infty}|\mathrm{Cov}(h(Y_0^{(i)}),h(Y_k^{(i)}))| < \infty \quad (i\in\{1,2\}) \tag{IV.14}$$

it holds that

$$\lim_{n\to\infty}\frac{1}{n}\mathrm{Var}\left(\sum_{k=0}^{n-1}h(Y_k^{(2)})\right) \leq \lim_{n\to\infty}\frac{1}{n}\mathrm{Var}\left(\sum_{k=0}^{n-1}h(Y_k^{(1)})\right).$$

We preface the proof of Theorem IV.17 by the following lemma, which may serve as a basis for the comparison of *homogeneous* Markov chains evolving according to P_iQ_i (or Q_iP_i), $i\in\{0,1\}$, where P_i and Q_i, $i\in\{0,1\}$, are kernels satisfying (**A1**) on some product space.

Lemma IV.18. Let P_i and Q_i, $i\in\{0,1\}$, be kernels satisfying (**A1**) on $(\mathsf{X},\mathcal{X})$, with $\mathsf{X}=\mathsf{Y}\times\mathsf{U}$ and $\mathcal{X}=\mathcal{Y}\otimes\mathcal{U}$. In addition, assume that for all $(y,u)\in\mathsf{X}$,

$$P_i(y,u;\{y\}\times\mathsf{U}) = 1 \quad (i\in\{0,1\}). \tag{IV.15}$$

Then for all $f\in\mathsf{L}^2(\pi)$ depending on only the first argument (i.e. $f(y,u)=h(y)$ for some h) and such that

$$\sum_{n=1}^{\infty}|\langle f,(P_iQ_i)^n f\rangle| < \infty \quad (i\in\{0,1\}) \tag{IV.16}$$

it holds that

$$v(f, P_1Q_1) = v(f, Q_1P_1) \leq v(f, P_0Q_0) = v(f, Q_0P_0).$$

Remark IV.19. The assumption (IV.15) is essential in Lemma IV.18. Indeed, let $\mathsf{X} = \{-1, 1\}$ and $\pi(\{1\}) = \pi(\{-1\}) = 1/2$, and define the kernels $P_0(x, \mathrm{d}x) = \delta_x(\mathrm{d}x')$, $Q_0(x, \mathrm{d}x') = \varepsilon\pi(\mathrm{d}x') + (1-\varepsilon)\delta_{-x}(\mathrm{d}x')$ for some $\varepsilon \in (0,1)$, $P_1(x, \mathrm{d}x') = \pi(\mathrm{d}x')$, and $Q_1 = Q_0$. Then the kernels P_i and Q_i, $i \in \{0,1\}$, satisfy (**A1**), and consequently Theorem IV.4 applies to the inhomogeneous chains evolving alternatingly according to the same. However, the similar result does not hold true for chains evolving according to the product kernels P_iQ_i and Q_iP_i, $i \in \{0,1\}$, as

$$v(f, P_0Q_0) = v(f, Q_0P_0) = \frac{\varepsilon}{2-\varepsilon} < 1 = v(f, P_1Q_1) = v(f, Q_1P_1),$$

with f being the identity mapping on X.

Proof of Lemma IV.18. Define Markov chains $\{X_k^{(i)}; k \in \mathbb{N}\}$, $i \in \{0,1\}$, evolving as

$$\cdots \xrightarrow{Q_i} X_{2k}^{(i)} = \begin{pmatrix} Y_k^{(i)} \\ U_k^{(i)} \end{pmatrix} \xrightarrow{P_i} X_{2k+1}^{(i)} = \begin{pmatrix} \check{Y}_k^{(i)} \\ \check{U}_k^{(i)} \end{pmatrix} \xrightarrow{Q_i} X_{2k+2}^{(i)} = \begin{pmatrix} Y_{k+1}^{(i)} \\ U_{k+1}^{(i)} \end{pmatrix} \xrightarrow{P_i} \cdots$$

with $X_0^{(i)} \sim \pi$. By construction,

$$\sum_{k=1}^{\infty} \left(|\mathrm{Cov}(f(X_0^{(i)}), f(X_k^{(i)}))| + |\mathrm{Cov}(f(X_1^{(i)}), f(X_{k+1}^{(i)}))| \right)$$
$$= \pi f^2 - \pi^2 f + 4 \sum_{k=1}^{\infty} |\mathrm{Cov}(h(Y_0^{(i)}), h(Y_k^{(i)}))| < \infty \quad (i \in \{0,1\}), \quad \text{(IV.17)}$$

where finiteness follows from the assumption (IV.16). Moreover, for all $n \in \mathbb{N}^*$ and $i \in \{0,1\}$,

$$\mathrm{Var}\left(\sum_{k=0}^{n-1} h(Y_k^{(i)}) \right) = \mathrm{Var}\left(\sum_{k=0}^{n-1} h(\check{Y}_k^{(i)}) \right) = \frac{1}{4} \mathrm{Var}\left(\sum_{k=0}^{2n-1} f(X_k^{(i)}) \right),$$

which implies, by (IV.17),

$$v(f, P_iQ_i) = v(f, Q_iP_i) = \frac{1}{2} \lim_{n\to\infty} \frac{1}{n} \mathrm{Var}\left(\sum_{k=0}^{n} f(X_k^{(i)}) \right) \quad (i \in \{0,1\}).$$

Finally, by (IV.17) we may now apply Theorem IV.4 to the chains $\{X_k^{(i)}; k \in \mathbb{N}\}$, $i \in \{0,1\}$, which establishes immediately the statement of the lemma. □

Proof of Theorem IV.17. We introduce the kernels

- $P_1(y, u; \mathrm{d}y' \times \mathrm{d}u') = \delta_{(y,u)}(\mathrm{d}y' \times \mathrm{d}u')$,
- $P_2(y, u; \mathrm{d}y' \times \mathrm{d}u') = \delta_y(\mathrm{d}y') R(y, \mathrm{d}u')$,
- $Q_1 = Q_2$ being defined implicitly as the transition kernel associated with the *freeze algorithm* (Algorithm 8).

It can be checked readily that the two sequences $\{Y_k^{(1)}; k \in \mathbb{N}\}$ and $\{Y_k^{(2)}; k \in \mathbb{N}\}$ generated by Algorithm 8 and Algorithm 9, respectively, have indeed the same distributions as the marginal processes (with respect to the first component) of homogeneous chains evolving according to the products P_1Q_1 and P_2Q_2, respectively.

In addition, all kernels P_i and Q_i, $i \in \{1, 2\}$, are π-reversible, as

- P_1 is reversible with respect to any probability measure (in particular, it is π-reversible),
- P_2 is π-reversible as a Gibbs-sampler sub-step transition kernel,
- $Q_1 = Q_2$ is π-reversible as a classical Metropolis-Hastings transition kernel.

Since P_1 has no off-diagonal component, it holds that $P_2 \succeq P_1$; moreover, trivially, $Q_2 = Q_1 \succeq Q_1$. Thus, we may complete the proof by applying Lemma IV.18 to the function $f(y, u) = h(y)$, for which the condition (IV.16) is satisfied (by (IV.14)). □

Case II : simulating R-transitions is infeasible

Pseudo-marginal algorithms (see [AR09] and [AV12]) are implemented using a Markov kernel $\check{R}$ on $\mathsf{Y} \times \mathcal{U}$ and a family $\{w_u; u \in \mathsf{U}\}$ of real-valued nonnegative functions on Y such that $\int \check{R}(y, \mathrm{d}u) w_u(y) = 1$ for all $y \in \mathsf{Y}$. We denote by $\check{r}$ the transition density of the kernel $\check{R}$ with respect to some dominating measure. Note that $R(y, \mathrm{d}u) := \check{R}(y, \mathrm{d}u) w_u(y)$ is a Markov transition kernel as well. The problem at hand is to sample the target distribution

$$\pi(\mathrm{d}y \times \mathrm{d}u) := \pi^*(\mathrm{d}y) R(y, \mathrm{d}u) = \pi^*(\mathrm{d}y) \check{R}(y, \mathrm{d}u) w_u(y)$$

under the assumption that

- for all $(y, u) \in \mathsf{Y} \times \mathsf{U}$, $\pi^*(y) \check{r}(y, u) w_u(y)$ is known up to a normalizing constant,
- for all $y \in \mathsf{Y}$, $\check{R}(y, \cdot)$ can be sampled from.

The particular case where $w_u(y) = 1$ for all $(y, u) \in \mathsf{Y} \times \mathsf{U}$ was discussed in the previous section, and we now turn to the case $w_u(y) \neq 1$ (i.e. sampling directly from R is infeasible). The solution provided by pseudo-marginal algorithms consists in replacing, in Algorithm 9, the operation (i) by the sampling $U \sim \check{R}(y, \cdot)$, and the computing the acceptance probability α (as defined in (IV.11)) via the formula

$$\alpha(y, u, \hat{y}, \hat{u}) := 1 \wedge \frac{\pi^*(\hat{y}) \check{r}(\hat{y}, \hat{u}) w_{\hat{u}}(\hat{y}) s(\hat{y}, \hat{u}; y) t(\hat{y}, \hat{u}, y; u)}{\pi^*(y) \check{r}(y, u) w_u(y) s(y, u; \hat{y}) t(y, u, \hat{y}; \hat{u})}.$$

The output of this algorithm, which will be referred to as the *noisy algorithm* in the following, is typically not—on the contrary to Algorithm 9—π^*-reversible due to the replacement of R by $\check{R}$. This justifies the denomination. However, when w is close to unity the noisy algorithm is close to Algorithm 9, which is, according to Theorem IV.17, more efficient than Algorithm 8 in terms of asymptotic variance.

Example 20 (Monte Carlo within Metropolis)**.** The Monte Carlo within Metropolis algorithm (MCWM ; see [AR09]) resembles closely the GIMH algorithm (see Example 12), however with the important difference that the importance sampling estimates $\pi_N^*(Y_k)$ (given by (IV.12)) are *not* stored and propagated through the algorithm along with the Y_k-values. Instead each estimate of the marginal density is recomputed using a "fresh" MC sample before the calculation of the acceptance probability. Thus, the MCWM algorithm can be cast into the framework of the noisy algorithm with $T = \check{R}$ and with the auxiliary variables U and $\hat{U}$ playing the roles of N-dimensional Monte Carlo samples.

Considering this, we now propose a novel algorithm which will be referred to as the *random refreshment algorithm* and which is a hybrid between Algorithm 9 and the noisy algorithm. This novel algorithm, which is described in Algorithm 10 below, targets *exactly* π^* and turns out to be more efficient than Algorithm 8.

Algorithm 10 The *random refreshment* algorithm

Given $(Y_k^{(3)}, U_k^{(3)}) = (y, u)$,

(i) (i.1) draw $U' \sim \check{R}(y, \cdot) \rightsquigarrow u'$,
(i.2) set

$$\check{U} \leftarrow \begin{cases} u' & \text{with probability } \varrho(y, u, u') := 1 \wedge \dfrac{w_{u'}(y)}{w_u(y)}, \\ u & \text{otherwise,} \end{cases} \rightsquigarrow \check{u}. \tag{IV.18}$$

(ii) draw $\hat{Y} \sim S(y, \check{u}; \cdot) \rightsquigarrow \hat{y}$,

(iii) draw $\hat{U} \sim T(y, \check{u}, \hat{y}; \cdot) \rightsquigarrow \hat{u}$,

(iv) set $(Y_{k+1}^{(3)}, U_{k+1}^{(3)}) \leftarrow \begin{cases} (\hat{y}, \hat{u}) & \text{with probability } \alpha(y, \check{u}, \hat{y}, \hat{u}), \\ (y, \check{u}) & \text{otherwise.} \end{cases}$

In Step (i) in Algorithm 10, the auxiliary variable $\check{U}$ can be either "refreshed", i.e. replaced by a new candidate U', or kept at the previous state $U_k^{(3)}$ according to an acceptance probability that turns out to be a standard Metropolis-Hastings acceptance probability (which will be seen in the proof of Theorem IV.22 below). Interestingly, this allows the desired distribution π as the target distribution of $\{(Y_k^{(3)}, U_k^{(3)}); k \in \mathbb{N}\}$. In comparison, the noisy algorithm described above differs only from Algorithm 10 by Step (i), in that the new candidate is always accepted in the noisy algorithm. This "systematic refreshment" makes actually the noisy algorithm imprecise in the sense that π is no longer the target distribution except when $w_u(y) = 1$ for all $(y, u) \in \mathsf{Y} \times \mathsf{U}$ (in which case $\varrho(y, u, u')$ in (IV.18) becomes identically equal to unity and Algorithm 10 translates into Algorithm 9. Compared to Algorithm 8, Step (i) allows the second component to be refreshed randomly according to the probability $\varrho(y, u, \check{u})$ whereas this component remains unchanged in Algorithm 8. Thus, in conformity with Algorithm 9, it is likely that Algorithm 10 has better mixing properties than Algorithm 8. That this is indeed the case may be established by reapplying the embedding technique developed in the previous part. Before formalizing this properly, we propose an example showing a typical situation where a Random Refreshment algorithm may be used.

Example 21 (Random refreshment GIMH-ABC)**.** In the paper [LAD12] (contributing to the discussion of [FP12]), the authors propose a novel algorithm, *rejuvenating GIMH-ABC* [LAD12, Algorithm 1], preventing the original *GIMH-ABC* [FP12, Algorithm 2] (termed *MCMC-ABC* in the paper in question) from falling into possible trapping states. The GIMH-ABC is an instance of Algorithm 8 targeting $\pi(\mathrm{d}y \times \mathrm{d}u \mid s_{\mathrm{obs}}) := \pi^*(\mathrm{d}y \mid s_{\mathrm{obs}})\check{R}(y, \mathrm{d}u)w_u(y, s_{\mathrm{obs}})$, where, in the ABC context,

- $\pi^*(\mathrm{d}y \mid s_{\mathrm{obs}})$ is the desired posterior of a parameter y given some observed data summary statistics s_{obs},
- $\check{R}(y, \cdot)$ is the likelihood of the data (from which sampling is assumed to be feasible),
- $w_u(y, s_{\mathrm{obs}}) := K[(s(u) - s_{\mathrm{obs}})/h] / \int \check{R}(y, \mathrm{d}u')K[(s(u') - s_{\mathrm{obs}})/h]$, where K is a kernel integrating to unity, providing the classical ABC discrepancy measure between the observed data summary statistics s_{obs} and that evaluated at the simulated data u.

Rejuvenating GIMH-ABC comprises an intermediate step in which the simulated data u, generated under the current parameter y, are refreshed systematically. However, since sampling from $R(y, \mathrm{d}u) := \check{R}(y, \mathrm{d}u)w_u(y, s_{\mathrm{obs}})$ is typically infeasible, the auxiliary variables are refreshed through $\check{R}$ in the spirit of Algorithm 9. Therefore, in accordance with Algorithm 10, a π-*reversible* alternative to rejuvenating GIMH-ABC is obtained by, instead of refreshing systematically the data, performing refreshment with probability (IV.18). Note that the fact that the constant in the denominator of $w_u(y, s_{\mathrm{obs}})$ is typically not computable does not prevent computation of (IV.18), since this constant appears in $w_u(y, s_{\mathrm{obs}})$ as well as $w_{u'}(y, s_{\mathrm{obs}})$. This provides a *random refreshment GIMH-ABC*, which can be compared quantitatively, via the Theorem IV.22 below, to the GIMH-ABC while at the same time avoiding the possible GIMH-ABC trapping states mentioned in [LAD12].

Theorem IV.22. Let $\{Y_k^{(1)}; k \in \mathbb{N}\}$ and $\{Y_k^{(3)}; k \in \mathbb{N}\}$ be the sequences of random variables generated by Algorithm 8 and Algorithm 10, respectively, where $(Y_0^{(i)}, U_0^{(i)}) \sim \pi$, $i \in \{1, 3\}$. Then the following holds true.

(i) The output of Algorithm 10 is π-reversible.

(ii) For all $h \in \mathsf{L}^2(\pi^*)$ satisfying

$$\sum_{k=1}^{\infty} |\mathrm{Cov}(h(Y_0^{(i)}), h(Y_k^{(i)}))| < \infty \quad (i \in \{1, 3\})$$

it holds that

$$\lim_{n\to\infty} \frac{1}{n} \mathrm{Var}\left(\sum_{k=0}^{n-1} h(Y_k^{(3)})\right) \leq \lim_{n\to\infty} \frac{1}{n} \mathrm{Var}\left(\sum_{k=0}^{n-1} h(Y_k^{(1)})\right).$$

Proof. Let the kernels P_1 and Q_1 be defined as in the proof of Theorem IV.17 and introduce furthermore

- P_3 defined implicitly by the transition $(Y_k^{(3)}, U_k^{(3)}) \to (Y_k^{(3)}, \check{U})$ according to Step (i) in Algorithm 10 (note that the first component is held fixed throughout the transition),
- $Q_3 = Q_1$.

In conformity with the proof of Theorem IV.17 it can be checked readily that the two sequences $\{Y_k^{(1)}; k \in \mathbb{N}\}$ and $\{Y_k^{(3)}; k \in \mathbb{N}\}$ generated by Algorithm 8 and Algorithm 10, respectively, have indeed the same distributions as the marginal processes (with respect to the first component) of homogeneous chains evolving according to the products P_1Q_1 and P_3Q_3, respectively. The π-reversibility of the kernels P_1 and $Q_1 = Q_3$ was established in the proof of Theorem IV.17. To verify π-reversibility of P_3 as well, note that P_3 is a Metropolis-Hastings kernel associated with the target distribution π, whose acceptance probability includes a Radon-Nikodym derivative of the type given in Proposition IV.10; it is therefore π-reversible. Indeed, note that P_3 updates only the second component according to $\check{R}(y, \mathrm{d}u')$ with the acceptance probability $\varrho(y, u, u')$. Assuming first that $\check{R}$ is dominated and denoting by $\check{r}$ its transition density, we have

$$\varrho(y, u, u') = 1 \wedge \frac{w_{u'}(y)}{w_u(y)} = 1 \wedge \frac{\pi(y, u')\check{r}(y, u)}{\pi(y, u)\check{r}(y, u')},$$

where $\pi(y, u) = \pi^*(y)\check{r}(y, u)w_u(y)$ in the density of the target π. This shows that $\varrho(y, u, u')$ is indeed the acceptance probability of a Metropolis-Hastings Markov chain targeting π,

with proposal kernel $\check{R}(y, \mathrm{d}u')\delta_y(\mathrm{d}y')$; the π-reversibility of P_3 follows. The proof can be adapted easily to the case where $\check{R}$ is not dominated. As a consequence, the product P_3Q_3 is also π-reversible, which establishes the statement (i) of the theorem. Finally, since P_1 has zero mass on the off-diagonal, it holds that $P_3 \succeq P_1$ and, clearly, $Q_3 = Q_1 \succeq Q_1$. The proof of (ii) is now concluded by applying Lemma IV.18 along the lines of the proof of Theorem IV.17. □

The algorithm of Carlin and Chib

In this section we apply Theorem IV.4 in the context of Gibbs samplers. In algorithms of Gibbs-type, the ordering in Definition IV.1 is usually not applicable since candidates are accepted with probability one, which implies that the chain never remains in the same state. The ordering is however meaningful when a component is discrete, in which case the analysis can again be cast into the framework of Theorem IV.4.

The algorithm of B. P. Carlin and S. Chib [CC95] was introduced in the context of model selection. For mixture models, the target is a probability distribution π^* on $(\mathsf{Y}, \mathcal{Y})$, where $\mathsf{Y} = \{1, \dots, n\} \times \mathsf{Z}$ and $\mathcal{Y}$ is the associated Borel sigma-field, and a π^*-distributed variable $Y = (M, Z)$ thus comprises a model index random variable M and a (typically continuous) random vector Z. A natural idea of sampling π^* consists in implementing a Gibbs sampler to obtain a Markov chain $\{Y_k^{(\mathrm{G})}; k \in \mathbb{N}\}$ with transitions as follows.

Algorithm 11 The Gibbs algorithm

Given $Y_k^{(\mathrm{G})} = (m, z)$,

(i) draw $M' \sim \pi^*(\,\cdot \mid z)$ and call the outcome m',

(ii) draw $Z' \sim \pi^*(\,\cdot \mid m')$ and call the outcome z',

(iii) set $Y_{k+1}^{(\mathrm{G})} = (m', z')$.

Remark IV.23. Since M is a discrete random variable, it is always possible to sample $M \sim \pi^*(\,\cdot \mid z)$. In contrast, sampling $Z \sim \pi^*(\,\cdot \mid m)$ is not always possible. In that case, one may replace (ii) by a Metropolis-Hastings step, yielding a Metropolis-within-Gibbs algorithm (see [RC04, section 10.3.3] for details).

Remark IV.24. Even though $\{Y_k^{(\mathrm{G})}; k \in \mathbb{N}\}$ is a Markov chain with stationary distribution π^*, it turns out in practice that the discrete component $\{M_k^{(\mathrm{G})}; k \in \mathbb{N}\}$ tends to be stuck in a few states. Indeed, when a variable z is sampled according to a model m, the probability of jumping to another model $m' \neq m$ is proportional to $\pi^*(m', z)$, which may be very low when the index component M is informative concerning the localization of Z.

The Carlin and Chib algorithm was introduced in [CC95] to circumvent this drawback by using some artificial auxiliary variables. It may be regarded as a data-augmentation algorithm based on a Gibbs sampler on the extended state space $\{1, \dots, n\} \times \mathsf{Z}^n$ which targets the distribution π defined for all $m \in \{1, \dots, n\}$ and $u = (u_1, \dots, u_n) \in \mathsf{Z}^n$ as

$$\pi(m, \mathrm{d}u) := \pi^*(m, \mathrm{d}u_m) \prod_{j \neq m} \rho_j(\mathrm{d}u_j), \tag{IV.19}$$

where $\{\rho_j\}_{j=1}^n$ are in [CC95] referred to as *pseudopriors* or *linking densities*. In what follows, we assume that $\pi^*(m, \cdot)$ and $\{\rho_j : j = 1, \dots, n\}$ are all dominated by some nonnegative measure ν. The choice of the pseudopriors should be tuned by the user,

provided that they are analytically tractable and can be sampled. Denote by $\{Y_k^{(\mathrm{C})}; k \in \mathbb{N}\}$ the Markov chain obtained by the Carlin and Chib algorithm, whose transitions comprise $n+1$ steps and can be described as follows.

Algorithm 12 The Carlin and Chib algorithm

Given $Y_k^{(\mathrm{C})} = (m, u_m)$,

(i) draw for all $j \neq m$, $U_j \sim \rho_j$ and call the outcomes u_j,

(ii) draw $M' \sim \pi(\,\cdot \mid u)$ and call the outcome m',

(iii) draw $U'_{m'} \sim \pi^*(\,\cdot \mid m')$ and call the outcome $u_{m'}$.

(iv) set $Y_{k+1}^{(\mathrm{C})} = (m', u'_{m'})$.

Similarly to Remark IV.23, one may use a Metropolis step instead of (iii) if sampling from $\pi^*(\,\cdot \,|\, m)$ is not possible.

Remark IV.25. Intuitively, the Carlin and Chib algorithm allows more visits to the different models than the Gibbs sampler ; indeed, in step (ii), the probability to jump to a model $m' \neq m$ is proportional to

$$\pi(m' \mid u) \propto \pi^*(m', u_{m'})\rho_m(u_m) \prod_{j \notin \{m, m'\}} \rho_j(u_j), \tag{IV.20}$$

where the three factors on the right hand side are large if the pseudopriors are chosen such that ρ_ℓ is close to $\pi^*(\,\cdot \,|\, \ell)$ for all $\ell \in \{1, \ldots, n\}$. The optimal case where $\rho_\ell(\,\cdot\,) = \pi^*(\,\cdot \mid \ell)$ implies, using (IV.20), that

$$\pi(m' \mid u) \propto \pi^*(m', u_{m'}) \prod_{j \neq m'} \pi^*(\,u_j \mid j).$$

It can be readily checked from the equation above that $\pi(m' \,|\, u) = \int \pi^*(m', \mathrm{d}z)$. Plugging this expression into step (ii) shows that we actually draw m' according to the exact marginal of the class index random variable regardless the value of u. Thus, the Carlin and Chib algorithm simulates *i.i.d.* samples according to π^*. However, this extreme and ideal situation requires that the quantity $\rho_\ell(\,\cdot\,) = \pi^*(\,\cdot \,|\, \ell)$ is tractable which is typically not the case.

In the light of the previous remark one may assume reasonably that the Carlin and Chib algorithm provides better estimates than the Gibbs sampler. However, since neither $\{Y_k^{(\mathrm{G})}; k \in \mathbb{N}\}$ nor $\{Y_k^{(\mathrm{C})}; k \in \mathbb{N}\}$ are π^*-reversible, [Tie95, Theorem 4] does not allow these two algorithms to be compared. Nevertheless, with an approach similar to that used in the *pseudo-marginal* context, we may provide a theoretical justification, based on Theorem IV.4, advocating the Carlin and Chib algorithm ahead of the Gibbs sampler. To do this we first embed $\{Y_k^{(\mathrm{G})}; k \in \mathbb{N}\}$ and $\{Y_k^{(\mathrm{C})}; k \in \mathbb{N}\}$ into two inhomogeneous π^*-reversible Markov chains $\{X_k^{(\mathrm{G})}; k \in \mathbb{N}\}$ and $\{X_k^{(\mathrm{C})}; k \in \mathbb{N}\}$ defined on $\mathsf{Y} = \{1, \ldots, n\} \times \mathsf{Z}$ through, for $i \in \{\mathrm{G}, \mathrm{C}\}$:

$$X_{2k}^{(i)} = \begin{pmatrix} M_k^{(i)} \\ Z_k^{(i)} \end{pmatrix} \xrightarrow{P_i} X_{2k+1}^{(i)} = \begin{pmatrix} \check{M}_k^{(i)} \\ \check{Z}_k^{(i)} \end{pmatrix} \xrightarrow{Q_i} X_{2k+2}^{(i)} = \begin{pmatrix} M_{k+1}^{(i)} \\ Z_{k+1}^{(i)} \end{pmatrix} \xrightarrow{P_i} \cdots \tag{IV.21}$$

Here

(i) P_{G} is defined implicitly by the transitions $\check{M}_k^{(\mathrm{G})} \sim \pi^*(\,\cdot \mid Z_k^{(\mathrm{G})})$ and $\check{Z}_k^{(\mathrm{G})} = Z_k^{(\mathrm{G})}$,

(ii) P_{C} is defined implicitly by the transitions consisting in
   – drawing the random vector $U = (U_1, \ldots, U_n) \in \mathsf{Z}^n$ such that

$$U_j \sim \begin{cases} \rho_j & \text{for } j \neq M_k^{(\mathrm{C})}, \\ \delta_{Z_k^{(\mathrm{C})}} & \text{for } j = M_k^{(\mathrm{C})}, \end{cases}$$

   – letting $\check{M}_k^{(\mathrm{C})} = m'$ with probability

$$\pi(m' \,|\, U) \propto \pi^*(m', U_{m'}) \prod_{j \neq m'} \rho_j(U_j),$$

   – letting $\check{Z}_k^{(\mathrm{C})} = U_{m'}$.

(iii) $Q_{\mathrm{G}} = Q_{\mathrm{C}}$ is defined implicitly by the transitions $Z_{k+1}^{(\mathrm{G})} \sim \pi^*(\,\cdot \mid \check{M}_k^{(\mathrm{G})})$ and $M_{k+1}^{(\mathrm{G})} = \check{M}_k^{(\mathrm{G})}$.

With $Y_k^{(i)} = (M_k^{(i)}, Z_k^{(i)})$, $i \in \{\mathrm{G}, \mathrm{C}\}$, it can be checked easily that $\{Y_k^{(\mathrm{G})}; k \in \mathbb{N}\}$ and $\{Y_k^{(\mathrm{C})}; k \in \mathbb{N}\}$ are sampled exactly as in the Gibbs sampler and the Carlin and Chib algorithm, respectively.

Remark IV.26. Since P_{G} and Q_{G} correspond to the one step Gibbs transitions (i) and (ii) in Algorithm 11, respectively, they are both π^*-reversible. Therefore, $Q_{\mathrm{C}} = Q_{\mathrm{G}}$ is π^*-reversible and we show in Appendix 8 .3 that P_{C} is also a π^*-reversible kernel.

Theorem IV.27. Let $\{X_k^{(i)}; k \in \mathbb{N}\}$, $i \in \{\mathrm{C}, \mathrm{G}\}$, be the Markov chains obtained through (IV.21) starting with $X_0^{(i)} \sim \pi^*$ for $i \in \{\mathrm{C}, \mathrm{G}\}$. In addition, assume that for all $m \neq m'$ and $z \in \mathsf{Z}$,

$$P_{\mathrm{C}}(m, z; m', \mathrm{d}z') \geq P_{\mathrm{G}}(m, z; m', \mathrm{d}z'). \tag{IV.22}$$

Then for all $f \in \mathsf{L}^2(\pi)$ such that for $i \in \{\mathrm{G}, \mathrm{C}\}$,

$$\sum_{k=1}^{\infty} \left(|\mathrm{Cov}(f(X_0^{(i)}), f(X_k^{(i)}))| + |\mathrm{Cov}(f(X_1^{(i)}), f(X_{k+1}^{(i)}))| \right) < \infty$$

it holds that

$$\lim_{n\to\infty} \frac{1}{n} \mathrm{Var}\left(\sum_{k=0}^{n-1} f(X_k^{(\mathrm{C})}) \right) \leq \lim_{n\to\infty} \frac{1}{n} \mathrm{Var}\left(\sum_{k=0}^{n-1} f(X_k^{(\mathrm{G})}) \right).$$

Assumption (IV.22) formalizes the fact that Algorithm 12 allows more jumps between models than Algorithm 11 (as noted in Remarks IV.24 and IV.25).

Proof. According to Remark IV.26, $\{X_k^{(\mathrm{G})}; k \in \mathbb{N}\}$ and $\{X_k^{(\mathrm{C})}; k \in \mathbb{N}\}$ are both inhomogeneous Markov chains that evolve alternatingly according to the π^*-reversible kernels P_i and Q_i, $i \in \{\mathrm{G}, \mathrm{C}\}$. Moreover, (IV.22) implies clearly that $P_{\mathrm{C}} \succeq P_{\mathrm{G}}$, and clearly $Q_{\mathrm{C}} = Q_G \succeq Q_{\mathrm{G}}$. The proof may now be concluded by applying Theorem IV.4 along the lines of the proofs of Theorem IV.17 or Theorem IV.22. □

5 Proof of Theorem IV.4

We preface the proof of Theorem IV.4 by some preliminary lemmas.

Lemma IV.28. Assume that $P_1, P_2, \ldots, P_n$ are π-reversible Markov transition kernels. Then, for all $(f, g) \in \mathsf{L}^2(\pi) \times \mathsf{L}^2(\pi)$,

$$\langle f, P_1 P_2 \cdots P_n g \rangle = \langle P_n \cdots P_2 P_1 f, g \rangle .$$

Proof. As each P_ℓ is π-reversible, it holds that $\langle P_\ell f, g \rangle = \langle f, P_\ell g \rangle$ for all $(f, g) \in \mathsf{L}^2(\pi) \times \mathsf{L}^2(\pi)$ and $\ell \in \{1, \ldots, n\}$. Applying repeatedly this relation yields

$$\begin{aligned}\langle f, P_1 P_2 \cdots P_n g \rangle &= \langle P_1 f, P_2 \cdots P_n g \rangle \\ &= \langle P_2 P_1 f, P_3 \cdots P_n g \rangle \cdots = \langle P_n \cdots P_2 P_1 f, g \rangle .\end{aligned}$$

□

Lemma IV.29. Let P and Q be Markov transition kernels on $(\mathsf{X}, \mathcal{X})$ such that $\pi P = \pi Q = \pi$ and let $\{X_k; k \in \mathbb{N}\}$ be a Markov chain evolving as

$$X_0 \xrightarrow{P} X_1 \xrightarrow{Q} X_2 \xrightarrow{P} X_3 \xrightarrow{Q} \cdots$$

with initial distribution $X_0 \sim \pi$. Then, for all $f \in \mathsf{L}^2(\pi)$ such that

$$\sum_{k=1}^{\infty} \left(|\mathrm{Cov}(f(X_0), f(X_k))| + |\mathrm{Cov}(f(X_1), f(X_{k+1}))| \right) < \infty, \tag{IV.23}$$

the limit, as n tends to infinity, of $n^{-1}\mathrm{Var}(\sum_{k=0}^{n-1} f(X_k))$ exists, and

$$\begin{aligned}\lim_{n\to\infty} \frac{1}{n} \mathrm{Var}\left(\sum_{k=0}^{n-1} f(X_k) \right) &= \pi f^2 - \pi^2 f \\ &\quad + \sum_{k=1}^{\infty} \mathrm{Cov}(f(X_0), f(X_k)) + \sum_{k=1}^{\infty} \mathrm{Cov}(f(X_1), f(X_{k+1})).\end{aligned} \tag{IV.24}$$

Proof. As covariances are symmetric,

$$\frac{1}{n} \mathrm{Var}\left(\sum_{k=0}^{n-1} f(X_k) \right) = \pi f^2 - \pi^2 f + 2n^{-1} \sum_{0 \le i < j \le n-1} \mathrm{Cov}(f(X_i), f(X_j)).$$

We now consider the limit, as n tends to infinity, of the last term on the right hand side. Let $\mathcal{E}$ and $\mathcal{O}$ denote the two complementary subsets of $\mathbb{N}$ consisting of the even and odd numbers, respectively. For all $(i, j) \in \mathbb{N}^2$ such that $i < j$, we have

$$\mathrm{Cov}(f(X_i), f(X_j)) = \begin{cases} \mathrm{Cov}(f(X_0), f(X_{j-i})) & \text{if } i \in \mathcal{E}, \\ \mathrm{Cov}(f(X_1), f(X_{j-i+1})) & \text{if } i \in \mathcal{O}. \end{cases}$$

This implies that

$$\begin{aligned}n^{-1} \sum_{\substack{0 \le i < j \le n-1 \\ i \in \mathcal{E}}} \mathrm{Cov}(f(X_i), f(X_j)) &= \\ &\sum_{k=1}^{n-1} n^{-1} \left(\left\lfloor \frac{n-1-k}{2} \right\rfloor + 1 \right) \mathrm{Cov}(f(X_0), f(X_k))\end{aligned}$$

and

$$n^{-1}\sum_{\substack{0\le i<j\le n-1\\ i\in\mathcal{O}}}\mathrm{Cov}(f(X_i),f(X_j))=\sum_{k=1}^{n-2}n^{-1}\left(\left\lfloor\frac{n-2-k}{2}\right\rfloor+1\right)\mathrm{Cov}(f(X_1),f(X_{k+1})).$$

Under (IV.23), the dominated convergence theorem applies, which provides that the limit, as n goes to infinity, of $n^{-1}\mathrm{Var}(\sum_{k=0}^{n-1}f(X_k))$ exists and is equal to (IV.24). □

Lemma IV.30. Let P_i and Q_i, $i\in\{0,1\}$, be π-reversible Markov kernels on $(\mathsf{X},\mathcal{X})$ such that $P_0\succcurlyeq P_1$ and $Q_0\succcurlyeq Q_1$. For all $n\in\mathbb{N}$ and $i\in\{0,1\}$, denote by $R_n^{(i)}$ the Markov kernel $R_n^{(i)}:=P_i\mathbb{1}_{\mathcal{E}}(n)+Q_i\mathbb{1}_{\mathcal{O}}(n)$. In addition, let $f\in\mathsf{L}^2(\pi)$ be such that for $i\in\{0,1\}$,

$$\sum_{k=1}^{\infty}\left|\left\langle f,R_0^{(i)}\cdots R_{k-1}^{(i)}f\right\rangle\right|<\infty. \tag{IV.25}$$

Then for all $\lambda\in(0,1)$,

$$\sum_{k=1}^{\infty}\lambda^k\left(\left\langle f,R_0^{(1)}\cdots R_{k-1}^{(1)}f\right\rangle+\left\langle f,R_1^{(1)}\cdots R_k^{(1)}f\right\rangle\right)\le\sum_{k=1}^{\infty}\lambda^k\left(\left\langle f,R_0^{(0)}\cdots R_{k-1}^{(0)}f\right\rangle+\left\langle f,R_1^{(0)}\cdots R_k^{(0)}f\right\rangle\right).$$

Proof. For all $n\in\mathbb{N}$ and all $\alpha\in(0,1)$, define $R_n^{(\alpha)}:=(1-\alpha)R_n^{(0)}+\alpha R_n^{(1)}$. In addition, set, for $\lambda\in(0,1)$, $K_\lambda(\alpha):=K_\lambda^{(\mathcal{E})}(\alpha)+K_\lambda^{(\mathcal{O})}(\alpha)$, where

$$K_\lambda^{(\mathcal{E})}(\alpha):=\sum_{k=1}^{\infty}\lambda^k\left\langle f,R_0^{(\alpha)}\cdots R_{k-1}^{(\alpha)}f\right\rangle,$$
$$K_\lambda^{(\mathcal{O})}(\alpha):=\sum_{k=1}^{\infty}\lambda^k\left\langle f,R_1^{(\alpha)}\cdots R_k^{(\alpha)}f\right\rangle.$$

Now, fix a distinguished $\lambda\in(0,1)$; we want show that for all $\alpha\in[0,1]$,

$$\frac{\mathrm{d}K_\lambda}{\mathrm{d}\alpha}(\alpha)\le 0. \tag{IV.26}$$

Thus, we start with differentiating $K_\lambda^{(\mathcal{E})}$:

$$\frac{\mathrm{d}K_\lambda^{(\mathcal{E})}}{\mathrm{d}\alpha}(\alpha)=\frac{\mathrm{d}}{\mathrm{d}\alpha}\sum_{k=1}^{\infty}\lambda^k\left\langle f,R_0^{(\alpha)}\cdots R_{k-1}^{(\alpha)}f\right\rangle. \tag{IV.27}$$

To interchange $\frac{\mathrm{d}}{\mathrm{d}\alpha}$ and $\sum_{k=1}^{\infty}$ in the previous equation, we first note that

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\alpha}\left\langle f,R_0^{(\alpha)}\cdots R_{k-1}^{(\alpha)}f\right\rangle&=\sum_{\ell=0}^{k-1}\frac{\partial}{\partial\alpha_\ell}\left\langle f,R_0^{(\alpha_0)}\cdots R_{k-1}^{(\alpha_{k-1})}f\right\rangle\Big|_{(\alpha_0,\ldots,\alpha_{k-1})=(\alpha,\ldots,\alpha)}\\&=\sum_{\ell=0}^{k-1}\left\langle f,R_{0\nearrow\ell-1}^{(\alpha)}(R_\ell^{(1)}-R_\ell^{(0)})R_{\ell+1\nearrow k-1}^{(\alpha)}f\right\rangle,\end{aligned}$$

where $R^{(\alpha)}_{s\nearrow t} := R^{(\alpha)}_s R^{(\alpha)}_{s+1} \cdots R^{(\alpha)}_t$ for $s \le t$ and $R^{(\alpha)}_{s\nearrow t} := \mathrm{id}$ otherwise. By (IV.1), $\|R^{(\alpha)}_n\| \le 1$, which implies that $\sup_{\alpha\in[0,1]} |\frac{\mathrm{d}}{\mathrm{d}\alpha}\langle f, R^{(\alpha)}_0 \cdots R^{(\alpha)}_{k-1} f\rangle| \le 2k\pi(f^2)$. Thus, as $\sum_{k=1}^{\infty} \lambda^k k < \infty$ we may interchange, in (IV.27), $\frac{\mathrm{d}}{\mathrm{d}\alpha}$ and $\sum_{k=1}^{\infty}$, yielding

$$\frac{\mathrm{d}K^{(\mathcal{E})}_\lambda}{\mathrm{d}\alpha}(\alpha) = \sum_{k=1}^{\infty} \lambda^k \sum_{\ell=0}^{k-1} \left\langle f, R^{(\alpha)}_{0\nearrow \ell-1} (R^{(1)}_\ell - R^{(0)}_\ell) R^{(\alpha)}_{\ell+1\nearrow k-1} f \right\rangle .$$

Similarly, it can be established that

$$\frac{\mathrm{d}K^{(\mathcal{O})}_\lambda}{\mathrm{d}\alpha}(\alpha) = \sum_{k=1}^{\infty} \lambda^k \sum_{\ell=1}^{k} \left\langle f, R^{(\alpha)}_{1\nearrow \ell-1} (R^{(1)}_\ell - R^{(0)}_\ell) R^{(\alpha)}_{\ell+1\nearrow k} f \right\rangle .$$

We now apply Lemma IV.28 to the two previous sums. For this purpose we will use the following notation $R^{(\alpha)}_{s\searrow t} := R^{(\alpha)}_s R^{(\alpha)}_{s-1} \cdots R^{(\alpha)}_t$ for $s \ge t$ and $R^{(\alpha)}_{s\searrow t} := \mathrm{id}$ otherwise. Then,

$$\begin{aligned}
&\frac{\mathrm{d}K_\lambda}{\mathrm{d}\alpha}(\alpha) \\
&= \sum_{k=1}^{\infty} \lambda^k \Bigg\{ \sum_{\ell=0}^{k-1} \left\langle R^{(\alpha)}_{\ell-1\searrow 0} f, (R^{(1)}_\ell - R^{(0)}_\ell) R^{(\alpha)}_{\ell+1\nearrow k-1} f \right\rangle \\
&\quad + \sum_{\ell=1}^{k} \left\langle R^{(\alpha)}_{\ell-1\searrow 1} f, (R^{(1)}_\ell - R^{(0)}_\ell) R^{(\alpha)}_{\ell+1\nearrow k} f \right\rangle \Bigg\} \\
&= \sum_{\ell=0}^{\infty} \sum_{m=0}^{\infty} \lambda^{\ell+m+1} \left\langle R^{(\alpha)}_{\ell-1\searrow 0} f, (R^{(1)}_\ell - R^{(0)}_\ell) R^{(\alpha)}_{\ell+1\nearrow \ell+m} f \right\rangle \\
&\quad + \sum_{\ell=1}^{\infty} \sum_{m=1}^{\infty} \lambda^{\ell+m-1} \left\langle R^{(\alpha)}_{\ell-1\searrow 1} f, (R^{(1)}_\ell - R^{(0)}_\ell) R^{(\alpha)}_{\ell+1\nearrow \ell+m-1} f \right\rangle .
\end{aligned}$$

Now, note that $R^{(\alpha)}_n = R^{(\alpha)}_{n'}$ for all $(n, n') \in \mathcal{O}^2$ and $R^{(\alpha)}_m = R^{(\alpha)}_{m'}$ for all $(m, m')^2 \in \mathcal{E}^2$; hence, separating, in the two previous sums, odd and even indices ℓ provides

$$\begin{aligned}
&\frac{\mathrm{d}K_\lambda}{\mathrm{d}\alpha}(\alpha) \\
&= \sum_{\ell\in\mathcal{E}} \sum_{m=0}^{\infty} \lambda^{\ell+m+1} \left\langle R^{(\alpha)}_{1\nearrow \ell} f, (R^{(1)}_0 - R^{(0)}_0) R^{(\alpha)}_{1\nearrow m} f \right\rangle \\
&\quad + \sum_{\ell\in\mathcal{E}\setminus\{0\}} \sum_{m=1}^{\infty} \lambda^{\ell+m-1} \left\langle R^{(\alpha)}_{1\nearrow \ell-1} f, (R^{(1)}_0 - R^{(0)}_0) R^{(\alpha)}_{1\nearrow m-1} f \right\rangle \\
&\quad + \sum_{\ell\in\mathcal{O}} \sum_{m=0}^{\infty} \lambda^{\ell+m+1} \left\langle R^{(\alpha)}_{0\nearrow \ell-1} f, (R^{(1)}_1 - R^{(0)}_1) R^{(\alpha)}_{0\nearrow m-1} f \right\rangle \\
&\quad + \sum_{\ell\in\mathcal{O}} \sum_{m=1}^{\infty} \lambda^{\ell+m-1} \left\langle R^{(\alpha)}_{0\nearrow \ell-2} f, (R^{(1)}_1 - R^{(0)}_1) R^{(\alpha)}_{0\nearrow m-2} f \right\rangle .
\end{aligned}$$

Finally, by combining the even and the odd sums,

$$\begin{aligned}
&\frac{\mathrm{d}K_\lambda}{\mathrm{d}\alpha}(\alpha) \\
&= \left\langle \sum_{\ell=0}^{\infty} \lambda^\ell R^{(\alpha)}_{1\nearrow \ell} f, (R^{(1)}_0 - R^{(0)}_0) \sum_{m=0}^{\infty} \lambda^m R^{(\alpha)}_{1\nearrow m} f \right\rangle \\
&\quad + \left\langle \sum_{\ell=0}^{\infty} \lambda^\ell R^{(\alpha)}_{0\nearrow \ell-1} f, (R^{(1)}_1 - R^{(0)}_1) \sum_{m=0}^{\infty} \lambda^m R^{(\alpha)}_{0\nearrow m-1} f \right\rangle .
\end{aligned}$$

Since $R_n^{(1)} \succcurlyeq R_n^{(0)}$, the operator $R_n^{(0)} - R_n^{(1)}$ is nonnegative on $\mathsf{L}^2(\pi)$ (by [Tie95, Lemma 3]), and for all $f \in \mathsf{L}^2(\pi)$ it holds that $\langle f, (R_n^{(1)} - R_n^{(0)})f\rangle \leq 0$. This shows (IV.26), which implies that the function $\alpha \mapsto K_\lambda(\alpha)$ is non-increasing on $(0,1)$. The proof is complete. □

Proof of Theorem IV.4. According to Lemma IV.29, for all functions $f \in \mathsf{L}^2(\pi)$ and $i \in \{0,1\}$,

$$v^{(i)}(f) = \pi f^2 - \pi^2 f + \sum_{k=1}^{\infty} \left(\mathrm{Cov}(f(X_0^{(i)}), f(X_k^{(i)})) + \mathrm{Cov}(f(X_1^{(i)}), f(X_{k+1}^{(i)}))\right). \quad \text{(IV.28)}$$

For the kernels P_i and Q_i, $i \in \{0,1\}$, in the statement of the theorem, let $\{R_k^{(i)}; k \in \mathbb{N}\}$, $i \in \{0,1\}$, be defined as in Lemma IV.30, which then implies that for all $\lambda \in (0,1)$,

$$\sum_{k=1}^{\infty} \left(\lambda^k \mathrm{Cov}(f(X_0^{(1)}), f(X_k^{(1)})) + \lambda^k \mathrm{Cov}(f(X_1^{(1)}), f(X_{k+1}^{(1)}))\right) \leq \sum_{k=1}^{\infty} \left(\lambda^k \mathrm{Cov}(f(X_0^{(0)}), f(X_k^{(0)})) + \lambda^k \mathrm{Cov}(f(X_1^{(0)}), f(X_{k+1}^{(0)}))\right). \quad \text{(IV.29)}$$

We conclude the proof by letting λ tend to one on each side of the previous inequality. Under (IV.5), we may, by the dominated convergence theorem, interchange limits with summation, which establishes the inequality (IV.29) also in the case $\lambda = 1$. Combining this with (IV.28) completes the proof. □

6 Conclusion

In this paper, we have extended successfully the theoretical framework proposed in [Pes73] and [Tie95] as a means of comparing the asymptotic variance of sample path averages for different Markov chains and, consequently, the efficiency of different MCMC algorithms to the context of inhomogeneous Markov chains evolving alternatingly according to two different Markov transition kernels. It turned out that this configuration covers, although not apparently, several popular MCMC algorithms such as Randomized MCMC [NFW12], Multiple-try Metropolis [LLW00] and its generalization [PBF10], and the pseudo-marginal algorithms [AR09, AV12]. It should be remarked however that our results do not take possible additional computational cost into consideration, which may be of importance in practical applications. While these algorithms are inapproachable for the standard tools provided in [Pes73] and [Tie95], our results allow, without heavy technical developments, rigorous theoretical justifications advocating the use of these algorithms. As illustrated by our novel *random refreshment* algorithm in the context of pseudo-marginal algorithms, the results of the present paper can also be used for designing new algorithms and improving, in terms of asymptotic variance, existing ones.

7 Appendix 1

7 .1 Proof of Proposition IV.9

First, set $\xi = P^n(x,\cdot) - \pi$; then by Jensen's inequality,

$$\|\xi\|_{V^{1/2}} = |\xi|(\mathsf{X})\frac{|\xi|(V^{1/2})}{|\xi|(\mathsf{X})} \le |\xi|(\mathsf{X})\left(\frac{|\xi|(V)}{|\xi|(\mathsf{X})}\right)^{1/2} = |\xi|^{1/2}(\mathsf{X})\,\|\xi\|_V^{1/2}\,,$$

and since $|\xi|(\mathsf{X}) \le 2$,

$$\|P^n(x,\cdot) - \pi\|_{V^{1/2}} \le (2C\rho^n V(x))^{1/2}\,. \tag{IV.30}$$

Now, without loss of generality we may assume that $\pi f = 0$, $|f|_{V^{1/2}} \le 1$, and $|Pf|_{V^{1/2}} \le 1$. Then applying (IV.30) yields for all $x \in \mathsf{X}$,

$$|(PQ)^n f(x)| \le (2C\rho^n V(x))^{1/2}\,.$$

Hence, for all $n \in \mathbb{N}$,

$$\begin{aligned}|\mathrm{Cov}(f(X_0), f(X_{2n}))| &= |\mathbb{E}(f(X_0)(PQ)^n f(X_0))| \\ &\le (2C\rho^n)^{1/2}\,\mathbb{E}\left(|f(X_0)|V^{1/2}(X_0)\right) \le (2C\rho^n)^{1/2}\,\pi V.\end{aligned}$$

In the same way, for all $n \ge 0$,

$$|\mathrm{Cov}(f(X_0), f(X_{2n+1}))| = |\mathbb{E}(f(X_0)(PQ)^n Pf(X_0))| \le (2C\rho^n)^{1/2}\,\pi V.$$

By applying successively the Cauchy-Schwarz and Jensen inequalities we obtain

$$\mathbb{E}\left(|f(X_1)|QV^{1/2}(X_1)\right) \le \left[\mathbb{E}\left(f^2(X_1)\right)\mathbb{E}\left(QV(X_1)\right)\right]^{1/2} \le \pi V,$$

where the last inequality follows from $f^2 \le V$ and $\pi P = \pi Q = \pi$. This implies that for all $n \in \mathbb{N}^*$,

$$\begin{aligned}|\mathrm{Cov}(f(X_1), f(X_{2n}))| &= |\mathbb{E}(f(X_1)Q(PQ)^{n-1} f(X_1))| \\ &\le \left(2C\rho^{n-1}\right)^{1/2}\mathbb{E}\left(|f(X_1)|QV^{1/2}(X_1)\right) \le \left(2C\rho^{n-1}\right)^{1/2}\pi V.\end{aligned}$$

In the same way, for all $n \in \mathbb{N}^*$ we have, using that $|Pf(x)| \le V^{1/2}(x)$,

$$|\mathrm{Cov}(f(X_1), f(X_{2n+1}))| = |\mathbb{E}(f(X_1)Q(PQ)^{n-1}Pf(X_1))| \le \left(2C\rho^{n-1}\right)^{1/2}\pi V.$$

The statement of the proposition follows.

7 .2 Proof of Proposition IV.13

Let K be the transition kernel of the Markov chain $\{Y_k^{(2)}; k \in \mathbb{N}\}$, i.e. for all $f \in \mathcal{F}(\mathcal{Y})$,

$$\begin{aligned}&\int f(y')K(y,\mathrm{d}y') \\ &\qquad = f(y)\beta(y) + \int f(y')R(y,\mathrm{d}u)S(y,u;\mathrm{d}y')T(y,u,y';\mathrm{d}u')\alpha(y,u,y',u'),\end{aligned}$$

where $\beta(y) := 1 - \int R(y, \mathrm{d}u)S(y, u; \mathrm{d}y')T(y, u, y'; \mathrm{d}u')\alpha(y, u, y', u')$. Thus, establishing π^*-reversibility of K amounts to verifying, for all f and g in $\mathcal{F}(\mathcal{Y})$,

$$\int f(y)g(y')\pi^*(\mathrm{d}y) \int R(y, \mathrm{d}u)S(y, u; \mathrm{d}y')T(y, u, y'; \mathrm{d}u')\alpha(y, u, y', u') = \int f(y)g(y')\pi^*(\mathrm{d}y') \int R(y', \mathrm{d}u')S(y', u'; \mathrm{d}y)T(y', u', y; \mathrm{d}u)\alpha(y', u', y, u). \quad \text{(IV.31)}$$

Indeed, by π-reversibility of $\{(Y_k^{(1)}, U_k^{(1)}); k \in \mathbb{N}\}$ it holds, for all $\bar{f}$ and $\bar{g}$ in $\mathcal{F}(\mathcal{Y} \otimes \mathcal{U})$,

$$\iint \bar{f}(y, u)\bar{g}(y', u')\pi(\mathrm{d}y \times \mathrm{d}u)S(y, u; \mathrm{d}y')T(y, u, y'; \mathrm{d}u')\alpha(y, u, y', u') = \iint \bar{f}(y, u)\bar{g}(y', u')\pi(\mathrm{d}y' \times \mathrm{d}u')S(y', u'; \mathrm{d}y)T(y', u', y; \mathrm{d}u)\alpha(y', u', y, u),$$

which establishes (IV.31) by letting $\bar{f}(y, u) = f(y)$ and $\bar{g}(y, u) = g(y)$. This completes the proof.

8 Appendix 2

8 .1 r-MCMC as a special case of Algorithm 9

As proposed initially by [NFW12], the r-MCMC algorithm generates a Markov chain $\{Y_k^{(2)}; k \in \mathbb{N}\}$ with transitions given by Algorithm 13 below. Denote by $\left|\frac{\partial f}{\partial u}(u)\right|$ the Jacobian determinant of a vector-valued transformation f. In this algorithm, f is any conti-

Algorithm 13 r-MCMC [NFW12]

Given $Y_k^{(2)} = y$,

(i) draw $\hat{Y} \sim \check{R}(y, \cdot) \rightsquigarrow \hat{y}$,

(ii) draw $U \sim \check{S}(y, \hat{y}; \cdot) \rightsquigarrow u$,

(iii) set

$$Y_{k+1}^{(2)} \leftarrow \begin{cases} \hat{y} & \text{w. pr. } \alpha^{(\mathrm{r})}(y, u, \hat{y}) := 1 \wedge \dfrac{\pi^*(\hat{y})\check{r}(\hat{y}, y)\check{s}(\hat{y}, y; f(u))}{\pi^*(y)\check{r}(y, \hat{y})\check{s}(y, \hat{y}; u)} \left|\dfrac{\partial f}{\partial u}(u)\right|, \\ y & \text{otherwise.} \end{cases} \quad \text{(IV.32)}$$

nuously differentiable involution on $\mathsf{U} = \mathbb{R}^d$. In addition, $\check{R}$ and $\check{S}$ are instrumental kernels on $(\mathsf{Y}, \mathcal{Y})$ and $(\mathsf{Y}^2, \mathcal{U})$, respectively, having transition densities $\check{r}$ and $\check{s}$ with respect to some dominating measure and Lebesgue measure on $\mathbb{R}^d$, respectively.

Proposition IV.31. The r-MCMC algorithm is a special case of Algorithm 9.

Proof. Since $\hat{Y}$ and U, obtained in Steps (i) and (ii) of Algorithm 13, are not drawn in the same order as in Algorithm 9, we first derive the expression of the corresponding kernels R and S, i.e.

$$R(y, \mathrm{d}u) = \left(\int \check{R}(y, \mathrm{d}\hat{y})\check{s}(y, \hat{y}; u)\right) \lambda_d(\mathrm{d}u) = r(y, u)\lambda_d(\mathrm{d}u),$$

$$S(y, u; \mathrm{d}\hat{y}) = \frac{\check{R}(y, \mathrm{d}\hat{y})\check{s}(y, \hat{y}; u)}{\int \check{R}(y, \mathrm{d}\hat{y})\check{s}(y, \hat{y}; u)},$$

where λ_d is Lebesgue measure on $\mathbb{R}^d$. Also note that

$$R(y, \mathrm{d}u)S(y, u; \mathrm{d}\hat{y}) = \check{R}(y, \mathrm{d}\hat{y})\check{s}(y, \hat{y}; u)\lambda_d(\mathrm{d}u). \tag{IV.33}$$

Moreover, introduce another auxiliary variable $\hat{U}$ taking values in U and being drawn according to $T(y, u, \hat{y}; \mathrm{d}\hat{u}) = \delta_{f(u)}(\mathrm{d}\hat{u})$. Note that the kernel T is not dominated by a common nonnegative measure regardless the value of u ; still, following Remark IV.11, the r-MCMC algorithm may be covered by Algorithm 9, provided that the ratio in the acceptance probability $\alpha^{(\mathrm{r})}(y, u, \hat{y})$ corresponds to the Radon-Nikodym derivative in Proposition IV.10 for

$$K^{(\mathrm{r})}(y, u; \mathrm{d}\hat{y} \times \mathrm{d}\hat{u}) = S(y, u; \mathrm{d}\hat{y})T(y, u, \hat{y}; \mathrm{d}\hat{u}) = S(y, u; \mathrm{d}\hat{y})\delta_{f(u)}(\mathrm{d}\hat{u})$$

and

$$\pi^{(\mathrm{r})}(\mathrm{d}y \times \mathrm{d}u) = \pi^*(\mathrm{d}y)R(y, \mathrm{d}u).$$

The proof is completed by applying Lemma IV.32 below. □

Lemma IV.32. The acceptance probability $\alpha^{(\mathrm{r})}$ in (IV.32) is equal to

$$\alpha^{(\mathrm{r})}(y, u, \hat{y}) = 1 \wedge \frac{\mathrm{d}\nu^{(\mathrm{r})}}{\mathrm{d}\mu^{(\mathrm{r})}}(x, \hat{x}), \tag{IV.34}$$

where $x := (y, u)$, $\hat{x} := (\hat{y}, \hat{u})$, and $\frac{\mathrm{d}\nu^{(\mathrm{r})}}{\mathrm{d}\mu^{(\mathrm{r})}}$ denotes the Radon-Nikodym derivative between the measures $\nu^{(\mathrm{r})}$ and $\mu^{(\mathrm{r})}$ defined by

$$\begin{aligned}
\nu^{(\mathrm{r})}(\mathrm{d}x \times \mathrm{d}\hat{x}) &:= \pi^{(\mathrm{r})}(\mathrm{d}\hat{y} \times \mathrm{d}\hat{u})K^{(\mathrm{r})}(\hat{y}, \hat{u}; \mathrm{d}y \times \mathrm{d}u), \\
\mu^{(\mathrm{r})}(\mathrm{d}x \times \mathrm{d}\hat{x}) &:= \pi^{(\mathrm{r})}(\mathrm{d}y \times \mathrm{d}u)K^{(\mathrm{r})}(y, u; \mathrm{d}\hat{y} \times \mathrm{d}\hat{u}).
\end{aligned}$$

Proof. Write $\alpha^{(\mathrm{r})}(y, u, \hat{y}) = 1 \wedge \gamma^{(\mathrm{r})}(y, u, \hat{y})$, where

$$\gamma^{(\mathrm{r})}(y, u, \hat{y}) := \frac{\pi^*(\hat{y})\check{r}(\hat{y}, y)\check{s}(\hat{y}, y; f(u))}{\pi^*(y)\check{r}(y, \hat{y})\check{s}(y, \hat{y}; u)} \left|\frac{\partial f}{\partial u}(u)\right|.$$

To show (IV.34) we will prove that for all bounded measurable functions G on $(\mathsf{Y} \times \mathsf{U})^2$ it holds that

$$\mathbb{E}_{\nu^{(\mathrm{r})}}[G(X, \hat{X})] = \int G(x, \hat{x})\nu^{(\mathrm{r})}(\mathrm{d}x \times \mathrm{d}\hat{x}) = \int G(x, \hat{x})\gamma^{(\mathrm{r})}(y, u, \hat{y})\mu^{(\mathrm{r})}(\mathrm{d}x \times \mathrm{d}\hat{x})$$

(where $x = (y, u)$ and $\hat{x} = (\hat{y}, \hat{u})$). Now, using the change of variables $u = f(\hat{u})$, which is equivalent to $\hat{u} = f(u)$ (since f is an involution) and using the relation (IV.33) we obtain

$$\begin{aligned}
&\mathbb{E}_{\nu^{(\mathrm{r})}}[G^{(\mathrm{r})}(X, \hat{X})] \\
&= \int G^{(\mathrm{r})}(y, f(\hat{u}), \hat{y}, \hat{u})\pi^*(\mathrm{d}\hat{y})r(\hat{y}, \hat{u})S(\hat{y}, \hat{u}; \mathrm{d}y)\lambda_d(\mathrm{d}\hat{u}), \\
&= \int G^{(\mathrm{r})}(y, u, \hat{y}, f(u))\pi^*(\mathrm{d}\hat{y})r(\hat{y}, f(u))S(\hat{y}, f(u); \mathrm{d}y)|(\partial f/\partial u)(u)|\lambda_d(\mathrm{d}u), \\
&= \int G^{(\mathrm{r})}(y, u, \hat{y}, f(u))\frac{\pi^*(\hat{y})\check{r}(\hat{y}, y)\check{s}(\hat{y}, y; f(u))}{\pi^*(y)\check{r}(y, \hat{y})\check{s}(y, \hat{y}; u)} \left|\frac{\partial f}{\partial u}(u)\right| \\
&\qquad\qquad \times \pi^*(\mathrm{d}y)\check{R}(y, \mathrm{d}\hat{y})\check{S}(y, \hat{y}; \mathrm{d}u) \\
&= \int G^{(\mathrm{r})}(x, \hat{x})\gamma^{(\mathrm{r})}(y, u, \hat{y})\mu^{(\mathrm{r})}(\mathrm{d}x \times \mathrm{d}\hat{x}),
\end{aligned}$$

which completes the proof. □

8 .2 GMTM as a special case of Algorithm 9

The GMTM algorithm proposed in [PBF10] generates a Markov chain $\{Y_k^{(2)}; k \in \mathbb{N}\}$ with transitions given by Algorithm 14 below. In Algorithm 14, the auxiliary variables

Algorithm 14 GMTM [PBF10]

Given $Y_k^{(2)} = y$,

(i) draw $(V_1, \dots, V_n) \sim_{\text{i.i.d.}} \check{R}(y, \cdot) \rightsquigarrow (v_1, \dots, v_n)$,

(ii) let J take the value $j \in \{1, 2, \dots, n\}$ w. p. $\omega(y, v_j)/\sum_{\ell=1}^n \omega(y, v_\ell)$,

(iii) let $\hat{y} \leftarrow v_j$,

(iv) draw $(\hat{V}_1, \dots, \hat{V}_{n-1}) \sim_{\text{i.i.d.}} \check{R}(\hat{y}, \cdot) \rightsquigarrow (\hat{v}_1, \dots, \hat{v}_{n-1})$,

(v) let $\hat{v}_n \leftarrow y$,

(vi) let

$$Y_{k+1}^{(2)} \leftarrow \begin{cases} \hat{y} & \text{with probability } \alpha^{(\mathrm{m})}(y, v, \hat{y}, \hat{v}) \\ & \qquad := 1 \wedge \dfrac{\pi^*(\hat{y})\check{r}(\hat{y}, y)\omega(\hat{y}, y)\sum_{k=1}^n \omega(y, v_k)}{\pi^*(y)\check{r}(y, \hat{y})\omega(y, \hat{y})\sum_{k=1}^n \omega(\hat{y}, \hat{v}_k)}, \\ y & \text{otherwise.} \end{cases} \tag{IV.35}$$

$V_1, \dots, V_n$ are defined on Y and for all $y \in \mathsf{Y}$ and $(v_1, \dots, v_n) \in \mathsf{Y}^n$, $\{\omega(y, v_k)/\sum_{\ell=1}^n \omega(y, v_\ell)\}_{k=1}^n$ are sample weights. Moreover, $\check{R}$ is an instrumental kernel defined on $(\mathsf{Y}, \mathcal{Y})$ having the transition density $\check{r}$ with respect to some dominating measure on $(\mathsf{Y}, \mathcal{Y})$.

Proposition IV.33. The GMTM algorithm is a special case of Algorithm 9.

Proof. Denoting by $V_1, \dots, V_n$ the random variables generated in Step (i) in Algorithm 14, the proposed candidate $\hat{Y}$ is obtained as V_J, where J is generated in Step (ii). Let $U = V_{-J}$, where

$$v_{-j} := (v_1, \dots, v_{j-1}, v_{j+1}, \dots, v_n).$$

To obtain the joint distribution of $(\hat{Y}, U)$ conditionally on $Y_k^{(2)}$, write, for any bounded measurable function G on Y^n,

$$\begin{aligned} &\mathbb{E}[G(\hat{Y}, U) \mid Y_k^{(2)} = y] \\ &= \sum_{j=1}^n \mathbb{E}[G(V_j, V_{-j})\mathbb{1}_{J=j} \mid Y_k^{(2)} = y] \\ &= \int \cdots \int \check{R}(y, \mathrm{d}\hat{y}) \prod_{k=1}^{n-1} \check{R}(y, \mathrm{d}u_k) \frac{n\omega(y, \hat{y})}{\sum_{\ell=1}^{n-1} \omega(y, u_\ell) + \omega(y, \hat{y})} G(\hat{y}, u) \\ &= \int \cdots \int R(y, \mathrm{d}u) S(y, u; \mathrm{d}\hat{y}) G(\hat{y}, u), \end{aligned}$$

where we introduced the kernels

$$R(y, \mathrm{d}u) := n \prod_{k=1}^{n-1} \check{R}(y, \mathrm{d}u_k) \int \frac{\check{R}(y, \mathrm{d}\hat{y})\omega(y, \hat{y})}{\sum_{\ell=1}^{n-1} \omega(y, u_\ell) + \omega(y, \hat{y})}, \tag{IV.36}$$

$$S(y, u; \mathrm{d}\hat{y}) := \frac{\check{R}(y, \mathrm{d}\hat{y})\omega(y, \hat{y})}{\sum_{\ell=1}^{n-1} \omega(y, u_\ell) + \omega(y, \hat{y})} \Bigg/ \int \frac{\check{R}(y, \mathrm{d}\hat{y})\omega(y, \hat{y})}{\sum_{\ell=1}^{n-1} \omega(y, u_\ell) + \omega(y, \hat{y})}. \tag{IV.37}$$

Now, set $\hat{U} = (\hat{V}_1, \dots, \hat{V}_{n-1})$ where the $\hat{V}_i$s are sampled in Step (iv). The distribution of $\hat{U}$ conditionally on $(Y_k^{(2)}, U, \hat{Y}) = (y, u, \hat{y})$ is given by

$$T(y, u, \hat{y}; \mathrm{d}\hat{u}) = \prod_{k=1}^{n-1} \check{R}(\hat{y}, \mathrm{d}\hat{u}_k). \tag{IV.38}$$

If $\check{R}$ is dominated by a nonnegative measure, then (IV.36), (IV.37), and (IV.38) show that the kernels R, S, and T are dominated as well. Denoting by r, s, and t the corresponding transition densities, it can be readily checked that

$$\frac{\pi^*(\hat{y}) r(\hat{y}, \hat{u}) s(\hat{y}, \hat{u}; y) t(\hat{y}, \hat{u}, y; u)}{\pi^*(y) r(y, u) s(y, u; \hat{y}) t(y, u, \hat{y}; \hat{u})} = \frac{\pi^*(\hat{y}) \check{r}(\hat{y}, y) \omega(\hat{y}, y) (\sum_{k=1}^{n-1} \omega(y, u_k) + \omega(y, \hat{y}))}{\pi^*(y) \check{r}(y, \hat{y}) \omega(y, \hat{y}) (\sum_{k=1}^{n-1} \omega(\hat{y}, \hat{u}_k) + \omega(\hat{y}, y))},$$

so that $\alpha^{(\mathrm{m})}$ defined in (IV.35) corresponds to the acceptance probability α defined in (IV.11) with these particular choices of r, s, and t. Consequently, the GMTM algorithm is a special case of Algorithm 9. □

Note that in the previous proof, we have chosen the auxiliary variable U as the vector of rejected candidates after Step (ii). Another natural idea would consist in choosing $U = (V_1, \dots, V_n)$, where the V_is are obtained in Step (i) ; however, since $\hat{Y}$ belongs to this set of candidates, the model would then not be dominated, which would make the proof more intricate.

8 .3 The Carlin and Chib Algorithm

Lemma IV.34. *P_{C} is a π^*-reversible Markov kernel.*

Proof. Recall that $\pi^*(m, \cdot)$ and $\{\rho_j\}_{j=1}^n$ are both dominated by the nonnegative measure ν. Then, using the notations $y = (m, z)$ and $\hat{y} = (\hat{m}, \hat{z})$ we have for all bounded measurable functions G on $\mathsf{Y} \times \mathsf{Y}$,

$$\begin{aligned}
&\int G(y, \hat{y})\, \pi^*(\mathrm{d}y)\, P_{\mathrm{C}}(y, \mathrm{d}\hat{y}) \\
&\quad = \int G(y, \hat{y})\, \pi^*(\mathrm{d}y) \left[\delta_z(\mathrm{d}u_m) \left(\prod_{j \neq m} \rho_j(\mathrm{d}u_j) \right) \right. \\
&\qquad\qquad \left. \frac{\pi^*(\hat{m}, u_{\hat{m}}) \prod_{j \neq \hat{m}} \rho_j(u_j)}{\sum_{k=1}^n \pi^*(k, u_k) \prod_{\ell \neq k} \rho_\ell(u_\ell)} \right] \delta_{u_{\hat{m}}}(\mathrm{d}\hat{z}), \\
&\quad = \int G(y, \hat{y}) A(m, \hat{m}, u)\, \nu(\mathrm{d}z)\, \delta_z(\mathrm{d}u_m)\, \delta_{u_{\hat{m}}}(\mathrm{d}\hat{z}) \prod_{j \neq m} \nu(\mathrm{d}u_j),
\end{aligned}$$

where

$$A(m, \hat{m}, u) := \pi^*(m, u_m) \pi^*(\hat{m}, u_{\hat{m}}) \frac{\prod_{j \neq m} \rho_j(u_j) \times \prod_{j \neq \hat{m}} \rho_j(u_j)}{\sum_{k=1}^n \pi^*(k, u_k) \prod_{\ell \neq k} \rho_\ell(u_\ell)}.$$

Plugging

$$\begin{aligned}
&A(m, \hat{m}, u) = A(\hat{m}, m, u), \\
&\nu(\mathrm{d}z)\, \delta_z(\mathrm{d}u_m)\, \delta_{u_{\hat{m}}}(\mathrm{d}\hat{z}) \prod_{j \neq m} \nu(\mathrm{d}u_j) = \delta_{u_m}(\mathrm{d}z)\, \delta_{u_{\hat{m}}}(\mathrm{d}\hat{z}) \prod_{j=1}^n \nu(\mathrm{d}u_j)
\end{aligned}$$

into the previous display shows that $\pi^*(\mathrm{d}y)\, P_{\mathrm{C}}(y, \mathrm{d}\hat{y})$ is a symmetric measure in y and $\hat{y}$, i.e. P_{C} is π^*-reversible. □

Conclusion

En dépit de la faible résolution des SIR d'aéronefs et du niveau de bruit élevé dû au fond, inhérents à notre cadre d'étude, notre travail a montré qu'il était possible de détecter la présence d'aéronefs dans une image multispectrale et de les classer parmi plusieurs catégories. Le principal défi qui consistait à prendre en compte simultanément la variabilité spectrale et spatiale des cibles a été relevé (i) en couplant au détecteur d'anomalies RX [RY90] une étude sur les ensembles de niveau de la transformée de Mahalanobis de l'image et (ii) en proposant un modèle d'observation dans lequel la distribution des SIR est la marginale d'une loi jointe faisant intervenir des déformations spatiales et spectrales aléatoires dont les réalisations ne sont pas observées. Les méthodes que nous avons mises au point respectent les impératifs liés au cadre expérimental de notre étude, à savoir un traitement séquentiel non supervisé des données, sans stockage aposteriori et en un temps suffisamment court pour permettre une implémentation de la solution en temps réel. Les simulations réalisées montrent que, comparé au monospectral, l'utilisation d'images multispectrales permet d'obtenir de meilleures performances tant en détection qu'en classification. L'explication de ce résultat se trouve essentiellement à deux niveaux : en détection, des anomalies présentes uniquement dans certaines bandes spectrales et non visibles en monospectral peuvent ainsi être identifiées et en classification, l'utilisation d'images multispectrales offre la possibilité de considérer des regroupements de bandes étroites où le rapport signal à bruit est plus faible qu'en bande large. Toutefois, ces performances sont fortement sensibles aux bandes spectrales retenues pour la détection et la classification. Pour remédier à ce problème, nous avons donc proposé une méthode d'optimisation, basée sur un algorithme génétique, identifiant le regroupement de bandes spectrales permettant d'atteindre le taux de détection optimal.

La méthodologie de détection proposée peut aisément s'adapter à d'autres scénarios et à d'autres sortes de cibles faiblement résolues sur d'autres fonds naturels de type forêt, désert, mer... Une piste d'amélioration consisterait à utiliser un estimateur de la matrice de covariance plus robuste aux perturbations et à un modèle de fond non-Gaussien, comme le permet par exemple un M-estimateur de la matrice de covariance [Hub64, Mar76]. La méthodologie de classification a été testée avec succès sur des images de chiffres manuscrits et sur des courbes de croissance, montrant ainsi qu'il était possible d'étendre cette approche sur d'autres types de données fonctionnelles. Travailler sur des données ayant une meilleure résolution spatiale et spectrale conduirait à de meilleures performances de classification dans la mesure où la méthode d'apprentissage développée exploite les informations spectrale et spatiale pour caractériser les différentes composantes du modèle de mélange. Toutefois, sous l'hypothèse que la dimension des données manquantes du modèle est proportionnelle à la taille des observations, l'utilisation de données fortement résolues ralentit le processus d'apprentissage et de classification : il y a donc un compromis à trouver. Les méthodes de détection et de classification doivent à présent être validées dans le cas d'un fond texturé multispectral : des mesures de spectres infrarouge de nuages sont envisagées

en vue de permettre l'élaboration d'un modèle de fond texturé corrélant les variations spectrales et spatiales. Enfin, il serait intéressant de disposer d'une méthode de recherche des bandes spectrales optimales pour la classification afin de les comparer avec les bandes spectrales optimales pour la détection et le cas échéant de trouver un compromis entre les deux. Ces éléments permettraient de spécifier les caractéristiques d'un capteur infrarouge multispectral, qui, couplé à nos méthodes de détection et de classification, serait capable de protéger un site sensible.

Au delà de la mise en oeuvre de ces solutions spécifiques au problème de détection et de classification initialement posé, cette thèse a également été l'occasion de réfléchir à des problématiques méthodologiques variées. De ce point de vue, notre contribution sur l'estimation séquentielle des paramètres dans des modèles à données manquantes a consisté à généraliser l'algorithme Online EM [CM07] à des situations où l'espérance conditionnelle est approchée par des méthodes de Monte Carlo par chaînes de Markov : le Monte Carlo online EM (MCoEM). Comparé à des méthodes similaires en bloc telles que l'algorithme SAEM [DLM99], le traitement séquentiel des données rend l'algorithme davantage sensible au bruit engendré par la méthode MCMC puisque la mise à jour des paramètres est réalisée après chaque nouvelle observation. Une attention particulière a par conséquent été apportée au choix de l'échantillonneur des données manquantes sous la loi aposteriori. Dans le cas des modèles de mélange, la solution proposée consiste à combiner l'algorithme MCoEM pour l'estimation des paramètres avec l'échantillonneur de Carlin et Chib [CC95] pour la simulation des données manquantes. Nous apportons une justification théorique quant à l'efficacité asymptotique de cet échantillonneur en démontrant une extension de l'ordre de Peskun [Tie95] à des chaînes inhomogènes. Sous certaines conditions, ce théorème nous permet d'affirmer que l'algorithme de Carlin et Chib possède une variance asymptotique plus faible que l'échantillonneur de Gibbs, autre méthode MCMC habituellement utilisée dans le cas de variables en grandes dimensions.

Une preuve de la convergence de la séquence d'estimateurs produite par le MCoEM vers un espace où la divergence de Kullback-Leibler entre la distribution du modèle et celle des observations est stationnaire, est actuellement en cours de réalisation. L'objectif sera de trouver des hypothèses relatives à la méthode d'approximation de l'espérance pour que la convergence du MCoEM soit similaire à celle de l'Online EM. Enfin, une autre perspective à explorer concerne la généralisation du Théorème IV.4 de comparaison établi pour des chaînes de Markov inhomogènes alternant entre deux noyaux π-réversibles à des chaînes de Markov alternant entre n noyaux π-réversibles. Un tel résultat permettrait l'étude de nombreuses méthodes MCMC parmi lesquelles les algorithmes de type Metropolis-within-Gibbs.

Bibliographie

[AAT07] S. Allassonnière, Y. Amit et A. Trouvé : Towards a coherent statistical framework for dense deformable template estimation. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 69(1):3–29, 2007.

[ADFDJ03] C. Andrieu, N. De Freitas, A. Doucet et M. I. Jordan : An introduction to MCMC for machine learning. *Machine Learning*, 50, No :1-2:5–43, 2003.

[ADH10] C. Andrieu, A. Doucet et R. Holenstein : Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 72(3):269–342, 2010.

[AFR97] E. Aleskerov, B. Freisleben et B. Rao : Cardwatch : A neural network based database mining system for credit card fraud detection. *In Computational Intelligence for Financial Engineering (CIFEr), 1997., Proceedings of the IEEE/IAFE 1997*, pages 220–226. IEEE, 1997.

[AGP91] Y. Amit, U. Grenander et M. Piccioni : Structural image restoration through deformable templates. *Journal of the American Statistical Association*, 86(414):376–387, 1991.

[AK10] S. Allassonnière et E. Kuhn : Stochastic Algorithm For Parameter Estimation For Dense Deformable Template Mixture Model. *ESAIM-PS*, 14:382–408, 2010.

[AKT10a] S. Allassonnière, E. Kuhn et A. Trouvé : Construction of Bayesian deformable models via a stochastic approximation algorithm : a convergence study. *Bernoulli*, 16(3):641–678, 2010.

[AKT10b] Stéphanie Allassonnière, Estelle Kuhn et Alain Trouvé : Models using stochastic algorithms : Applications to medical images. *Journal de la Société Française de Statistique*, 151(1):1–16, 2010.

[AM06] C. Andrieu et É. Moulines : On the ergodicity properties of some adaptive MCMC algorithms. *The Annals of Applied Probability*, 16(3): 1462–1505, 2006.

[Amb03] T. Ambwani : Multi class support vector machine implementation to intrusion detection. *In Neural Networks, 2003. Proceedings of the International Joint Conference on*, volume 3, pages 2300–2305. IEEE, 2003.

[AMP05] C. Andrieu, E. Moulines et P. Priouret : Stability of stochastic approximation under verifiable conditions. *SIAM Journal on control and optimization*, 44(1):283–312, 2005.

[AR05] Y. F. Atchadé et J. S. Rosenthal : On adaptive Markov chain Monte Carlo algorithms. *Bernoulli*, 11:815–828, 2005.

[AR09] C. ANDRIEU et G. O. ROBERTS : The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37(2):697–725, 2009.

[Ash07] J. ASHBURNER : A fast diffeomorphic image registration algorithm. *Neuroimage*, 38(1):95–113, 2007.

[AV12] C. ANDRIEU et M. VIHOLA : Convergence properties of pseudo-marginal Markov chain Monte Carlo algorithms. *arXiv preprint arXiv :1210.1484*, 2012.

[BAR+11] D. BORGHYS, V. ACHARD, S. R. ROTMAN, N. GORELIK, C. PERNEEL et E. SCHWEICHER : Hyperspectral anomaly detection : A comparative evaluation of methods. *In Proc. of IEEE URSI GASS*, pages 1– 4, 2011.

[BC11] J. BIGOT et B. CHARLIER : On the consistency of fréchet means in deformable models for curve and image analysis. *Electronic Journal of Statistics*, 5:1054–1089, 2011.

[Bea03] M. A. BEAUMONT : Estimation of population growth or decline in genetically monitored populations. *Genetics*, 164(3):1139–1160, 2003.

[BG13] J. BIGOT et X. GENDRE : Minimax properties of fréchet means of discretely sampled curves. *The Annals of Statistics*, 41(2):923–956, 2013.

[BGL09] J. BIGOT, S. GADAT et J-M. LOUBES : Statistical m-estimation and consistency in large deformable models for image warping. *Journal of Mathematical Imaging and Vision*, 34(3):270–290, 2009.

[BH99] J. G. BOOTH et J. P. HOBERT : Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 61(1):265–285, 1999.

[Big11] J. BIGOT : Fréchet means of curves for signal averaging and application to ecg data analysis. *arXiv preprint arXiv :1111.1855*, 2011.

[BMTY05] M. F. BEG, M. I. MILLER, A. TROUVÉ et L. YOUNES : Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International journal of computer vision*, 61(2):139–157, 2005.

[Bod09] S. BODA : *Feature-based image registration.* Thèse de doctorat, 2009.

[Boo97] F. L. BOOKSTEIN : *Morphometric tools for landmark data : geometry and biology.* Cambridge University Press, 1997.

[BPS05] J. A. BENEDIKTSSON, J. A. PALMASON et J. R. SVEINSSON : Classification of hyperspectral data from urban areas based on extended morphological profiles. *Geoscience and Remote Sensing, IEEE Transactions on*, 43(3):480–491, 2005.

[Bro92] L. G. BROWN : A survey of image registration techniques. *ACM computing surveys (CSUR)*, 24(4):325–376, 1992.

[BSH00] S. G. BEAVEN, D. STEIN et L. HOFF : Comparison of gaussian mixture and linear mixture models for classification of hyperspectral data. *In Geoscience and Remote Sensing Symposium, 2000. Proceedings. IGARSS 2000. IEEE 2000 International*, volume 4, pages 1597–1599. IEEE, 2000.

[BWJ01] D. BARBARA, N. WU et S. JAJODIA : Detecting novel network intrusions using bayes estimators. *In First SIAM Conference on Data Mining.* Citeseer, 2001.

[CAM04] N. P. CASTELLANOS, P. L. D. ANGEL et V. MEDINA : Nonrigid medical image registration technique as a composition of local warpings. *Pattern Recognition*, 37(11):2141–2154, 2004.

[CB05] P-J. CHUNG et J. F. BOHME : Recursive EM and SAGE-inspired algorithms with application to DOA estimation. *Signal Processing, IEEE Transactions on*, 53(8):2664–2677, 2005.

[CBK09] V. CHANDOLA, A. BANERJEE et V. KUMAR : Anomaly detection : A survey. *ACM Computing Surveys (CSUR)*, 41(3):15, 2009.

[CC95] B. P. CARLIN et S. CHIB : Bayesian model choice via Markov chain Monte Carlo. *J. R. Statist. Soc.B*, 57:473–484, 1995.

[CC02] C. I. CHANG et S. S. CHIANG : Anomaly detection and classification for hyperspectral imagery. *IEEE Trans. Geosci. Remote Sensing*, 40:1314 –1325, 2002.

[CCM06] O. CAPPÉ, M. CHARBIT et E. MOULINES : Recursive EM algorithm with applications to DOA estimation. *In Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 3, pages III–III. IEEE, 2006.

[CD85] G. CELEUX et J. DIEBOLT : The SEM algorithm : a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. *Computational statistics quarterly*, 2(1):73–82, 1985.

[CG95] S. CHIB et E. GREENBERG : Understanding the Metropolis-Hastings algorithm. *The American Statistician*, 49(4):327–335, 1995.

[CGG88] H. CHEN, L. GUO et A. GAO : Convergence and robustness of the Robbins-Monro algorithm truncated at randomly varying bounds. *Stochastic Processes and their Applications*, 27:217–231, 1988.

[Cha11a] B. CHARLIER : *Etude des propriétés statistiques des moyennes de Fréchet dans des modèles de déformations pour l'analyse de courbes et d'images en grande dimension.* Thèse de doctorat, Université de Toulouse, Université Toulouse III-Paul Sabatier, 2011.

[Cha11b] B. CHARLIER : Necessary and sufficient condition for the existence of a fréchet mean on the circle. *arXiv preprint arXiv :1109.1986*, 2011.

[Chr99] G. E. CHRISTENSEN : Consistent linear-elastic transformations for image matching. *In Information processing in medical imaging*, pages 224–237, 1999.

[CLM95] J-F. CARDOSO, M. LAVIELLE et E. MOULINES : Un algorithme d'identification par maximum de vraisemblance pour des données incomplètes. *Comptes rendus de l'Académie des sciences. Série 1, Mathématique*, 320(3):363–368, 1995.

[CM07] O. CAPPÉ et E. MOULINES : Online EM Algorithm for Latent Data Models. *J. R. Statist. Soc.B*, 71:593–613, 2007.

[CMR05] O. CAPPÉ, E. MOULINES et T. RYDÉN : *Inference in hidden Markov models.* Springer, 2005.

[CR99] G. CASELLA et C.P. ROBERT : *Monte Carlo statistical methods.* Springer-Verlag, New York, 1999.

[CRM96] G. E. CHRISTENSEN, R. D. RABBITT et M. I. MILLER : Deformable templates using large deformation kinematics. *Image Processing, IEEE Transactions on*, 5(10):1435–1447, 1996.

[CVGCMM+06] G. CAMPS-VALLS, L. GOMEZ-CHOVA, J. MUÑOZ-MARÍ, J. VILA-FRANCÉS et J. CALPE-MARAVILLA : Composite kernels for hyperspectral image classification. *Geoscience and Remote Sensing Letters, IEEE*, 3(1):93–97, 2006.

[d'A63] W. T. D'ARCY : *On growth and form*. Cambridge : University Press ; New York : Macmillan, 1963.

[DFN02] P. DELLAPORTAS, J. J. FORSTER et I. NTZOUFRAS : On Bayesian model and variable selection using MCMC. *Statistics and Computing*, 12(1):27–36, 2002.

[DG02] M. DAVY et S. GODSILL : Detection of abrupt spectral changes using support vector machines an application to audio signal segmentation. *In Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, volume 2, pages II–1313. IEEE, 2002.

[DI96] J DIEBOLT et E. H. S. IP : A stochastic EM algorithm for approximating the maximum likelihood estimate. *Markov chain Monte Carlo in practice*, 1996.

[DJC98] M. J. DESFORGES, P. J. JACOB et J. E. COOPER : Applications of probability density estimation to the detection of abnormal conditions in engineering. *Proceedings of the Institution of Mechanical Engineers, Part C : Journal of Mechanical Engineering Science*, 212(8):687–703, 1998.

[DLJ04] B. DAVIS, P. LORENZEN et S. C. JOSHI : Large deformation minimum mean squared error template estimation for computational anatomy. *In ISBI*, volume 4, pages 173–176, 2004.

[DLM99] B. DELYON, M. LAVIELLE et E. MOULINES : Convergence of a stochastic approximation version of the EM algorithm. *Annals of Statistics*, pages 94–128, 1999.

[DLR77] A. P. DEMPSTER, N. M. LAIRD et D. B. RUBIN : Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

[DMM00] A. DESOLNEUX, L. MOISAN et J-M. MOREL : Meaningful alignments. *International Journal of Computer Vision*, 40(1):7–23, 2000.

[DMM01] A. DESOLNEUX, L. MOISAN et J-M. MOREL : Edge detection by helmholtz principle. *Journal of Mathematical Imaging and Vision*, 14(3):271–284, 2001.

[DMM03] A. DESOLNEUX, L. MOISAN et J-M. MOREL : A grouping principle and four applications. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(4):508–513, 2003.

[DPH00] J. R. DELLER, J. G. PROAKIS et J. H. L. HANSEN : *Discrete-time processing of speech signals*. IEEE New York, NY, USA :, 2000.

[DPK05] F. DIBOS, S. PELLETIER et G. KOEPFLER : Real-time segmentation of moving objects in a video sequence by a contrario detection. *In Image Processing, 2005. ICIP 2005. IEEE International Conference on*, volume 1, pages I–1065. IEEE, 2005.

[DS10] B. J. DANIEL et A. P. SCHAUM : Urchin : An RX-derivative accounting for anisotropies in whitened clutter. *In Proc. SPIE*, volume 7695, page 769504, 2010.

[DSSV00] C. DE STEFANO, C. SANSONE et M. VENTO : To reject or not to reject : that is the question-an answer in case of neural classifiers. *Systems, Man, and Cybernetics, Part C : Applications and Reviews, IEEE Transactions on*, 30(1):84–94, 2000.

[Esk00] E. ESKIN : Anomaly detection over noisy data using learned probability distributions. *In In Proceedings of the International Conference on Machine Learning*, pages 255–262, 2000.

[Fau07] M. FAUVEL : *Spectral and spatial methods for the classification of urban remote sensing data.* Thèse de doctorat, 2007.

[FJ03] B.J. FREY et N. JOJIC : Transformation-invariant clustering using the EM alogrithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003.

[Fle87] R. FLETCHER : *Practical Methods of Optimization.* Wiley, 1987.

[FM03] G. FORT et E. MOULINES : Convergence of the Monte Carlo expectation maximization for curved exponential families. *The Annals of Statistics*, 31(4):1220–1259, 2003.

[FP12] P. FEARNHEAD et D. PRANGLE : Constructing summary statistics for approximate Bayesian computation : semi-automatic approximate Bayesian computation. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 74(3):419–474, 2012.

[Fré48] M. FRÉCHET : Les éléments aléatoires de nature quelconque dans un espace distancié. *In Annales de l'institut Henri Poincaré*, volume 10, pages 215–310. Presses universitaires de France, 1948.

[Gau81] G. GAUFFRE : Aircraft infrared radiation modeling. *La Recherche Aerospatiale, July-Aug. 1981, p. 245-265. In French.*, 1:245–265, 1981.

[GG84] S. GEMAN et D. GEMAN : Stochastic relaxation, Gibbs distributions, and the Bayesian restauration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):721–741, 1984.

[GJ06] J. A. GLAUNÈS et S. JOSHI : Template estimation form unlabeled point set data and surfaces for computational anatomy. *In Proceedings of the First International Workshop on Mathematical Foundations of Computational Anatomy-Geometrical and Statistical Methods for Modelling Biological Shape Variability*, 2006.

[GLM07] F. GAMBOA, J-M. LOUBES et E. MAZA : Semi-parametric estimation of shifts. *Electronic Journal of Statistics*, 1:616–640, 2007.

[GM96] C. A. GLASBEY et N. J. MARTIN : Multimodal microscopy by digital image processing. *Journal of Microscopy*, 181(3):225–237, 1996.

[GM09] B. GROSJEAN et L. MOISAN : A-contrario detectability of spots in textured backgrounds. *Journal of Mathematical Imaging and Vision*, 33(3):313–337, 2009.

[Gol89] D.E. GOLDBERG : *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison Wesley, 1989.

[Goo91] C. GOODALL : Procrustes methods in the statistical analysis of shape. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 285–339, 1991.

[Gos05] A. A. GOSHTASBY : *2-D and 3-D image registration : for medical, remote sensing, and industrial applications*. Wiley, 2005.

[Gre93] U. GRENANDER : *General pattern theory : A mathematical study of regular structures*. Clarendon Press Oxford, 1993.

[Gre07] R. L. GREGORY : Helmholtz's principle. *Perception*, 36(6):795–796, 2007.

[GRS96] W. R. GILKS, S. RICHARDSON et D. J. SPIEGELHALTER : *Markov chain Monte Carlo in practice*, volume 2. CRC press, 1996.

[GS04] S. GAFFNEY et P. SMYTH : Joint probabilistic curve clustering and alignment. *In In Advances in Neural Information Processing Systems 17*, 2004.

[Har75] J. A. HARTIGAN : *Clustering algorithms*. John Wiley & Sons, Inc., 1975.

[Has70] W. K. HASTINGS : Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[HCYW96] L. E. HOFF, A. M. CHEN, X. YU et E. M. WINTER : Enhanced classification performance from multiband infrared imagery. *IEEE Proceedings of ASILOMAR-29*, pages 837 –841, 1996.

[HG09] B. A. S. HASAN et J. Q. GAN : Sequential EM for unsupervised adaptive Gaussian mixture model based classifier. *In Machine Learning and Data Mining in Pattern Recognition*, pages 96–106. Springer, 2009.

[HM11] M. HAIRER et J. C. MATTINGLY : Yet another look at Harris ergodic theorem for Markov chains. *In Seminar on Stochastic Analysis, Random Fields and Applications VI*, pages 109–117. Springer, 2011.

[HR07] O. HÄGGSTRÖM et J. S. ROSENTHAL : On variance conditions for Markov chain CLTs. *Elect. Comm. in Probab.*, 12:454–464, 2007.

[Hub64] P. J. HUBER : Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1):73–101, 1964.

[Hul94] J. J. HULL : A database for handwritten text recognition research. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(5): 550–554, 1994.

[HZ00] N. E. HECKMAN et R. H. ZAMAR : Comparing the shapes of regression functions. *Biometrika*, 87(1):135–144, 2000.

[JD06] M. JOHANSSON et M. DALENBRING : SIGGE, a prediction tool for aeronautical IR signatures, and its applications. *In 9th AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, volume 3276, 2006.

[JM00] S. C. JOSHI et M. I. MILLER : Landmark matching via large deformation diffeomorphisms. *Image Processing, IEEE Transactions on*, 9(8):1357–1370, 2000.

[Jol05] I. JOLLIFFE : *Principal component analysis*. Wiley Online Library, 2005.

[JR95] J-J. JACQ et C. ROUX : Registration of 3-D images by genetic optimization. *Pattern Recognition Letters*, 16(8):823–841, 1995.

[Kay98] S. M. KAY : *Fundamentals of Statistical signal processing, Volume 2 : Detection theory*. Prentice Hall PTR, 1998.

[KG92] A. KNEIP et T. GASSER : Statistical tools to analyze data representing a sample of curves. *The Annals of Statistics*, 20(3):1266–1305, 1992.

[KL04] E. KUHN et M. LAVIELLE : Coupling a stochastic approximation version of EM with an MCMC procedure. *ESAIM : Probability and Statistics*, 8(1):115–131, 2004.

[KL05] E. KUHN et M. LAVIELLE : Maximum likelihood estimation in nonlinear mixed effects models. *Computational Statistics & Data Analysis*, 49(4): 1020–1038, 2005.

[KN05] H. KWON et N. M. NASRABADI : Kernel RX-algorithm : a nonlinear anomaly detector for hyperspectral imagery. *Geoscience and Remote Sensing, IEEE Transactions on*, 43(2):388 – 397, feb. 2005.

[KR02] J. KARLHOLM et I. RENHORN : Wavelength band selection method for-multispectral target detection. *Applied Optics*, 41:6786 –6795, 2002.

[KTAT03] K. KADOTA, D. TOMINAGA, Y. AKIYAMA et K. TAKAHASHI : Detecting outlying samples in microarray data : A critical assessment of the effect of outliers on sample classification. *Chem-Bio Informatics*, 3(1):30–45, 2003.

[KTKPB02] P. KAEW TRA KUL PONG et R. BOWDEN : An improved adaptive background mixture model for real-time tracking with shadow detection. *In Video-Based Surveillance Systems*, pages 135–144. Springer, 2002.

[KTNU00] J. KYBIC, P. THÉVENAZ, A. NIRKKO et M. UNSER : Unwarping of unidirectionally distorted epi images. *Medical Imaging, IEEE Transactions on*, 19(2):80–93, 2000.

[KV86] C. KIPNIS et S. R. S. VARADHAN : Central limit theorem for additive functionals of reversible Markov processes and applications to simple exclusions. *Communications in Mathematical Physics*, 104(1):1–19, 1986.

[LACM06] Z. LIU, J. ALMHANA, V. CHOULAKIAN et R. MCGORMAN : Online EM algorithm for mixture with application to internet traffic modeling. *Computational statistics & data analysis*, 50(4):1052–1071, 2006.

[LAD12] A. LEE, C. ANDRIEU et A. DOUCET : Discussion of a paper by P. Fearnhead and D. Prangle. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 74(3):419–474, 2012.

[LAJ+12] S. LEFEBVRE, S. ALLASSONNIÈRE, J. JAKUBOWICZ, T. LASNE et É. MOULINES : Aircraft classification with a low resolution infrared sensor. *Machine Vision and Application Journal*, 24(1):175–186, 2012.

[Lan95] K. LANGE : A gradient algorithm locally equivalent to the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 425–437, 1995.

[Lég00] S. LÉGER : *Analyse stochastique de signaux multi-fractaux et estimations de paramètres*. Thèse de doctorat, Université d'Orléans, 2000.

[LEK+03] A. LAZAREVIC, L. ERTOZ, V. KUMAR, A. OZGUR et J. SRIVASTAVA : A comparative study of anomaly detection schemes in network intrusion detection. *In Proceedings of the third SIAM international conference on data mining*, volume 3, pages 25–36. Siam, 2003.

[LF04] R. A. LEVINE et J. FAN : An automated (Markov chain) Monte Carlo em algorithm. *Journal of Statistical Computation and Simulation*, 74(5):349–360, 2004.

[LGW04] Q. LI, C. N. GEORGHIADES et X. WANG : Blind multiuser detection in uplink CDMA with multipath fading : A sequential EM approach. *Communications, IEEE Transactions on*, 52(1):71–81, 2004.

[LLW00] J. S. LIU, F. LIANG et W. H. WONG : The Multiple-try method and local optimization in Metropolis sampling. *Journal of the American Statistical Association*, 95(449):121–134, 2000.

[LMM95] H. LI, B. S. MANJUNATH et S. K. MITRA : A contour-based approach to multisensor image registration. *Image Processing, IEEE Transactions on*, 4(3):320–334, 1995.

[LPP02] W. LEIGH, M. PAZ et R. PURVIS : An analysis of a hybrid neural network and pattern recognition technique for predicting short-term increases in the nyse composite index. *Omega*, 30(2):69–76, 2002.

[ŁRR13] K. ŁATUSZYŃSKI, G. O. ROBERTS et J. S. ROSENTHAL : Adaptive Gibbs samplers and related MCMC methods. *The Annals of Applied Probability*, 23(1):66–98, 2013.

[LRVD10a] S. LEFEBVRE, A. ROBLIN, S. VARET et G. DURAND : Metamodeling of aircraft infrared signature dispersion. *AStA Advances in Statistical Analysis*, 94(4):405–422, 2010.

[LRVD10b] S. LEFEBVRE, A. ROBLIN, S. VARET et G. DURAND : A methodological approach for statistical evaluation of aircraft infrared signature. *Reliability Engineering & System Safety*, 95(5):484–493, 2010.

[LT07] M. LI et J. TIAN : Anomaly detection for hyperspectral images based on improved RX algorithm. *Proc. SPIE*, 6787, 2007.

[Lue03] D. G. LUENBERGER : *Linear and nonlinear programming*. Springer, 2003.

[LWK94] J. S. LIU, W. H. WONG et A KONG : Covariance structure and convergence rate of the Gibbs sampler with various scans. *Biometrika*, 81(1):27–40, 1994.

[LY09] X. LIU et M. C. K. YANG : Simultaneous curve registration and clustering for functional data. *Computational Statistics & Data Analysis*, 53(4): 1361–1376, 2009.

[Man05] D. MANOLAKIS : Taxonomy of detection algorithms for hyperspectral imaging applications. *Optical Engineering*, 44(6):066403–066403, 2005.

[Mar76] R. A. MARONNA : Robust M-estimators of multivariate location and scatter. *The annals of statistics*, pages 51–67, 1976.

[Mat75] G. MATHERON : *Random sets and integral geometry*. New York : John Wiley & Sons, 1975.

[MC94] W. J. MOROKOFF et R. E. CAFLISCH : Quasi-random sequences and their discrepancies. *SIAM Journal on Scientific Computing*, 15(6):1251–1279, 1994.

[MDC10] S. MATTEOLI, M. DIANI et G. CORSINI : A tutorial overview of anomaly detection in hyperspectral images. *Aerospace and Electronic Systems Magazine, IEEE*, 25(7):5–28, 2010.

[MG99] A. MIRA et C. J. GEYER : Ordering Monte Carlo Markov chains. *School of Statistics, University of Minnesota. technical report*, 1999.

[MG00] P. MONASSE et F. GUICHARD : Fast computation of a contrast invariant image representation. *IEEE Transactions on Image Processing*, 9:860–872, 2000.

[Mir01] A. MIRA : Ordering and improving the performance of monte carlo markov chains. *Statistical Science*, pages 340–350, 2001.

[MK07] G. MCLACHLAN et T. KRISHNAN : *The EM algorithm and extensions*, volume 382. John Wiley & Sons, 2007.

[ML06] M. L. MARX et R. J. LARSEN : *Introduction to mathematical statistics and its applications*. Pearson/Prentice Hall, 2006.

[ML09] A. MIRA et F. LEISEN : Covariance ordering for discrete and continuous time Markov chains. *Statistica Sinica*, 19(2):651, 2009.

[MM88] R. H. MYERS et D. C. MONTGOMERY : Response surface methodology, 1988.

[MM98] S. MASNOU et J. M. MOREL : Level lines based disocclusion. *In Proc. of IEEE ICIP*, volume 3, pages 259 –263, 1998.

[MMS03] D. MANOLAKIS, D. MARDEN et G. A. SHAW : Hyperspectral image processing for automatic target detection applications. *Lincoln Laboratory Journal*, 14(1):79–116, 2003.

[MMTY08] J. MA, M. I. MILLER, A. TROUVÉ et L. YOUNES : Bayesian template estimation in computational anatomy. *NeuroImage*, 42(1):252–261, 2008.

[MR93] X-L. MENG et D. B. RUBIN : Maximum likelihood estimation via the ECM algorithm : A general framework. *Biometrika*, 80(2):267–278, 1993.

[MRBL08] P. R. MCCULLOUGH, M. REGAN, L. BERGERON et K. LINDSAY : Quantum efficiency and quantum yield of an hgcdte infrared sensor array. *Quantum*, 120(869):759–776, 2008.

[MRR+53] N. METROPOLIS, A. W. ROSENBLUTH, M. N. ROSENBLUTH, A. H. TELLER et E. TELLER : Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21:1087, 1953.

[MS02] D. MANOLAKIS et G. A. SHAW : Detection algorithms for hyperspectral imaging applications. *Signal Processing Magazine, IEEE*, 19(1):29–43, 2002.

[MT96] T. MCINERNEY et D. TERZOPOULOS : Deformable models in medical image analysis : a survey. *Medical image analysis*, 1(2):91–108, 1996.

[MT09] S. S. P. MEYN et R. L. TWEEDIE : *Markov chains and stochastic stability*. Cambridge University Press, 2009.

[MZHS08] F. MEI, C. ZHAO, H. HUO et Y. SUN : An adaptive kernel method for anomaly detection in hyperspectral imagery. *In Intelligent Information Technology Application, 2008. IITA'08. Second International Symposium on*, volume 1, pages 874–878. IEEE, 2008.

[NFW12] G. K. NICHOLLS, C. FOX et A. M. WATT : Coupled MCMC with a randomized acceptance probability. *arXiv preprint arXiv :1205.6857*, 2012.

[NH98] R. M. NEAL et G. E. HINTON : A view of the EM algorithm that justifies incremental, sparse, and other variants. *In Learning in graphical models*, pages 355–368. Springer, 1998.

[NKSS91] M. NOAH, J. KRISTL, J. SCHROEDER et B. P. SANDFORD : NIRATAM-NATO infrared air target model. *In Surveillance Technologies, Proceedings of SPIE*, volume 1479, pages 275– 282, 1991.

[NP92] J. NEYMAN et E. S. PEARSON : *On the problem of the most efficient tests of statistical hypotheses*. Springer, 1992.

[OP03] S. OSHER et N. PARAGIOS : *Geometric level set methods in imaging, vision, and graphics*. Springer, 2003.

[PB00] J. C. PINHEIRO et D. M. BATES : *Linear mixed-effects models : basic concepts and examples*. Springer, 2000.

[PBF10] S. PANDOLFI, F. BARTOLUCCI et N. FRIEL : A generalization of the Multiple-try Metropolis algorithm for Bayesian estimation and model selection. *In International Conference on Artificial Intelligence and Statistics*, pages 581–588, 2010.

[PD12] A. PETRALIAS et P. DELLAPORTAS : An MCMC model search algorithm for regression problems. *Journal of Statistical Computation and Simulation*, (ahead-of-print):1–19, 2012.

[Pes73] P. H. PESKUN : Optimum Monte-Carlo sampling using Markov chains. *Biometrika*, 60(3):607–612, 1973.

[PJ92] B. T. POLYAK et A. B. JUDITSKY : Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992.

[PLL07] D. POKRAJAC, A. LAZAREVIC et L. J. LATECKI : Incremental local outlier detection for data streams. *In Computational Intelligence and Data Mining, 2007. CIDM 2007. IEEE Symposium on*, pages 504–515. IEEE, 2007.

[PMPP05] A. PLAZA, P. MARTINEZ, J. PLAZA et R. PEREZ : Dimensionality reduction and classification of hyperspectral image data using sequences of extended morphological transformations. *Geoscience and Remote Sensing, IEEE Transactions on*, 43(3):466–479, 2005.

[Pol90] B. T. POLYAK : New method of stochastic approximation type. *Automat. Remote Control*, 51:937–946, 1990.

[PP07] A. PATCHA et J-M. PARK : An overview of anomaly detection techniques : Existing solutions and latest technological trends. *Computer Networks*, 51(12):3448–3470, 2007.

[QLdP99] F. A. QUINTANA, J. S. LIU et G. E. del PINO : Monte Carlo EM with importance reweighting and its applications in random effects models. *Computational statistics & data analysis*, 29(4):429–444, 1999.

[Ram98] J. O. RAMSAY : Estimating smooth monotone functions. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 60(2):365–375, 1998.

[Ram06] J. O. RAMSAY : *Functional data analysis*. Wiley Online Library, 2006.

[RC03] F. J-P. RICHARD et L. D. COHEN : A new image registration technique with free boundary constraints : application to mammography. *Computer Vision and Image Understanding*, 89(2):166–196, 2003.

[RC04] C. P. ROBERT et G. CASELLA : *Monte Carlo statistical methods*, volume 319 de *Springer Texts in Statistics*. Springer, 2004.

[RFH+07] F. ROUSSEAU, S. FAISAN, F. HEITZ, J-P. ARMSPACH, Y. CHEVALIER, F. BLANC, J. DE SÈZE, L. RUMBACH *et al.* : Une approche a contrario pour la détection de changements dans des images irm multimodales 3d. 2007.

[RFKN92] F.C. ROBEY, D.R. FUHRMANN, E.J. KELLY et R. NITZBERG : A CFAR adaptive matched filter detector. *Aerospace and Electronic Systems, IEEE Transactions on*, 28(1):208 –216, jan 1992.

[RL98] J. O. RAMSAY et X. LI : Curve registration. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 60(2):351–363, 1998.

[RM51] H. ROBBINS et S. MONRO : A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407, 1951.

[RM05] A. RAO et S. P. MAHULIKAR : Aircraft powerplant and plume infrared signature modelling and analysis. *In 43rd AIAA Aerospace Sciences Meeting and Exhibit*, volume 221, 2005.

[RMHM+10] A. ROBIN, L. MOISAN, L. HEGARAT-MASCLE *et al.* : An a-contrario approach for subpixel change detection in satellite imagery. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(11):1977–1993, 2010.

[Rob96] C. P. ROBERT : *Méthodes de Monte Carlo par chaînes de Markov*. Economica, 1996.

[RR04] G. O. ROBERTS et J. S. ROSENTHAL : General state space Markov chains and MCMC algorithms. *Probability Surveys*, 1:20–71, 2004.

[RSRD07] H. RINGBERG, A. SOULE, J. REXFORD et C. DIOT : Sensitivity of pca for traffic anomaly detection. *ACM SIGMETRICS Performance Evaluation Review*, 35(1):109–120, 2007.

[Rup88] D. RUPPERT : Efficient estimations from a slowly convergent robbins-monro process. Rapport technique, Cornell University Operations Research and Industrial Engineering, 1988.

[RY90] I. S. REED et X. YU : Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 38(10):1760 –1770, oct 1990.

[SBC+02] P. SIMONEAU, R. BERTON, K. CAILLAULT, G. DURAND, T. HUET, L. LABARRE, C. MALHERBE, C. MIESCH, A. ROBLIN et B. ROSIER : Matisse : advanced earth modeling for imaging and scene simulation. *In International Symposium on Remote Sensing*, pages 39–48. International Society for Optics and Photonics, 2002.

[SBH+02] D. W. J. STEIN, S. G. BEAVEN, L. E. HOFF, E. M. WINTER, A. P. SCHAUM et A. D. STOCKER : Anomaly detection from hyperspectral imagery. *Signal Processing Magazine, IEEE*, 19(1):58–69, 2002.

[SC78] H. SAKOE et S. CHIBA : Dynamic programming algorithm optimization for spoken word recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 26(1):43–49, 1978.

[SCE96] C. R. SCHWARTZ, J. N. CEDERQUIST et M. T. EISMANN : Target detection using infrared spectral sensors. *In SPIE's 1996 International Symposium on Optical Science, Engineering, and Instrumentation*, pages 182–194. International Society for Optics and Photonics, 1996.

[Sch03] Robert E SCHAPIRE : The boosting approach to machine learning : An overview. *Lecture Notes In Statistics*, pages 149–172, 2003.

[Sch07] A. P. SCHAUM : Hyperspectral anomaly detection beyond RX. *In Defense and Security Symposium*, pages 656502–656502. International Society for Optics and Photonics, 2007.

[SCS+00] A. SALTELLI, K. CHAN, E. M. SCOTT *et al.* : *Sensitivity analysis*, volume 134. Wiley New York, 2000.

[Ser82] J. SERRA : *Image analysis and mathematical morphology.* London : Academic Press. Mathematics, 1982.

[SI00] M-A. SATO et S. ISHII : On-line EM algorithm for the normalized Gaussian network. *Neural Computation*, 12(2):407–432, 2000.

[Sil85] B. W. SILVERMAN : Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society. Series B*, 47:1–52, 1985.

[SL05] H. E. SOLBERG et A. LAHTI : Detection of outliers in reference distributions : performance of horn' s algorithm. *Clinical chemistry*, 51(12):2326–2332, 2005.

[SM04] S. SINGH et M. MARKOU : An approach to novelty detection applied to the classification of image regions. *Knowledge and Data Engineering, IEEE Transactions on*, 16(4):396–407, 2004.

[SPS01] C. SPENCE, L. PARRA et P. SAJDA : Detection, synthesis and compression in mammographic image analysis with a hierarchical image probability model. *In Mathematical Methods in Biomedical Image Analysis, 2001. MMBIA 2001. IEEE Workshop on*, pages 3–10. IEEE, 2001.

[SRY90] A. D. STOCKER, I. S. REED et X. YU : Multidimensional signal processing for electro-optical target detection. *In OE/LASE'90, 14-19 Jan., Los Angeles, CA*, pages 218–231. International Society for Optics and Photonics, 1990.

[SS97] A. D. STOCKER et A. P. SCHAUM : Application of stochastic mixing models to hyperspectral detection problems. *In AeroSense'97*, pages 47–60. International Society for Optics and Photonics, 1997.

[ST94] G. SAMORODNITSKY et M. S. TAQQU : Stable non-Gaussian processes. *Stochastic Models with Infinite Variance, Chapman & Hall, New York*, 1994.

[Ste02] M. L. STEIN : Fast and exact simulation of fractional brownian surfaces. *Journal of Computational and Graphical Statistics*, 11(3):587–599, 2002.

[TF03] S. TEZUKA et H. FAURE : I-binomial scrambling of digital nets and sequences. *Journal of complexity*, 19(6):744–757, 2003.

[TGB10] Y. P. TAITANO, B. A. GEIER et K. W. BAUER : A locally adaptable iterative RX detector. *EURASIP Journal on Advances in Signal Processing*, 2010:11, 2010.

[Thi00] E. THIÉMARD : *Sur le calcul et la majoration de la discrépance à l'origine.* Thèse de doctorat, Ecole polytechnique fédérale de Lausanne, 2000.

[TI08] D. TELESCA et L.Y.T. INOUE : Bayesian hierarchical curve registration. *Journal of the American Statistical Association*, 103:328–339, 2008.

[Tie95] L. TIERNEY : A note on Metropolis-Hastings kernels for general state spaces. *Annals of Applied Probability*, 8:1–9, 1995.

[Tit84] D. M. TITTERINGTON : Recursive parameter estimation using incomplete data. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 257–267, 1984.

[TRU98] P. THÉVENAZ, U. E. RUTTIMANN et M. UNSER : A pyramid approach to subpixel registration based on intensity. *Image Processing, IEEE Transactions on*, 7(1):27–41, 1998.

[TS54] R. D. TUDDENHAM et M. M. SNYDER : Physical growth of California boys and girls from birth to eighteen years. *Publications in child development. University of California, Berkeley*, 1(2):183, 1954.

[Tuf97] B. TUFFIN : *Simulation accélérée par les méthodes de Monte Carlo et quasi-Monte Carlo : théorie et applications*. Thèse de doctorat, 1997.

[TW87] M. A. TANNER et W. H. WONG : The calculation of posterior distributions by data augmentation. *Journal of the American statistical Association*, 82(398):528–540, 1987.

[TY05] A. TROUVÉ et L. YOUNES : Local geometry of deformable templates. *SIAM journal on mathematical analysis*, 37(1):17–59, 2005.

[Vap98] V. N. VAPNIK : *Statistical learning theory*. Wiley, 1998.

[Var10] S. VARET : *Développement de méthodes statistiques pour la prédiction d'un gabarit de signature infrarouge*. Thèse de doctorat, Université Toulouse III, Paul Sabatier, 2010.

[VP11] H. VISSER et A. C. PETERSEN : Inferences on weather extremes and weather-related disasters : a review of statistical methods. *Climate of the past Discussions*, 7(5):2893–2935, 2011.

[WG97] K. WANG et T. GASSER : Alignment of curves by dynamic time warping. *The Annals of Statistics*, 25(3):1251–1276, 1997.

[Wol93] R. WOLFINGER : Laplace's approximation for non linear mixed models. *Biometrika*, 80, No :4:791–795, 1993.

[WT90] G. C. G. WEI et M. A. TANNER : A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. *Journal of the American Statistical Association*, 85(411):699–704, 1990.

[Wu83] C. F. WU : On the convergence properties of the EM algorithm. *The Annals of Statistics*, 11(1):95–103, 1983.

[WZ06] S. WANG et Y. ZHAO : Almost sure convergence of Titterington's recursive estimator for mixture models. *Statistics & probability letters*, 76(18):2001–2006, 2006.

[YHR+97] X. YU, L. E. HOFF, I. S. REED, A. M. CHEN et L. B. STOTTS : Automatic target detection and recognition in multiband imagery : A unified ml detection and estimation approach. *Image Processing, IEEE Transactions on*, 6(1):143–156, 1997.

[YRS93] X. YU, I.S. REED et A.D. STOCKER : Comparative performance analysis of adaptive multispectral detectors. *Signal Processing, IEEE Transactions on*, 41(8):2639 –2656, aug 1993.

[ZC07] J-T. ZHANG et J. CHEN : Statistical inferences for functional data. *The Annals of Statistics*, 35(3):1052–1079, 2007.

[ZF03] B. ZITOVA et J. FLUSSER : Image registration methods : a survey. *Image and vision computing*, 21(11):977–1000, 2003.

[Zho08] Z. ZHONG : *Curve registration in functional data analysis*. ProQuest, 2008.

[Zie77] H. ZIEZOLD : On expected figures and a strong law of large numbers for random elements in quasi-metric spaces. *In Transactions of the Seventh Prague Conference on Information Theory, Statistical Decision Functions, Random Processes and of the 1974 European Meeting of Statisticians*, pages 591–602. Springer, 1977.