Bayesian algorithms in high dimension, application to cosmology

Gabriel Ducrocq

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2022IPPAG001

Thèse de doctorat

ENSAE

IP PARIS

Bayesian algorithms in high dimension, application to cosmology

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à l'École Nationale de la Statistique et de l'Administration Économique

École doctorale n°574 École doctorale de mathématiques Hadamard (EDMH)
Spécialité de doctorat: Mathématiques appliquées

Thèse présentée et soutenue à Palaiseau, le 10/05/2022, par

GABRIEL DUCROCQ

Composition du Jury :

Radek Stompor Directeur de recherche, CNRS (Laboratoire d'Astroparticule et de cosmologie)	Président
Julyan Arbel Chargé de recherche, INRIA Grenoble Rhône-Alpes	Rapporteur
Krzysztof Łatuszyński Professeur, université de Warwick (Département de Statistiques)	Rapporteur
Hans-Kristian Eriksen Professeur, Université d'Oslo (Institute of Theoretical Astrophysics)	Examinateur
Anna Korba Professeur, ENSAE (CREST)	Examinateur
Rémi Bardenet Chargé de recherche, Université de Lille (CRIStAL)	Examinateur
Nicolas Chopin Professeur, ENSAE (CREST)	Directeur de thèse

Nous n'avons pas trop peu de temps, nous en avons beaucoup de perdu [...] Oui, brève est la vie, non que nous recevons, mais que nous nous sommes faite.

SÉNÈQUE

Remerciements

Je remercie d'abord Nicolas, pour sa patience, ses conseils et sa bienveillance. Merci de m'avoir donné ma chance quand je toquais à ta porte. Je remercie ensuite Radek, pour ses contributions à ma thèse et pour avoir toujours pris le temps de m'écouter et de faire des remarques pertinentes. Merci à Josquin, qui a pris le temps de vulgariser la cosmologie au débutant que je suis. Merci aussi pour les heures de débuggage. Merci à Clément, à qui j'ai pu poser les questions les plus stupides sans jamais qu'il ne me juge. Merci plus généralement à tout le projet B3DCMB pour cette thèse passionnante.

I would also like to thank Hans-Kristian Eriksen, for his valuable inputs when I was lost, dealing with the cosmology libraries.

Hors de ce projet, j'aimerais remercier Pierre, qui m'a soutenu lorsque je souhaitais faire une thèse. Je n'en aurais sûrement pas fait s'il n'avait pas été là. Merci à Cristina de s'être souvenue de moi et de m'avoir transmis l'offre de thèse de Nicolas. Merci à Mohamed et Marco Cannone de m'avoir accueilli dans leur bureau quand je cherchais des thèses. Merci à mes co-bureaux pour ces heures agréables passées à discuter, de statistiques et d'autres choses: Jules, Lucie, Nicolas, Badr. Merci à tous ceux grâce à qui l'ambiance au CREST était et reste excellente. Une liste désordonnée et non exhaustive: Avo, Julien, Amir, Jérémy, Flore, François-Pierre, Boris, Yannick, Geoffrey, Etienne, Hugo, Nayel, Arnak, Gauthier, Lionel...

Merci à Johan, qui, autour d'un verre, a toujours patiemment écouté le récit technique de mes déboires de thèse. Merci à toi aussi, Damien, ces après-midi passées avec les loutrons et toi étaient un puissant divertissement à mes galères. Merci plus généralement à tous mes amis, qui m'ont distrait quand j'en avais besoin.

Thank you, Nit. More than anyone else, you have been my true confidant during these three years. Thank you for being the person you are. Wubba lubba dub dub !

Enfin, merci à Mathilde, ma soeur, et à son mari, Paul, dont les vidéos de Sixtine me redonnent toujours le sourire. Merci à Joseph, mon frère. Grâce à ton humour, nos retrouvailles étaient et sont toujours source de réjouissances pour moi, qu'importent les aléas de la vie. Merci à mes parents, dont le soutien n'a jamais failli, quelle que soit la situation. Je n'aurais jamais pu arriver là sans eux.

Contents

Chapter 1.

Introduction à la partie statistique (in French)

Cette section présente les concepts nécessaire à la bonne compréhension de cette thèse. D'abord, nous discutons des statistiques Bayésiennes and des méthodes de Monte-Carlo par chaînes de Markov (MCMC en Anglais). Ensuite, nous détaillons expliquons les bases de l'analyse du Fond Diffus Cosmologique et l'usage des méthodes MCMC dans ce context. Finalement, nous résumons nos contributions dans les deux dernières sections.

1.1. Statistiques Bayésiennes

Le but de l'inférence Bayésienne est d'extraire de la connaissance à partir d'expériences faites dans le monde réel, voir Robert (2007) pour une revue de littérature. Chaque expérience peut être décrite mathématiquement comme un n-uplet $(\mathcal{Y}, \mathcal{B}(\mathcal{Y}), \{\mathbb{P}_\theta, \theta \in \Theta\})$ où $\mathcal{Y}$ est l'espace d'observation et $\mathcal{B}(\mathcal{Y})$ la tribu Borélienne de $\mathcal{Y}$. L'ensemble Θ est appelé l'espace des paramètres et $\{\mathbb{P}_\theta, \theta \in \Theta\}$ est une famille de mesures sur $(\mathcal{Y}, \mathcal{B}(\mathcal{Y}))$. Dans le reste de ce manuscrit nous considérons seulement $\mathcal{Y} = \mathbb{R}^d$ avec $d \in \mathbb{N}^\star$, $\Theta \subseteq \mathbb{R}^m$ et nous supposons que $\mathbb{P}_\theta$ est dominée par la mesure de Lebesgue $\mathrm{d}y$ quelque soit $\theta \in \Theta$.

L'inférence Bayésienne quantifie l'incertitude sur θ en mettant à jour la croyance a priori de l'expérimentateur grâce aux données. Si nous considérons que θ est une variable aléatoire sur $(\Theta, \mathcal{B}(\Theta), \mathrm{d}\theta)$, l'expérimentateur peut formaliser sa croyance a priori concernant le paramètre, avant toute observation, grâce à la mesure a priori $p_0(\theta)\mathrm{d}\theta$, où $p_0(\theta)$ est la densité de θ par rapport à $\mathrm{d}\theta$. Supposons qu'on observe y, une réalization de la variable aléatoire $Y \in \mathcal{Y}$, nous pouvons définir la vraisemblance:

$$\begin{aligned} p\colon \Theta \times \mathcal{Y} &\to [0, +\infty) \\ (\theta, y) &\mapsto p(y|\theta) \end{aligned} \tag{1.1}$$

Les données y peuvent contraindre l'incertitude a priori grâce à la vraisemblance. Nous pouvons décrire l'incertitude a posteriori sur le paramètre grâce à la distribution:

$$\pi(\theta|y) = \frac{p(y|\theta)p_0(\theta)}{Z(y)} \tag{1.2}$$

où

$$Z(y) = \int_\Theta p(y|\theta)p_0(\theta)\mathrm{d}\theta \tag{1.3}$$

est appelée l'évidence ou la vraisemblance marginale. Puisque p_0 est une densité de probabilité, nous avons $Z(y) < \infty$. Il est possible de choisir p_0 telle que:

$$\int_\Theta p_0(\theta)\mathrm{d}\theta = \infty.$$

Dans ce cas il est nécessaire de vérifier que $Z(y) < \infty$ pour que Eq. (1.3) soit bien définie.
En pratique, les moments de la distribution a posteriori nous intéressent:

$$\mathbb{E}[h(\theta)|y] = \int_\Theta h(\theta)\pi(\theta|y)\mathrm{d}\theta \tag{1.4}$$

où $h : \Theta \to \mathbb{R}$ par exemple, et nous voulons aussi estimer l'évidence Eq. (1.3). Nous souhaitons aussi un estimateur ponctuel de notre paramètre. Pour celà, nous prenons l'estimateur du maximum a posteriori (MAP):

$$\theta^{\mathrm{MAP}} \in \arg\max_{\theta} \pi(\theta|y).$$

Manifestement, faire de l'inférence Bayésienne suppose que nous soyons capables de calculer des espérances comme Eq. (1.4). C'est difficile en général et la distribution a posteriori Eq. (1.2) est souvent difficilement calculable. Dans ce cas, nous avons deux options: caractériser la distribution a posteriori à travers un échantillonage ou l'approximer grâce à une famille de distribution calculables. Dans la prochaine section, nous nous concentrons sur la première option. Pour la seconde approche, le lecteur peut lire Martin J Wainwright (2008).

1.2. Méthodes de Monte-Carlo par chaînes de Markov

Puisque $\pi(\theta|y)$ est une densité sur Θ, nous abandonnons la dépendance en y par soucis de clarté. Nous considérons donc $\pi(\theta)$ au lieu de $\pi(\theta|y)$.

Les méthodes de Monte-Carlo par chaînes de Markov (MCMC) ont pour objectif de calculer des espérances sur l'espace d'état Θ:

$$\mathbb{E}_{\pi}[h(\theta)] = \int_{\Theta} h(\theta)\pi(\theta)\mathrm{d}\theta \tag{1.5}$$

où π est une densité sur Θ et, par exemple, $h : \Theta \to \mathbb{R}$. Pour ce faire, les méthodes MCMC produisent des échantillons corrélés de π et estiment Eq. (1.5) grâce à la moyenne empirique de ces échantillons. Plus précisément, les méthodes MCMC construisent une chaîne de Markov $(\theta)_{n\geqslant 1}$ telle que:

$$\frac{1}{N}\sum_{n=0}^{N} h(\theta_n) \xrightarrow[N\to\infty]{\mathrm{a.s}} \mathbb{E}_{\pi}[h(\theta)]. \tag{1.6}$$

Si le noyau de transition P de la chaîne de Markov laisse π invariante:

$$\int_{\Theta} \pi(\theta)P(\theta, \mathrm{d}\theta')\mathrm{d}\theta = \pi(\theta')\mathrm{d}\theta'$$

et que la chaîne est irréductible et apériodique, alors nous savons que pour $\pi - \mathrm{p.s}$ tout $\theta_0 \in \Theta$:

$$||P^n(\theta_0, \mathrm{d}\theta) - \pi(\theta)\mathrm{d}\theta||_{\mathrm{TV}} \xrightarrow[n\to\infty]{} 0$$

où $||.||_{\mathrm{TV}}$ est la norme de variation totale, voir e.g Roberts and Rosenthal (2004). Cela signifie que pour $\pi - \mathrm{p.s}$ tout point de départ, la chaîne converge vers sa distribution invariante. Sous la condition plus forte d'Harris récurrence, ce résultat est valable pour tout $\theta_0 \in \Theta$, voir e.g Roberts and Rosenthal (2004). En plus de ce résultat, nous pouvons également caractériser la vitesse de convergence vers la distribution invariante: si la chaîne possède un "small set" and satisfait une condition de "drift" sur cet ensemble, elle est géométriquement ergodique, c'est à dire:

$$||P^n(\theta_0, \mathrm{d}\theta) - \pi(\theta)\mathrm{d}\theta||_{\mathrm{TV}} \leqslant M(\theta_0)\rho^n$$

pour un $\rho \in [0, 1[$ et $M(\theta_0) < \infty$ pour $\pi - \mathrm{p.s}$ tout $\theta_0 \in \Theta$. Dans le cas où M ne dépend pas de θ_0, la chaîne est dite uniformément ergodique. Finalement, nous avons un théorème de la limite centrale:

$$\frac{1}{\sqrt{N}}\sum_{n=0}^{N} \{h(X_n) - \mathbb{E}_{\pi}[h(\theta)]\} \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}(0, \sigma_h^2)$$

pour $\pi - \text{p.s}$ tout $\theta_0 \in \Theta$, où

$$\sigma_h^2 = \text{Var}_\pi(h) + 2\sum_{n=1}^{\infty} \text{Cov}(h(\theta_0), h(\theta_n)) \tag{1.7}$$

sous la condition que la chaîne est uniformément ergodique et que $\mathbb{E}_\pi[h(\theta)] < \infty$, voir Meyn and Tweedie (2014). Dans le cas où la chaîne n'est que géométriquement ergodique, le théorème de la limite centrale tient toujours si $\mathbb{E}_\pi[h(\theta)] < \infty$ et si la chaîne est réversible, c'est à dire:

$$\pi(\text{d}\theta_0)P(\theta_0, \text{d}\theta_1) = \pi(\text{d}\theta_1)P(\theta_1, \text{d}\theta_0)$$

pour n'importe quel $\theta_0, \theta_1 \in \Theta$. Finalement, si

$$L_0^2(\pi) = \{h(\theta) : \mathbb{E}_\pi[h(\theta)] = 0; \text{Var}_\pi[h(\theta)] < \infty\}$$

est l'espace de Hilbert des fonctions scalaires de moyenne nulle, de variance finie et de produit scalaire $< h(\theta), t(\theta) >= \mathbb{E}_\pi[h(\theta)t(\theta)]$ tel que $||h(\theta)||_{L_0^2(\pi)} =< h(\theta), h(\theta) >$, nous pouvons définir l'opérateur "forward" comme:

$$F^n h(x) = \mathbb{E}[h(\theta_n)|\theta_0 = x] = \int_\Theta h(\theta)P^n(x, \text{d}\theta) \tag{1.8}$$

pour n'importe quel $n \in \mathbb{N}^\star$. Liu et al. (1995) ont montré qu'à stationnarité, c'est à dire lorsque nous supposons que $\theta_0 \sim \pi$ et que $\theta_n|\theta_0 \sim P^n(\theta_0, \text{d}\theta)$, nous avons:

$$||F^n|| = \gamma_n \tag{1.9}$$

où $||.||$ est la norme opérateur définie comme:

$$||F|| = \sup_{t \in L_0^2(\pi)} \frac{||Ft||_{L_0^2(\pi)}}{||t||_{L_0^2(\pi)}}$$

et

$$\gamma_n := \sup_{f,g} \text{Corr}(f(\theta_0), g(\theta_n)) \tag{1.10}$$

est le coefficient de corrélation maximal entre θ_0 et θ_n, où le supremum est pris sur l'ensemble des fonctions scalaires dont la variance est finie. Nous pouvons voir que les normes d'opérateur des puissances de F, toutes inférieures à 1, sont directement bornées par les autocorrélations "lag-n" de la chaîne. Finalement, Liu et al. (1995) ont montré que si la chaîne est réversible, F est auto-adjoint, et dans ce cas:

$$||F||^n = \gamma_n.$$

Dans les deux prochaines sections, nous donnons des exemples d'algorithmes de Monte-Carlo par chaînes de Markov ayant les propriétés que nous venons de décrire: l'algorithme de Métropolis-Hastings et l'échantillonneur de Gibbs.

1.2.1. L'algorithme de Métropolis-Hastings

Une façon connue de construire des chaînes de Markov comme décrit dans dans la section précédente est l'algorithme de Métropolis-Hatings, décrit par Hastings (1970): à chaque étape $n \in \mathbb{N}$, un nouvel état θ' est proposé grâce à la distribution instrumentale $Q(\theta_{n-1}, \text{d}\theta)$. Nous acceptons ce nouvel état, c'est à dire $\theta_n = \theta'$, avec une probabilité qui dépend du ratio de la densité évaluée en ce nouvel état sur la densité évaluée en l'état précédent θ_{n-1}.
En fonction du choix de Q, cet algorithme a différentes propriétés. En général, les praticiens choisissent $Q(\theta_{n-1}, \text{d}\theta) \overset{\mathcal{L}}{=} \mathcal{N}(\theta_{n-1}, \tau\Sigma)$, où $\tau \in \mathbb{R}_+$ est le paramètre d'échelle et $\tau\Sigma$ est la matrice de covariance de

la distribution instrumentale, tous les deux choisis par l'utilisateur. Il faut choisir Σ de sorte que la matrice de covariance de la distribution cible corresponde à peu près à la matrice de covariance de la distribution instrumentale. Cela nous aidera à proposer de nouveaux états le long des principales directions de la distribution cible. Le paramètre τ est choisi tel que l'algorithme a le taux d'acceptation voulu. Cet algorithme est appelé l'algorithme de Métropolis-Hastings par marche aléatoire, voir Tierney (1994): à chaque étape nous perturbons l'état actuel θ_n avec un bruit Gaussien et nous obtenons un nouvel état θ_{n+1}. Si l'état proposé est dans une région de plus grande probabilité que l'état précent, nous acceptons automatiquement ce nouvel état. Sinon, nous l'acceptons avec une probabilité proportionnelle au ratio des densités évaluées en le nouveau et l'ancien état, voir Algorithm 1. Manifestement, nous devons être

Algorithm 1: Etape n de l'algorithme de Métropolis-Hastings par marche aléatoire

Input: Etat actuel θ_{n-1}, densité cible π
Output: Etat suivant θ_n

Simuler $\theta^{'} \sim \mathcal{N}(\theta_{n-1}, \tau\Sigma)$
Calculer $r(\theta_{n-1}, \theta^{'}) = \min\left\{\frac{\pi(\theta^{'})}{\pi(\theta_{n-1})}, 1\right\}$
Simuler $u \sim \mathcal{U}[0,1]$
if $u < r(\theta_{n-1}, \theta^{'})$ **then**
    définir $\theta_n \leftarrow \theta^{'}$
else
    définir $\theta_n \leftarrow \theta_{n-1}$

capable d'évaluer au moins une version non normalisée de π pour implémenter cet algorithme.

Cet algorithme laisse la loi cible π invariante par construction. En effet, son noyau de transition est donné par:

$$\begin{aligned} P_{\mathrm{RWMH}}(\theta_{n-1}, \mathrm{d}\theta_n) =& r(\theta_{n-1}, \theta_n) Q(\theta_{n-1}; \mathrm{d}\theta_n) + \delta_{\theta_{n-1}}(\mathrm{d}\theta_n) \\ & \times \left\{1 - \int_{\Theta} r(\theta_{n-1}, \theta^{'}) Q(\theta_{n-1}; \mathrm{d}\theta^{'})\right\} \end{aligned}$$

où la fonction r est définie par Algorithme 1. Il est facile de démontrer que:

$$\pi(\mathrm{d}\theta_{n-1}) P_{\mathrm{RWMH}}(\theta_{n-1}, \mathrm{d}\theta_n) = \pi(\mathrm{d}\theta_n) P_{\mathrm{RWMH}}(\theta_n, \mathrm{d}\theta_{n-1}) \tag{1.11}$$

c'est à dire, P_{RWMH} vérifie les "detailed balance conditions". En intégrant les deux côtés de Eq. (1.11) par rapport à θ_{n-1} nous donne la π-invariance de P_{RWMH} et puisque la distribution intrumentale est continue et positive sur $\Theta \times \Theta$, elle est aussi irréductible. Il s'ensuit que l'algorithme est aussi irréductible. Tierney (1994) décrit les propriétés théoriques de cet algorithme.

Dans la section suivante nous discutons d'une autre façon de construire des chaînes de Markov avec les bonnes propriétés: l'échantillonneur de Gibbs.

1.2.2. L'échantillonneur de Gibbs

Si $\Theta \subseteq \mathbb{R}^m$ est multidimensionnel et que nous connaissons les lois conditionnelles $\pi(\theta^i|\theta^{-i})$ de la loi cible π, où θ^i est la i-ème composante de θ et θ^{-i} dénote toutes les composantes excepté la i-ème, nous pouvons implémenter un échantillonneur de Gibbs, voir Algorithme 2. Cet algorithme est appelé l'échantillonneur de Gibbs par balayage systématique. La variante par balayage aléatoire consiste à sélectionner aléatoirement une des conditionnelles à échantillonner, ce qui rend l'algorithme réversible. Les deux algorithmes laissent π invariante. Cependant, nous devons vérifier que l'algorithme

Algorithm 2: Etape n de l'échantillonneur de Gibbs

Input: Etat actuel θ_{n-1}, densité cible π
Output: Etat suivant θ_n

1 Echantillonner $\theta_n^1 \sim \pi(\theta^1|\theta_{n-1}^{-1})$
2 Echantillonner $\theta_n^2 \sim \pi(\theta^2|\theta_n^1, \theta_{n-1}^3, \ldots, \theta_{n-1}^m)$
3 Echantillonner $\theta_n^3 \sim \pi(\theta^3|\theta_n^1, \theta_n^2, \theta_{n-1}^4, \ldots, \theta_{n-1}^m)$
$\vdots$
4 Echantillonner $\theta_n^m \sim \pi(\theta^m|\theta_n^{-m})$

est irréductible et apériodique. Si une distribution conditionnelle de π n'a pas de forme connue, nous pouvons remplacer un échantillonnage direct par un algorithme de Métropolis-Hastings ciblant cette conditionnelle pour un nombre fixé d'étapes. Cet algorithme est appelé "Metropolis-within-Gibbs". Puisque l'algorithme de Métropolis-Hastings laisse invariante la bonne distribution, l'algorithme "Metropolis-within-Gibbs" laisse π invariante. Bien que l'échantillonneur de Gibbs ne requiert pas la calibrations de paramètres, l'introduction d'un pas de Métropolis pour échantillonner une des lois conditionnelles introduit la nécessité d'un calibrage de la loi instrumentale. En règle générale, calibrer un algorithme "Metropolis-within-Gibbs" est plus difficile que de calibrer un algorithme de Métropolis-Hastings: à chaque itération, les paramètres de la loi conditionnelle cible changent. Cela signifie une calibration différente pour la loi instrumentale à chaque itération, tandis que l'algorithme de Métropolis-Hastings ne requiert la calibration de la loi instrumentale qu'une fois pour toutes.

Au lieu d'échantillonner chaque loi conditionnelle univariée, nous pouvons partitionner les coordonnées en sous-ensembles et échantillonner chacun des ces sous-ensembles conditionnellement aux autres. Cet algorithme est similaire à l'Algorithm 2 excepté que les distributions conditionnelles sont maintenant multivariées. Liu et al. (1995) a montré que sous certaines conditions, l'échantillonneur de Gibbs par balayage systématique est géométriquement ergodique avec un taux de convergence:

$$\rho = r := \lim_{n\to\infty} ||F^n||^{1/n} < 1$$

où F est l'opérateur "forward" de la chaîne définie en Eq. (1.8) et r est appelé le rayon spectral de l'opérateur F. Sous certaines conditions, l'échantillonneur de Gibbs par balayage aléatoire est géométriquement ergodique avec un taux:

$$\rho = ||F|| < 1.$$

Ces deux derniers résultats ensembles avec Eq. (1.9) suggèrent que nous pourrions utiliser les autocorrélations "lag-n" de l'échantillonneur de Gibbs avec balayage systématique et les autocorrelations "lag-1" de l'échantillonneur de Gibbs par balayage aléatoire pour estimer leur taux de convergence géométrique respectifs. Le cas ou les coordonnées sont partitionnées en deux sous-ensembles $\theta = (\theta^1, \theta^2)$ est bien étudié et compris, voir Liu (1994) et Liu et al. (1994). Par simplicité, nous considérons le cas où $\Theta \subseteq \mathbb{R}^2$. Dans ce cas, les processus $(\theta_n^1)_{n\geqslant 1}$ et $(\theta_n^2)_{n\geqslant 1}$ sont des chaînes de Markov réversibles. Il a été montré qu'à stationnarité:

$$\gamma_1 = \gamma_\pi \tag{1.12}$$

où γ_1 est définie en Eq. (2.10) et

$$\gamma_\pi = \sup_{f,g} \mathrm{Corr}(f(\theta^1), g(\theta^2))$$

où le supremum est pris sur toutes les fonctions à valeurs réelles de variance finie et (θ^1, θ^2) est distribué selon π. De plus, nous avons:

$$\gamma_{\pi_1} = \gamma_{\pi_2} = \gamma_\pi^2 \tag{1.13}$$

avec, pour $i \in \{1, 2\}$:

$$\gamma_{\pi_i} = \sup_{f,g} \mathrm{Corr}(f(\theta_n^i), g(\theta_{n+1}^i))$$

où le supremum est pris sur toutes les fonctions à valeurs réelles avec variance finie et où nous supposons la stationnarité. Il s'ensuit que:

$$||F||^2 = ||F_1|| = ||F_2|| = \gamma_\pi^2$$

où γ est le coefficient de corrélation maximale définie en Eq. (1.10) et F, F_1 et F_2 sont les opérateurs "forward" de chaînes $(\theta_n)_{n\geqslant 1}$, $(\theta_n^1)_{n\geqslant 1}$ et $(\theta_n^2)_{n\geqslant 1}$ respectivement. Le résultat Eq. (1.12) énonce que la corrélation maximale entre deux états successifs de l'échantillonneur de Gibbs est égal à la corrélation maximale de deux composantes de la loi cibles π. Le résultat Eq. (1.13) implique que le coefficient de corrélation maximale entre deux état successifs des sous chaînes $(\theta_n^1)_{n\geqslant 1}$ et $(\theta_n^2)_{n\geqslant 1}$ dépend directement de la corrélation maximale entre les deux composantes de la loi cible. De plus, nous savons qu'à stationnarité, l'autocorrélation "lag-1" est donnée par:

$$\gamma_{\pi_1} = \sup_{h:\, \mathrm{Var}(h(\theta))<\infty} \left\{ 1 - \frac{\mathbb{E}[\mathrm{Var}(h(\theta^1)|\theta^2)]}{\mathrm{Var}[h(\theta^1)]} \right\} \tag{1.14}$$

où le terme de droite est appelé fraction d'information manquante, voir Liu (1994).
De plus, Liu (1994) et Liu et al. (1994) ont démontré que les rayons spectraux des opérateurs "forward" des chaînes $(\theta_n^1)_{n\geqslant 1}$, $(\theta_n^2)_{n\geqslant 1}$ et $(\theta_n)_{n\geqslant 1}$ sont tous égaux à γ_π^2. Cela signifie que le taux de convergence géométrique de l'échantillonneur de Gibbs et des sous chaînes depend de la force des corrélations entre ses deux composantes. De plus par Eq. (1.13), la force de ces corrélations se manifeste au travers des variances conditionnelles: si θ^1 et θ^2 sont très corrélées, le support de de la distribution de θ^1 sachant θ^2 est petit comparé à la variance de la loi marginale a posteriori de θ^1 et l'échantillonneur de Gibbs n'explore pas efficacement le support de la loi jointe a posteriori. Cela se traduit par une fraction d'information manquante proche de un et donc un taux de convergence géométrique proche de un.

1.2.3. Control variates

Nous pouvons réduire la variance des estimateurs de Monte-Carlo grâce à des variables de contrôle, voir Robert and Casella (2004). Supposons que nous avons $J \in \mathbb{N}$ fonctions h_j telles que, quelque soit $j \in \{1, \ldots, J\}$:

$$\mathbb{E}_{\pi(\theta|y)}[h_j(\theta)] = 0$$

appelées variables de contrôle. Alors l'estimateur

$$\hat{p}_\beta := \frac{1}{N} \sum_{n=1}^{N} \{f(\theta_n) + \beta^t h(\theta_n)\} \tag{1.15}$$

est tel que

$$\hat{p}_\beta \xrightarrow[N\to\infty]{\text{p.s}} \mathbb{E}[f(\theta) + \beta^t h(\theta)] = \mathbb{E}[f(\theta)].$$

où $\beta \in \mathbb{R}^J$ et $h(\theta) = (h_1(\theta), \ldots, h_J(\theta)) \in \mathbb{R}^J$. Nous devons choisir β et h tel que la variance de Eq. (1.15) est réduite comparée la moyenne ergodique usuelle Eq. (1.6). Deux scenarii sont possibles.

Le premier où les points $(\theta)_{n\geqslant 0}$ ont été échantillonnés de façon i.i.d comme dans le cas de la méthode du rejet où de l'échantillonnage d'importance. Dans ce cas:

$$\mathrm{Var}(\hat{p}_\beta) = \frac{1}{N}\{\mathrm{Var}(f(\theta)) + \beta^t \mathrm{Var}(h(\theta))\beta + 2\beta^t \mathrm{Cov}(h(\theta), f(\theta))\} \tag{1.16}$$

où $\mathrm{Var}(h(\theta))$ est la matrice $J \times J$ de variance du vecteur $h(\theta)$ et $\mathrm{Cov}(h(\theta), f(\theta))$ est le vecteur $J \times 1$ avec $\mathrm{Cov}(h(\theta), f(\theta))_{i,1} = \mathrm{Cov}(h_i(\theta), f(\theta))$. En différentiant par rapport à β pour trouver la valeur atteignant la variance minimale, nous avons:

$$\beta^\star = -\mathrm{Var}(h(\theta))^{-1}\mathrm{Cov}(h(\theta), f(\theta)).$$

Bien sûr, les deux quantités du terme de droite ne sont pas nécessairement connues et nous devons estimer $\beta^\star$ en utilisant l'échantillon $(\theta)_{n\geqslant 0}$ comme fait dans Owen (2013):

$$\hat{\beta} = -\hat{\Sigma}_{h,h}^{-1}\hat{\Sigma}_{h,f}$$

où $\hat{\Sigma}_{h,h}$ et $\hat{\Sigma}_{h,f}$ sont des estimateurs de $\mathrm{Var}(h(\theta))$ et $\mathrm{Cov}(h(\theta), f(\theta))$ basés sur $(\theta)_{n\geqslant 0}$, respectivement. Notons que le nouvel estimateur:

$$\hat{p}_{\hat{\beta}} := \frac{1}{N}\sum_{n=1}^{N}\{f(\theta_n) + \hat{\beta}^t h(\theta_n)\}$$

est asymptotiquement sans biais, voir Owen (2013).

Dans le second scénario $(\theta)_{n\geqslant 0}$ ont été échantillonnés avec des méthodes de Monte-Carlo par chaînes de Markov. Les points échantillonnés ne sont plus indépendants et la variance de l'estimateur Eq. (1.15) n'est plus Eq. (1.16) mais plutôt de la forme Eq. (1.7). Nous voulons maintenant minimiser la variance asymptotique Eq. (1.15):

$$\begin{aligned}\sigma^2_{f+\beta^t h} =& \mathrm{Var}_\pi(f(\theta_n) + \beta^t h(\theta_n)) \\ &+ 2\sum_{n=1}^{\infty}\mathrm{Cov}(f(\theta_0) + \beta^t h(\theta_0), f(\theta_n) + \beta^t h(\theta_n))\end{aligned}$$

en tant que fonction de β. C'est en général difficile. Quand le kernel de transition associé à la chaîne de Markov est réversible par rapport à la loi cible, Dellaportas and Kontoyiannis (2011) proposent une façon d'estimer le $\beta^\star$ optimal. Une autre option est de simplement ignorer les corrélations entre les points successifs et de retourner au premier scénario. Le $\beta^\star$ ainsi estimé ne minimize plus la variance asymptotique dans ce cas.

Plusieurs choix pour h ont été proposés dans la littérature. Quand les moyennes conditionnelles de la loi cible sont connues, Dellaportas and Kontoyiannis (2011) proposent les variables de contrôle:

$$h_i(\theta) = \theta^i - \mathbb{E}[\theta^i|\theta^{-i}]$$

où θ^i dénote la i-ème composante de θ, θ^{-i} dénote toute les composantes de θ excepté la i-ème et l'espérance est prise sous $\pi(\theta^i|\theta^{-i}, y)$.

Quand la fonction de score:

$$s_\pi(\theta) = \nabla_\theta \log \pi(\theta|y)$$

est disponible, les fonctions:

$$h(x) = \nabla_\theta \cdot \phi(\theta) + \phi(\theta) \cdot s_\pi(\theta)$$

sont des variables de contrôle, pour toute fonction ϕ telle que

$$\oint_{\partial\Theta} p(\theta)\phi(\theta) \cdot n(\theta)S(\mathrm{d}\theta) = 0$$

où $\oint_{\partial\Theta}$ dénote l'intégrale sur le bord de Θ, et $S(\mathrm{d}\theta)$ est l'élément de surface en $\theta \in \partial\Theta$ et sous la condition que la densité de probabilité $\pi(.|y) \in C^1(\Theta, \mathbb{R})$.

Nous ne sommes pas obligés de nous restreindre à une familles finie de variables de contrôle $\{h_1(\theta), \ldots, h_J(\theta)\}$. Par exemple Oates et al. (2016) proposent de construire une approximation non paramétrique de f. Ils divisent l'échantillon $(\theta)_{0 \leqslant n \leqslant N}$ en deux sous-ensembles disjoints $\mathcal{D}_0$ et $\mathcal{D}_1$ de taille $m+1$ et $N-m-1$ respectivement, et utilisent $\mathcal{D}_0$ pour avoir une approximation non paramétrique de f:

$$s_{f,\mathcal{D}_0} := \arg\min_{g \in \mathcal{H}_+} \left\{ \frac{1}{m} \sum_{n=0}^{m} (f(\theta_n) - g(\theta_n))^2 + \lambda ||g||_{\mathcal{H}_+} \right\}$$

où $\lambda > 0$, $\mathcal{H}_+$ est un espace de Hilbert de fonctions écrites comme la somme d'une fonction constante plus une autre fonction vérifiant le "Stein trick" et $||.||_{\mathcal{H}_+}$ est la norme sur $\mathcal{H}_+$. Si nous définissons:

$$\mu(s_{f,\mathcal{D}_0}) := \mathbb{E}_{\pi(.|y)}[s_{f,\mathcal{D}_0}(\theta)]$$

sous certaines conditions, parmi lesquelles $f \in \mathcal{H}_+$, la variance du nouvel estimateur:

$$\hat{\mu}(\mathcal{D}_0, \mathcal{D}_1; f) = \frac{1}{N-m-1} \sum_{n=m+1}^{N} \{f(\theta_n) - s_{f,\mathcal{D}_0}(\theta_n)\} + \mu(s_{f,\mathcal{D}_0})$$

est $\mathcal{O}(N^{-7/6})$, surperformant le $\mathcal{O}(N^{-1})$ de la traditionnelle moyenne ergodique des méthodes de MCMC.

Une méthode similaire est employée par Mira et al. (2013) pour construire des variables de ccontrôle menant à un estimateur de variance nulle. Supposons que nous choisissons un opérateur Hermitien H agissant sur les fonctions infiniment différentiables à support compact tel que:

$$H\sqrt{\pi} = 0$$

et une fonction ψ à support compact et infiniment différentiable. Alors:

$$\tilde{f}(\theta) = f(\theta) + \frac{H\psi}{\sqrt{\pi(\theta|y)}}$$

est telle que:

$$\mathbb{E}_{\pi(\theta|y)}[\tilde{f}(\theta)] = \mathbb{E}_{\pi(\theta|y)}[f(\theta)].$$

Le meilleur choix possible du couple (H, ψ) est tel que $\tilde{f}$ a une variance nulle, ce qui arrive lorsque:

$$H\psi = -\sqrt{\pi(\theta|y)} \left\{ f(\theta) - \mathbb{E}_{\pi(\theta|y)}[f(\theta)] \right\}. \tag{1.17}$$

Mira et al. (2013) proposent plusieurs choix de fonctions ψ qui vérifient Eq. (1.17). Malheureusement, pour un H donné la fonction optimale ψ ne peut en général pas être obtenue explicitement. Nous pouvons alors restreindre ψ à une famille de fonctions paramétrique et minimiser la variance de $\tilde{f}$ par rapport à ces paramètres. Dans ce cas, la variance du nouvel estimateur n'est plus zéro et nous devons vérifier que le nouvel estimateur est non biaisé, voir Mira et al. (2013) poour plus de détails.

1.3. Data Augmentation

Il arrive parfois que l'on puisse réécrire la vraisemblance Eq. (1.1) comme:

$$\pi(y|\theta) = \int_{\mathcal{X}} p(y|x,\theta) p(x|\theta) \mathrm{d}x \tag{1.18}$$

où $p(x|\theta)$ est une densité sur l'espace mesuré $(\mathcal{X}, \mathcal{B}(\mathcal{X}), \mathrm{d}x)$ où $\mathcal{X} \subseteq \mathbb{R}^q$ avec $q \in \mathbb{N} \backslash \{0\}$. Cela revient à introduire une nouvelle variable aléatoire X sur $\mathcal{X}$ avec distribution conditionnelle $p(x|\theta)\mathrm{d}x$, appelée

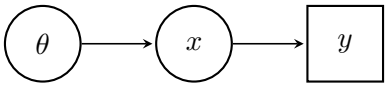

Figure 1.1.: Graph orienté acyclique de Eq. (1.18)

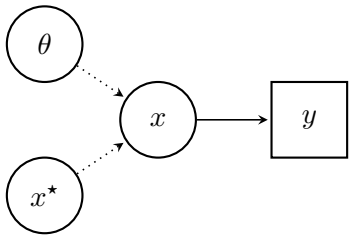

Figure 1.2.: Graph orienté acyclique de la paramétrisation non centrée.

variable latente.
Au lieu de cibler la loi à posteriori Eq. (1.18), nous pouvons maintenant cibler la loi jointe a posteriori:

$$\pi(\theta, x|y) = \frac{p(y|x,\theta)p(x|\theta)p(\theta)}{Z(y)}.$$

Il y a plusieurs raisons pour faire cela: la vraisemblance observée Eq. (1.1) peut être incalculable, X peut avoir un intérêt scientifique ou les méthodes MCMC peuvent être plus efficaces en ciblant la loi jointe a posteriori plutôt que la loi a posterior des paramètres. Cibler la loi jointe a posteriori est appelé "Data augmentation", voir Liu et al. (1994). Dans la suite, nous supposerons que les données et les paramètres sont indépendant sachant les variables latentes:

$$p(y|x,\theta) = p(y|x)$$

et nous dessinons le Graph Orienté Acyclique (DAG) du modèle Fig. 1.1. Dans ce contexte, un DAG est une représentation graphique d'un modèle hiérarchique: un noeud rond dénote une variable non observée, un noeud carré dénote une variable observée et une flèche pleine représente une dépendance stochastique entre deux variables. Notons qu'il ne peut pas y avoir de cycle dans un tel graph, voir Whittaker (1990).

Lorsque nous pouvons échantillonner x sachant (θ, y) ainsi que θ sachant (x, y), nous pouvons implémenter un échanillonneur de Gibbs bivarié. Comme discuté dans la section précédente, plus les corrélations a posteriori entre les paramètres et les variables latentes sont grandes, plus le taux de convergence géométrique de l'échantillonneur de Gibbs est élevé. Dans ce cas, il faut casser les corrélations a posteriori afin d'améliorer ce taux de convergence. Pour ce faire, nous pouvons reparamétrer le modèle: nous trouvons une paire de variables aléatoires $(X^\star, \theta)$ avec une loi jointe a priori $p_0(X^\star, \theta)$ et une fonction η telle que:

$$x = \eta(x^\star, \theta).$$

voir Papaspiliopoulos et al. (2007) par exemple. Notons que η, pour θ fixé, ne doit pas nécessairement être bijective. Nous pouvons alors cibler la loi jointe a posteriori $\pi(\theta, x^\star|y)$ et si nous faisons un bon choix de η, la corrélation a posteriori entre θ et $X^\star$ devrait être inférieur à la corrélation a posteriori entre θ and X. Une reparamétrisation généralement utile est donnée par la paramétrisation non-centrée, dépeinte en Fig. 1.2. Dans cette paramétrisation, le paramètre et les variables latentes sont maintenant a priori indépendants et les corrélations entre les deux ne viennent que des données.

Quand l'échantillonneur de Gibbs sur la paramétrisation centrée, appelé l'échantillonneur de Gibbs centré, mélange bien, l'échantillonneur de Gibbs sur la paramétrisation non centrée, appelé échantillonneur

de Gibbs non centré, mélange mal: quand les données sont très informatives sur les variables latentes, elles peuvent casser les corrélations a priori entre θ et x et l'échantillonneur de Gibbs centré aura un taux de convergence géométrique bas. Tandis que l'échantillonneur de Gibbs non centré mélangera mal puisque θ et x^* seront fortement corrélés a posteriori, car la vraisemblance est très contraignante.

L'opposé est généralement vrai: quand la distribution a priori corrèle θ et x plus fortement que la vraisemblance n'identifie x, l'échantillonneur de Gibbs sera inefficace à cause d'une fraction d'information manquante élevée Eq. (1.14). L'échantillonneur de Gibbs non centré mélangera mieux puisque nous cassons les corrélations a priori entre θ et $x^\star$.

Considérons par exemple le modèle hiérarchique linéaire:

$$\begin{aligned} X &\sim \mathcal{N}(\theta, \sigma_x^2) \\ Y &\sim \mathcal{N}(X, \sigma_y^2) \end{aligned} \tag{1.19}$$

où X, Y, θ sont des variables aléatoires à valeurs réelles σ_x, σ_y sont des réels strictement positifs. Nous utilisons une loi a priori plate sur θ et nous nous intéressons à sa loi a posteriori sachant les variables observées $\pi(\theta|y)$. Il est facile de montrer que $\int_{\mathbb{R}} \pi(\theta|y)\mathrm{d}\theta < \infty$. Supposons de plus que nous utilisons une stratégie de data augmentation et que nous ciblons la loi jointe a posteriori $\pi(\theta, x|y)$ avec un échantillonneur de Gibbs. Un calcul simple montre que la loi jointe a posteriori de (θ, X) est une loi Gaussienne avec matrice de précision:

$$\boldsymbol{Q}_{\mathrm{c}} = \begin{pmatrix} \frac{1}{\sigma_x^2} & -\frac{1}{\sigma_x^2} \\ -\frac{1}{\sigma_x^2} & \frac{1}{\sigma_x^2} + \frac{1}{\sigma_y^2} \end{pmatrix}$$

ce qui donne la matrice de covariance:

$$\boldsymbol{\Sigma}_{\mathrm{c}} = \begin{pmatrix} \sigma_x^2 + \sigma_y^2 & \sigma_y^2 \\ \sigma_y^2 & \sigma_y^2 \end{pmatrix}.$$

Par Section 1.2.2 nous savons que le taux de convergence de cet échantillonneur de Gibbs est donné par la corrélation maximale a posteriori entre les variables X et θ. De plus, puisqu'elles sont jointement Gaussiennes, cette corrélations maximale et atteinte par les fonctions linéaires de X et θ et parce que les fonctions linéaires n'affectent pas les corrélation entre variables Gaussiennes, le taux de convergence de l'échantillonneur de Gibbs est donné par:

$$\rho_{\mathrm{c}} = \frac{\sigma_y^2}{\sigma_y^2 + \sigma_x^2}.$$

Nous savons que ce taux de converge est élevé quand la loi a priori est beaucoup plus informative que la vraisemblance (c'est à dire quand la variance a priori σ_x^2 est plus faible que σ_y^2). L'opposé est vrai: le taux de convergence est bas lorsque la vraisemblance est beaucoup plus informative que la loi a priori. Nous pouvons aussi réécrire: Eq. (1.19) dans une paramétrisation non centrée:

$$\begin{aligned} X^\star &\sim \mathcal{N}(0, \sigma_x^2) \\ Y &\sim \mathcal{N}(\theta + X^\star, \sigma_y^2). \end{aligned}$$

Les variables θ et X sont maintenant a priori indépendantes et toutes les corrélations a posteriori viennent de la vraisemblance. En suivant le même calcul que pour la paramétrisation centrée, nous obtenons le taux de convergence pour l'échantillonneur de Gibbs bivarié avec la paramétrisation non centrée:

$$\rho_{\mathrm{nc}} = \frac{\sigma_x^2}{\sigma_y^2 + \sigma_x^2} = 1 - \rho_{\mathrm{c}}.$$

Il est clair que la paramétrisation non centrée a le comportement opposé à la paramétrisation centrée: lorsque la vraisemblance est très informative comparée à la loi a priori, l'échantillonneur de Gibbs mélange très mal. Au contraire, lorsque la loi a priori est plus informative que la vraisemblance, l'échantillonneur de Gibbs mélange bien.

Une façon de bénéficier des propriétés de l'échantillonneur de Gibbs centré et non centré est d'utiliser l'algorithme d'entrelacement: nous échantillonnons d'abord θ et x avec l'échantillonneur de Gibbs centré, nous utilisons ensuite une transformation η pour changer de paramétrisation afin d'obtenir $(\theta, x^\star)$ et finalement nous échantillonnons θ sachant $x^\star$ comme dans l'échantillonneur de Gibbs non centré, voir Yu and Meng (2011). Ce faisant, nous avons "le meilleur des deux mondes".

Algorithm 3: Etape n de l'algorithme d'entrelacement

Input: Etat actuel θ_{n-1}, densité cible π
Output: Prochain état θ_n

1 Simuler $x_n \sim \pi(x|\theta_{n-1}, y)$
2 Simuler $\theta_{n-0.5} \sim \pi(\theta|x_n, y)$
3 Calculer $x_n^\star = \eta^{-1}(x_n, \theta_{n-0.5}, y)$
4 Simuler $\theta_n \sim \pi(\theta|x_n^\star, y)$

Chapter 2.

Introduction

This section introduces the concepts needed to understand the rest of this thesis. We first discuss Bayesian statistics and Markov chain Monte-Carlo (MCMC) methods. Then, we explain the basics of Cosmic Microwave Background (CMB) data analysis and the use of MCMC methods in that context. Finally, we summarize the contributions of the two papers in the last two sections.

2.1. Bayesian statistics

The purpose of Bayesian inference is to extract knowledge about the world from experiments, see Robert (2007) for an in-depth review. Each experiment can be mathematically described as a tuple $(\mathcal{Y}, \mathcal{B}(\mathcal{Y}), \{\mathbb{P}_\theta, \theta \in \Theta\})$ where $\mathcal{Y}$ is the observational space and $\mathcal{B}(\mathcal{Y})$ is the Borel sigma-algebra of $\mathcal{Y}$. The set Θ is called the parameter space and $\{\mathbb{P}_\theta, \theta \in \Theta\}$ is a family of probability measures on $(\mathcal{Y}, \mathcal{B}(\mathcal{Y}))$. In the rest of this manuscript we will only consider $\mathcal{Y} = \mathbb{R}^d$ with $d \in \mathbb{N}^\star$, $\Theta \subseteq \mathbb{R}^m$ and assume that $\mathbb{P}_\theta$ is dominated by the Lebesgue measure $\mathrm{d}y$ for any $\theta \in \Theta$.

Bayesian inference quantifies the uncertainty on the parameter θ by updating the prior belief of the experimenter thanks to the data. If we regard θ as a random variable on $(\Theta, \mathcal{B}(\Theta), \mathrm{d}\theta)$, the experimenter can formalize its prior belief about the parameter, before any observation, through the prior measure $p_0(\theta)\mathrm{d}\theta$, where $p_0(\theta)$ is the density of θ with respect to $\mathrm{d}\theta$. Assuming we observe y a realization of the random variable $Y \in \mathcal{Y}$, we can define the likelihood function:

$$\begin{aligned} p\colon \Theta \times \mathcal{Y} &\to [0, +\infty) \\ (\theta, y) &\mapsto p(y|\theta) \end{aligned} \tag{2.1}$$

The data y can constrain the prior uncertainty through the likelihood function. We can describe the a posteriori uncertainty about the parameter through the posterior distribution:

$$\pi(\theta|y) = \frac{p(y|\theta)p_0(\theta)}{Z(y)} \tag{2.2}$$

where

$$Z(y) = \int_\Theta p(y|\theta)p_0(\theta)\mathrm{d}\theta \tag{2.3}$$

is called the evidence or marginal likelihood. Note that since p_0 is a probability density function, we necessarily have $Z(y) < \infty$. It is possible to choose p_0 such that:

$$\int_\Theta p_0(\theta)\mathrm{d}\theta = \infty.$$

In this case, it is necessary to check that $Z(y) < \infty$ so that Eq. (2.3) is well defined.
In practice, we are interested in the moments of the posterior distribution:

$$\mathbb{E}[h(\theta)|y] = \int_\Theta h(\theta)\pi(\theta|y)\mathrm{d}\theta \tag{2.4}$$

where $h : \Theta \to \mathbb{R}$ for example, and in estimating the marginal likelihood Eq. (2.3). We may also want a point estimate of our parameter and take the Maximum A Posteriori (MAP) estimator:

$$\theta^{\mathrm{MAP}} \in \arg\max_{\theta} \pi(\theta|y).$$

Obviously, our ability to do Bayesian inference relies on our ability to compute expectations like Eq. (2.4). This is difficult in general and oftentimes the posterior distribution Eq. (2.2) is not even tractable. In this case we have two options: characterizing the posterior distribution through sampling or approximating the posterior distribution through a tractable family of distributions. In the next section we focus on the former option. The interested reader can read Martin J Wainwright (2008) for an introduction to the latter.

2.2. Markov chain Monte-Carlo methods

Since $\pi(\theta|y)$ is a density over Θ, in this section we drop the dependence on y for the sake of brevity. So we consider a density $\pi(\theta)$ instead of $\pi(\theta|y)$.

The Markov chain Monte-Carlo (MCMC) methods aim at computing expectations over the state space Θ:

$$\mathbb{E}_{\pi}[h(\theta)] = \int_{\Theta} h(\theta)\pi(\theta)\mathrm{d}\theta \tag{2.5}$$

where π is a density over Θ and, for example, $h : \Theta \to \mathbb{R}$. To do so, MCMC methods produce dependent samples from π and estimate Eq. (2.5) with the empirical mean of the sample. More precisely, MCMC methods build a Markov chain $(\theta)_{n\geqslant 1}$ such that:

$$\frac{1}{N}\sum_{n=0}^{N} h(\theta_n) \xrightarrow[N\to\infty]{\mathrm{a.s}} \mathbb{E}_{\pi}[h(\theta)]. \tag{2.6}$$

If the transition kernel P of the Markov chain leaves π invariant:

$$\int_{\Theta} \pi(\theta)P(\theta, \mathrm{d}\theta')\mathrm{d}\theta = \pi(\theta')\mathrm{d}\theta'$$

and the chain is irreducible and aperiodic, then we know that for $\pi - \mathrm{a.e}$ every $\theta_0 \in \Theta$:

$$||P^n(\theta_0, \mathrm{d}\theta) - \pi(\theta)\mathrm{d}\theta||_{\mathrm{TV}} \xrightarrow[n\to\infty]{} 0$$

where $||.||_{\mathrm{TV}}$ is the total variation norm, see e.g Roberts and Rosenthal (2004). This means that for $\pi - \mathrm{a.e}$ every starting points, the chain converges to its invariant distribution. Under the stronger assumption that the chain is Harris recurrent, this result holds for every $\theta_0 \in \Theta$, see e.g Roberts and Rosenthal (2004). On top of that result, we can also characterize the speed of convergence to the stationary distribution: if the chain possesses a small set and satisfies the drift condition on this set, it is geometrically ergodic, that is:

$$||P^n(\theta_0, \mathrm{d}\theta) - \pi(\theta)\mathrm{d}\theta||_{\mathrm{TV}} \leqslant M(\theta_0)\rho^n$$

for some $\rho \in [0, 1[$ and $M(\theta_0) < \infty$ for $\pi - \mathrm{a.e}$ every $\theta_0 \in \Theta$. In the case M does not depend on θ_0, the chain is said to be uniformly ergodic. Finally, we have a Central Limit Theorem (CLT):

$$\frac{1}{\sqrt{N}}\sum_{n=0}^{N} \{h(X_n) - \mathbb{E}_{\pi}[h(\theta)]\} \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}(0, \sigma_h^2)$$

for $\pi - \mathrm{a.e}$ every $\theta_0 \in \Theta$, where

$$\sigma_h^2 = \mathrm{Var}_{\pi}(h) + 2\sum_{n=1}^{\infty} \mathrm{Cov}(h(\theta_0), h(\theta_n)) \tag{2.7}$$

under the assumption that the chain is uniformly ergodic and that $\mathbb{E}_\pi[h(\theta)] < \infty$, see Meyn and Tweedie (2014). In the case the chain is only geometrically ergodic, the CLT still holds if $\mathbb{E}_\pi[h(\theta)] < \infty$ and the chain is reversible, that is:

$$\pi(\mathrm{d}\theta_0)P(\theta_0, \mathrm{d}\theta_1) = \pi(\mathrm{d}\theta_1)P(\theta_1, \mathrm{d}\theta_0)$$

for any $\theta_0, \theta_1 \in \Theta$. Finally, if

$$L_0^2(\pi) = \{h(\theta) : \mathbb{E}_\pi[h(\theta)] = 0; \mathrm{Var}_\pi[h(\theta)] < \infty\}$$

is the Hilbert space of all zero-mean scalar functions with finite variance and inner product $< h(\theta), t(\theta) >= \mathbb{E}_\pi[h(\theta)t(\theta)]$ such that $||h(\theta)||_{L_0^2(\pi)} =< h(\theta), h(\theta) >$, we can define the forward operator associated to the chain as:

$$F^n h(x) = \mathbb{E}[h(\theta_n)|\theta_0 = x] = \int_\Theta h(\theta)P^n(x, \mathrm{d}\theta) \tag{2.8}$$

for any $n \in \mathbb{N}^\star$. Liu et al. (1995) have shown that at stationarity, that is when we assume that $\theta_0 \sim \pi$ and that $\theta_n|\theta_0 \sim P^n(\theta_0, \mathrm{d}\theta)$, we have:

$$||F^n|| = \gamma_n \tag{2.9}$$

where $||.||$ is the operator norm defined as:

$$||F|| = \sup_{t \in L_0^2(\pi)} \frac{||Ft||_{L_0^2(\pi)}}{||t||_{L_0^2(\pi)}}$$

and

$$\gamma_n := \sup_{f,g} \mathrm{Corr}(f(\theta_0), g(\theta_n)) \tag{2.10}$$

is the maximal correlation coefficient between θ_0 and θ_n, with the supremum being taken over all scalar functions with finite variance. We see that the operator norms of the powers of F, all inferior or equal to 1, are directly bounding the lag-n autocorrelations of the chain. Finally, Liu et al. (1995) have shown that if the chain is reversible, F is self-adjoint, and in that case:

$$||F||^n = \gamma_n.$$

In the next two subsections we give two examples of Markov chain Monte Carlo algorithms with the properties we just described: the Metropolis-Hastings algorithm and the Gibbs sampler algorithm.

2.2.1. Metropolis-Hastings algorithm

A popular way to construct a Markov chain as described in the previous section is the Metropolis-Hastings algorithm described by Hastings (1970): at each step $n \in \mathbb{N}$, a new state θ' is proposed according to a proposal distribution $Q(\theta_{n-1}, \mathrm{d}\theta)$. We accept the new state, that is $\theta_n = \theta'$, with a probability that depends on the ratio of the density evaluated in the proposed move on the density of the previous state θ_{n-1}.
Depending on the choice of Q, the algorithm has different properties. Practitioners usually choose $Q(\theta_{n-1}, \mathrm{d}\theta) \overset{\mathcal{L}}{=} \mathcal{N}(\theta_{n-1}, \tau\Sigma)$, where $\tau \in \mathbb{R}_+$ is the scale parameter and $\tau\Sigma$ is the covariance matrix of the proposal distribution, both user-chosen. We must choose Σ so that the covariance of the target distribution roughly matches the covariance of the proposal. This will help proposing moves along the principal direction of the target distribution. The parameter τ is chosen so that the algorithm has the desired acceptance rate. This algorithm is called the Random-Walk Metropolis-Hastings algorithm, see Tierney (1994): at each time step we perturb the current state θ_n with Gaussian noise and get a proposed state θ_{n+1}. If the proposed state is in a region of higher density than the previous state, we always accept the move. Otherwise, we accept it with a probability proportional to the ratio of the densities evaluated in the old and proposed state, see Algorithm 4. Obviously, we need to be able to evaluate at least the unnormalized

Algorithm 4: Step n of RWMH algorithm

Input: Current state θ_{n-1}, target density π
Output: Next state θ_n

Sample $\theta^{'} \sim \mathcal{N}(\theta_{n-1}, \tau\Sigma)$
Compute $r(\theta_{n-1}, \theta^{'}) = \min\left\{\frac{\pi(\theta^{'})}{\pi(\theta_{n-1})}, 1\right\}$
Sample $u \sim \mathcal{U}[0,1]$
if $u < r(\theta_{n-1}, \theta^{'})$ **then**
    set $\theta_n \leftarrow \theta^{'}$
else
    set $\theta_n \leftarrow \theta_{n-1}$

version of π to implement a Metropolis-Hastings algorithm.

This algorithm leaves the target π invariant by construction. Indeed, its Markov transition kernel is given by:

$$\begin{aligned} P_{\mathrm{RWMH}}(\theta_{n-1}, \mathrm{d}\theta_n) =& r(\theta_{n-1}, \theta_n) Q(\theta_{n-1}; \mathrm{d}\theta_n) + \delta_{\theta_{n-1}}(\mathrm{d}\theta_n) \\ & \times \left\{1 - \int_{\Theta} r(\theta_{n-1}, \theta^{'}) Q(\theta_{n-1}; \mathrm{d}\theta^{'})\right\} \end{aligned}$$

where the function r is defined in Algorithm 4. It is then straightforward to show that:

$$\pi(\mathrm{d}\theta_{n-1}) P_{\mathrm{RWMH}}(\theta_{n-1}, \mathrm{d}\theta_n) = \pi(\mathrm{d}\theta_n) P_{\mathrm{RWMH}}(\theta_n, \mathrm{d}\theta_{n-1}) \tag{2.11}$$

that is, P_{RWMH} verifies the detailed balance condition. Integrating both sides of Eq. (2.11) with respect to θ_{n-1} gives the π-invariance of P_{RWMH} and since the proposal distribution is continuous and positive on $\Theta \times \Theta$, it is also irreducible. It follows that the algorithm is also aperiodic. Tierney (1994) describes the theoretical properties of this algorithm.

In the next subsection we discuss another popular way of building a Markov chain with the desired properties: the Gibbs sampler algorithm.

2.2.2. The Gibbs sampler algorithm

If $\Theta \subseteq \mathbb{R}^m$ is multidimensional and we know the conditionals $\pi(\theta^i|\theta^{-i})$ of the target π, where θ^i denotes the i-th components of θ and θ^{-i} denotes all the components excepted the i-th one, we can implement a Gibbs sampler, see Algorithm 5. This algorithm is called the systematic scan Gibbs sampler. The random scan

Algorithm 5: Step n of the Gibbs sampler

Input: Current state θ_{n-1}, target density π
Output: Next state θ_n

Sample $\theta_n^1 \sim \pi(\theta^1|\theta_{n-1}^{-1})$
Sample $\theta_n^2 \sim \pi(\theta^2|\theta_n^1, \theta_{n-1}^3, \ldots, \theta_{n-1}^m)$
Sample $\theta_n^3 \sim \pi(\theta^3|\theta_n^1, \theta_n^2, \theta_{n-1}^4, \ldots, \theta_{n-1}^m)$
$\vdots$
Sample $\theta_n^m \sim \pi(\theta^m|\theta_n^{-m})$

variant consists in drawing randomly the conditional to sample from, making the algorithm reversible. Both

algorithms leave π invariant. However, we must check that the algorithm is irreducible and aperiodic. If a conditional distribution of π has no known form, we can replace a direct sampling by a Metropolis-Hastings algorithm targeting this conditional for any predetermined number of steps, implementing a Metropolis-within-Gibbs algorithm. Since the Metropolis-Hastings algorithm leaves the right conditional invariant, the Metropolis-within-Gibbs algorithm still leaves π invariant. However, while the Gibbs sampler is tuning-free, introducing a Metropolis-Hastings step to sample from one of the conditional distribution introduces the need of tuning its proposal distribution. In general, the tuning of a Metropolis-within-Gibbs algorithm is more difficult than the tuning of a Metropolis-Hastings aglorithm: at each iteration, the parameters of the conditional distribution targeted by the Metropolis step changes. This means that a different tuning of the proposal distribution is required at each iteration, while the Metropolis-Hastings algorithm requires tuning the proposal distribution once and for all.

Instead of sampling from each univariate conditional, we can partition the coordinates into subsets and sample in turn each subset given the others. This algorithm is similar as Algorithm 5 except that the conditional distributions are now multivariate. Liu et al. (1995) have shown that under mild conditions, the systematic scan Gibbs sampler is geometrically ergodic with rate:

$$\rho = r := \lim_{n\to\infty} ||F^n||^{1/n} < 1$$

where F is the forward operator of the chain defined in Eq. (2.8) and r is called the spectral radius of the operator F. Under mild conditions, the random scan Gibbs sampler is geometrically ergodic with rate:

$$\rho = ||F|| < 1.$$

These two previous results together with Eq. (2.9) suggest that we could use the lag-n autocorrelations of the systematic scan Gibbs sampler and the lag-1 autocorrelation of the random scan Gibbs sampler to estimate their respective geometric rate of convergence.

The case where the coordinates are partitioned into two subsets $\theta = (\theta^1, \theta^2)$ is well studied and understood, see Liu (1994) and Liu et al. (1994). For simplicity, we will consider the case where $\Theta \subseteq \mathbb{R}^2$.
In this case the processes $(\theta_n^1)_{n\geqslant 1}$ and $(\theta_n^2)_{n\geqslant 1}$ are reversible Markov chains. It has been shown that at stationarity:

$$\gamma_1 = \gamma_\pi \tag{2.12}$$

where γ_1 is defined in Eq. (2.10) and

$$\gamma_\pi = \sup_{f,g} \mathrm{Corr}(f(\theta^1), g(\theta^2))$$

where the supremum is taken over all scalar functions of finite variance and (θ^1, θ^2) is distributed according to π. In addition, we have:

$$\gamma_{\pi_1} = \gamma_{\pi_2} = \gamma_\pi^2 \tag{2.13}$$

with, for $i \in \{1, 2\}$:

$$\gamma_{\pi_i} = \sup_{f,g} \mathrm{Corr}(f(\theta_n^i), g(\theta_{n+1}^i))$$

where the supremum is taken over all scalar function with finite variance and stationarity is assumed. It follows that:

$$||F||^2 = ||F_1|| = ||F_2|| = \gamma_\pi^2$$

where γ is the coefficient of maximal correlation defined in Eq. (2.10) and F, F_1 and F_2 are the forward operators of the chains $(\theta_n)_{n\geqslant 1}$, $(\theta_n^1)_{n\geqslant 1}$ and $(\theta_n^2)_{n\geqslant 1}$ respectively. The result Eq. (2.12) states that the maximal correlation between two successive Gibbs samples is equal to the maximal correlation of the two components of the target density π. The result Eq. (2.13) implies that the maximal correlation coefficient between successive samples from the subchains $(\theta_n^1)_{n\geqslant 1}$ and $(\theta_n^2)_{n\geqslant 1}$ directly depend on the maximal

correlation between the two components of the target distribution. In addition, we know that at stationarity, the lag-1 autocorrelation is given by:

$$\gamma_{\pi_1} = \sup_{h:\, \mathrm{Var}(h(\theta))<\infty} \left\{ 1 - \frac{\mathbb{E}[\mathrm{Var}(h(\theta^1)|\theta^2)]}{\mathrm{Var}[h(\theta^1)]} \right\} \tag{2.14}$$

where the right hand term is called the fraction of missing information, see Liu (1994).
In addition, it has been shown by Liu (1994) and Liu et al. (1994) that the spectral radii of the forward operators of the chains $(\theta_n^1)_{n\geqslant 1}$, $(\theta_n^2)_{n\geqslant 1}$ and $(\theta_n)_{n\geqslant 1}$ are all equal to γ_π^2. This means that the geometric rate of convergence of the Gibbs sampler and its subchains will depend on the strength of the correlations between its two components. In addition, by Eq. (2.13), the strength of these correlations appear through the conditional variance: if θ^1 and θ^2 are highly correlated, the support of the distribution of θ^1 given θ^2 is small compared to the marginal posterior variance of θ^1 and the Gibbs sampler does not explore efficiently the entire support of the joint distribution. This translates into a fraction of missing information close to one and hence a geometric convergence rate close to one.

2.2.3. Control variates

We can reduce the variance of Monte Carlo estimates using control variates, see Robert and Casella (2004). Suppose we have $J \in \mathbb{N}$ functions h_j such that, for any $j \in \{1, \ldots, J\}$:

$$\mathbb{E}_{\pi(\theta|y)}[h_j(\theta)] = 0$$

called control variates. Then the estimator

$$\hat{p}_\beta := \frac{1}{N} \sum_{n=1}^{N} \{ f(\theta_n) + \beta^t h(\theta_n) \} \tag{2.15}$$

is such that

$$\hat{p}_\beta \xrightarrow[N\to\infty]{\text{a.s}} \mathbb{E}[f(\theta) + \beta^t h(\theta)] = \mathbb{E}[f(\theta)].$$

where $\beta \in \mathbb{R}^J$ and $h(\theta) = (h_1(\theta), \ldots, h_J(\theta)) \in \mathbb{R}^J$ .We must choose β and h so that the variance of Eq. (2.15) is reduced compared to the usual ergodic average in Eq. (2.6). Two scenarii are possible.

The first one arises when the sampled points $(\theta)_{n\geqslant 0}$ have been sampled i.i.d like in rejection sampling or importance sampling. In this case:

$$\mathrm{Var}(\hat{p}_\beta) = \frac{1}{N} \{ \mathrm{Var}(f(\theta)) + \beta^t \mathrm{Var}(h(\theta))\beta + 2\beta^t \mathrm{Cov}(h(\theta), f(\theta)) \} \tag{2.16}$$

where $\mathrm{Var}(h(\theta))$ is the $J \times J$ variance matrix of the vector $h(\theta)$ and $\mathrm{Cov}(h(\theta), f(\theta))$ is the $J \times 1$ vector with $\mathrm{Cov}(h(\theta), f(\theta))_{i,1} = \mathrm{Cov}(h_i(\theta), f(\theta))$. Differentiating against β to find the value leading to the minimum variance we get:

$$\beta^\star = -\mathrm{Var}(h(\theta))^{-1} \mathrm{Cov}(h(\theta), f(\theta)).$$

Of course, the two quantities in the right hand term are not necessarily known and we may need to estimate $\beta^\star$ using the sample $(\theta)_{n\geqslant 0}$ as done by Owen (2013):

$$\hat{\beta} = -\hat{\Sigma}_{h,h}^{-1} \hat{\Sigma}_{h,f}$$

where $\hat{\Sigma}_{h,h}$ and $\hat{\Sigma}_{h,f}$ are estimates of $\mathrm{Var}(h(\theta))$ and $\mathrm{Cov}(h(\theta), f(\theta))$ based on $(\theta)_{n\geqslant 0}$, respectively. One should note that the new estimator:

$$\hat{p}_{\hat{\beta}} := \frac{1}{N} \sum_{n=1}^{N} \{ f(\theta_n) + \hat{\beta}^t h(\theta_n) \}$$

is only asymptotically unbiased, see Owen (2013).

The second scenario happens when $(\theta)_{n\geqslant 0}$ have been sampled with MCMC methods. The sampled points are no longer independent and the variance of the estimator Eq. (2.15) is no longer Eq. (2.16) but rather of the form Eq. (2.7). We now want to minimize the asymptotic variance of Eq. (2.15):

$$\begin{aligned}\sigma^2_{f+\beta^t h} =& \mathrm{Var}_\pi(f(\theta_n) + \beta^t h(\theta_n)) \\ &+ 2\sum_{n=1}^{\infty} \mathrm{Cov}(f(\theta_0) + \beta^t h(\theta_0), f(\theta_n) + \beta^t h(\theta_n))\end{aligned}$$

as a function of β. This is not easy in general. When the transition kernel associated to the Markov chain is reversible with respect to the target, Dellaportas and Kontoyiannis (2011) propose a way to estimate the optimal $\beta^\star$. Another option is to simply ignore the correlations between the successive samples and fall back on the first scenario. However, the estimated $\beta^\star$ does not minimize the asymptotic variance in this case.

Several choices for h have been proposed in the literature. When the conditional means of the target densities are known, Dellaportas and Kontoyiannis (2011) propose to take:

$$h_i(\theta) = \theta^i - \mathbb{E}[\theta^i|\theta^{-i}]$$

where θ^i denotes the i-th component of θ, θ^{-i} denotes all components of θ except the i-th one and the expectation is taken under $\pi(\theta^i|\theta^{-i}, y)$.

When the score function:

$$s_\pi(\theta) = \nabla_\theta \log \pi(\theta|y)$$

is available, the functions:

$$h(x) = \nabla_\theta \cdot \phi(\theta) + \phi(\theta) \cdot s_\pi(\theta)$$

are control variates, for any function ϕ such that

$$\oint_{\partial\Theta} p(\theta)\phi(\theta) \cdot n(\theta) S(\mathrm{d}\theta) = 0$$

where $\oint_{\partial\Theta}$ denotes the integral over the boundary of Θ, and $S(\mathrm{d}\theta)$ is the surface element at $\theta \in \partial\Theta$ and under the condition that the probability density $\pi(.|y) \in C^1(\Theta, \mathbb{R})$.

We do not have to restrict ourselves to a finite basis of control variates functions $\{h_1(\theta), \ldots, h_J(\theta)\}$. For example Oates et al. (2016) propose to build a non-parametric approximation of f. They split the sample $(\theta)_{0\leqslant n\leqslant N}$ in two disjoint subsets $\mathcal{D}_0$ and $\mathcal{D}_1$ of size $m+1$ and $N-m-1$ respectively, and use $\mathcal{D}_0$ to get a non parametric approximation of f:

$$s_{f,\mathcal{D}_0} := \arg\min_{g\in\mathcal{H}_+} \left\{ \frac{1}{m}\sum_{n=0}^{m} (f(\theta_n) - g(\theta_n))^2 + \lambda ||g||_{\mathcal{H}_+} \right\}$$

where $\lambda > 0$, $\mathcal{H}_+$ is a Hilbert space of functions written as the sum of a constant function plus another function verifying the Stein trick and $||.||_{\mathcal{H}_+}$ is the norm on $\mathcal{H}_+$. If we denote:

$$\mu(s_{f,\mathcal{D}_0}) := \mathbb{E}_{\pi(.|y)}[s_{f,\mathcal{D}_0}(\theta)]$$

under some conditions, among which that $f \in \mathcal{H}_+$, the variance of the new estimator:

$$\hat{\mu}(\mathcal{D}_0, \mathcal{D}_1; f) = \frac{1}{N-m-1}\sum_{n=m+1}^{N} \{f(\theta_n) - s_{f,\mathcal{D}_0}(\theta_n)\} + \mu(s_{f,\mathcal{D}_0})$$

is $\mathcal{O}(N^{-7/6})$, outperforming the $\mathcal{O}(N^{-1})$ of the traditional MCMC ergodic average.

A similar method is employed by Mira et al. (2013) for building zero-variance control variates. Suppose we choose a Hermitian operator H acting on the infinitely differentiable functions with compact support such that:

$$H\sqrt{\pi} = 0$$

and a function ψ compactly supported and infinitely differentiable. Then:

$$\tilde{f}(\theta) = f(\theta) + \frac{H\psi}{\sqrt{\pi(\theta|y)}}$$

is such that:

$$\mathbb{E}_{\pi(\theta|y)}[\tilde{f}(\theta)] = \mathbb{E}_{\pi(\theta|y)}[f(\theta)].$$

The best possible choice of a couple (H, ψ) is such that $\tilde{f}$ has zero variance, which happens when:

$$H\psi = -\sqrt{\pi(\theta|y)}\left\{f(\theta) - \mathbb{E}_{\pi(\theta|y)}[f(\theta)]\right\}. \tag{2.17}$$

Mira et al. (2013) propose several choices of H and then to find the function ψ that verifies Eq. (2.17). Unfortunately, for a given H the optimal function ψ cannot be obtained explicitly in general. We can then restrict ψ to a parametric family of functions and minimize the variance of $\tilde{f}$ with respect to these parameters. In this case, the variance of the new estimator is not zero anymore and we must verify that the new estimator is unbiased, see Mira et al. (2013) for more details.

2.3. Data Augmentation

It happens sometimes that we can rewrite the likelihood Eq. (2.1) as:

$$\pi(y|\theta) = \int_{\mathcal{X}} p(y|x,\theta)p(x|\theta)\mathrm{d}x \tag{2.18}$$

where $p(x|\theta)$ is a density on the measure space $(\mathcal{X}, \mathcal{B}(\mathcal{X}), \mathrm{d}x)$ where $\mathcal{X} \subseteq \mathbb{R}^q$ for $q \in \mathbb{N}\backslash\{0\}$. This amounts to introducing a new random variable X on $\mathcal{X}$ with conditional distribution $p(x|\theta)\mathrm{d}x$, called the latent variable.

Instead of targeting the posterior distribution Eq. (3.3), we may now target the joint posterior distribution:

$$\pi(\theta, x|y) = \frac{p(y|x,\theta)p(x|\theta)p(\theta)}{Z(y)}.$$

We have several reasons to do that: the observed likelihood Eq. (2.1) may be intractable, we may have a scientific interest in doing inference on X or the MCMC methods would be more efficient targeting the joint posterior distribution rather than the posterior on the parameter. Targeting the joint posterior distribution is called a Data Augmentation scheme, see Liu et al. (1994). In the following we will assume that the data and the parameters are independent given the latent variable:

$$p(y|x,\theta) = p(y|x)$$

and we draw the Directed Acyclic Graph (DAG) of the model Fig. 2.1. In this context, a DAG is a graphical representation of a hierarchical model: a round node depicts an unobserved variable, a square node depicts an observed variable, plain arrows represent stochastic dependence relationships between two variables and dashed arrows represent deterministic relationships. Note that there cannot be directed cycles in such a graph, see Whittaker (1990).

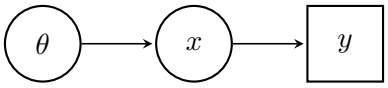

Figure 2.1.: Directed acyclic graph of the Eq. (2.18)

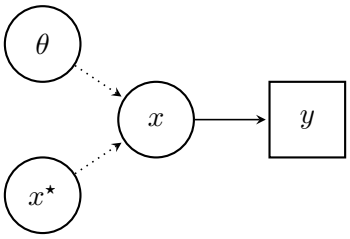

Figure 2.2.: Directed acyclic graph of the non centered parametrization.

When we can sample x given (θ, y) as well as θ given (x, y), we can implement a two-steps Gibbs sampler. As discussed in the previous section, the higher the posterior correlations between the latent variable and the parameter, the poorer the geometrical convergence rate of the Gibbs sampler. When this happens, we need to break the posterior correlations to improve the rate of convergence of the Gibbs sampler. To do this, we can reparametrize the model: we find a random pair $(X^\star, \theta)$ with joint prior density $p_0(X^\star, \theta)$ and a function η such that:

$$x = \eta(x^\star, \theta).$$

see Papaspiliopoulos et al. (2007) for example. Note that η, for a fixed θ, need not to be one-to-one. We can then target the posterior distribution $\pi(\theta, x^\star|y)$ and if we made a good choice of η, the posterior correlations between θ and $X^\star$ should be lower than the posterior correlations between θ and X. A usually useful parametrization is the non-centered parametrization, depicted in Fig. 2.2. In this parametrization, the parameter and the latent variable are now a priori independent and the correlations between the two are only coming from the data.

When the Gibbs sampler on centered parametrization, called the centered Gibbs, mixes well, the Gibbs sampler on the non centered parametrization, called the non-centered Gibbs, usually mixes badly: when the data is very informative about the latent variable, it may be sufficient to break the prior correlations between θ and x and the centered Gibbs sampler will have a low geometrical rate of convergence. While the non-centered Gibbs will mix poorly because θ and x^* will be strongly correlated a posteriori, since the likelihood is very constraining.

The opposite is usually true: when the prior distribution shows stronger correlations between θ and x than the likelihood identifies x, the centered Gibbs sampler will be inefficient because of a high fraction of missing information Eq. (2.14). The non-centered Gibbs will mix better since we broke the prior correlations between θ and $x^\star$.

Consider for example the simple linear hierarchical model:

$$\begin{aligned} X &\sim \mathcal{N}(\theta, \sigma_x^2) \\ Y &\sim \mathcal{N}(X, \sigma_y^2) \end{aligned} \tag{2.19}$$

where X, Y, θ are real-valued random variables and σ_x, σ_y are strictly positive real numbers. We set a flat a prior on θ and we are interested in its posterior distribution given the observed variable $\pi(\theta|y)$. It is straightforward to show that $\int_{\mathbb{R}} \pi(\theta|y)\mathrm{d}\theta < \infty$. Suppose in addition that we use a data augmentation scheme and target the joint posterior distribution $\pi(\theta, x|y)$ with a Gibbs sampler. A simple calculation

shows that this joint posterior distribution of (θ, X) is a Gaussian distribution with precision matrix:

$$\boldsymbol{Q}_{\mathrm{c}} = \begin{pmatrix} \frac{1}{\sigma_x^2} & -\frac{1}{\sigma_x^2} \\ -\frac{1}{\sigma_x^2} & \frac{1}{\sigma_x^2} + \frac{1}{\sigma_y^2} \end{pmatrix}$$

which yields the covariance matrix:

$$\boldsymbol{\Sigma}_{\mathrm{c}} = \begin{pmatrix} \sigma_x^2 + \sigma_y^2 & \sigma_y^2 \\ \sigma_y^2 & \sigma_y^2 \end{pmatrix}.$$

We know from Section 2.2.2 that this Gibbs sampler is reversible and that its rate of convergence is given by the maximal posterior correlation between X and θ. In addition, since they are jointly Gaussian, this maximal correlation is attained by linear functions of X and θ and because linear functions do not affect the correlations between two jointly Gaussian distributed variables, the rate of convergence of the Gibbs sampler is given by:

$$\rho_{\mathrm{c}} = \frac{\sigma_y^2}{\sigma_y^2 + \sigma_x^2}.$$

We see that this rate of convergence is high when prior is much more informative than the likelihood (that is when the prior variance σ_x^2 is much lower than σ_y^2). The converse is true: the rate of convergence is low when the likelihood is much more informative than the prior. We can also rewrite Eq. (2.19) in a non centered parametrization:

$$X^\star \sim \mathcal{N}(0, \sigma_x^2)$$

$$Y \sim \mathcal{N}(\theta + X^\star, \sigma_y^2).$$

The variables θ and X are now a priori independent and all the posterior correlations come from the likelihood. Following the same calculation as for the centered parametrization, we get the following convergence rate for the two step Gibbs sampler under the non centered parametrization:

$$\rho_{\mathrm{nc}} = \frac{\sigma_x^2}{\sigma_y^2 + \sigma_x^2} = 1 - \rho_{\mathrm{c}}.$$

It is obvious that the non centered parametrization has the opposite behavior as the centered one: when the likelihood is very informative compared to the prior term, the Gibbs sampler mixes very badly. On the contrary, when the prior term is more informative than the likelihood, the Gibbs sampler mixes well.

A way to enjoy the properties of the centered and the non-centered Gibbs samplers is to use the interweaving algorithm: we first sample θ and x with the centered Gibbs sampler, then use our transformation η to change parametrization and get $(\theta, x^\star)$ and finally sample θ given $x^\star$ as in the non-centered Gibbs sampler, see Yu and Meng (2011). Doing so, we have the best of both worlds.

Algorithm 6: Step n of the interweaving algorithm

Input: Current state θ_{n-1}, target density π
Output: Next state θ_n

1 Sample $x_n \sim \pi(x|\theta_{n-1}, y)$
2 Sample $\theta_{n-0.5} \sim \pi(\theta|x_n, y)$
3 Compute $x_n^\star = \eta^{-1}(x_n, \theta_{n-0.5}, y)$
4 Sample $\theta_n \sim \pi(\theta|x_n^\star, y)$

2.4. Cosmology

We now introduce the basics of Cosmic Microwave Background (CMB) analysis. First, we introduce the general context of the problem. Second, we give the technical details of the CMB experiments. Finally, we discuss the statistical model and approaches that have been used so far to make inference about the CMB.

2.4.1. Generalities

Today's cosmologists estimate the age of the universe to be 13 billions years, the birth of which is called the Big Bang. Some 380,000 years after that Big Bang, the universe cooled down, allowing the electrons to bond with protons to form hydrogen, making the universe transparent, and allowing light to travel through the universe. Looking far away in our universe today, we can still observe this light and this signal is called the Cosmic Microwave Background (CMB).

Arno Penzias and Robert W. Wilson accidentally discovered this phenomenon in 1964 while testing a radio equipment for Bell Labs, see Penzias and Wilson (1965). Since this moment, several space missions have been launched to detect the CMB. The first one, called Cosmic Background Explorer (COBE), was a NASA-funded satellite launched in 1989. Other missions were launched afterwards, like the Wilkinson Microwave Anisotropy Probe space observatory (WMAP) in 2001 and the Planck telescope mission in 2009. The next mission consists in a satellite called LiteBird which should be launched in 2027.

In theory, the CMB signal should have faint fluctuations, called anisotropies. In practice, we observe that while the main background is at $2.73K$ where K denotes the Kelvin unit, the anistropies on top are as small as $100\mu K$. The successive missions needed highly sensitive instruments and each mission pushed the sensitivities and resolutions of its instrument to a new level, see Fig. 2.3. In addition, since we are observing light, not only can we record its intensity, but also its polarization.

The intensity and polarization of this light tell us about the early universe. From the map of CMB fluctuations, we can deduce its power spectrum, that is, the intensity of the different wavelengths composing this signal. Once this power spectrum is obtained, we use it to discriminate between cosmological models and cosmological parameters: different models and different parameters predict different power spectra, which we can test against the observed power spectrum. Among the cosmological parameters of interest are the quantity of dark matter, the quantity of dark energy, the age of the universe etc... In the rest of this thesis we will focus on the power spectrum and leave the cosmological parameters aside.

2.4.2. The Cosmic Microwave Background signal

The signal is collected by an instrument spinning around an axis and pointing in different directions across time, covering its observation area (the entire sky for some missions or part of it for others) in a given period of time. This produces the time-ordered data (TOD): the value provided by the detector as a function of time, see Fig. 2.4. The detectors are not perfect though, and they provide a noisy measurement, where the noise also depends on the time and is time-correlated. At this stage, we do not have a map of the sky yet, but only a noisy time-series. Once we have this, we can project it into a noisy spherical map of the sky. For simplicity, it is often assumed that the noise is Gaussian for each pixel with a known standard deviation, mean zero and that the correlations between pixels are negligible. We make this assumption in the rest of the present thesis.

On top of the noisy measurements, a detector does not observe an infinitely precise point in a given direction. Instead it gets information from a small area around that direction. This is what is called the beam. In practice it takes the form of a convolution of the CMB sky map with a a beam function B,

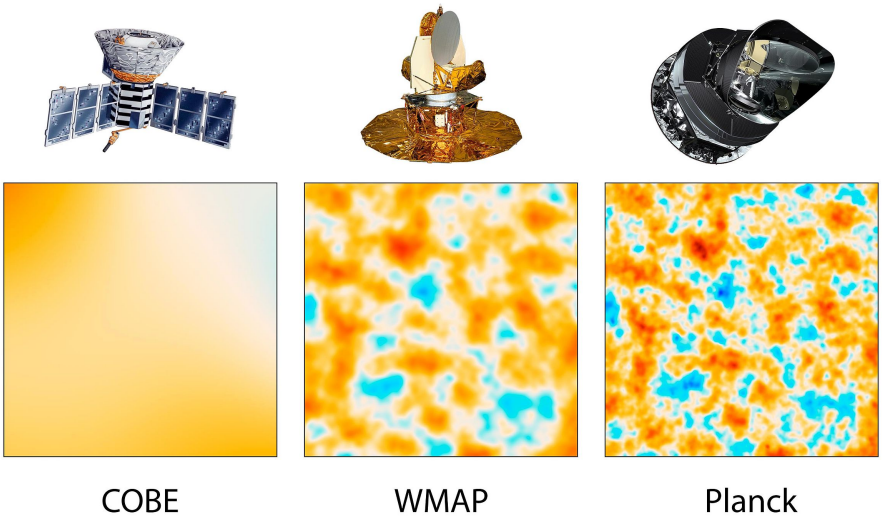

Figure 2.3.: Evolution of the resolution through the missions. The missions are in chronological order from left to right. We can notice the increasing resolution. This is a NASA/JPL-Caltech/ESA illustration. Link: https://photojournal.jpl.nasa.gov/catalog/PIA16874

expressed in spherical coordinates:

$$s_{\text{smoothed}}(\theta,\phi) = \int_{[0,2\pi[}\int_{[0,\pi]} B(\theta-\theta',\phi-\phi')s(\theta,\phi)d\theta' d\phi' \tag{2.20}$$

where $\theta \in [0,\pi]$ is the azimuthal angle (longitude), $\phi \in [0,2\pi[$ is the polar angle (colatitude), s is the noise-free skymap, s_{smoothed} is the convoluted skymap, and

$$B : [0,\pi] \times [0,2\pi] \to \mathbb{R}$$

is expressed in spherical coordinates and is assumed to be independent on the direction the detector is pointing at. Different detectors may have different beam functions.

Another reason why we do not observe exactly the CMB signal is the presence of foreground components. Other sources emit radiation and this pollutes our signal. Examples of such pollution are synchrotron, free-free and thermal dust emissions. In the rest of this thesis we consider that the foregrounds components have been removed, thanks to component separation techniques, see Ade et al. (2014b) for example.

2.4.3. Spherical harmonics

To characterize the strength of the CMB fluctuations on the sphere, it is natural to decompose it in the spherical harmonic basis as described by Müller (1966), the equivalent of the Fourier transform but on the sphere. More precisely, any signal $s(\theta,\phi)$ on the sphere can be written as:

$$s(\theta,\phi) = \sum_{\ell=0}^{\infty}\sum_{m=-\ell}^{\ell} a_{\ell m}Y_{\ell m}(\theta,\phi) \tag{2.21}$$

where $a_{\ell m} \in \mathbb{C}$ and the spherical harmonic function:

$$Y_{\ell m} : [0,\pi] \times [0,2\pi] \to \mathbb{C}$$

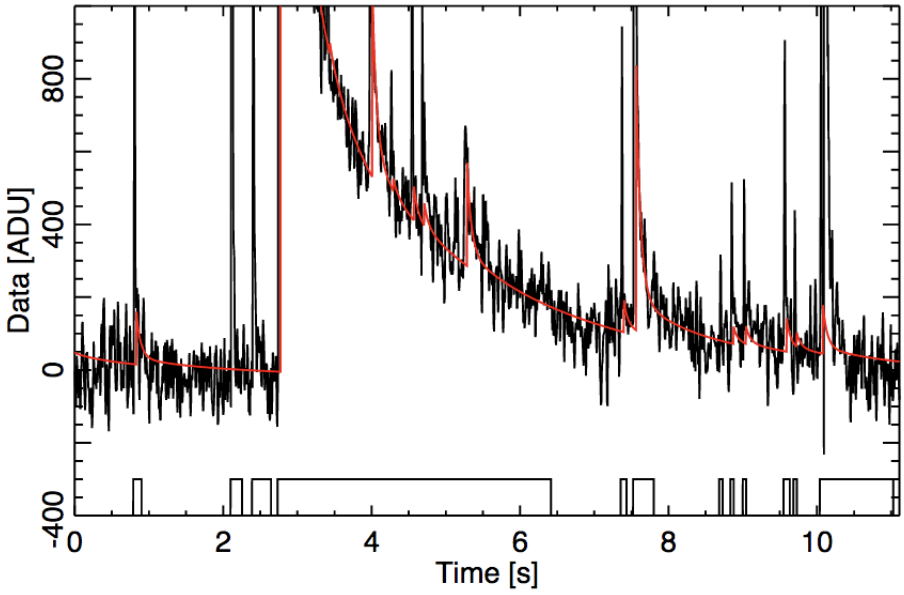

Figure 2.4.: Example of time-ordered data of a detector, that is, the value of the detector against time. Every time a particle hits the detector, its temperature increases and it takes some time to cool down. We fit a template to these spikes and remove then. The remaining fluctuations are the CMB signal and the noise. This figure comes from Ade et al. (2014a).

from an orthonormal basis on the sphere, with:

$$Y_{\ell m}(\theta,\phi) = \sqrt{\frac{2\ell+1}{4\pi}\frac{(\ell-m)!}{(\ell+m)!}} P_\ell(\cos\theta) e^{im\phi}$$

where $P_\ell(\cos\theta)$ are the associated Legendre polynomials, which for our purposes can be treated as black box functions. Note that we can invert equation Eq. (2.21) and make a spherical harmonic analysis:

$$a_{\ell m} = \int_{[0,\pi]} \int_{[0,2\pi[} s(\theta,\phi) Y^\star_{\ell m}(\theta,\phi) d\theta d\phi \tag{2.22}$$

for any $0 \leqslant \ell \leqslant$ and $-\ell \leqslant m \leqslant \ell$, where $Y^\star_{\ell m}$ is the complex conjugate of $Y_{\ell m}$. In addition, the angular power spectrum $\{C_\ell\}_{\ell \geqslant 0}$ of the signal is the "empirical variance" of the $a_{\ell,m}$ coefficients:

$$C_\ell := \frac{1}{2\ell+1} \sum_{m=-\ell}^{\ell} |a_{\ell,m}|^2$$

In practice, we observe a discretization of the signal on the sphere in HEALPix format, see Gorski et al. (2005). The minimal resolution consists in 12 pixels of same size on the sphere. Finer and finer resolutions can be obtained by further dividing each pixel into four pixels of same size. So a specific resolution is given by $N_{\text{pix}} = 12N_{\text{side}}^2$, $N_{\text{side}} = 2^n$ with $n \in \mathbb{N}$. This format has two main features:

- The sky is discretized in rings of iso-latitude pixels.
- Since the surface of the unit sphere is 4π and each pixel has the same size, the size of one pixel is given by:

  $$w := \frac{4\pi}{N_{\text{pix}}}.$$

  where N_{pix} is the number of pixel of the grid.

In the rest of this thesis we assume that the sphere is discretized into N_{pix} such pixels and n_i will denote the spherical coordinates of the center of the i-th pixel. We can identify any pixel to the coordinates of its center, so we will use n_i to denote the pixel number i, for $i \in \{1, \dots, N_{\text{pix}}\}$.

In practice we can only expand the signal in spherical harmonic coefficients up to $\ell = \ell_{\max} < \infty$. In this case the spherical harmonic synthesis for a given pixel n_i is given by:

$$s(n_i) = \sum_{\ell=0}^{\ell_{\max}} \sum_{m=-\ell}^{\ell} a_{\ell m} Y_{\ell m}(n_i).$$

If we define $\boldsymbol{Y}$ to be the matrix whose columns are the vectors

$$\overrightarrow{Y}_{\ell m} := (Y_{\ell m}(n_1), \dots, Y_{\ell m}(n_{N_{\text{pix}}})) \in \mathbb{C}^{N_{\text{pix}}} \tag{2.23}$$

for $0 \leqslant \ell \leqslant \ell_{\max}$ and $-\ell \leqslant m \leqslant \ell$ and s the vector of $(a_{\ell m})$ arranged in the same order, then we can write the transformation of a map from the harmonic basis to the pixel domain, called a synthesis, as:

$$s_{\text{pix}} = \boldsymbol{Y} s.$$

As for the transformation of a map from the pixel domain to the harmonic basis, called an analysis, on the discretized sphere we can approximate the integral Eq. (2.22) numerically:

$$a_{\ell m} \approx \frac{4\pi}{N_{\text{pix}}} \sum_{i=1}^{N_{\text{pix}}} s_{\text{pix}}(n_i) Y^{\star}_{\ell m}(n_i).$$

In matrix form, this is equivalent to:

$$s = \frac{4\pi}{N_{\text{pix}}} \boldsymbol{Y}^{\star} s_{\text{pix}}$$

where $\boldsymbol{Y}^{\star}$ is the conjugate transpose of $\boldsymbol{Y}$. In addition, we know that the $Y_{\ell m}$ functions are orthonormal on the sphere. This means that for any ℓ_1, ℓ_2, $-\ell_1 \leqslant m_1 \leqslant \ell_1$ and $-\ell_2 \leqslant m_2 \leqslant \ell_2$ we have, for a number of pixels N_{pix} sufficiently large:

$$\begin{aligned}
\frac{4\pi}{N_{\text{pix}}} \overrightarrow{Y}^{*}_{\ell_1 m_1} \overrightarrow{Y}_{\ell_2 m_2} &= \frac{4\pi}{N_{\text{pix}}} \sum_{i=1}^{N_{\text{pix}}} Y^{*}_{\ell_1 m_1}(n_i) Y_{\ell_2 m_2}(n_i) \\
&\approx \int_{[0,\pi]} \int_{[0,2\pi[} Y^{*}_{\ell_1 m_1}(\theta, \phi) Y_{\ell_2 m_2}(\theta, \phi) d\theta d\phi \\
&= \delta_{\ell_1 \ell_2} \delta_{m_1 m_2}
\end{aligned}$$

where δ_{xy} denotes the Dirac function that is one if $x = y$ and zero otherwise. This in turn implies that:

$$\boldsymbol{Y}^{\star} \boldsymbol{Y} \approx \frac{N_{\text{pix}}}{4\pi} \boldsymbol{I} \tag{2.24}$$

where $\boldsymbol{I}$ is the identity matrix in dimension $(\ell_{\max} + 1)^2$. This means that in the numerical implementation, we can make a spherical synthesis of a signal expressed in spherical harmonics domain followed by a spherical analysis and get the exact same signal in the spherical harmonics domain. Note that in practice, instead of forming the matrix $\boldsymbol{Y}$, we rely on numerical routines to perform spherical synthesis and analysis. Exploiting the HEALPix grid format to perform fast Fourier transform on each one of the iso-latitude ring, these routines scale as $\mathcal{O}(\ell^3_{\max})$ when the number of iso-latitude rings is roughly equal to $\ell_{\max}$, see Reinecke (2011) for more details.

Finally, when we observe the polarization of the light and its intensity, we no longer observe a vector $s_{\text{pix}} \in \mathbb{R}^{N_{\text{pix}}}$ corresponding to a discretized skymap, but a vector of three maps $(s^I_{\text{pix}}, s^Q_{\text{pix}}, s^U_{\text{pix}}) \in \mathbb{R}^{3N_{\text{pix}}}$

with $s^I_{\rm pix}, s^Q_{\rm pix}, s^U_{\rm pix} \in \mathbb{R}^{N_{\rm pix}}$ and I denotes intensity of the light and Q and U are its polarization in different directions. This signal now decomposes as a vector $(a^{I,P}_{\ell m})_{\ell, -\ell \leqslant m \leqslant \ell}$ in the harmonic basis where:

$$a^{I,P}_{\ell m} := (a^T_{\ell m}, a^E_{\ell m}, a^B_{\ell m})$$

and we write:

$$a^P_{\ell,m} := (a^E_{\ell m}, a^B_{\ell m}).$$

The pixel maps and the spherical harmonics maps are related by a spherical synthesis:

$$s^Q_{\rm pix}(n_i) = \sum_{\ell=0}^{\ell_{\max}} \sum_{m=-\ell}^{\ell} a^P_{\ell m} \cdot Y_{E,\ell m}(n_i)$$

and

$$s^U_{\rm pix}(n_i) = \sum_{\ell=0}^{\ell_{\max}} \sum_{m=-\ell}^{\ell} a^P_{\ell m} \cdot Y_{B,\ell m}(n_i).$$

Here we have:

$$Y_{E,2,\ell m} = \frac{1}{2} \begin{bmatrix} Y_{2,\ell m} + Y_{-2,\ell m} \\ -i(Y_{2,\ell m} - Y_{-2,\ell m}) \end{bmatrix}$$

and

$$Y_{B,2,\ell m} = \frac{1}{2} \begin{bmatrix} i(Y_{2,\ell m} - Y_{-2,\ell m}) \\ Y_{2,\ell m} + Y_{-2,\ell m} \end{bmatrix}$$

where $Y_{2,\ell m}$ and $Y_{-2,\ell m}$ are spin-weighted spherical harmonics basis functions. We can treat these functions as black-boxes and the reader can see Grain et al. (2009) for more details. We can also do a spherical analysis:

$$a^E_{\ell m} \approx \frac{4\pi}{N_{\rm pix}} \sum_{i=1}^{N_{\rm pix}} s^{Q,U}_{\rm pix}(n_i) Y^\star_{E,\ell m}(n_i)$$

and

$$a^B_{\ell m} \approx \frac{4\pi}{N_{\rm pix}} \sum_{i=1}^{N_{\rm pix}} s^{Q,U}_{\rm pix}(n_i) Y^\star_{B,\ell m}(n_i)$$

where

$$s^{Q,U}_{\rm pix}(n_i) = (s^Q_{\rm pix}(n_i), s^U_{\rm pix}(n_i))$$

and the angular power spectrum is now:

$$C_\ell := \frac{1}{2\ell + 1} \sum_{m=-\ell}^{\ell} a^{I,P}_{\ell,m} (a^{I,P}_{\ell,m})^\star.$$

2.4.4. The statistical model

We can now write a statistical model for the observed sky. We assume that our skymap is of dimension $N_{\text{pix}} = 12N_{\text{side}}^2$, $N_{\text{side}} = 2^n$ with $n \in \mathbb{N}^\star$, we set $\ell_{\max} \leqslant 2N_{\text{side}}$ and we ignore the monopole and dipole components, which means that we consider $C_0 = C_1 = 0$.

Given a power spectrum $\{C_\ell\}_{2\leqslant\ell\leqslant\ell_{\max}}$ we can summarize the steps generating the observed signal:

1. We assume that the CMB signal s expressed in the harmonic domain has distribution:

$$s \sim \mathcal{N}(0, \boldsymbol{C})$$

where $\boldsymbol{C}$ is the diagonal (block diagonal in the case of temperature and polarization) matrix where each C_ℓ is repeated $2\ell + 1$ times such that:

$$\text{Cov}(a_{\ell_1 m_2}, a_{\ell_2 m_2}) = C_{\ell_1}\delta_{\ell_1\ell_2}\delta_{m_1 m_2}$$

where δ_{xy} is the Dirac delta function equal to 1 if $x = y$ and 0 otherwise. If we observe the polarization as well as the temperature, we have:

$$\text{Cov}(a^{I,P}_{\ell_1 m_2}, a^{I,P}_{\ell_2 m_2}) = \boldsymbol{C}_{\ell_1}\delta_{\ell_1\ell_2}\delta_{m_1 m_2}$$

where:

$$\boldsymbol{C}_\ell = \begin{pmatrix} C_\ell^{TT} & C_\ell^{TE} & C_\ell^{TB} \\ C_\ell^{TE} & C_\ell^{EE} & C_\ell^{EB} \\ C_\ell^{TB} & C_\ell^{EB} & C_\ell^{BB} \end{pmatrix}$$

and $\{C_\ell^{TT}\}$, $\{C_\ell^{EE}\}$,$\{C_\ell^{BB}\}$, $\{C_\ell^{TE}\}$, $\{C_\ell^{TB}\}$, $\{C_\ell^{EB}\}$ are the temperature, E-mode, B-mode and cross correlations power spectra. So if we observe the intensity and the polarization, the covariance matrix $\boldsymbol{C}$ is block diagonal with blocks $\boldsymbol{C}_\ell$.

2. Since we do not observe this signal in the harmonic domain directly, we need to change to the pixel domain:

$$s_{\text{pix}} = \boldsymbol{Y}s$$

where $\boldsymbol{Y}$ is defined in Eq. (2.23).

3. As explained Section 2.4.2, we actually observe a smoothed version of s_{pix}, obtained according to Eq. (2.20). In the rest of this thesis we make the assumption that the beam is spherically symmetric, which implies that its expansion writes:

$$B(\theta) = \sum_{\ell=2}^{\ell_{\max}} b_\ell Y_{\ell m}(\theta, 0)$$

where $b_\ell \in \mathbb{R}$ for $2 \leqslant \ell \leqslant \ell_{\max}$. In the rest of this thesis we assume that the beam writes:

$$b_\ell = \exp\{-\ell(\ell+1)\sigma^2_{\text{FWHM}}/(8\log(2)\} \tag{2.25}$$

In addition, a convolution between two functions amounts to a multiplication of their coefficients in the expansion in the spherical harmonics basis. If we write $\boldsymbol{B}$ the diagonal matrix of dimension $(\ell_{\max}+1)^2 - 4$ with the coefficient b_ℓ repeated $2\ell + 1$ times on the diagonal, the pixel map, once smoothed, writes:

$$s_{\text{pix}} = \boldsymbol{Y}\boldsymbol{B}s = \tilde{\boldsymbol{Y}}s$$

where $\tilde{\boldsymbol{Y}} := \boldsymbol{Y}\boldsymbol{B}$

4. However we are not observing the smoothed pixel map directly. We must add a Gaussian noise, with mean zero and covariance matrix $\boldsymbol{N}$ in the pixel domain, as explained in Section 2.4.2. So the map d we are observing is given by:
$$d = s_{\rm pix} + n$$
where $n \sim \mathcal{N}(0, \boldsymbol{N})$ and $\boldsymbol{N}$ is the instrumental noise covariance matrix.

Overall, the statistical model generating the observed data $d \in \mathbb{R}^{N_{\rm pix}}$ is a linear normal hierarchical model:

$$\begin{aligned} s|\{C_\ell\} &\sim \mathcal{N}(0, \boldsymbol{C}) \\ d|s &\sim \mathcal{N}(\tilde{\boldsymbol{Y}} s, \boldsymbol{N}) \end{aligned} \tag{2.26}$$

The likelihood of this model writes straightforwardly in the pixel domain:

$$\mathcal{L}(d|\{C_\ell\}) \propto \frac{\exp\left\{-(1/2) d^t (\tilde{\boldsymbol{Y}} \boldsymbol{C} \tilde{\boldsymbol{Y}}^t + \boldsymbol{N})^{-1} d\right\}}{|\tilde{\boldsymbol{Y}} \boldsymbol{C} \tilde{\boldsymbol{Y}}^t + \boldsymbol{N}|^{1/2}}$$

where we can compute the covariance matrix in the following way:

$$\begin{aligned} \boldsymbol{C}_{\rm pix} := \mathbb{E}[dd^t] &= \mathbb{E}[(s_{\rm pix} + n)(s_{\rm pix} + n)^t] \\ &= \boldsymbol{N} + \sum_{\ell_1, m_1} \sum_{\ell_2, m_2} \mathbb{E}[a^*_{\ell_1, m_1} a_{\ell_2, m_2}] Y^*_{\ell_1, m_1} Y_{\ell_2, m_2} b_{\ell_1} b_{\ell_2} \\ &= \boldsymbol{N} + \sum_{\ell, m} Y^*_{\ell, m} Y_{l, m} C_\ell b_\ell^2 \\ &= \boldsymbol{N} + \sum_{\ell} \boldsymbol{P}^\ell C_\ell. \end{aligned} \tag{2.27}$$

The last equality follows from the addition theorem for spherical harmonics, see Müller (1966), the dependence of $\boldsymbol{C}_{\rm pix}$ on $\{C_\ell\}$ is dropped, and the matrix $\boldsymbol{P}^\ell$ is such that

$$\boldsymbol{P}^\ell_{i,j} = \frac{(2\ell+1) b_\ell^2}{4\pi} P_\ell(r_i \cdot r_j) = \frac{(2\ell+1) b_\ell^2}{4\pi} P_\ell(\cos\theta_{ij})$$

where r_i, r_j are unit vectors pointing to the pixels $\hat{n}_i$ and $\hat{n}_j$ respectively and θ_{ij} denotes the angle between these two vectors. Note that the expression Eq. (2.27) is valid whether we are observing the entire sky or not.

If we observe the entire sky and $\boldsymbol{N} = \alpha^2 \boldsymbol{I}_{N_{\rm pix}}$, where $\boldsymbol{I}_{N_{\rm pix}}$ is the identity matrix in dimension $N_{\rm pix}$, we can simplify the model by writing it in the spherical harmonics basis entirely:

$$\begin{aligned} s|\{C_\ell\} &\sim \mathcal{N}(0, \boldsymbol{C}) \\ d' &= s + n' \end{aligned}$$

where

$$d' = \boldsymbol{B}^{-1} \boldsymbol{Y}^t d$$

and

$$n' \sim \mathcal{N}(0, \alpha^2 \frac{N_{\rm pix}}{4\pi} \boldsymbol{B}^{-2})$$

The log-likelihood becomes:

$$\begin{aligned}\log \mathcal{L}(d'|\{C_\ell\}) = -\frac{1}{2} d'^T \left(\boldsymbol{C} + \alpha^2 \frac{N_{\rm pix}}{4\pi} \boldsymbol{B}^{-2} \right)^{-1} d' \\ -\frac{1}{2} \log \left| \boldsymbol{C} + \alpha^2 \frac{N_{\rm pix}}{4\pi} \boldsymbol{B}^{-2} \right|\end{aligned} \tag{2.28}$$

up to an additive constant, where $|\boldsymbol{M}|$ denotes the determinant of a matrix $\boldsymbol{M}$. The reader can see that the covariance matrix in this likelihood is diagonal and so it is computationally cheap to compute. Unfortunately, the assumptions of full sky coverage and diagonal noise covariance matrix never hold in practice, and this makes the CMB data analysis computationally much more expensive in practice. This "ideal" situation still provides useful insights about the model and we will refer to it in later sections.

2.5. Power spectrum inference

In this section we use the statistical model described in Section 2.4 to perform inference on the angular power spectrum. We first introduce the case of full sky observations. Then, we discuss inference approaches in the more realistic case where we do not observe the entire sky, first describing the pseudo-C_ℓ approach. We then explain in detail the likelihood approximations that have been used and the quadratic maximum likelihood method. Finally, we take a Bayesian viewpoint and describe the Gibbs sampler approach.

2.5.1. Entire sky observation

Suppose that we are observing the entire sky. Thanks to the orthogonality property of the spherical harmonic basis Eq. (2.24), we can multiply the last line of Eq. (3.1) and write it in the spherical harmonic domain, as done in Eq. (2.28). If we neglect the noise, the estimator:

$$\hat{C}_\ell = \frac{1}{2\ell+1} \sum_{m=-\ell}^{\ell} |d_{\ell m}|^2 \tag{2.29}$$

defined for $2 \leqslant \ell \leqslant \ell_{\rm max}$ is obviously an unbiased estimator of C_ℓ. In addition, we have:

$$\begin{aligned}\mathrm{Var}\left(\frac{\hat{C}_\ell}{C_\ell}\right) =& \mathbb{E}\left[\frac{\hat{C}_\ell^2}{C_\ell^2}\right] - \mathbb{E}\left[\frac{\hat{C}_\ell}{C_\ell}\right]^2 \\ =& -1 + \frac{1}{(2\ell+1)^2 C_\ell^2} \mathbb{E}\left[\sum_{m,m'} d_{\ell m} d^\star_{\ell m} d^\star_{\ell m'} d_{\ell m'}\right] \\ =& -1 + \frac{1}{(2\ell+1)^2 C_\ell^2} \sum_m \mathbb{E}[d_{\ell m} d^\star_{\ell m} d_{\ell m} d^\star_{\ell m}] \\ &+ \sum_{m,m',m\neq m'} \mathbb{E}[d_{\ell m} d^\star_{\ell m} d^\star_{\ell m'} d_{\ell m}] \\ =& -1 + \frac{1}{(2\ell+1)^2 C_\ell^2} [3C_\ell^2 (2\ell+1) \\ &+ 2\ell C_\ell^2 (2\ell+1)] \\ =& \frac{2}{(2\ell+1)}\end{aligned}$$

where the fifth equality comes from Wick's theorem, see Isserlis (1918). This means that:

$$\mathrm{Var}(\hat{C}_\ell) = \frac{2}{2\ell+1} C_\ell^2.$$

In addition, in the presence of noise with a covariance matrix proportional to identity $\boldsymbol{N} = \alpha^2 \boldsymbol{I}_{N_{\mathrm{pix}}}$, we can write the noise power spectrum as:

$$N_\ell = \alpha^2 \frac{N_{\mathrm{pix}}}{4\pi} b_\ell^{-2} \tag{2.30}$$

and the pseudo-C_ℓ estimator now writes:

$$\hat{C}_\ell = \frac{1}{2\ell+1} \sum_{m=-\ell}^{\ell} |d_{\ell m}|^2 - N_\ell$$

and is unbiased. Its variance is given by:

$$\mathrm{Var}(\hat{C}_\ell) = \frac{2}{2\ell+1} (C_\ell + N_\ell)^2.$$

The part of the variance coming from the power spectrum C_ℓ is called the cosmic variance and is an irreducible source of uncertainty. For very large angular scales (low ℓ), the variance $\hat{C}_\ell$ is large because of the cosmic variance. This is not surprising since $\hat{C}_\ell$ is an average of a few $d_{\ell m}$. For very low angular scales (large ℓ), the variance is high because the exponential drop of the beam given in Eq. (2.25) sharply increases the noise term given in Eq. (2.30). Note that the estimator Eq. (2.29) is also the maximum likelihood estimator of C_ℓ. In addition, since the variable:

$$\hat{Y}_\ell = (2\ell+1) \frac{\hat{C}_\ell + N_\ell}{C_\ell + N_\ell}$$

is a sum of $\nu = 2\ell+1$ independent standard normal variables, it is distributed according to a χ^2 distribution with ν degrees of freedom. From which it follows that the distribution of $\hat{D}_\ell = \hat{C}_\ell + N_\ell$ is given by:

$$p(\hat{D}_\ell | D_\ell) \propto D_\ell^{-1} \left(\frac{\hat{D}_\ell}{D_\ell} \right)^{\nu/2-1} \exp\left\{ -\frac{\nu}{2} \frac{\hat{D}_\ell}{D_\ell} \right\}. \tag{2.31}$$

where $D_\ell = C_\ell + N_\ell$. So we see that:

$$\hat{D}_\ell | D_\ell \sim \Gamma\left(\frac{\nu}{2}, \frac{2C_\ell}{(2\ell+1)} \right)$$

which has a mode in $(\nu-2)C_\ell/\nu$, not corresponding to its mean. This means that the distribution of $\hat{D}_\ell | D_\ell$ is skewed in general. However, as $l \to \infty$, this distribution tends to a Gaussian distribution, since $\hat{D}_\ell$ is an average of $2\ell+1$ i.i.d terms.

We can also take a Bayesian viewpoint and setting a flat prior p_0 on $\{C_\ell\}$, we see from Eq. (2.31) the posterior distribution of the D_ℓ is an inverse gamma distribution:

$$D_\ell | d' \sim \Gamma^{-1}\left(\frac{\nu}{2} - 1, \frac{\nu \hat{D}}{2} \right). \tag{2.32}$$

These results remain true if we ignore the instrumental noise, except that $\hat{D}_\ell$ and D_ℓ are replaced by $\hat{C}_\ell$ and C_ℓ respectively and that the noise terms N_ℓ are removed.

2.5.2. Pseudo-C_ℓ

For simplicity, in this section we neglect the noise term and the beam effect in Eq. (2.26) to derive our results. At the end of the section we provide the estimators with noise included. We follow Hivon et al. (2002a) for the details of the results.

When we do not observe the entire sky, we can no longer use the orthogonality property Eq. (2.24) to write our model in the harmonic spherical basis. Instead, the mask introduces coupling between the $\tilde{a}_{\ell m}$ we recover from the incomplete skymap. If we denote by $W(\theta, \phi)$ the function that is one on unmasked pixels and zero on masked pixels, and consider the skymap without the noise, we have:

$$\begin{aligned}
\tilde{a}_{\ell m} &= \int_{[0,\pi]} \int_{[0,2\pi[} s_{\mathrm{pix}}(\theta,\phi) W(\theta,\phi) Y_{\ell m}(\theta,\phi) \mathrm{d}\theta \mathrm{d}\phi \\
&= \sum_{\ell' m'} a_{\ell' m'} \int_{[0,\pi]} \int_{[0,2\pi[} Y_{\ell' m'}(\theta,\phi) Y_{\ell m}(\theta,\phi) W(\theta,\phi) \mathrm{d}\theta \mathrm{d}\phi \\
&= \sum_{\ell' m'} a_{\ell' m'} \boldsymbol{K}_{\ell m \ell' m'}(W)
\end{aligned}$$

where $K_{\ell m \ell' m'}(W)$ is called the coupling kernel. Since the $\tilde{a}_{\ell m}$ are linear combinations of independent Gaussian variables, they also are Gaussian variables. But they are not independent anymore. In matrix form, calling $\tilde{s}$ the vector of $\tilde{a}_{\ell m}$ we get:

$$\tilde{s} = \boldsymbol{K}(W) s$$

where K is called the coupling matrix. Unfortunately, this matrix is non invertible and we cannot recover the true s from $\tilde{s}$. Instead, we define the pseudo-C_ℓ estimator:

$$\tilde{C}_\ell = \frac{1}{2\ell + 1} \sum_m |\tilde{a}_{\ell m}|^2 \tag{2.33}$$

and compute its expectation:

$$\begin{aligned}
\mathbb{E}[\tilde{C}_{\ell_1}] &= \frac{1}{2\ell_1 + 1} \sum_{\ell_1 m_1} \mathbb{E}[\tilde{a}_{\ell_1 m_1} \tilde{a}^*_{\ell_1 m_1}] \\
&= \frac{1}{2\ell_1 + 1} \sum_{m_1=-\ell_1}^{\ell_1} \sum_{\ell_2 m_2} \sum_{\ell_3 m_3} \mathbb{E}[a_{\ell_2 m_2} a^*_{\ell_3 m_3}] \\
&\quad \times \boldsymbol{K}_{\ell_1 m_1 \ell_2 m_2}(W) \boldsymbol{K}^*_{\ell_1 m_1 \ell_3 m_3}(W) \\
&= \frac{1}{2\ell_1 + 1} \sum_{m_1=-\ell_1}^{\ell_1} \sum_{\ell_2} C_{\ell_2} \sum_{m_2=-\ell_2}^{\ell_2} |\boldsymbol{K}_{\ell_1 m_1 \ell_2 m_2}(W)|^2 \\
&= \sum_{\ell_2} \boldsymbol{M}_{\ell_1 \ell_2} C_{\ell_2}.
\end{aligned}$$

where the sums on ℓ_2, ℓ_3 run over $0, \ldots, \ell_{\max}$ and the sums over m_1, m_2, m_3 run over $-\ell_1, \ldots, \ell_1$, $-\ell_2, \ldots, \ell_2$, $-\ell_3, \ldots, \ell_3$. If we stack the pseudo-C_ℓ estimators as a vector $\tilde{C}$ and similarly for the right hand side of the last equality $\boldsymbol{B}^2 C + \tilde{N}$, we get:

$$\mathbb{E}[\tilde{C}] = \boldsymbol{M} C$$

where $\boldsymbol{M}$ is a matrix made of the $\boldsymbol{M}_{\ell_1 \ell_2}$ elements:

$$\boldsymbol{M}_{\ell_1 \ell_2} = \frac{2\ell_2 + 1}{4\pi} \sum_{l_3} (2\ell_3 + 1) \mathcal{W}_{\ell_3} \begin{pmatrix} \ell_1 & \ell_2 & \ell_3 \\ 0 & 0 & 0 \end{pmatrix}^2,$$

$\{\mathcal{W}_\ell\}$ denotes the power spectrum of the window function and the $3j$ Wigner symbol is given by:

$$\begin{pmatrix} \ell_1 & \ell_2 & \ell_3 \\ 0 & 0 & 0 \end{pmatrix} = (-1)^{L/2} \left[\frac{(L-2\ell_1)!(L-2\ell_2)!(L-2\ell_3)!}{(L+1)!} \right]^{1/2} \times \frac{(L/2)!}{(L/2-\ell_1)!(L/2-\ell_2)!(L/2-\ell_3)!}$$

and $L = \ell_1 + \ell_2 + \ell_3$. See, e.g Hivon et al. (2002b) and Efstathiou (2004) for a detailed derivation. For small sky cuts, the M matrix is invertible, but not for larger sky cuts, see Mortlock et al. (2002).

It is assumed, see Hivon et al. (2002b), that the estimators $\tilde{C}_\ell$ have the same distribution as their full sky counterpart, except that the number of freedom is updated. More precisely, they are distributed according to a χ^2 distribution with a number of freedom given by:

$$\nu = (2\ell + 1) f_{\rm sky} \frac{w_2^2}{w_4},$$

where:

$$w_i = \frac{1}{4\pi} \int_{[0,\pi]} \int_{[0,2\pi[} W^i(\theta, \phi) \mathrm{d}\theta \mathrm{d}\phi.$$

When taking the presence of noise and beam into account, the mean of the estimator Eq. (2.33) becomes:

$$\mathbb{E}[\hat{C}_\ell] = \sum_{\ell_2} \boldsymbol{M}_{\ell_1 \ell_2} b_\ell^2 C_{\ell_2} + \tilde{N}_\ell$$

where $\tilde{N}_\ell$ is the average noise power spectrum given by:

$$\tilde{N}_\ell = \frac{1}{4\pi} \sum_{i,j}^{N_{\rm pix}} \boldsymbol{N}_{ij} w^2 P_\ell(\theta_{ij})$$

where $w = 4\pi / N_{\rm pix}$ and θ_{ij} is the angle between pixel i and j, see Hivon et al. (2002b) and Efstathiou (2004). In order to use $\tilde{C}$ as an unbiased estimator of the power spectrum, we need to know the beam B, the noise power spectrum $\tilde{N}$ and be able to compute and invert the matrix M.

Since in general we do not know the power spectrum of the noise, we need to estimate it. Following Hivon et al. (2002b), we can have an approximation of the noise time correlation during the sky observation and deduce its temporal power spectrum. We can then sample a Gaussian noise in the time ordered data, see Section 2.4.2, and project it into a skymap. We can then expand it into spherical harmonics coefficients. Doing this many times allows us to get a good approximation of the noise power spectrum $\tilde{N}$. An unbiased estimator of the power spectrum is then given by:

$$\hat{C} := \boldsymbol{B}^{-2} \boldsymbol{M}^{-1} (\tilde{C} - \tilde{N}_{\rm MC})$$

where $\tilde{N}_{\rm MC}$ is a Monte-Carlo estimator of the noise power spectrum.

2.5.3. Likelihood approximations

For the sake of completeness, we mention the likelihood approximations based on the pseudo-C_ℓ for cosmological parameters inference.

When we apply a sky mask, the likelihood of $\hat{C}_\ell$ as a function of C_ℓ is no longer an inverse gamma probability distribution function, see Upham et al. (2019) for a derivation of the new likelihood function. In addition, using a brute force maximum likelihood estimation based on this likelihood for a broad range

of multipoles would be too time consuming for high resolution maps. Instead, many approximations have been devised to approximate the likelihood of $\{C_\ell\}$ given $\{\hat{C}_\ell\}$. We detail a few of them as an example. The interested reader can see Hamimeche and Lewis (2008), Hamimeche and Lewis (2009) and Gerbino et al. (2020) for further details.

The general idea is to start from the true likelihood function Eq. (2.31) for full sky and develop an approximation that is quadratic in a function of $\{C_\ell\}$ and that can be generalized to the cut sky situation:

$$-2\log\mathcal{L}(X_C) \overset{C}{=} (Z_C - \hat{Z}_C)^t \boldsymbol{M}^{-1}(Z_C - \hat{Z}_C) + \log|Y|$$

where $\overset{C}{=}$ means equality up to an additive constant, Z_C and $\hat{Z}_C$ are functions of $\{C_\ell\}$ and $\{\hat{C}_\ell\}$ respectively and $\boldsymbol{M}$ is a chosen covariance matrix. We now give several examples, for any $\ell \in \{2, \dots, \ell_{\max}\}$:

- The symmetric Gaussian approximation. It is an approximation that is Gaussian in the true power spectrum, with the covariance fixed at $\hat{C}_\ell$:

$$-2\log\mathcal{L}_{\text{symmetric}}(C_\ell|\hat{C}_\ell) \overset{C}{=} \frac{2\ell+1}{2}\left[\frac{\hat{C}_\ell - C_\ell}{\hat{C}_\ell}\right]^2.$$

- The fiducial Gaussian approximation. It is the same as the symmetric Gaussian approximation, except that the covariance is considered fixed at a $C_{\ell,\text{fid}}$ value:

$$-2\log\mathcal{L}_{\text{fid}}(C_\ell|\hat{C}_\ell) \overset{C}{=} \frac{2\ell+1}{2}\left[\frac{C_\ell - C_\ell}{\hat{C}_{\ell,\text{fid}}}\right]^2.$$

- The improper Gaussian approximation. The approximation is still quadratic in C_ℓ, this time the covariance is set to C_ℓ. It is called improper because a determinant term is missing:

$$-2\log\mathcal{L}_{\text{improper}}(C_\ell|\hat{C}_\ell) \overset{C}{=} \frac{2\ell+1}{2}\left[\frac{\hat{C}_\ell - C_\ell}{C_\ell}\right]^2.$$

- The log-normal approximation is given by:

$$-2\log\mathcal{L}_{\text{LN}}(C_\ell|\hat{C}_\ell) \overset{C}{=} \frac{2\ell+1}{2}\left[\log\left(\frac{\hat{C}_\ell}{C_\ell}\right)\right]^2.$$

- The one-third two-third approximation is a weighted sum of the two previous ones:

$$-2\log\mathcal{L}_{\text{WMAP}}(C_\ell|\hat{C}_\ell) \overset{C}{=} \frac{1}{3}\log\mathcal{L}_{\text{improper}}(C_\ell|\hat{C}_\ell) + \frac{2}{3}\log\mathcal{L}_{\text{LN}}(C_\ell|\hat{C}_\ell).$$

Ideally, these approximations should match the skewness of the likelihood for low ℓ and be approximately Gaussian for high ℓ in order to be a good approximation of the likelihood function. Percival and Brown (2006) have studied these approximations through their expansion around the maximum of the likelihood function Eq. (2.32). We now give some examples of this, where for simplicity we ignore the noise term. The maximum in C_ℓ of Eq. (2.32) is $\hat{C}_\ell$, so taking $C_\ell = (1+\epsilon)\hat{C}_\ell$ we get that the log likelihood writes:

$$-2\log\mathcal{L}_{\text{exact}}(C_\ell|\hat{C}_\ell) = \nu\left[\frac{\epsilon^2}{2} - \frac{2\epsilon^3}{3} + \mathcal{O}(\epsilon^4)\right]$$

up to an additive constant. Doing the same development to the approximations, we get the following results.

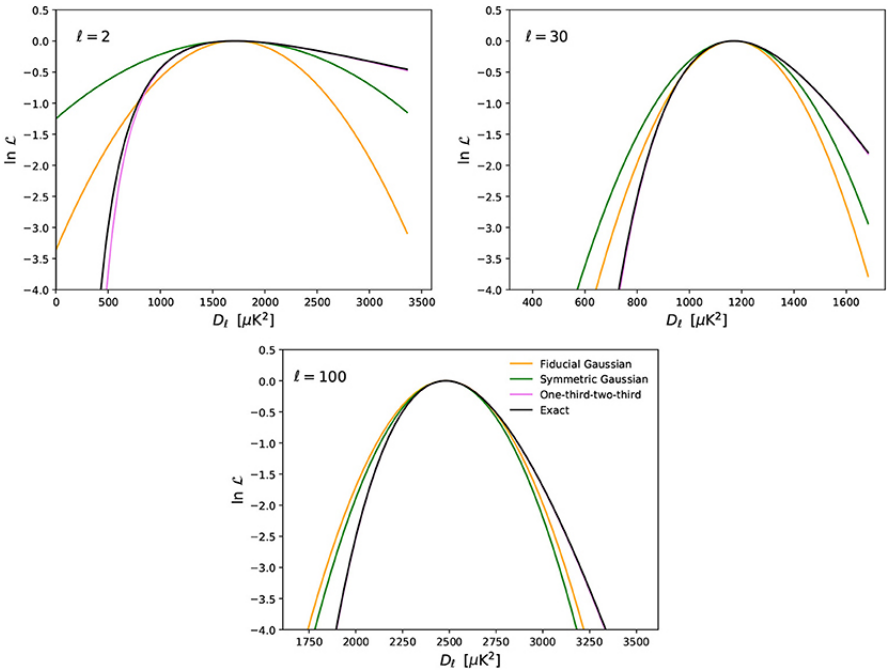

Figure 2.5.: Plot of the true likelihood of $D_\ell = \frac{\ell(\ell+1)}{2\pi}C_\ell$ and different likelihood approximation. We assume full sky observation and no noise. This figure is taken from Gerbino et al. (2020)

- Development of the symmetric Gaussian approximation gives:

$$-2\log\mathcal{L}_{\text{symmetric}} = \nu\frac{\epsilon^2}{2}.$$

  We see that this development matches the one of the likelihood up to the second term only. In addition, we have

$$-2\log\mathcal{L}_{\text{symmetric}} < -2\log\mathcal{L}_{\text{exact}}(C_\ell|\hat{C}_\ell)$$

  for $\epsilon > 0$. The oppostive is true for $\epsilon < 0$. So this likelihood approximation is biased low. Finally, since it is quadratic, it fails in capturing the skewness of the likelihood for low ℓ.

- Since all of the approximations match the likelihood at least up to the second order, we give only another one example. The development of the one-third two-third approximation gives:

$$-2\log\mathcal{L}_{\text{WMAP}}(C_\ell|\hat{C}_\ell) = \nu\left[\frac{\epsilon^2}{2} - \frac{2\epsilon^3}{3} + \mathcal{O}(\epsilon^4)\right].$$

  We see that it matches the likelihood up to the third order and should yield less biased results.

Many other approximations have been devised and studied, see e.g Gerbino et al. (2020). As an example, we reproduce the plot of the likelihood approximations given in that paper, see Fig. 2.5.

We can use the likelihood approximation of our choice to make inference about the cosmological parameters, for example using maximum likelihood estimation or MCMC algorithms. Of course the biases introduced by the approximation of the likelihood C_ℓ is likely to bias the results on the cosmological parameters. This is studied in Hamimeche and Lewis (2008) and Hamimeche and Lewis (2009).

2.5.4. Quadratic maximum likelihood method

The idea behind the Quadratic Maximum Likelihood (QML) estimator, see e.g Tegmark (1997), is that since the pair products of the unmasked pixels are a linear combination of the power spectrum on average, one of the best solution for the estimation of the power spectrum should be a linear combination of the the pair products of the unmasked pixels. That is, we are looking for an estimator of the form:

$$\hat{C}_\ell = d^t \boldsymbol{E}^\ell d - \alpha_\ell \tag{2.34}$$

for $\ell \in \{0, \dots, \ell_{\max}\}$ where $\boldsymbol{E}^\ell$ is a matrix and α_ℓ is a scalar, both depending on ℓ. We are looking for such matrices and scalars that produce an unbiased estimator of the power spectrum and that is optimal, in the sense that it minimizes the variance of the estimator Eq. (2.34). Since (Tegmark, 1997) we have:

$$\mathbb{E}[d^t \boldsymbol{E}^\ell d] = \mathrm{Tr}(\boldsymbol{E}^\ell \boldsymbol{C}_{\mathrm{pix}}) = \sum_{\ell'} \mathrm{Tr}(\boldsymbol{P}^{\ell'} \boldsymbol{E}^\ell) C_{\ell'} + \mathrm{Tr}[N \boldsymbol{E}^\ell]$$

show that for:

$$\alpha_\ell = \mathrm{Tr}[N \boldsymbol{E}^\ell]$$

we have

$$\mathbb{E}[\hat{C}_\ell] = \sum_{\ell'} \mathrm{Tr}(\boldsymbol{P}^{\ell'} \boldsymbol{E}^\ell) C_{\ell'}.$$

If we arrange the $\{\hat{C}_\ell\}$ in vector $\hat{C}$ and the corresponding $\{C_\ell\}$ in a vector C, this means we have:

$$\mathbb{E}[\hat{C}] = \boldsymbol{W} C$$

for a matrix $\boldsymbol{W}$. Which in turn implies that

$$\tilde{C} = \boldsymbol{W}^{-1} \hat{C}$$

is an unbiased estimator of the power spectrum if W is invertible. What is left is to choose the matrices $\boldsymbol{E}^\ell$. Tegmark (1997) show that if we choose

$$\boldsymbol{E}^\ell = \frac{1}{2} \boldsymbol{C}_{\mathrm{pix}}^{-1} \boldsymbol{P}^\ell \boldsymbol{C}_{\mathrm{pix}}^{-1}$$

then $W = \boldsymbol{F}$ and the estimators $\hat{C}$ and $\tilde{C}$ have covariance matrix $\boldsymbol{F}$ where:

$$\boldsymbol{F} = \frac{1}{2} \mathrm{Tr} \left\{ \boldsymbol{C}_{\mathrm{pix}}^{-1} \boldsymbol{P}^\ell \boldsymbol{C}_{\mathrm{pix}}^{-1} \boldsymbol{P}^{\ell'} \right\}$$

is the Fisher information matrix associated to the likelihood $\mathcal{L}(d|\{C_\ell\}$, that is:

$$\boldsymbol{F} := \mathbb{E}[\nabla_{\{C_\ell\}} \log \mathcal{L}(d|\{C_\ell\}) \nabla_{\{C_\ell\}} \log \mathcal{L}(d|\{C_\ell\})^t].$$

The fact that the covariance matrix of the estimator is equal to the Fisher information matrix means that we have an estimator achieving the Cramer-Rao bound: the lowest possible variance for an unbiased estimator, see Nielsen (2013).

The computation of the QML estimator poses two problems: first, all the computation happens in the pixel domain and the matrices involved are of size $N_{\mathrm{pix}} \times N_{\mathrm{pix}}$. This can make the procedure costly, since $\boldsymbol{C}_{\mathrm{pix}}$ is dense and need to be inverted. The QML estimator scales as $\mathcal{O}(N_{\mathrm{pix}}^3)$. The second problem is that the procedure achieves an optimal variance when all the matrices are evaluated at "the true power spectrum". But this is the quantity we want to estimate. In practice, we need to guess a fiducial power spectrum and compute the matrices in this power spectrum, leading to a potentially suboptimal estimator.

2.5.5. The Bayesian viewpoint

We can also take a Bayesian viewpoint and use MCMC algorithms to sample from the posterior on the power spectrum. Setting a prior p_0 on the power spectrum the model writes:

$$\begin{aligned} \{C_\ell\} &\sim p_0 \\ s|\{C_\ell\} &\sim \mathcal{N}(0, \boldsymbol{C}) \\ d|s &\sim \mathcal{N}(\tilde{\boldsymbol{Y}}s, \boldsymbol{N}) \end{aligned}$$

and we would like to sample from the posterior distribution:

$$\pi(\{C_\ell\}|d) \propto \mathcal{L}(d|\{C_\ell\}) p_0(C_\ell).$$

However, when the resolution of the map is high, that is $N_{\rm pix}$ is high, the computation of the likelihood is very expensive because of the inversion of a $N_{\rm pix} \times N_{\rm pix}$ dense matrix and the computation of its determinant. Fortunately, Eq. (3.1) is a hierarchical model and we can straightforwardly use a data augmentation scheme. As explained in Section 2.3 and done in e.g Jewell et al. (2004), Hajian (2007) and Wandelt et al. (2004), with a flat prior p_0, we can target the joint posterior distribution:

$$\pi(\{C_\ell\}, s|d) \propto p(d|s)\pi(s|\{C_\ell\} p_0(\{C_\ell\}) \tag{2.35}$$

where:

$$\log p(d|s) = -(d - \tilde{\boldsymbol{Y}}s)^t \boldsymbol{N}^{-1}(d - \tilde{\boldsymbol{Y}}s)/2 + c_1$$

and

$$\log p(s|\{C_\ell\}) = -s^t \boldsymbol{C}^{-1} s/2 - \log|\boldsymbol{C}|/2 + c_2$$

with c_1, c_2 – real valued constants and $|\mathbf{M}|$ denoting the absolute value of the determinant of a matrix $\mathbf{M}$. We can then use a two-step Gibbs sampler as described in Section 2.2.2 to target the joint distribution Eq. (2.35). This algorithm has the advantage of being easy to implement and tuning-free.

1. Indeed, the first step, called the power spectrum sampling step, samples from the conditional

   $$\pi(\{C_\ell\}|s, d) = \pi(\{C_\ell\}|s)$$

   where this equality follows from the fact that the power spectrum is independent from the observed sky map given latent map s. As explained in Section 2.5.1 and ignoring the noise term, this step amounts to sampling from $\ell_{\rm max} - 2$ independent inverse gamma distributions:

   $$p(\{C_\ell\}|s) \propto \prod_{\ell=2}^{\ell_{\rm max}} \frac{\exp\{-(2\ell+1)\sigma_\ell/2C_\ell\}}{C_\ell^{(2\ell+1)/2}} \tag{2.36}$$

2. The second step, called constrained realization step, is also conceptually simple and tuning-free. It is straightforward to show that:

   $$s|d, \{C_\ell\} \sim \mathcal{N}(\mu, \boldsymbol{\Sigma}) \tag{2.37}$$

   where

   $$\begin{aligned} \boldsymbol{\Sigma} &:= (\tilde{\boldsymbol{Y}}^t \boldsymbol{N}^{-1} \tilde{\boldsymbol{Y}} + \boldsymbol{C}^{-1})^{-1} \\ \mu &:= \boldsymbol{\Sigma} \tilde{\boldsymbol{Y}}^t \boldsymbol{N}^{-1} d. \end{aligned} \tag{2.38}$$

The covariance matrix $\boldsymbol{\Sigma}$ is dense in the presence of a sky mask and highly dimensional. Since it depends on the power spectrum, we must compute its Cholesky decomposition at each iteration, which is too costly. Instead, people have been solving the following system:

$$(\tilde{\boldsymbol{Y}}^t \boldsymbol{N}^{-1} \tilde{\boldsymbol{Y}} + \boldsymbol{C}^{-1}) x = \tilde{\boldsymbol{Y}}^t \boldsymbol{N}^{-1/2} w_0 + \boldsymbol{C}^{-1/2} w_1 + \tilde{\boldsymbol{Y}}^t \boldsymbol{N}^{-1} d \tag{2.39}$$

where $w_0, w_1 \sim \mathcal{N}(0, \boldsymbol{I})$ and $\boldsymbol{I}$ is the identity matrix in dimension $(\ell_{\max}+1)^2 - 4$, see e.g Eriksen et al. (2004). The exact resolution of this system leads to a solution distributed according to Eq. (2.37). It is too costly to invert the matrix $\boldsymbol{Q} := \boldsymbol{\Sigma}^{-1}$ at each Gibbs iteration. In practice people have been using iterative algorithms like the preconditionned conjugate gradient (PCG) agorithm, see Polyak (2021), to solve the system Eq. (2.39) approximately. To accelerate this resolution, different preconditioners have been used. For example, we can use a diagonal preconditioner, consisting in computing the diagonal of the inverse of $\boldsymbol{Q}$ or the dense preconditioner, consisting in computing a block of the inverse of $\boldsymbol{Q}$ for a subset of components and the inverse of the diagonal for the the other components, see e.g Seljebotn et al. (2019), Eriksen et al. (2004) and Papež et al. (2018) for a review.

Since we are using a two-step Gibbs sampler, the results of Section 2.2.2 apply. In particular, and for full sky observations and noise matrix $\boldsymbol{N} = \alpha \boldsymbol{I}_{N_{\text{pix}} \times N_{\text{pix}}}$, we roughly have:

$$\begin{aligned} \text{Var}(C_\ell | d) &\propto (C_\ell + N_\ell)^2 \\ \text{Var}(C_\ell | s) &\propto C_\ell^2 \end{aligned}$$

where $N_\ell = \alpha \frac{N_{\text{pix}}}{4\pi} \boldsymbol{B}^{-2}$. This implies that the fraction of missing information Eq. (2.14) is low for the components ℓ with $C_\ell \gg N_\ell$, called the high signal to noise ratio (SNR) component, and high for the components ℓ such that $C_\ell \ll N_\ell$, called the low SNR components. So we can expect this algorithm to mix well for high SNR components and badly for low SNR components. This is what is observed in practice, see Jewell et al. (2009). To circumvent this problem, Jewell et al. (2009) have used a non centered parametrization as explained in Section 2.3 to break the correlation between the power spectrum and the latent variables coming from the prior. This new algorithm samples the low SNR components efficiently but in turn mixes very badly on the high SNR components.

A Fortran code, called COMMANDER, implements the Gibbs sampler described in this section, with a range of preconditionners, see e.g Eriksen et al. (2008), Eriksen et al. (2004), Wandelt et al. (2004), Chu et al. (2005) and Larson et al. (2007).

2.6. Summary of the contributions

2.6.1. Summary of our work regarding the analysis of the CMB data

In Section 2.5.5 we discussed the use of the Gibbs sampler made in e.g Jewell et al. (2004), to sample the joint posterior distribution of the power spectrum and the true underlying skymap. We also explained that this Gibbs sampler is inefficient in sampling the low signal-to-noise ratio power spectrum because of the strong correlations between these power spectrum modes and the skymap. To break these correlations, Jewell et al. (2009) have been using the results in Section 2.3 and build a non centered version of the Gibbs sampler. That is, rewriting Eq. (3.1) as:

$$\begin{aligned} \{C_\ell\} &\sim p_0 \\ \tilde{s} &\sim \mathcal{N}(0, \boldsymbol{I}) \\ d &= \tilde{\boldsymbol{Y}} \boldsymbol{C}^{1/2} \tilde{s} + n \end{aligned}$$

they use a Gibbs sampler to target the joint posterior density $\pi(\{C_\ell\}, \tilde{s}|d)$. This effectively breaks the posterior dependencies between $\{C_\ell\}$ and $\tilde{s}$ for the small angular scales. However, this Gibbs sampler is very inefficient in sampling the large angular scales because of Metropolis-within-Gibbs move, see Jewell et al. (2009). Our first contribution consists in using an interweaving scheme, described in Section 2.3. Using this, we are able to sample efficiently the entire signal-to-noise ratio range for roughly the same computational cost as the non centered Gibbs sampler.

The second contribution regards the highly dimensional system resolution, Eq. (2.39). First, we use a Metropolis ratio after the resolution of this system, which enables us to solve it even more approximately while still being sure that the Gibbs sampler targets the right distribution. This method has been described by Gilavert et al. (2015). A second way to deal with this system is to bypass it altogether. To achieve this, we augment the conditional posterior distribution $\pi(s|d, \{C_\ell\})$ with a Gaussian distributed auxiliary variable z such that sampling from $\pi(s|d, \{C_\ell\}, z)$ and $\pi(z|d, \{C_\ell\}, s)$ is easy. This enables us to use a Gibbs sampler targeting $\pi(z, s|d, \{C_\ell\})$. Marginalizing over z after a predefined number of Gibbs steps gives us a MCMC scheme targeting $\pi(s|d, \{C_\ell\})$. Unfortunately, the high signal-to-noise ratio components of s are strongly correlated with z and this algorithm does not sample efficiently this part of the map s. To improve the mixing of this algorithm, we use an overrelaxation step on top of the auxiliary Gibbs steps. Overrelaxation is known to suppress the random-walk behavior of the Gibbs sampler in the presence of strong correlations, see Neal (1998). The use of such an auxiliary variable drastically reduces the computational cost of the Gibbs sampler: from few hundred of spherical harmonics transform (on top of the computation of preconditioner) to only 2 such operations.

We tested the Gibbs sampler using different combinations of the steps described above, that is interweaving with auxiliary step, interweaing with full system resolution, centered Gibbs sampling with auxiliary Gibbs step... We evaluated their respective performances on liteBird like experiment, with two different sky masks: one covering about 80% of the sky and the second one covering roughly 30% of the sky, thus inducing greater correlations in the multipoles. In both cases we found that the Centered Gibbs sampler with the auxiliary variable step outperforms any algorithm in terms of ESS per second. Compared to the centered Gibbs sampler, its ESS per second is almost 10 times better on the EE polarization power spectrum on average and almost 100 times better on BB polarization power spectrum.

2.6.2. Summary of our work regarding the compression of MCMC outputs

We usually want to discard part of the correlated output produced by a MCMC method. There may be several reasons for that: we may want to discard the first b output of the chain, corresponding to the "non-stationary part" of the chain, effectively reducing the bias of the ergodic average given in Eq. (2.6). It is also customary to keep one state every t states to reduce the memory footprint of the MCMC algorithm or reducing the computational cost of a potential post-processing. This procedure is called thinning. Of course, throwing some output away means throwing some information away, and we would like to keep as much information about the target distribution as possible. To do so, we propose a two-step thinning procedure. Suppose the MCMC algorithm targets a density $p : \mathbb{R}^d \mapsto \mathbb{R}$ and that we are interested in evaluating the integral:

$$p(f) = \mathbb{E}_p[f(X)] \tag{2.40}$$

where $f : \mathbb{R}^d \mapsto \mathbb{R}$ is a function. Suppose in addition that the MCMC algorithm yields an output $\{X_1, \dots, X_N\}$. First, using J available control variates $h(X) = (h_1(X), \dots, h_J(X))^t$, we obtain an estimator of the expectation Eq. (2.40) of interest as a weighted sum of the original chain:

$$\hat{p}_\star(f) = \sum_{n=1}^{N} w_n f(X_n)$$

such that the weights $\{w_n\}_{1 \leqslant n \leqslant N}$ sum to one, are independent of f and such that $\sum_{n=1}^{N} w_n h_j(X_n) = 0$ for any $j \in \{1, \dots, J\}$.

The second step is to resample the weighted chain $\{(w_n, X_n)\}_{1 \leqslant n \leqslant N}$ based on the weights. To perform this step, we use the Cube method, which is a survey sampling method, see Deville (2004). We use this method to find a subsample $\{Z_m\}_{1 \leqslant m \leqslant M} \subset \{X_n\}_{1 \leqslant n \leqslant N}$ with $M \ll N$ such that

$$\sum_{m=1}^{M} h_j(Z_m) = 0 \tag{2.41}$$

approximately for any $j \in \{1, \dots, J\}$. Since we know that $\mathbb{E}_p[h_j(X)] = 0$ for any $j \in \{1, \dots, J\}$, we can hope that the sample $\{Z_m\}_{1 \leqslant m \leqslant M}$ verifying Eq. (2.41) is "representative" of p.

We also provide two different ways to build control variates: the first one when the score function is available, called the Stein trick, see Oates et al. (2016). The second one is based on the conditional means of p, see Dellaportas and Kontoyiannis (2011).

Finally, we evaluate our method against the regular thinning method and the kernel Stein Discrepency (KSD) thinning method, see Riabiz et al. (2020) for the detail of this algorithm. We use three metrics to compare these three methods: the kernel Stein discrepency, see Riabiz et al. (2020), the energy distance described by Mak and Joseph (2018) and a star discrepency that we define in a subsequent section. The numerical results showed that the Cube thinning method performs worse than the KSD thinning in terms of KSD. This was expected since KSD thinning greedily minimizes the kernel Stein Discrepency. In terms of star discrepency, our cube thinning method tends to perform better than the regular thinning procedure and the KSD thinning method. In addition, the Cube thinning method outperforms the KSD thinning procedure in terms of energy distance. Note that our method tends to perform differently depending on the choice of control variate and the KSD thinning procedure performs differently depending on the kernel choice, see Riabiz et al. (2020). In addition, our Cube method scales at worst as $\mathcal{O}(NJ^3 + 2^J)$ where J is the number of control variates, while the KSD thinning procedure scales as $\mathcal{O}(NM^2)$ where M is the subsample size. This means that we can apply our procedure very quickly with a complexity independent of the subsample size while the KSD thinning is very costly for subsample sizes $M \gg 1$.

Chapter 3.

Amended Gibbs samplers for Cosmic Microwave Background power spectrum estimation

Joint work with Nicolas Chopin, Josquin Errard and Radek Stompor, appeared in *Physical Review D*, 105, 103501.

We study different variants of the Gibbs sampler algorithm from the perspective of their applicability to the estimation of power spectra of the cosmic microwave background (CMB) anisotropies. These include approaches studied earlier in the CMB literature as well as new ones which are proposed in this work. We demonstrate all these variants on full and cut sky simulations and compare their performance, assessing both their computational and statistical efficiency. For this we employ a consistent comparison metric, an effective sample size (ESS) per second. We show that one of the proposed approaches, referred to as Centered overrelax, which capitalizes on additional, auxiliary variables to minimize computational time needed per sample, and uses overrelaxation to decorrelate subsequent samples, performs better than the standard Gibbs sampler by a factor between one and two orders of magnitude in the nearly full-sky, satellite-like cases. It therefore potentially provides an interesting alternative to the currently favored approaches.

3.1. Introduction

In the past few decades, the analysis of the Cosmic Microwave Background (CMB) has made a lot of progress. Numerous, novel and advanced statistical and numerical techniques have been proposed and implemented for virtually every step down the CMB data analysis pipeline. In particular, an entire slew of very diverse methods have been designed to produce estimates of the temperature or polarization power spectra or estimates of the cosmological parameters from a set of noisy CMB maps. We can divide these in three broad categories. The first one includes the so-called pseudo-C_ℓ approaches, see e.g., Upham et al. (2019); Hamimeche and Lewis (2008, 2009); Grain et al. (2009); Hivon et al. (2002b), which compute the power spectra directly from the noisy observed maps of the CMB sky. See Gerbino et al. (2020) for a review. The second category involves the maximum likelihood methods, see Gjerløw et al. (2015); Tegmark and de Oliveira-Costa (2000), which maximize the likelihood of the observed CMB maps with respect to the sought-after coefficients of the CMB power spectra. The third category comprises the Bayesian approaches using Monte Carlo sampling methods, which directly target the posterior distribution of the estimated parameters, such as power spectra, given the observed data. A number of such techniques exist and some have been applied either for the power spectra or cosmological parameter estimation. These include the Metropolis-Hastings sampler, see Lewis and Bridle (2002); Wraith et al. (2009); Eriksen et al. (2008). the Hamiltonian Monte Carlo sampler see Taylor et al. (2008); Hajian (2007), or the Gibbs sampler, see Eriksen et al. (2004); Larson et al. (2007); Racine et al. (2016); Jewell et al. (2009); Wandelt et al. (2004).

Out of those, the pseudo-C_ℓ methods are computationally very efficient but require careful character-

isation of their statistical properties and a design of a corresponding pseudo-likelihood to allow for a meaningful interpretation of the estimated spectra. They are often a method of choice for the analysis of spectra at angular scales much smaller than the observed sky area, when such a pseudo-likelihood construction is more straightforward, see Gerbino et al. (2020).

The maximum likelihood methods are statistically more robust. However, they are computationally heavy and typically require approximations to provide a meaningful description of the power spectrum likelihood. They are typically applicable only to downsized data sets providing constraints on the power spectra on large angular scales, see Gjerløw et al. (2015); Tegmark and de Oliveira-Costa (2000).

The Monte Carlo sampling techniques can provide a robust description of the posterior distribution of the estimated spectra in the full range of angular scales. They do so by generating chains of samples which encode the statistical properties of the posterior. Some of these techniques can also cut significantly on the computational load of the maximum likelihood methods. Out of potential methods, Gibbs samplers has been found particularly well adapted to the context of the CMB power spectrum estimation, in e.g., Eriksen et al. (2004); Racine et al. (2016); Jewell et al. (2009), and this is a Gibbs sampler which is implemented in the most advanced, existing, Bayesian CMB power spectrum estimation code, see Racine et al. (2016). Gibbs samplers have also thorough statistical underpinning. In particular, the efficiency of the two-steps Gibbs sampler for linear hierarchical models, i.e., as used in the CMB context, have been extensively studied in the statistical literature,e.g., Liu (1994).

The current implementations of the Gibbs sampler however remain computationally demanding. This often imposes practical limits on the number or the size of test and validation runs which can be afforded, and frequently requires approximations in modelling input CMB data in order to simplify the calculations. The computational gain here comes however at the potential risk of increased statistical uncertainties, presence of biases or both, in the final results of such analyses. More efficient Gibbs algorithms are required in order to bypass such limitations.

There are two factors determining sampler's run time: the time needed to draw a single sample and the overall number of samples required to provide sufficient sampling of the posterior. How good the posterior sampling is, is best quantified by a number of effective, uncorrelated samples. This number is smaller, and typically much smaller, than the number of actual samples, which are usually correlated. The stronger the correlations, the less efficiently the samples explore the volume of the posterior, and consequently, more samples are needed to reach the same number of the effective samples. This effect is referred to in the statistical literature as a bad mixing of the algorithm, in e.g Robert and Casella (2004).

In this paper we present several new ideas aiming at enhancing the performance of the Gibbs sampler as applied to the CMB power spectrum estimation. These include methods, which aim at cutting the number of actual samples required to characterize reliably the posterior, i.e., improving mixing properties of the algorithms, as well as methods which attempt to trade the time needed to compute samples for their number, potentially leading to a net gain in the overall performance. The third possibility of improving numerical algorithms and their implementation to cut on the computational time of each sample is not considered in this work. To compare the different methods, we evaluate the number of effective, uncorrelated samples which these methods can produce per unit time. This metric, referred to as an effective sample size per second, ESS, is defined and discussed in Sect. 3.7.

We organize this paper as follows. In Section 3.2 we review the adopted data model and introduce the basic formalism. In Section 3.3 we present the standard Gibbs sampler as considered in early CMB power spectrum estimation literature and discuss its deficiencies. In Section 3.4 and 3.5 we discuss techniques aiming at decreasing the number of necessary samples, while keeping the sample computations unchanged. In Section 3.6, we discuss ways to suppress time needed for the single sample computations, compensating by an increased number of samples. Finally, in Section 3.7, we describe our experiments and compare the performance of all the presented Gibbs variants. We show that on nearly full-sky, satellite-like data, one of the proposed algorithms performs (in terms of the effective sample size per second) one order of magnitude better on the EE power spectrum and two orders of magnitude better

on the BB power spectrum than our baseline algorithm. We conclude our findings in Section 3.8.

3.2. Basic formalism

3.2.1. Data model

We assume throughout this work that the input data set consists of noisy maps including only the CMB signal and we focus on the estimation of its power spectra from such maps. The maps typically cover only part of the entire sky and can be of one, two or three Stokes parameters, corresponding to the total intensity, I, or Stokes parameters, Q and U only, or all three Stokes parameters, I, Q, and U, respectively. The CMB signal is assumed to be Gaussian, with the covariance given by matrix $\boldsymbol{C}$. The noise in the maps is also Gaussian with the covariance given by $\boldsymbol{N}$. The data model underlying the maps is therefore hierarchical, (see Eriksen et al. (2008, 2004); Wandelt et al. (2004); Chu et al. (2005)), and reads,

$$\begin{aligned} \{\mathbf{C}_\ell\} &\sim p_0, \\ s|\{\mathbf{C}_\ell\} &\sim \mathcal{N}(0, \boldsymbol{C}), \\ d|s &\sim \mathcal{N}(\tilde{\boldsymbol{Y}}s, \boldsymbol{N}), \end{aligned} \tag{3.1}$$

Here, $\sim$ denotes a sample drawn from the distribution on the right hand side. p_0 is the flat prior and $\mathcal{N}(m, \mathbf{\Sigma})$ denotes the Gaussian distribution with mean m and covariance $\mathbf{\Sigma}$.

The set $\{\mathbf{C}_\ell\}_{2\leqslant\ell\leqslant\ell_{\rm max}}$ denotes a set of all relevant power spectra coefficients numbered by a multiple number, ℓ, (with the monopole and dipole ignored in the case of total intensity power spectrum). These uniquely define the CMB covariance matrix, $\boldsymbol{C}$. Each $\mathbf{C}_\ell$ can be a number, i.e., in the case of the total intensity maps, or a matrix, in the case of multiple Stokes parameter maps, as elaborated below, equation (3.2).

The variable s is the sky map expressed in the spherical harmonic basis. Hereafter, we follow the convention that we use two real numbers (one for the coefficients with $m = 0$) corresponding to the real and imaginary part of the sky harmonic coefficients, instead of a single complex one, and separate them into two real vectors, see Seljebotn (2010) for an extensive justification.

The matrix $\tilde{\boldsymbol{Y}}$ is the product of the spherical harmonics synthesis matrix $\boldsymbol{Y}$ and the (diagonal) Gaussian beam matrix $\boldsymbol{B}$, in the spherical harmonic domain, which is assumed diagonal corresponding to an axially symmetric beam. Consequently, $\tilde{\boldsymbol{Y}}\,s$ stands for the beam-smoothed CMB map in the pixel domain computed from the harmonic coefficients, s, drawn from the Gaussian distribution with covariance $\boldsymbol{C}$. In general, it covers only observed part of the sky.

We assume that for a full sky map,

$$\frac{4\pi}{N_{\rm pix}}\boldsymbol{Y}^T\boldsymbol{Y} = \boldsymbol{I},$$

where $\boldsymbol{I}$ denotes the identity matrix, This means that the adapted pixelization used to discretize the map objects is such that all the spherical harmonics all the way up the band limit, $\ell_{\rm max}$, are orthogonal on the grid made of the pixel centers.

The data vector, $d \in \mathbb{R}^{N_{\rm pix}}$, is the noisy sky map in the pixel domain and $N_{\rm pix}$ is the number of pixels of the map.

We note that the noise covariance, $\boldsymbol{N}$, is given in the pixel domain and assumed hereafter to be diagonal (though not necessarily to be proportional to the unit matrix). In contrast, the signal covariance, $\boldsymbol{C}$, is defined in the harmonic domain, and is block diagonal (diagonal in the case of total intensity only). For example, in the case of inference on total intensity and polarization, the blocks of the signal covariance

are,

$$\boldsymbol{C}_\ell = \begin{pmatrix} C_\ell^{TT} & C_\ell^{TE} & C_\ell^{TB} \\ C_\ell^{TE} & C_\ell^{EE} & C_\ell^{EB} \\ C_\ell^{TB} & C_\ell^{EB} & C_\ell^{BB} \end{pmatrix} \tag{3.2}$$

where $\{C_\ell^{TT}\}$, $\{C_\ell^{EE}\}$,$\{C_\ell^{BB}\}$, $\{C_\ell^{TE}\}$, $\{C_\ell^{TB}\}$, $\{C_\ell^{EB}\}$ are the temperature, E-mode, B-mode and cross correlations power spectra. For the standard, parity-invariant cosmology, adopted in this work, $\{C_\ell^{TB}\} = \{C_\ell^{EB}\} = 0$. In the rest of this paper, for simplicity, we will drop the dependency of the signal covariance matrix on the power spectrum and define $\boldsymbol{C} := \boldsymbol{C}(\{\mathbf{C}_\ell\})$.

For the sake of transparency, hereafter we present our algorithms specialized for the case of the total intensity as the generalization to include polarization is straightforward, see Larson et al. (2007) for example. We however include polarization in all our numerical experiments experiments in Section 3.7.

3.2.2. Likelihood

If the observed data are normally distributed given the power spectrum, the likelihood of the observed data reads,

$$\mathcal{L}(d|\{C_\ell\}) \propto \frac{\exp\left\{-(1/2)d^T(\tilde{\boldsymbol{Y}}\mathbf{C}\tilde{\boldsymbol{Y}}^T + \boldsymbol{N})^{-1}d\right\}}{|\tilde{\boldsymbol{Y}}\mathbf{C}\tilde{\boldsymbol{Y}}^T + \boldsymbol{N}|^{1/2}},$$

where $|\dots|$ denotes the absolute value of the determinant of a matrix, and $\tilde{\boldsymbol{Y}}\mathbf{C}\tilde{\boldsymbol{Y}}^T$ is the signal covariance of the cut-sky map in the pixel domain.

The full covariance matrix of this likelihood is dense in the pixel domain and, in the case of partial sky coverage and noise covariance matrix not proportional to the identity, is also dense in the harmonic domain, therefore inverting it and computing its determinant is time consuming as soon as the dimension, i.e., the number of the observed sky pixels, is high. Hence the computation of the likelihood, and therefore the maximum likelihood approach, becomes quickly prohibitive. We can however rely on the Bayesian approach instead.

3.2.3. Bayesian approach

Adopting the Bayesian viewpoint and putting an improper flat prior $p_0(\{C_\ell\})$ on the power spectrum, we can derive the posterior distribution of the power spectrum coefficients,

$$\pi(\{C_\ell\}|d) \propto \mathcal{L}(d|\{C_\ell\}). \tag{3.3}$$

Unfortunately, evaluating this posterior is as computationally involved as the computation of the likelihood and making application of the sampling algorithms difficult or, as in the case of Metropolis-Hastings sampler, directlt infeasible.

To bypass this difficulty we can augment our data model and consider a joint posterior over the power spectrum and sky map, as done in previous works, for exampke in Eriksen et al. (2008, 2004); Wandelt et al. (2004); Chu et al. (2005); Larson et al. (2007):

$$\pi(\{C_\ell\}, s|d) \propto p(d|s)p(s|\{C_\ell\}). \tag{3.4}$$

where

$$\log p(d|s) = -(d - \tilde{\boldsymbol{Y}}s)^T\boldsymbol{N}^{-1}(d - \tilde{\boldsymbol{Y}}s)/2 + c_1$$

and

$$\log p(s|\{C_\ell\}) = -s^T\boldsymbol{C}^{-1}s/2 - \log|\boldsymbol{C}|/2 + c_2$$

with c_1, c_2 – real valued constants. We note that s denotes the set of spherical harmonic coefficients and is therefore equivalent to the full sky map in the pixel domain, notwithstanding the fact that d may correspond only to a partial sky for which the data are available. Consequently, the number of elements of s can be much larger than the number of the data points collected in d. The elements of s are referred to as latent variables, as they are introduced to facilitate the computation and will be eventually discarded. As their covariance matrix, $\boldsymbol{C}$, is very structured, its determinant and its inverse are both straightforwardly computable. Hence, we could apply the Metropolis-Hastings algorithm to this joint distribution, however, we do not expect it to be efficient due to the high dimensionality of the problem and the strong correlations between the variables. However, as first proposed in Jewell et al. (2004); Wandelt et al. (2004), we can sample from the respective conditional posterior distributions of this joint posterior and can apply a Gibbs sampler instead. We discuss this in detail in the next section.

We note that in general using an improper prior distribution may lead to an improper posterior distribution - that is one with infinite mass - creating troubles for MCMC algorithms as discussed in the statistical literature, see e.g., Hobert and Casella (1996). However, in our application and in the case of full-sky data it can be shown, see appendix A, that the improper flat prior, $p(\{C_\ell\})$, results in a proper posterior distribution. This is consistent with the previous CMB literature on MCMC applications, see Eriksen et al. (2008, 2004); Wandelt et al. (2004); Chu et al. (2005); Larson et al. (2007), which have reported no pathological cases. In contrast, as also shown in appendix A, using Jeffrey's prior, described in Harold (1946), on this model, as also suggested in some previous CMB works, e.g Larson et al. (2007) and Eriksen et al. (2008), leads to an improper posterior distribution in the case of full sky observation and thus results in a non-valid MCMC algorithm. Given that, and following the accepted convention in the field, we adopt the improper flat prior on the power spectrum throughout this work.

3.3. Gibbs Sampling

3.3.1. The algorithm

The principle of Gibbs sampling for data augmentation is to sample iteratively from the conditional distributions of the parameters and the latent variables, see, e.g., Tanner and Wong (1987). Algorithm 7 shows one iteration of this algorithm applied to the joint posterior distribution in equation (3.4).

Algorithm 7: Iteration t of Gibbs sampling for Data Augmentation

Input: $(\{C_\ell\}_t, s_t)$

Output: $(\{C_\ell\}_{t+1}, s_{t+1})$

1 $s_{t+1} \sim p(s|d, \{C_\ell\}_t)$ `// Constrained Realization step`

2 $\{C_\ell\}_{t+1} \sim p(\{C_\ell\}|d, s_{t+1})$ `// Power Sampling step`

The first step of drawing a sample of the sky signal, s_{t+1}, given the data and the power spectrum is called the constrained realization step. The second step is the power spectrum sampling step as it draws a sample of the power spectrum given the data and the sky signal. This type of algorithms has been widely used for CMB data analysis, in e.g Eriksen et al. (2008, 2004); Wandelt et al. (2004); Chu et al. (2005); Larson et al. (2007). The hierarchical data model underlying Algorithm 7 can be represented graphically by a directed acyclic graph (DAG) shown in Figure 3.1.

3.3.2. Constrained Realization step

The distribution of the sky map, conditional on the observed map and the power spectrum, is given by,

$$s|d, \{C_\ell\} \sim \mathcal{N}(\mu, \boldsymbol{\Sigma}) \tag{3.5}$$

where

$$\begin{aligned} \boldsymbol{\Sigma} &:= \boldsymbol{Q}^{-1} = (\tilde{\boldsymbol{Y}}^T \boldsymbol{N}^{-1} \tilde{\boldsymbol{Y}} + \boldsymbol{C}^{-1})^{-1} \\ \mu &:= \boldsymbol{\Sigma} \tilde{\boldsymbol{Y}}^T \boldsymbol{N}^{-1} d. \end{aligned} \tag{3.6}$$

However, in the case of an inhomogeneous noise and/or an incomplete sky coverage, the covariance matrix in equation (3.5), $\boldsymbol{\Sigma}$, is dense and highly dimensional. Hence it is costly to invert it or to compute its Cholesky decomposition.

In order to sample from this Gaussian distribution, we can rely instead on an algorithm proposed in the CMB context in Wandelt et al. (2004) and known in the statistical literature as the Perturbation-Optimization algorithm, see Orieux et al. (2012). The steps are,

- Draw $w_0, w_1 \sim \mathcal{N}(0, \boldsymbol{I})$
- Solve for x:

$$(\tilde{\boldsymbol{Y}}^T \boldsymbol{N}^{-1} \tilde{\boldsymbol{Y}} \;+\; \boldsymbol{C}^{-1})x \quad = \quad \tilde{\boldsymbol{Y}}^T \boldsymbol{N}^{-1/2} w_0 \;+\; \boldsymbol{C}^{-1/2} w_1 \;+\; \tilde{\boldsymbol{Y}}^T \boldsymbol{N}^{-1} d \tag{3.7}$$

where $\boldsymbol{M}^{1/2}$ denotes any matrix satisfying:

$$\boldsymbol{M} = \boldsymbol{M}^{1/2} (\boldsymbol{M}^{1/2})^T.$$

Obviously, the right-hand term of equation (3.7) is a normal variable with distribution $\mathcal{N}(\tilde{\boldsymbol{Y}}^T \boldsymbol{N}^{-1} d, \mathbf{Q})$ and the solution of this system is a random variable drawn from the distribution in equation (3.5). Since this system may be very high-dimensional and badly conditioned, in practice the system in equation (3.7) is solved using an iterative solver such as preconditioned conjugate gradient (PCG) algorithm. This indeed has been the standard way of making the constrained realization step in the context of CMB data analysis, see Eriksen et al. (2008, 2004); Wandelt et al. (2004); Chu et al. (2005); Larson et al. (2007); Jewell et al. (2009), however, see, e.g., Elsner and Wandelt (2013), for alternative solvers, and Papež et al. (2018) for their comparison. In the following, we introduce the Truncated Perturbation-Optimization (TPO) algorithm, which is a Perturbation-Optimization algorithm using an iterative method to solve the linear system, which is terminated after a predetermined number of iterations or reaching a precision threshold and therefore potentially failing to attain sufficient accuracy.

3.3.3. Power spectrum sampling

The second step of the Gibbs sampler in Algorithm 7 consists in sampling the power spectrum conditionally on the sky signal, s, and the observed data, d. As visualized in Figure 3.1, the sampling is in fact independent on the data as $p(\{C_\ell\}|s, d) = p(\{C_\ell\}|s)$ and given by,

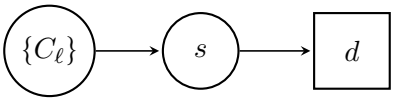

Figure 3.1.: Directed acyclic graph of model 3.1. Circles and squares represent unobserved and observed variables respectively. Plain arrows represent stochastic dependence.

$$p(\{C_\ell\}|s) \quad \propto \quad \frac{\exp\left\{-(1/2) s^T \boldsymbol{C}^{-1} s\right\}}{|\boldsymbol{C}|^{1/2}}. \tag{3.8}$$

In the case of either temperature or polarization, separately, this corresponds to a product of inverse gamma distributions, see Eriksen et al. (2008, 2004); Wandelt et al. (2004); Chu et al. (2005); Larson et al. (2007),

$$p(\{C_\ell\}|s) \quad \propto \quad \prod_{\ell=2}^{\ell_{\max}} \frac{\exp\left\{-(2\ell+1)\sigma_\ell/2C_\ell\right\}}{C_\ell^{(2\ell+1)/2}}, \tag{3.9}$$

where,

$$\sigma_\ell := \frac{1}{2\ell+1} \sum_{-\ell_{\max} \leqslant m \leqslant \ell_{\max}} |a_{\ell,m}|^2$$

is the empirical power spectrum. In the case of temperature and polarization, we must instead sample from independent inverse Wishart distributions, see Eriksen et al. (2008, 2004); Wandelt et al. (2004); Chu et al. (2005); Larson et al. (2007). Hereafter, we continue presenting the formalism for the total intensity case and include polarization only in the numerical experiments in Section 3.7.

3.3.4. Shortcomings

While being straightforward to implement and tuning-free, this algorithm has two major issues that can prohibit its application in many cases of interest. These are: the high computational cost of the constrained realization step and the strong correlations between the sky map and the power spectrum. This is this last property, referred to in the statistical literature as bad mixing of the algorithm, which drives the number of samples needed to sample the full volume of the posterior. Both these factors tend to inflate the overall computational time of the algorithm potentially limiting its applicability. We discuss each of them in more detail below.

Constrained realizations

The resolution of the system in equation (3.7) is costly in general. Depending on the preconditioner that is being used, between $\mathcal{O}(350)$ and $\mathcal{O}(1000)$ spherical harmonics transforms were required for a WMAP-like experiment, with eight frequency bands, see Eriksen et al. (2004). In this work, we find that the resolution of the system takes $\mathcal{O}(240)$ spherical harmonics transforms for a lower resolution, LiteBIRD-like experiment with an 80% Planck galactic mask, assuming the standard, Block-Jacobi preconditioner.

More sophisticated preconditioners could speed up the convergence, however they typically require expensive precomputation and extra time to apply them. These can significantly offset any gain in the number of iterations they may bring. We note that in principle we need highly accurate solutions, what exacerbates the computational problem. The high accuracy is necessary to ensure that the solutions are really drawn from the required distribution. So while it may be tempting to compromise on the solution precision in the interest of the time, for low accuracy solutions, we may not even know what is the true underlying distribution they have been effectively drawn from, potentially invalidating the entire procedure.

This is a real issue for the Gibbs sampler, since if we are not sampling from the correct conditional distributions at each iteration, we have no idea what effective joint distribution the Gibbs sampler is simulating from or even whether this distribution exists at all.

Power spectrum sampling

The second problem concerns the sampling of the power spectrum conditional on the sky map, that is, the second step of our Gibbs sampler.

We define the lag-1 autocorrelation for any function f with finite second order moment under π, i.e., for which $\int f^2(x)\,\pi(x)\,dx$ is finite, as

$$\gamma_f = \frac{\mathrm{Cov}(f(\{C_\ell\}_0), f(\{C_\ell\}_1)|d)}{\mathrm{Var}(f(\{C_\ell\})|d)},$$

where $\{C_\ell\}_0 \sim \pi(\{C_\ell|d)$ and $\{C_\ell\}_1$ are two consecutive power spectrum samples computed once the stationarity has been reached. It has been shown in the statistical literature, see Liu (1994), that in the case of data augmentation as in the case under consideration, at stationarity, the lag-1 autocorrelation, γ_f, can be expressed as,

$$\gamma_f = 1 - \frac{\mathbb{E}\{\mathrm{Var}(f(\{C_\ell\})|s,d)|d\}}{\mathrm{Var}(f(\{C_\ell\})|d)}. \tag{3.10}$$

Following the statistical literature results, see Liu et al. (1995) and Liu et al. (1994), it can be shown that the geometric rate of convergence of the Gibbs sampler – see equation (B.1) in Appendix B for a definition, γ, reads,

$$\gamma = \sup_f \gamma_f = \{\sup_{f,g} \mathrm{Corr}(f(C_\ell), g(s)|d)\}^2 \tag{3.11}$$

where the supremum is taken over all functions with finite second order moment under π, and Cov, Var, Corr, and $\mathbb{E}$ stand respectively for covariance, variance, correlation, and expectation value of the arguments. Equation (3.10) shows that the lag-1 autocorrelation is determined by the fraction of the "conditional variance" over the posterior variance. If the conditional variance of the power spectrum given the sky is very small compared to the posterior variance of the power spectrum, then the lag-1 autocorrelation is high, leading to an inefficient sampling of the posterior and the bad mixing of the algorithm. equation (3.11) states that this happens when $\{C_\ell\}$ and s are highly correlated.

This is actually intuitive: when the variance of the conditional distribution is small compared to the posterior one, sampling from this conditional distribution will make only "small steps", changing very little the power spectrum compared to the full range of potential posterior values. This in turn will lead to a small change as compared to the full posterior when sampling the signal conditionally on the power spectrum and so on. Consequently, the algorithm will not explore the posterior distribution efficiently.

Unfortunately, we encounter this problem in our application. Indeed, let us consider the case where we observe the full sky and have an isotropic noise covariance matrix: in this case the matrix $(\mathbf{C} + \tilde{\boldsymbol{Y}}^T \mathbf{N} \tilde{\boldsymbol{Y}})^{-1}$ is diagonal in the harmonic domain and the posterior distribution is a product of inverse translated Gamma distribution and we have roughly:

$$\begin{aligned}\mathrm{Var}(C_\ell|d) &\propto (C_\ell + N_\ell)^2 \\ \mathrm{Var}(C_\ell|s) &\propto C_\ell^2.\end{aligned}$$

Hence, the lag-1 autocorrelation for multipole ℓ reads,

$$\gamma_f^{(\ell)} \approx 1 - \left(\frac{C_\ell}{C_\ell + N_\ell}\right)^2 = 1 - \left(\frac{\mathrm{SNR}_\ell}{\mathrm{SNR}_\ell + 1}\right)^2, \tag{3.12}$$

where SNR_ℓ stands for the signal-to-noise ratio of the power spectrum coefficient corresponding to multipole ℓ, defined as,

$$\mathrm{SNR}_\ell = \frac{C_\ell}{N_\ell}$$

Consequently, the standard Gibbs sampler will not sample the low signal-to-noise components efficiently, as for $\mathrm{SNR}_\ell \ll 1$, $\gamma_f^{(\ell)} \sim 1$, indicating, following on the previous discussion, that the posterior variance will be much bigger than the conditional variance. What, in turn, from equation (3.11) is related to the fact that the correlation between the power spectrum coefficients and the sky maps in this regime are strong.

For high signal-to-noise cases, $\gamma_f^{(\ell)} \sim 0$, and the conditional and posterior variances are comparable, the correlations between the power spectra and the sky are expected to be significantly lower, and we expect that the algorithm will mix well for these components.

All these observations are graphically summarized in Fig. 3.2.

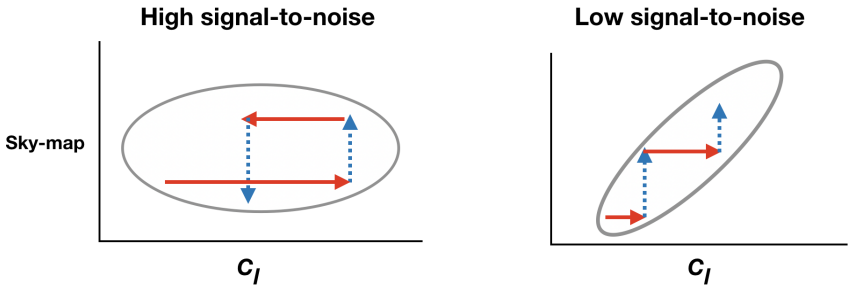

Figure 3.2.: Example of a sequence of consecutive samples of the Gibbs sampler in the centered parametrization. For low signal-to-noise power spectrum coefficients, shown in the right panel, the sky map and the power spectrum are strongly correlated. This leads to *the bad mixing of the algorithm* in this regime and a large number of samples is needed to explore the posterior for such components. This is not the case of the high signal-to-noise components shown in the left panel. Here, the correlations are small and *the resulting mixing of the algorithm is good* with many fewer samples needed to explore the posterior. In both panels the red plain arrows depict sampling of the power spectrum given the sky map and the blue dotted arrows sampling the sky map given the power spectrum.

3.4. Non Centered Gibbs sampling

3.4.1. Algorithm

To circumvent this problem, we reparametrize the model in equation (3.1) to break the dependencies between the signal and the power spectrum. Such an approach was studied in the statistical literature, in e.g. Papaspiliopoulos and Roberts (2003); Papaspiliopoulos et al. (2007); Agapiou et al. (2014), and the CMB context in Jewell et al. (2009). The new model reads,

$$\begin{aligned} \{C_\ell\} &\sim p_0 \\ \tilde{s} &\sim \mathcal{N}(0, \boldsymbol{I}) \\ d &= \tilde{\boldsymbol{Y}} \boldsymbol{C}^{1/2} \tilde{s} + n \end{aligned} \tag{3.13}$$

where $n \sim \mathcal{N}(0, \boldsymbol{N})$ and $\boldsymbol{I}$ is the identity matrix of dimension $(\ell_{\max}+1)^2 - 4$. We plot its directed acyclic graph representation in Figure 3.3.

In this parametrization, the power spectrum, $\{C_\ell\}$, and the signal, $\tilde{s}$, are now independent a priori and all the posterior correlations come from the likelihood of the model.

In order to sample from that model we are also using a Gibbs sampler. Algorithm 8 shows one iteration of the algorithm.

The first step is implemented like the first step of the centered Gibbs sampler except that at its conclusion we change the parametrization: we simulate $s_{t+1} \sim p(s|d, \{C_\ell\}_t)$ and then set $\tilde{s}_{t+1} = \boldsymbol{C}_t^{-1/2} s_{t+1}$. The second step, however, is different. This is because the power spectrum and the observed sky map are

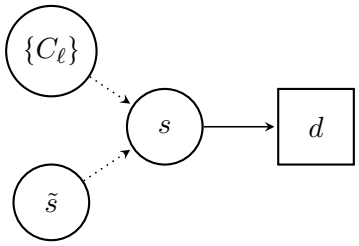

Figure 3.3.: Directed acyclic graph of the model in equation (3.13). Circles and squares represent unobserved and observed variables respectively. Plain arrows represent stochastic dependence. Dashed arrows represent deterministic dependence.

Algorithm 8: Iteration t of the non centered Gibbs sampler

Input: $(\{C_\ell\}_t, \tilde{s}_t)$
Output: $(\{C_\ell\}_{t+1}, \tilde{s}_{t+1})$
1 $\tilde{s}_{t+1} \sim p(\tilde{s}|d, \{C_\ell\}_t)$
2 $\{C_\ell\}_{t+1} \sim p(\{C_\ell\}|d, \tilde{s}_{t+1})$

not independent when conditioned on the signal map. The second conditional density takes the following form,

$$\log p(\{C_\ell\}|\tilde{s}, d) = -\frac{1}{2}(d - \tilde{\boldsymbol{Y}}\boldsymbol{C}^{1/2}\tilde{s})^T\boldsymbol{N}^{-1}(d - \tilde{\boldsymbol{Y}}\boldsymbol{C}^{1/2}\tilde{s}) + c \tag{3.14}$$

where c is a constant. Since we are unable to sample directly from this conditional, we rely on a Metropolis step. This is implemented as follows,

- Propose $\{C_\ell\}_{\text{new}} \sim q(\cdot|\{C_\ell\}_t)$
- Set $\{C_\ell\}_{t+1} = \{C_\ell\}_{\text{new}}$ with probability
$$r = \min(1, \alpha),$$
where
$$\alpha = \frac{\exp\left\{-(d - \tilde{\boldsymbol{Y}}\boldsymbol{C}_{\text{new}}^{1/2}\tilde{s}_t)^T\boldsymbol{N}^{-1}(d - \tilde{\boldsymbol{Y}}\boldsymbol{C}_{\text{new}}^{1/2}\tilde{s}_t)/2\right\}}{\exp\left\{-(d - \tilde{\boldsymbol{Y}}\boldsymbol{C}_t^{1/2}\tilde{s}_t)^T\boldsymbol{N}^{-1}(d - \tilde{\boldsymbol{Y}}\boldsymbol{C}_t^{1/2}\tilde{s}_t)/2\right\}} \times \frac{q(\{C_\ell\}_t|\{C_\ell\}_{\text{new}})}{q(\{C_\ell\}_{\text{new}}|\{C_\ell\}_t)},$$
otherwise set $\{C_\ell\}_{t+1} = \{C_\ell\}_t$.

Here $q(.|\{C_\ell\}_t)$ is the proposal distribution assumed to be normal with a diagonal covariance matrix, centered in $\{C_\ell\}_t$, whose components are truncated to real positive numbers. This algorithm has already been implemented in the context of CMB data analysis in Jewell et al. (2009). In addition, since the problem is very high-dimensional, we decompose $\{C_\ell\}$ into disjoint subsets and we sample each them in turn, one-by-one, while keeping all others fixed following the approach of Jewell et al. (2009). Consequently, we are implementing a Gibbs sampler targeting the distribution in equation (3.14), however each Gibbs step is performed thanks to the Metropolis step. We also follow Jewell et al. (2009) in order to tune the diagonal elements of the covariance matrix of the proposal distribution, q.

3.4.2. Shortcomings

We can already expect this algorithm to suffer from two main shortcomings. First, we still have to solve a high-dimensional linear system, as described in Section 3.3.4. The problems are the same, namely, the

high computational cost of the algorithm and the fact that it may not always converge to a solution which is sufficiently accurate.

The second problem of the non-centered Gibbs sampler is related to the sampling of the power spectrum conditionally on the observed data and the signal map as discussed in, e.g., Jewell et al. (2009). Indeed, when looking at the distribution in equation (3.14) we see that for the low signal-to-noise ratio components, we can make large moves in the parameters space and the value of the density will not change much because the noise is much bigger. Unfortunately the opposite is true for high signal-to-noise ratio components: when the noise is small compared to the power spectrum, making large moves will make large changes in the value of the density, leading to a small acceptance rate in the Metropolis-Hasting algorithm. This is in addition to the fact that a mere use of the non centered parametrization already worsens the mixing properties of the Gibbs sampler on the high signal-to-noise ratio components as visualized in Figure 3.4. This intuition is confirmed by the experiments made in Jewell et al. (2009).

Consequently, we still need to find an alternative algorithm that is capable of sampling efficiently the high and low SNR simultaneously.

3.5. Interweaving

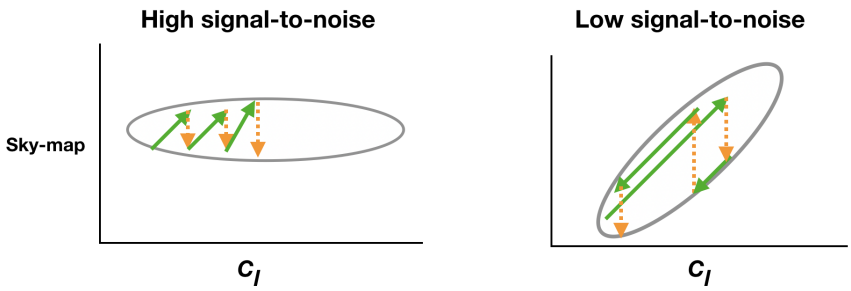

Figure 3.4.: Example of a sequence of samples of the non-centered Gibbs sampler. For low signal-to-noise power spectrum components, the sky map and the power spectrum are strong correlated, right panel. This is not the case of the high signal-to-noise components, left panel. The green plain arrows depict sampling the power spectrum given the sky map and the orange dotted arrows the sky map given the power spectrum. As shown, the non-centered Gibbs sampler explores the high signal-to-noise components much less efficiently, left panel, than the low signal-to-noise components, right panel.

3.5.1. Algorithm

The idea of interweaving, also called ASIS in the statistical literature, see Yu and Meng (2011), allows us to capitalize on the properties of the centered and non-centered Gibbs algorithm presented earlier. This is done by combining both these samplers together rather than simply alternating between them. In this section we apply this idea to the power spectrum estimation.

The interweaving scheme proposed here applies first the Gibbs kernel as described in Section 3.3, followed by changing the variable to get a non-centered version of the algorithm and finally concluding by sampling the power spectrum as explained in Section 3.4. These steps are implemented in Algorithm 9. The first two steps of the algorithm are the usual centered Gibbs sampler, Section 3.3. The third step constitutes a change of variable that shifts to the non-centered version of the Gibbs sampler. The fourth

Algorithm 9: Iteration t of ASIS

Input: $(\{C_\ell\}_t, s_t)$
Output: $(\{C_\ell\}_{t+1}, s_{t+1})$

1 $s_{t+0.5} \sim p(s|d, \{C_\ell\}_t)$
2 $\{C_\ell\}_{t+0.5} \sim p(\{C_\ell\}|s_{t+0.5})$
3 $\tilde{s}_{t+0.5} = \mathbf{C}(\{C_\ell\}_{t+0.5})^{-1/2} s_{t+0.5}$
4 $\{C_\ell\}_{t+1} \sim p(\{C_\ell\}|d, \tilde{s}_{t+0.5})$
5 $s_{t+1} = \mathbf{C}(\{C_\ell\}_{t+1})^{1/2} \tilde{s}_{t+0.5}$

step effectively samples the power spectrum from the non-centered parametrization, while the fifth goes back to the centered one.

We note that we can look at the interweaving algorithm as an Alternating Subspace-Spanning Resampling algorithm (ASSR), see Liu (2003) with the underlying MCMC algorithm being the centered Gibbs sampler and the mapping defined as $\mathcal{M}(\{C_\ell\}, s) = (\{C_\ell\}, \tilde{s})$.

Intuitively, the algorithm will have better mixing properties than the centered and non-centered Gibbs sampler algorithms. First, interweaving will mix as well as the centered Gibbs sampler on the high SNR components, thanks to Steps 1 and 2, see Section 3.3. It will also mix as well as the non-centered Gibbs sampler on the low SNR components, thanks to the change of variable and sampling in Steps 3 and 4. Second, we are not only exploiting the strength of each algorithm. We can expect interweaving to show a "compound effect": the high SNR components will still benefit a bit from the non-centered step, however inefficient it may be, and vice-versa.

So far we have proposed an algorithm that we expect to behave nicely on a broad range of signal-to-noise ratios. But the constrained realization step is still a problem: whatever the mixing properties of the algorithm we are using, the cost of one iteration is still very high and this is expected to continue to be a major hindrance for the applications.

3.6. Constrained realization step

Solving the constrained realization equation, equation (3.7), is a problem for several reasons. First, this system is high dimensional and dense and computing explicitly the inverse of its system matrix, Q, would be really time consuming, not to mention the memory requirements to store it. These issues can be efficiently handled by the use of an iterative solver, most commonly a preconditioned conjugate gradient (PCG) algorithm. However, iterative algorithms solve the system only up to some pre-defined accuracy and require sometimes a large number of iterations to provide a sufficiently precise solution. Trading on this may speed up time to solution but can result in a bias, effects of which are hard to quantify.

One solution would be to add a Metropolis-Hastings step after we proposed a new sky map sample using equation (3.7). This indeed would ensure that the accepted constrained realization solutions conform indeed with the desired posterior. However, such a naive implementation would lower the acceptance rate resulting into high autocorrelations between the successive samples.

We propose two alternative solutions to these two problems in this section. First, we present an auxiliary variable scheme that allows us to add a Metropolis-Hastings step without reducing the efficiency of the sampler. Second, we introduce another auxiliary variable that allows us to eliminate altogether any need to sample exactly from a high dimensional normal distribution with a dense covariance matrix.

These two algorithms leave distribution in equation (3.5) invariant, leading therefore to a valid Gibbs Sampler.

3.6.1. Reversible jump perturbation optimisation step

Our first approach is based on an algorithm called in the statistical literature the Reversible-Jump Perturbation Optimization (RJPO) algorithm described in Gilavert et al. (2015).

We start from augmenting the model with an auxiliary variable z such that

$$z|s \sim \mathcal{N}(\boldsymbol{Q}s + \boldsymbol{Q}\mu, \boldsymbol{Q})$$

where $\boldsymbol{Q}$, μ are defined equation (3.6) and s is distributed according to equation (3.5). We then perform a Metropolis-Hastings move on this augmented target.

Our proposal consists in the following deterministic, differentiable and reversible transformation:

$$\phi(s, z) = (-s + f(z), z) = (s', z).$$

Following Gilavert et al. (2015), the Metropolis acceptance rate for this proposal writes:

$$\min(1, e^{-r(z)^t(s-s')}),$$

where $r(z) := z - \boldsymbol{Q}f(z)$.

On choosing $f(z) = \boldsymbol{Q}^{-1}z$, the acceptance rate of the Metropolis-Hastings scheme is one, and we accept every proposed move. In addition, as shown in Gilavert et al. (2015), this choice of $f(z)$ leads to uncorrelated successive samples.

Note that in this case $s' = -s + \boldsymbol{Q}^{-1}z = -s + \boldsymbol{Q}^{-1}(\boldsymbol{Q}s + \eta) = \boldsymbol{Q}^{-1}\eta$ where $\eta \sim \mathcal{N}(\boldsymbol{Q}\mu, \boldsymbol{Q})$ which means we are solving the exact same system as for sampling from equation (3.5) in the usual centered Gibbs sampler.

As we explained before, the problem is that we are unable to solve this system exactly and instead we have to relay on some iterative algorithms that in the interest of time we stop once some predefined precision has been reached. The scheme considered here allows to account on such effects.

Indeed, instead of defining $f(z) = \boldsymbol{Q}^{-1}z$, we can define it as the output of a truncated iterative solver like the preconditioned conjugate gradient algorithm. applied to the system $\boldsymbol{Q}f(z) = z$. The acceptance rate of the corresponding Metropolis-Hastings algorithm will not be 1 and the correlations between two successive samples will not be 0 anymore. Since the ratio depends on $r(z)$, the more precisely we solve the system, the higher the acceptance rate, but so is the computational cost. With this algorithm, we are facing a computational efficiency/autocorrelation tradeoff.

Let us denote $\hat{u}$ the approximate solution of $\boldsymbol{Q}f(z) = z$. Now we have $s' = -s + \hat{u}$ and $r(z) = z - \boldsymbol{Q}\hat{u} = z - \boldsymbol{Q}(s + s') = \eta - \boldsymbol{Q}s'$. Finally the algorithm reads as Algorithm 10.

Algorithm 10: RJPO algorithm

1 Sample $\eta \sim \mathcal{N}(\boldsymbol{Q}\mu, \boldsymbol{Q})$
2 Solve $\boldsymbol{Q}\hat{s} = \eta$ approximately
3 Compute $\alpha = \min(1, e^{-r(z)^t(s-\hat{s})})$ where $r(z) := \eta - \boldsymbol{Q}\hat{s}$.
4 With probability α, set $s' = \hat{s}$, otherwise set $s' = s$.

It is remarkable that the first and second steps of this RJPO algorithm are exactly the same system as for sampling from distribution in equation (3.5). We are just adding a Metropolis-Hastings step to ensure that we are leaving this distribution invariant, however approximately we solve the system.

The presence of such a Metropolis step allows us to solve the system with an arbitrary precision without biasing the Metropolis-within-Gibbs algorithm. Thus, we can spare some computation time by decreasing the precision required to solve the system.

If one chooses to solve the system exactly, the RJPO algorithm always accepts the proposed move and the successive samples are uncorrelated. In this case, RJPO is exactly the same as the PO algorithm used to sample from equation (3.5) in Section 3.3.2.

If instead we decide to solve the system only approximately, we introduce correlations between successive samples and the acceptance rate will depend on how approximate we solve it: the more precise we are, the higher the acceptance rate.

Even though the RJPO algorithm has nice properties, it still involves solving a very high dimensional system, at least very approximately. We also have to arbitrate between a lower computing time and higher autocorrelations. In the next section we present another auxiliary variable scheme that completely bypasses such inconveniences.

3.6.2. Augmented Gibbs step

Instead of shortening the computing time needed to solve the constrained realization linear system as we did in the previous subsection, we may avoid it completely and rely on a different MCMC scheme. The dimension is huge though, and we would like to avoid the computation of an acceptance ratio. A Gibbs sampler seems a natural solution. The relevant algorithm has been originally proposed in the statistical literature in Marnissi et al. (2018). In this Section we describe it and adapt the generic algorithm to the specific case of the CMB power spectrum estimation.

Gibbs step

Instead of sampling directly from the conditional distribution, equation (3.5):

$$\pi(s|\{C_\ell\}, d)$$

we augment it with an auxiliary variable v so that sampling from $\mathcal{L}(v|s, \{C_\ell\}, d)$ and $\mathcal{L}(s|v, \{C_\ell\}, d)$ is easier. We choose a v such that:

$$v|s, \{C_\ell\}, d \sim \mathcal{N}(\mathbf{\Gamma}\tilde{\boldsymbol{Y}}s, \mathbf{\Gamma}) \tag{3.15}$$

where $\mathbf{\Gamma} := (\beta \boldsymbol{I} - \boldsymbol{N}^{-1})$ and β is a scalar chosen so that $\mathbf{\Gamma}$ is positive definite. This gives us the following conditional distribution (up to an irrelevant prior on v),

$$s|v, \{C_\ell\}, d) \sim \mathcal{N}(\boldsymbol{M}\tilde{\boldsymbol{Y}}^T(v + \boldsymbol{N}^{-1}d), \boldsymbol{M}) \tag{3.16}$$

where $\boldsymbol{M} := (\frac{\beta N_{\rm pix}}{4\pi}\boldsymbol{B}^2 + \boldsymbol{C}^{-1})^{-1}$. Note that both $\mathbf{\Gamma}$ and $\boldsymbol{M}$ are diagonal – or block diagonal matrices in the case of temperature and polarization.

We note that we can sample efficiently from these two conditional distributions, and consequently we are able to sample from the distribution in equation (3.5) as well and to do so without any need for solving the constrained realization problem. Indeed, we can simply use the Gibbs sampler and draw pairs of (s, v) consecutively from their conditional distributions and since $\int \pi(s, v|d, \{C_\ell\})dv = \pi(s|d, \{C_\ell\})$, we merely discard v at the end. Such a scheme leaves distribution in equation (3.5) invariant.

Even though this augmented Gibbs step is computationally efficient, its overall performance will mainly depend on the correlations between v and s. If we write the joint distribution of (s, v), we realize that they are jointly Gaussian with covariance matrix,

$$\left[\begin{array}{c|c} \mathbf{\Sigma} & \mathbf{\Sigma}\tilde{\boldsymbol{Y}}^T\mathbf{\Gamma} \\ \hline \mathbf{\Gamma}\tilde{\boldsymbol{Y}}\mathbf{\Sigma} & \mathbf{\Gamma} + \mathbf{\Gamma}\tilde{\boldsymbol{Y}}^T\mathbf{\Sigma}\tilde{\boldsymbol{Y}}\mathbf{\Gamma} \end{array}\right] \tag{3.17}$$

Looking at equation (3.17), we see that $\mathbf{\Sigma}$, defined in equation (3.5), is influencing the correlations between s and v. From our earlier analysis, see also Eriksen et al. (2004), this may be a problem since $\mathbf{\Sigma}$ may be dense for high signal-to-noise ratio components. We can expect this Gibbs step to show poor mixing on these components. However, for lower SNR ratio components, $\mathbf{\Sigma}$ tend to be band diagonal and we can expect this Gibbs move to be much more efficient. We confirm this expectation with help of numerical experiments in Section 3.7.2.

3.6.3. Overrelaxation

The overrelaxation method, see Neal (1998) for the statistical background, is a way around these strong correlations. Instead of sampling successively from distributions in equations (3.15) and (3.16), we are going to sample from,

$$v^{t+1} = \mathbf{\Gamma}\tilde{\boldsymbol{Y}} s^t + \gamma(v^t - \mathbf{\Gamma}\tilde{\boldsymbol{Y}} s^t) + \mathbf{\Gamma}^{1/2}(1-\gamma^2)^{1/2} Z_1 \tag{3.18}$$

and,

$$s^{t+1} = \boldsymbol{M}\tilde{\boldsymbol{Y}}^T (v^{t+1} + \boldsymbol{N}^{-1} d) + \gamma(s^t - \boldsymbol{M}\tilde{\boldsymbol{Y}}^T (v^{t+1} + \boldsymbol{N}^{-1} d)) + \boldsymbol{M}^{1/2}(1-\gamma^2)^{1/2} Z_2 \tag{3.19}$$

where Z_1, Z_2 are two independent standard normal variables, Z_1 has the dimension of v and Z_2 of s. Here $\gamma \in]-1, 1[$ is a parameter chosen by the user.
It is straightforward to show that the move in equation (3.18) leaves the distribution in equation (3.15) invariant, i.e., the distribution of v_{t+1} is given by equation (3.15) if that of v_t is, and that the move in equation (3.19) leaves the distribution in equation (3.16) invariant. In addition, it has been argued in the statistical literature, see Neal (1998), that such a "symmetrical" conditional move around the mean make it possible for the Gibbs sampler to move in a consistent direction in the presence of correlations, thus suppressing the random-walk behavior of the Gibbs sampler.

3.7. Experiments

In this section we consider several experiments. For the first comparison of our algorithms, we assume that we observe the entire sky. This way, the covariance matrices are diagonal, the centered, non-centered and interweaving algorithms are computationally cheap and we can easily draw many samples.

In the second round of experiments we assume exactly the same setting as the first one, except that we apply the 80% Planck mask leading to a posteriori coupled multipoles.

In both cases in order to test our algorithms in the circumstances reflecting potential future applications we assume noise levels and the resolution reflecting roughly those of the future CMB satellite mission, LiteBird, see Hazumi and Group. (2020).

The final set of experiments is designed to mimic a ground-based setup. We take here very roughly the parameters of the 90GHz frequency channel of the Simon Observatory, see Ade et al. (2019). We assume a sky coverage of 37%, what leads to even more strongly coupled multipoles.

3.7.1. Polarization full-sky experiment

For this first experiment comparing interweaving and the centered and non centered Gibbs algorithms, we assume we observe the entire sky and that the noise covariance matrix writes $N = \alpha^2 I$, where I is the identity matrix of dimension N_{pix}. For ease of implementation we are doing inference on EE and BB components of the power spectrum only, assuming only Q and U maps are observed. In this case we can exactly sample a map from the constrained realization step in equation (3.5) at no cost since the covariance matrix of the normal distribution is diagonal. In addition, the power spectrum components are a posteriori independent and we have an analytical expression for each marginal distribution.

Regarding the set-up, we choose $\mathrm{NSIDE} = 256$ with $\ell_{\max} = 512$ and we apply an instrumental beam of 30-arcmin fwhm. We choose a rms noise of $\alpha = 0.2\mu K$-arcmin. Since the BB components have a very low signal-to-noise ratio for the highest multipoles, we make progressively wider bins, starting at $l = 396$ - corresponding to $\mathrm{SNR} = 0.24$ - and get a total of 412 multipoles instead of the 512 initial ones.

To make the comparison, we first run 10 chains of length 300 for each algorithm – with an exception of the centered Gibbs – and use these samples to calibrate the proposal distributions of the non-centered power sampling step. Once we have the covariance matrices of the proposal distributions, we run 10 chains of 10^4 iterations for each algorithm and compute the relevant metrics on this basis.

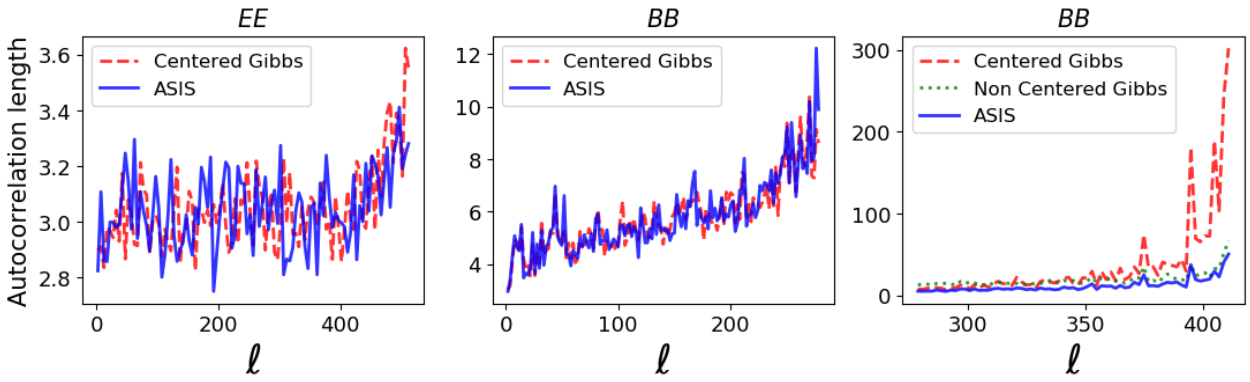

Figure 3.5.: Integrated autocorrelation times against multipole for each algorithm. The BB components are split into two graphs for readability, because of the broad range of scales they cover.

In order to evaluate the respective performances of these three algorithms, we look at the respective Integrated Autocorrelation Time (IAT) for each one of them accross all the multipoles, where

$$\mathrm{IAT}_\ell := 1 + 2 \sum_{k=1}^{N_{\mathrm{lag}}} \rho_k^\ell$$

and ρ_k^ℓ is the autocorrelation of the chain at lag k for C_ℓ defined at stationarity as:

$$\rho_k^\ell := \frac{\mathrm{Cov}(C_{\ell,0}, C_{\ell,k})}{\mathrm{Var}(C_\ell|d)}$$

with $C_{\ell,0} \sim p(C_\ell|d)$ and $C_{\ell,k}$ is obtained after k iterations of Gibbs sampler, starting at $C_{\ell,0}$.

Figures 3.5 and 3.6 show the IAT for each algorithm against multipoles index and the logarithm of the signal-to-noise ratio respectively. Since it is known, see Jewell et al. (2009), that the non-centered Gibbs sampler does not mix well – however good is our tuning – for medium to high SNR, we only tuned it on components having signal to noise ratio inferior to 1. For readability, we only show its performances on these components and we split the BB components in two parts, which have very different scales, and display them in two different panels. As expected, the interweaving algorithm mixes as well as the centered Gibbs on signal-to-noise ratio superior to 1. However, when the SNR starts to be low, the integrated autocorrelation times of the centered Gibbs algorithm increases sharply compared to those of the interweaving. Note also that for the lowest signal-to-noise ratios, the non-centered version of the Gibbs sampler performs better than the centered version, as expected, and that the interweaving have even lower integrated autocorrelation times than non centered Gibbs. These results are in agreement with the analysis of Section 3.5.

In order to have a clearer picture of the respective performances of the algorithms, we plot the ratio of $\mathrm{IAT}^{\mathrm{centered}}/\mathrm{IAT}^{\mathrm{asis}}$ and $\mathrm{IAT}^{\mathrm{noncentered}}/\mathrm{IAT}^{\mathrm{asis}}$ against the multipoles in Figures 3.7 and 3.8, respectively. We produce the same graphics against the log signal-to-noise ratio Figures 3.9 and 3.10. It is clear from these plots that the interweaving algorithm inherits the excellent mixing properties of the centered Gibbs on high signal-to-noise ratio components while outperforming it on the lower ones. In addition, interweaving outperforms the non-centered Gibbs algorithm on lower signal-to-noise range. We provide example of histograms for a wide range of signal-to-noise ratios in Appendix C.1.

This simple experiment shows how good the mixing properties of the interweaving algorithm are on the full range of multipoles as it is able to sample efficiently for high and low signal-to-noise ratios. However, because of the absence of sky-cut, these algorithms are very cheap computationally. Our analysis of the respective efficiencies of the algorithms does not take into account the computing time. In addition, in the

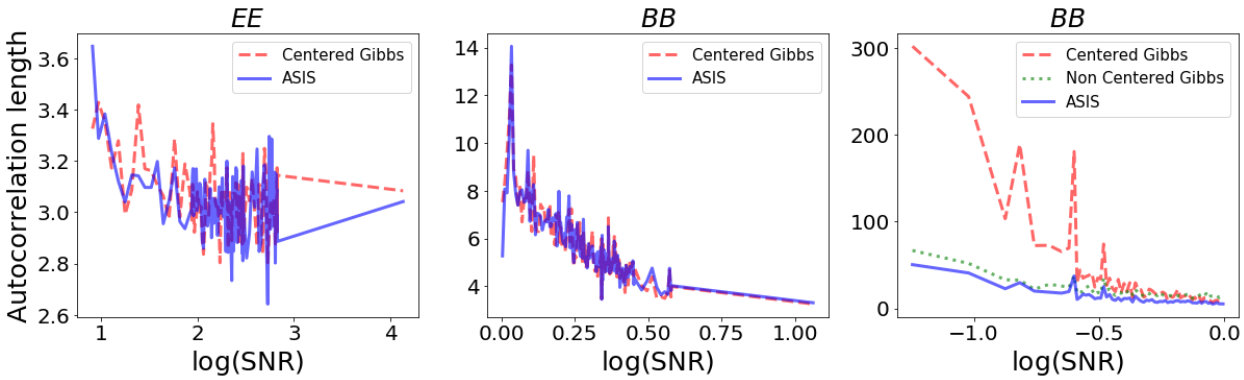

Figure 3.6.: Integrated autocorrelation time against log(SNR) for each algorithm. The BB components are split into two graphs for readability, because of the very different scales.

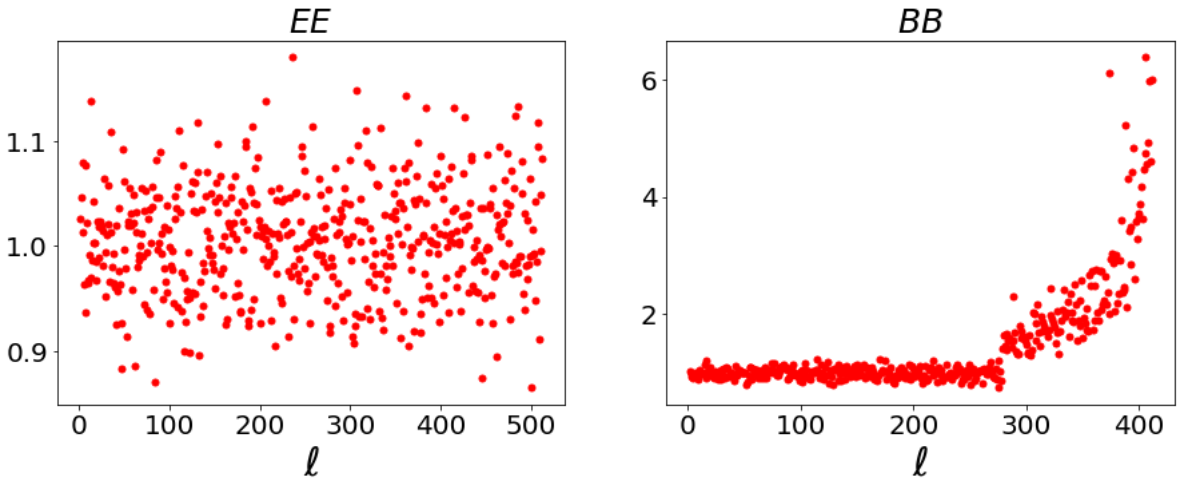

Figure 3.7.: Ratios of Integrated Autocorrelation Time against multipole. Numerator: centered Gibbs. Denominator: interweaving. A ratio superior to 1 indicates that centered Gibbs is performing better than interweaving in terms of autocorelation time.

presence of a sky-mask, the $\{C_\ell\}$ are no longer independent and this may hinder the non-centered power spectrum sampling step. In order to test the algorithms in more realistic contexts, we consider a second experiment with a cut-sky in the next section.

3.7.2. Nearly full-sky polarization experiment

The set-up of this experiment is exactly the same as the one Section 3.7.1 except that we apply the 80% Planck sky mask, that we plot in Appendix C.2. We use the same binning and blocking schemes as in the previous section.

Because we do not analyze the full sky, we cannot access the true posterior distribution and the system to solve for the constrained realization step is no longer diagonal: we have to rely on a TPO algorithm using a preconditioned conjugate gradient (PCG) solver, see Gilavert et al. (2015), with a diagonal preconditioner, as done in previous works, in e.g Jewell et al. (2009), Eriksen et al. (2004) and explained in Section 3.3.2.

We also try a centered Gibbs sampler and an interweaving algorithm with a Gibbs step on the augmented conditional instead of doing the usual system resolution. We test the interweaving algorithm with a RJPO, see Gilavert et al. (2015), constrained realization step. We described each algorithm in Section 3.6. We

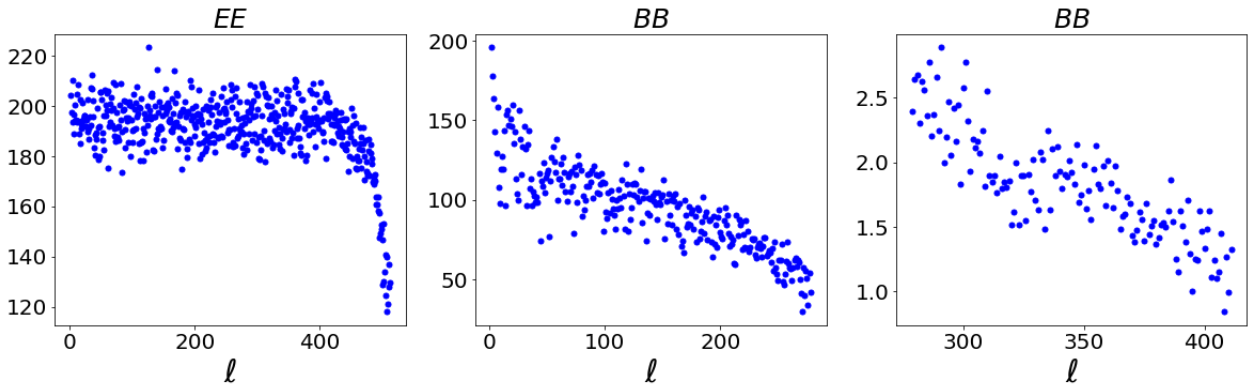

Figure 3.8.: Ratios of Integrated Autocorrelation Time against multipole of non centered Gibbs on interweaving. Note that the BB components are split in two graphs for readability, because of the very broad range of scales.

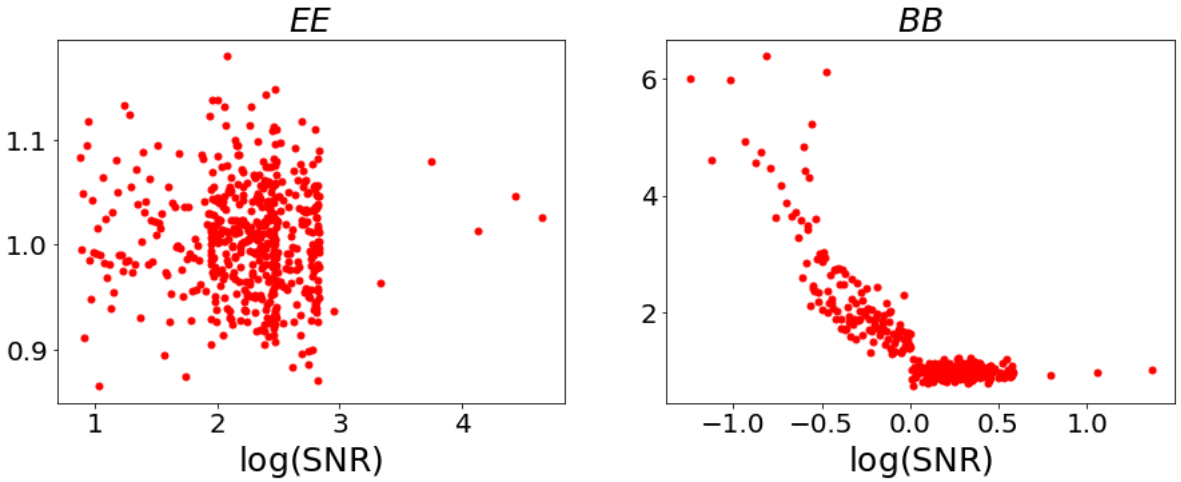

Figure 3.9.: Ratios of Integrated Autocorrelation Time against log signal-to-noise ratio of centered Gibbs on interweaving.

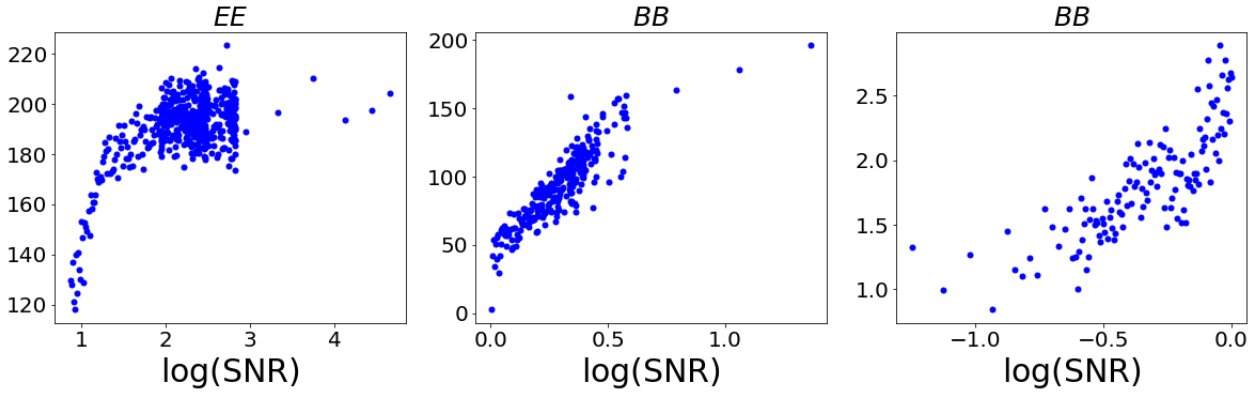

Figure 3.10.: Ratios of Integrated Autocorrelation Time against signal-to-noise ratio for non centered Gibbs on interweaving. Note that the BB components are split in two graphs for readability, because of the very different scales.

Algorithm	**Constrained realization**	**power spectrum sampling**
Centered	PCG	centered move
ASIS	PCG	centered + non-centered moves
ASIS RJPO	RJPO	centered + non-centered moves
ASIS 1	auxiliary variable	centered + non-centered moves
ASIS 20	auxiliary variable	centered + non-centered moves
ASIS 65	auxiliary variable	centred + non-centered moves
Centered 1	auxiliary variable	centered move
Centered overrelax	overrelaxation	centered move

Table 3.1.: Summary of the approaches used to address constrained realization and power spectrum sampling steps for each of the algorithms studied in this work.

follow Eriksen et al. (2004) and set the error threshold to 10^{-6} when using the TPO algorithm, except for ASIS RJPO, where it is set to 10^{-5}.

In the following, "Centered" will denote the usual centered Gibbs algorithm, "ASIS" and "ASIS RJPO" will mean interweaving and interweaving with a RJPO step respectively. The format "algorithm name + integer" will denote the algorithm "algorithm name" with the PCG solver being replaced by the augmented Gibbs sampler for the "integer" number of iterations. For example, "Centered 1" is the usual centered Gibbs algorithm with the PCG step replaced by the augmented Gibbs sampler for one iteration only, setting $\beta = \alpha^{-2} + 10^{-14}$. The name "Centered overrelax" denotes the centered Gibbs algorithm with the PCG step replaced by two iterations of overrelaxation plus one iteration of classical augmented Gibbs step, with $\gamma = -0.995$ chosen to be close to -1 to deal with the strong correlations of the covariance matrix, equation (3.17). Table 3.1 summarizes the algorithms.

We tune the algorithms as follows. We run each one of them for a few hundreds iterations. Based on the results, we estimate the covariances of the marginal of each multipole and use them as the proposal covariances – multiplied by a scalar inferior to one – for the actual run, targeting a 25% acceptance rate. After tuning, every algorithm is run for 10^3 iterations, except for Centered overrelax and Centered 1 which are run for 10^5 iterations since they are computationally cheaper.

We first look at the Effective Sample Sizes (ESS) per second of each algorithm. If we run the algorithm for N iterations, the ESS for component ℓ is defined as:

$$\mathrm{ESS}_\ell := \frac{N}{\mathrm{IAT}_\ell}$$

where IAT_ℓ is defined in Section 3.7.1. The ESS per second, for each component, is then defined as the ESS for the N iterations divided by the CPU time in second needed for the N iterations. Obviously, for any ℓ, the greater ESS_ℓ per second, the better.

We plot the ESS per second in Figure 3.11. Centered 1 and Centered overrelax outperform the other algorithms in term of ESS per second, whatever the SNR. Otherwise, the algorithms using the PCG sampling step seem to be performing better on the EE components than the ones using an auxiliary Gibbs step, especially compared to ASIS 1 which underperforms. The opposite is true on BB. We can easily explain these observations: the augmented Gibbs constrained realization step is much cheaper than a PCG resolution of the system, but it also leads to much worse mixing properties on EE but not on BB. We are facing a trade off between computing time and mixing on EE: the smaller the number of augmented Gibbs step, the faster the algorithm but the greater the autocorrelations. The same holds for ASIS RJPO. We must then find the number of Gibbs steps maximizing the ESS per second. But overall it seems these algorithms will not perform as well as their PCG counterparts on EE.

The reader should also note that the non-centered step of interweaving comes with a cost that cannot be reduced: in our case roughly 130 spherical harmonic synthesis operations. That is why ASIS 1 performs much worse than ASIS 20 and ASIS 65: as the algorithm has to perform at least 130 spherical harmonic transforms, one could as well do 20 augmented Gibbs constrained realization step instead of 1, improving the mixing properties of the algorithm without increasing the computing time so much, leading to a better ESS per second.

The picture is different for the BB components: the augmented Gibbs constrained realization step is mixing much better on such lower signal-to-noise ratios and hence the algorithms using such a step have a better ESS per second than their PCG counterparts. Note that Centered 1 and Centered overrelax are outperforming all the other algorithms by far. That is because it comes at almost no cost – only one spherical harmonic analysis and one synthesis per iteration. This has to be compared to the heavy cost of the PCG solver, typically 150 PCG iterations, complemented by 2 spherical harmonics transform per iteration, of the Centered, ASIS and ASIS RJPO algorithms, and to the incompressible cost of the non centered step of the ASIS 1, ASIS 20 and ASIS 65 algorithms. Since the augmented Gibbs step mixes well on this SNR, Centered 1 and Centered overrelax are good mixing and cheap algorithms, hence their ESS per second is much better. One must be careful though: this behavior tends to fade on very low SNR: the centered parametrization has greater and greater autocorrelations as the SNR decreases.

In order to get a better idea of the relative performances of the algorithms, we examine the ratios of ESS per second of each algorithm on the ESS per second of the usual centered Gibbs and of the interweaving algorithm, Tables 3.2 and 3.3.

These tables confirm the behavior we described above: on EE components ASIS 1 and ASIS 65 are outperformed by Centered and ASIS in terms of ESS per second, while ASIS 20 seems to perform similarly. Note that the algorithms tend to perform worse in comparison to Centered than compared to ASIS: thought ASIS and Centered have roughly the same mixing on EE, ASIS is more expensive than Centered. In addition, Centered is outperforming ASIS on EE because it is a bit cheaper. On BB however, each algorithm seems to outperform Centered. Again, this is because the augmented Gibbs constrained realization step leads to as good a mixing as the PCG resolution while dramatically reducing the overall cost of the algorithms, leading in turn to a much better ESS per second. As for ASIS, its ESS per second is greater than the one of Centered only for the lowest signal-to-noise ratio components. This indicates that we could probably have applied the non-centered step on these components only: the algorithm would have been cheaper while still having good mixing properties on low SNR components, leading to a much better ESS per second, on both EE and BB.

We should pay a closer attention to Centered 1 and Centered overrelax. These algorithms are computationally cheap compared to any other algorithm. Hence, whatever their mixing properties on EE components and on very low SNR components, their ESS per second is much higher. In addition, the ESS per second of Centered overrelax is higher than the one of Centered 1 on EE components, showing that these step is handling the strong correlations better.

Finally, Figure 3.12 shows the empirical mean posterior of Centered overrelax with the two standard deviations intervals. The solid black lines denotes the true spectrum. The recovered spectrum seems to

Algorithm	5th	25th	50th	75th	95th
ASIS	0.544	0.685	0.772	0.88	1.061
ASIS 1	0.095	0.126	0.16	0.225	0.9
ASIS 20	0.287	0.477	0.697	1.044	1.997
ASIS 65	0.396	0.575	0.786	0.988	1.288
ASIS RJPO	0.647	0.79	0.896	1.013	1.217
Centered 1	1.015	1.862	2.915	5.303	29.31
Centered overrelax	2.843	4.708	6.925	11.265	28.635

Table 3.2.: For each algorithm, percentiles of their ESS per second relative to the ESS per second of the Centered algorithm for EE components.

match the true spectrum well.

3.7.3. Polarization cut sky experiment

The set-up of this second cut-sky experiment is the same as in the preceding section, except that we apply the Simon Observatory-motivated 37% sky mask, see Ade et al. (2019), that we plot in Appendix C.2, and set the noise rms to $\sigma = 0.28\mu K$ per pixel for both EE and BB components. Since we have very low SNR components we start binning the BB multipoles at $\ell = 320$ into progressively wider bins. After binning is applied, we are left with 331 bins out of the 512 initial multipoles. Regarding the blocking scheme for the non centered power spectrum sampling step, we make one block for multipoles $2 \leqslant \ell \leqslant 280$ and make blocks of size one for $280 < \ell \leqslant 331$.

As in the previous section, we make tuning runs of 10 parallel chains of 300 iterations. Then, we run all algorithms for 10 parallel chains of 10^3 iterations, except for the Centered 1 and Centered overrelax cases, for which we run 10 parallel chains of length 10^5. Still following Eriksen et al. (2004), we set the threshold for the PCG algorithm to 10^{-6}, except for the interweaving algorithm with RJPO step, for which we set the threshold to 10^{-5}.

Figure 3.13 shows the ESS per CPU second against $\log(\mathrm{SNR})$. ASIS, ASIS RJPO and Centered algorithms tend to perform the same, except on the lower range of SNR where Centered algorithm is outperformed by ASIS and ASIS RJPO. We also note ASIS RJPO performs better than ASIS on the low SNR components.

Clearly, the Centered 1 and Centered overrelax variants outperform any other algorithm, sometimes by several orders of magnitude and over almost the entire range of SNR: that is because it is computationally very cheap compared to the other algorithms. Note however that its mixing properties degrade with too

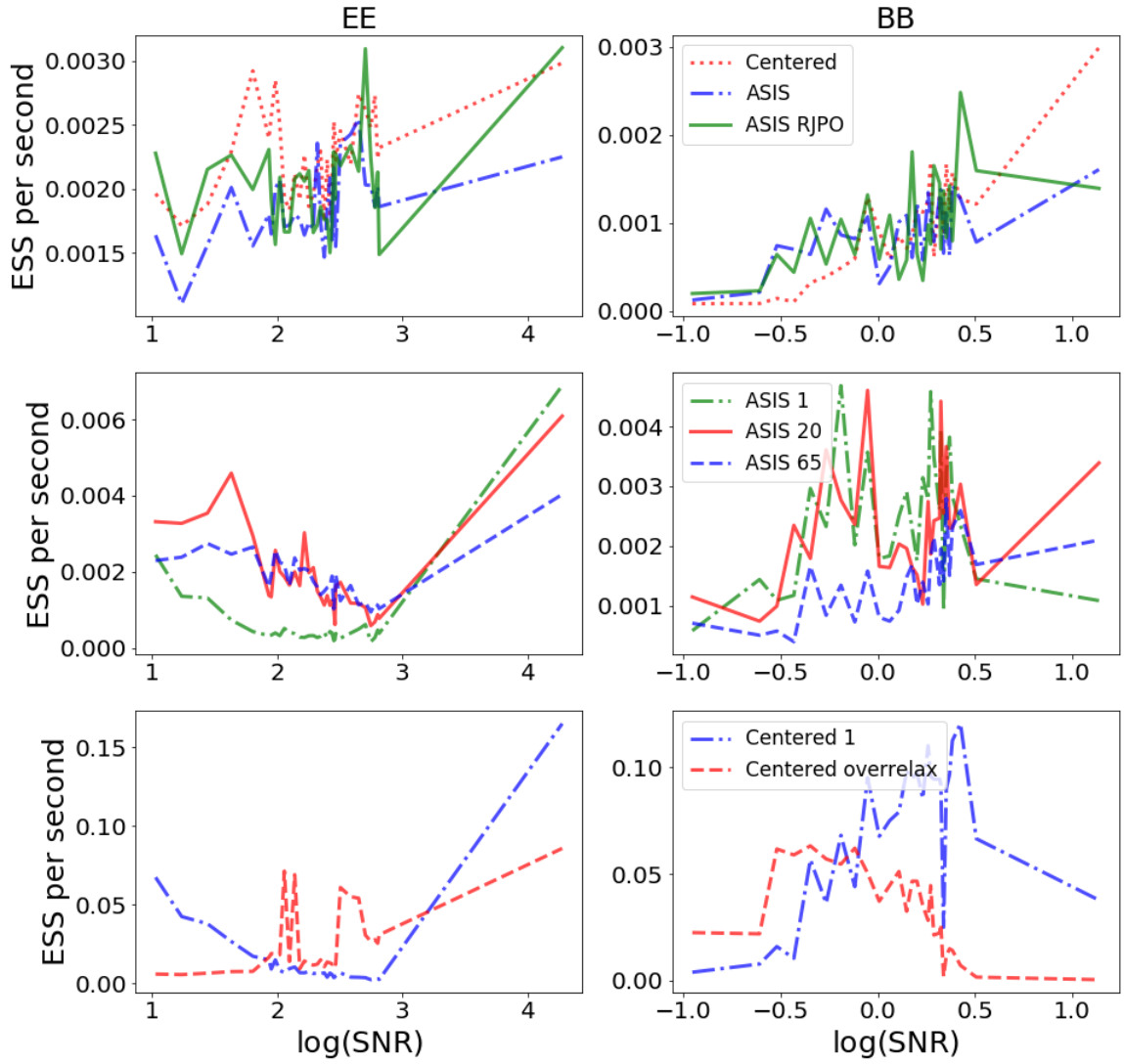

Figure 3.11.: Effective Sample Size per second against $\log(\mathrm{SNR})$ for each algorithm. For the sake of clarity, we group the algorithms with a similar performance and plot different groups in separate panels. The left columns shows the results for the EE and the right one for the BB spectra.

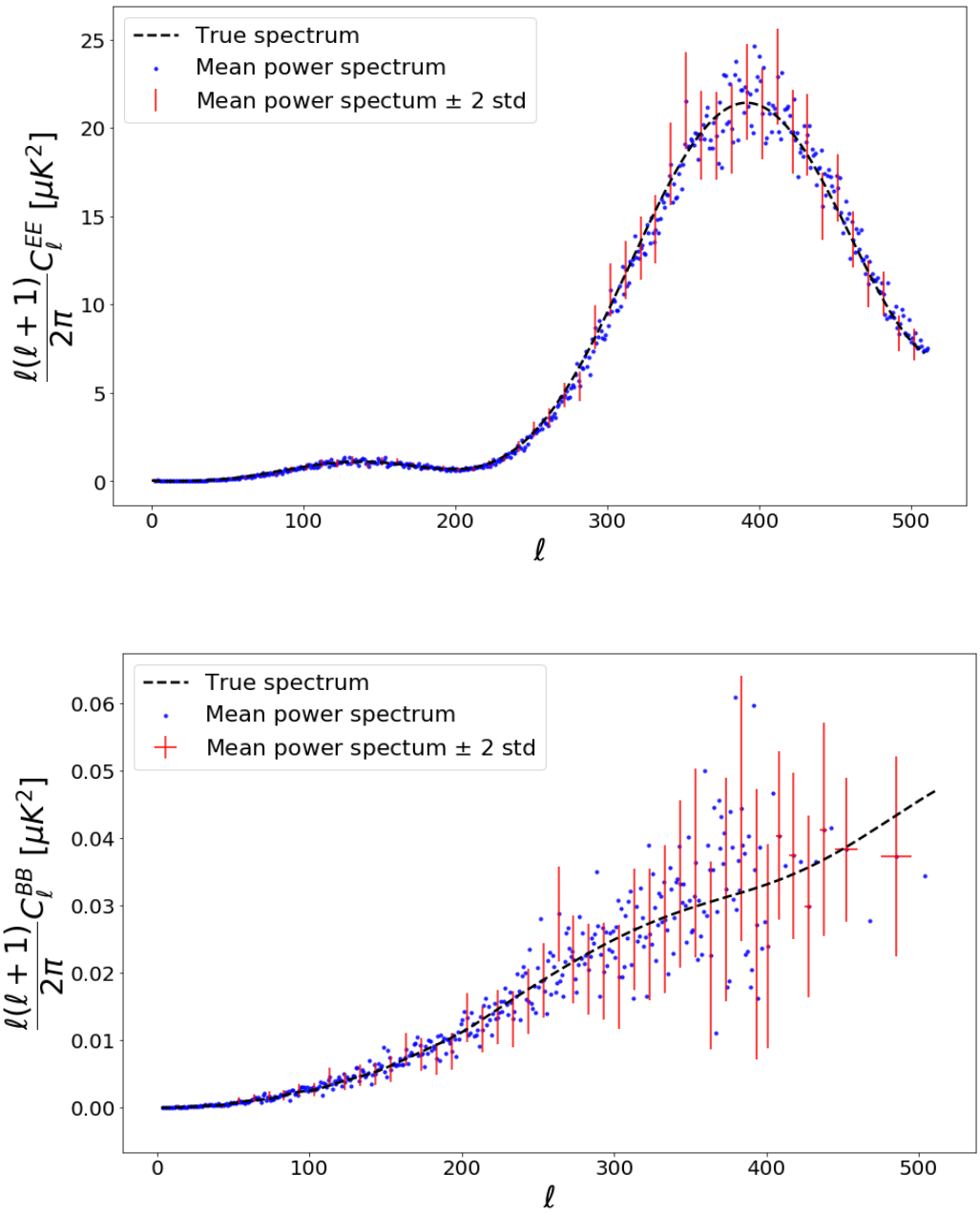

Figure 3.12.: Comparison of posterior power spectrum to the true power spectrum for the LiteBird-like experiment. The blue points correspond to the mean power spectrum. The top and bottom of the horizontal bars of the crosses correspond to the mean plus/minus two standard deviation. The horizontal bars correspond to the ℓ range spanned by the binning scheme. We only plot the crosses for one every ten multipoles on the non binned part. On the binned part, we only plot the crosses for one every two multipoles. The panels show EE (top) and BB (bottom) power spectra.

Algorithm	5th	25th	50th	75th	95th
ASIS	0.376	0.697	1.016	1.625	3.966
ASIS 1	1.04	1.694	2.696	4.936	14.132
ASIS 20	1.115	1.689	2.466	3.972	10.15
ASIS 65	0.567	0.967	1.477	2.522	6.103
ASIS RJPO	0.391	0.644	1.007	1.719	4.097
Centered 1	38.62	63.838	83.914	116.926	169.758
Centered overrelax	2.173	14.331	36.227	100.185	492.866

Table 3.3.: For each algorithm, percentiles of their ESS per second relative to the ESS per second of the Centered algorithm for BB components.

high and too low SNR. That is because the auxiliary step mixes worse on high SNR while the centered parametrization provides a bad mixing on low SNR.

Finally, Tables 3.4 and 3.5 summarize the distribution of the ratios of ESS per second. On average, on EE, the Centered 1 algorithm performs 14 times better than the ASIS and the Centered ones, with a minimum of 0.17 and a maximum of 416. The few multipoles for which the ESS per second is worse than that of the ASIS and Centered cases are the ones corresponding to the highest SNR, where the auxiliary variable step mixes very badly. On BB, the Centered 1 variant performs on average 214 times better than the ASIS and Centered approaches, with a minimum at 5 and a maximum at 1147. Note that the Centered overrelax algorithm performs better than the Centered 1 one on EE but not on BB components. However, it still performs much better on the BB components than the ASIS and Centered variants.

Figure 3.14 shows the empirical mean posterior distribution of Centered overrelax with the two standard deviation interval. The solid black line denotes the true power spectrum.

3.8. Conclusion

We have discussed and compared a number of the Gibbs samplers implemented in the context of the CMB power spectrum estimation.

Two of the studied cases, the centered Gibbs see Eriksen et al. (2008) and non centered Gibbs see Jewell et al. (2004) samplers, have been previously applied to the inference of the power spectrum of the CMB signal. While both the variants have been demonstrated to be feasible, they have been also found to be computationally very demanding. Two main reasons behind it have been identified by Eriksen et al. (2008); Jewell et al. (2004). First, both these algorithms display poor sampling efficiency, with the centered Gibbs

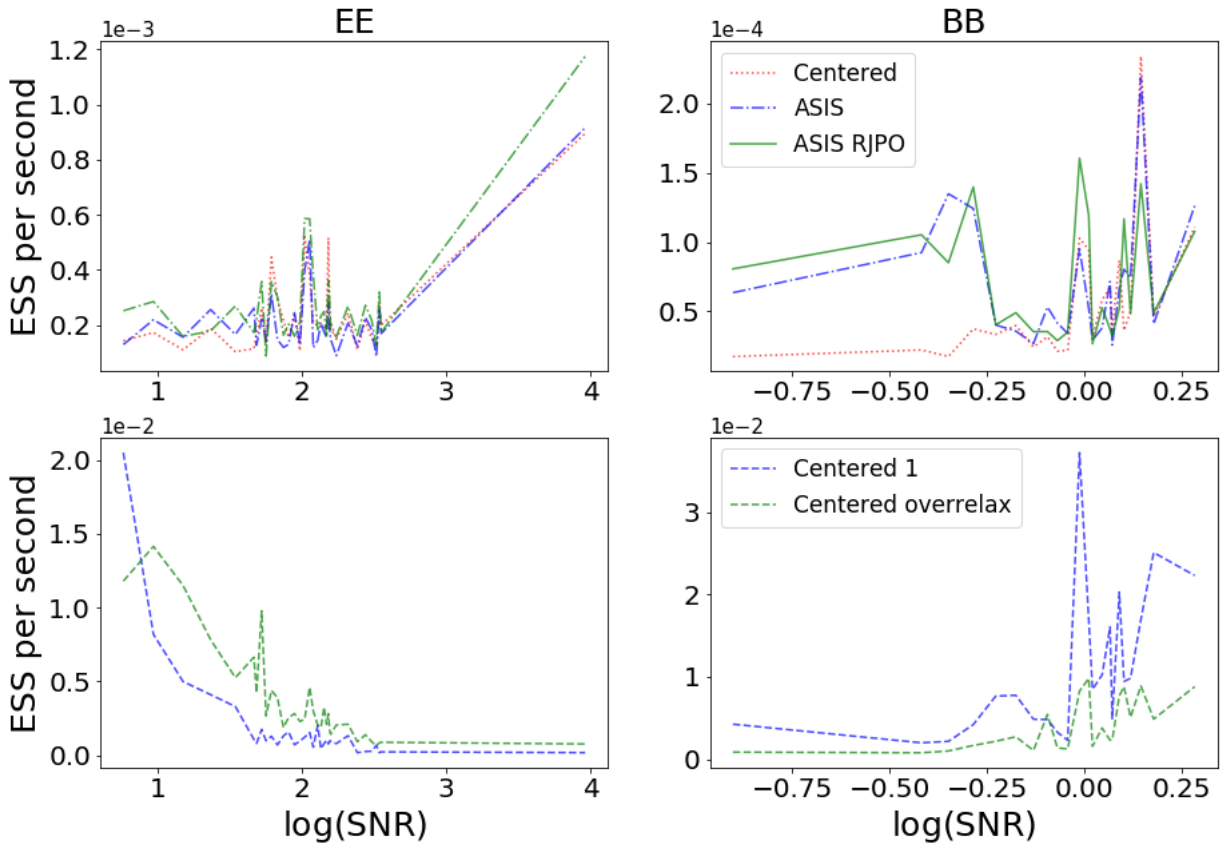

Figure 3.13.: Effective Sample Size per second against $\log(\mathrm{SNR})$ for each algorithm. For the sake of clarity, we plot the results for the Centered 1 case, separately.

Algorithm	5th	25th	50th	75th	95th
ASIS	0.549	0.738	0.991	1.248	1.703
ASIS RJPO	0.633	0.88	1.111	1.349	1.973
Centered 1	1.172	2.819	5.469	10.991	60.119
Centered overrelax	3.854	7.75	12.709	23.646	93.642

Table 3.4.: For each algorithm, percentiles of their ESS per second relative to the ESS per second of the Centered algorithm for EE components.

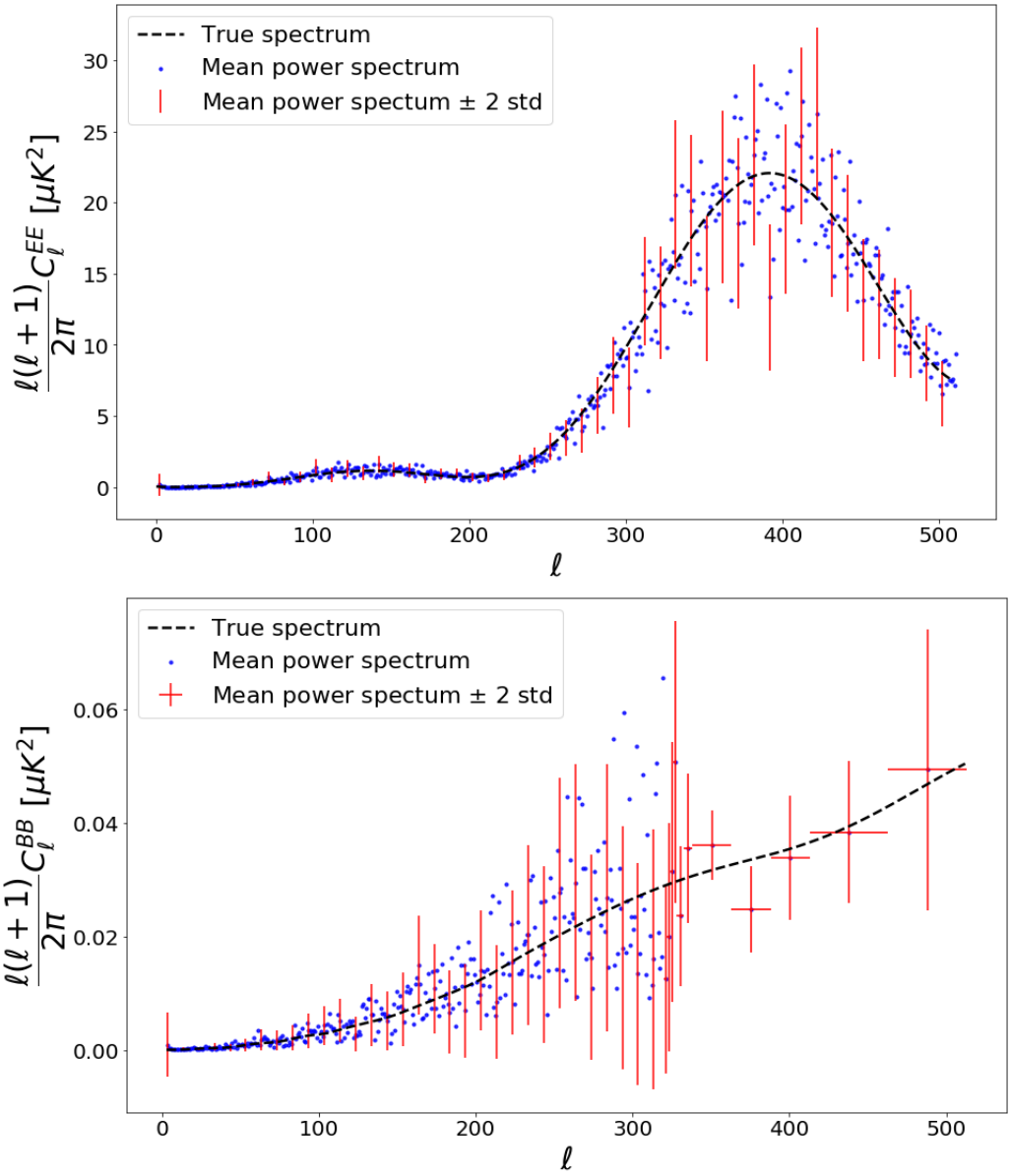

Figure 3.14.: An example of the constraints on the power spectra derived using the Centered overrelax algorithm in the case of the ground-based experiment discussed in the text. The black dashed lines show the true input spectra. The blue points show the best power estimates in each bin equal to the mean power computed over the generated chains. The vertical bars of the crosses correspond to the mean plus/minus one standard deviation, and the horizontal bars the corresponding bins in ℓ. We only plot the crosses for one every ten multipoles on the non binned part. We plot the crosses for every multipole on the binned part. The top panel shows the EE power spectrum and the bottom – the BB one. The input CMB maps assumed the standard cosmological model with the assumed tensor-to-scalar ratio $r = 0.001$.

Algorithm	5th	25th	50th	75th	95th
ASIS	0.548	0.785	1.067	1.497	4.4
ASIS RJPO	0.621	0.982	1.239	1.819	4.922
Centered 1	75.717	129.257	183.94	279.193	516.07
Centered overrelax	28.877	52.064	75.548	104.045	157.7

Table 3.5.: For each algorithm, percentiles of their ESS per second relative to the ESS per second of the Centered algorithm for BB components.

failing on the low SNR components and the non-centered Gibbs on the high SNR components. Second, both these algorithms require significant computations for every sky signal sample due to the need for solving the constrained realization system of equations. We have elaborated on both these factors from the theoretical perspective and demonstrated them via numerical experiments.

We have subsequently proposed a number of possible extensions aiming at improving the overall performance of these two methods.

First, we have looked at improving the sampling efficiency of the standard algorithms. To this end, we have introduced the interweaving concept proposed earlier in the statistical literature, and implemented it in the CMB power spectrum estimation context to improve on the mixing of the over the entire range of signal-to-noise ratio, enabling a more efficient sampling of the entire power spectrum. While potentially promising the improvement comes at the cost of increased computational time per sample.

Second, we have looked at the ways of lowering the cost of single sample computations via statistical means. We have considered two approaches here. Our first proposal, the RJPO algorithm allows for approximate solutions to the constrained realization problem without introducing biases to the final results and does not increase the sample autocorrelation length, if proper tuning is ensured. Our second proposal alleviates the need for solving the constrained realization system altogether by introducing an auxiliary variable. The algorithm is easy to implement and tuning free, but comes at the price of increased dimensionality of the problem.

We have compared and studied all these variants on simulated CMB maps with full and cut-sky coverage.

We have found that for the cases with full and nearly full sky coverage show that the Centered overrelax algorithm performed, on average, an order of magnitude better on EE and two orders of magnitude better on BB in terms of ESS per second than the other algorithms.

This Centered overrelax algorithm It exhibits, however, some drawbacks: for very low or very high signal-to-noise ratio, it produces long autocorrelations. For the very low ratios, it is because of the centered parametrization, while for the very high ones, it is because of the bad mixing of the auxiliary Gibbs sampler step. To solve the first problem, we face a trade-off: we may improve on the mixing by using a non-centered step on the lowest SNR ratio components, but this would increase the cost of the algorithm. To solve the second problem, we would need to find a better mixing algorithm for the constrained realization step, or to find more efficient ways to solve the system.

We note that another MCMC algorithm addressing these problems and seemingly efficient for the entire range of SNR ratios has been developed by Racine et al. (2016). However, to our knowledge, this algorithm has only been used to make inference on the cosmological parameters instead of the power spectrum. In addition, it requires two resolutions of the high dimensional constrained realization system and is thus very costly and requires the tuning of proposal distributions.

3.9. Acknowledgements

We thank Clément Leloup for comments, careful reading of the manuscript, and stimulating discussions. This work was performed in the context of the B3DCMB project and benefited from its inspiring, multidisciplinary environment. We acknowledge the use of `healpy`, see Zonca et al. (2019), `Healpix`, see Gorski et al. (2005), `numpy`, see Harris and et al. (2020) and `maplotlib`, see Hunter (2007). This research was supported by the French National Research Agency (ANR) grant, ANR-B3DCMB, (ANR-17- CE23-0002). JE and RS acknowledge additional support of the ANR-BxB grant (ANR-17-CE31-0022).

Appendix A.

Improper priors

In this appendix we show that in the case of a full sky observation and a noise matrix proportional to identity $\boldsymbol{N} = \alpha \boldsymbol{I}$, using a flat prior over the power spectrum lead to a proper posterior distribution while Jeffrey's prior leads to an improper posterior distribution.

Since we assume full sky coverage and a noise matrix proportional to identity, we can rewrite Model 3.1 in harmonic domain:

$$d = s + n$$

where s is the signal map expressed in the spherical harmonics basis - the vector of $(a_{l,m})_{2 \leqslant l \leqslant \ell_{\max}, 0 \leqslant m \leqslant l}$ coefficients, that is $s \sim \mathcal{N}(0, \boldsymbol{C}(\{C_\ell\}))$. We also now have $n \sim \mathcal{N}(0, \boldsymbol{B}^{-2} \alpha w)$ where $w = \dfrac{4\pi}{N_{\text{pix}}}$, α being the noise matrix in spherical harmonics basis. It follows that d is the observed skymap expressed in harmonic domain too.

In this case, the likelihood straightforwardly writes as:

$$\mathcal{L}(d|\{C_\ell\}) = \prod_{\ell=2}^{\ell_{\max}} \frac{\exp\left(-(1/2)\dfrac{||d_\ell||_2^2}{C_\ell + b_\ell^{-2}\alpha w}\right)}{|C_\ell + b_\ell^{-2}\alpha w|^{(2\ell+1)/2}} \times \mathbf{1}_{\{C_\ell > 0\}}$$

Let us suppose we are using a flat prior on the power spectrum. In this case, we have $\pi(\{C_\ell\}|d) \propto \mathcal{L}(d|\{C_\ell\})$. Then, doing the following change of variable: $y_l = C_\ell + b_l^{-2}\alpha w$ we have:

$$\int_0^\infty \mathcal{L}(d|\{C_\ell\}) dC_2 \dots dC_{\ell_{\max}} \propto \int_0^\infty \prod_{l=2}^{\ell_{\max}} p_\gamma(y_l; \alpha_l, \beta_l) \times \mathbf{1}_{\{y_l > b_l^{-2}\alpha w\}} dy_2 \dots dy_{\ell_{\max}}$$

up to a positive multiplicative constant. Here p_γ means inverse Gamma distribution with parameters $\beta_l = \dfrac{||d_l||_2^2}{2}$ and $\alpha_l = \dfrac{2l-1}{2}$. Since $\mathbf{1}_{\{y_l > b_l^{-2}\alpha w\}} \leqslant \mathbf{1}_{\{y_l > 0\}}$, we have:

$$\int_0^\infty \mathcal{L}(d|\{C_\ell\}) dC_2 \dots dC_{\ell_{\max}} \lesssim \int_0^\infty \prod_{l=2}^{\ell_{\max}} p_\gamma(y_l; \alpha_l, \beta_l) dy_2 \times \dots \times dy_{\ell_{\max}}$$

And the right-hand term of this equation is integrable as the product of independant inverse Gamma densities. Hence, the posterior distribution we obtain with a flat prior is proper.

Now, with Jeffrey's prior $p(\{C_\ell\}) = \prod_{l=2}^{\ell_{\max}} \dfrac{1}{C_\ell}$, things are different:

$$\int_0^1 \mathcal{L}(d|\{C_\ell\}) p(\{C_\ell\}) dC_2 \dots dC_{\ell_{\max}} = \int_0^1 \prod_{l=2}^{\ell_{\max}} \times \frac{\exp\left(-(1/2)\dfrac{||d_l||_2^2}{C_\ell + b_l^{-2}\alpha w}\right)}{|C_\ell + b_l^{-2}\alpha w|^{(2l+1)/2}} \frac{1}{C_\ell} \mathbf{1}_{\{C_\ell > 0\}}.$$

But, on $]0, 1]$ and for any $l \in \{2, \dots, \ell_{\max}\}$ we have:

$$\exp\left(-(1/2)\frac{||d_l||_2^2}{C_\ell + b_l^{-2}\alpha w}\right) \geqslant \exp\left(-(1/2)\frac{||d_l||_2^2}{b_l^{-2}\alpha w}\right)$$

and

$$\frac{1}{|C_\ell + b_l^{-2}\alpha w|^{(2l+1)/2}} \geqslant \frac{1}{|1 + b_l^{-2}\alpha w|^{(2l+1)/2}}$$

Hence we have

$$\int_0^1 \mathcal{L}(d|\{C_\ell\})p(\{C_\ell\})dC_2 \dots dC_{\ell_{\max}} \gtrsim \int_0^1 \prod_{l=2}^{\ell_{\max}} \frac{1}{C_\ell} dC_2 \times \cdots \times dC_{\ell_{\max}}$$

And obviously the right-hand side diverges to infinity. Since the integrand of the left-hand side is positive on $]0, \infty[$, this proves that the posterior distribution is improper if we use Jeffrey's prior on the power spectrum.

Appendix B.

Mixing

In this appendix we provide a intuitive understanding of what we call the "mixing" of a MCMC algorithm. As a toy example, suppose we wish to sample the Gaussian vector (X, Y) with mean zero and covariance matrix:

$$\mathbf{\Sigma} = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

where $\rho \in]-1, 1[$.
Now we set $\rho = 0$ and we use a Gibbs sampler to sample from this distribution. We can plot the trajectory of the algorithm, see Figure B.1.

We can also suppose that $\rho = 0.99$, in which case we get another trace plot, see Figure B.2.

We plot the autocorrelations of the Gibbs sampler for X and Y in Figure B.3.

We can see on Figures B.1 and B.2 that the Gibbs sampler with $\rho = 0$ explores the target distribution much more efficiently than when $\rho = 0.99$. We can also see on Figure B.3 that the autocorrelations are much longer when $\rho = 0.99$ than when $\rho = 0$. More precisely, the Gibbs sampler samples independently when $\rho = 0$ while when $\rho = 0.99$, the sampled points are still correlated after 150 steps. When an algorithm explores the target distribution and shows low autocorrelations like the Gibbs sampler when $\rho = 0$, we say it is mixing well. On the contrary, when an algorithm behaves like the Gibbs sampler when $\rho = 0.99$, we say it is mixing badly. Here, the term "mixing" does not have a precise definition and we use it loosely.

Even though we use the term "mixing" loosely, we can still characterize the convergence of a Markov chain with state space $\mathcal{X}$, invariant distribution π and transition kernel P. We usually want our Markov chain to converge geometrically to the invariant distribution π, that is:

$$||P^n(x, \mathrm{d}y) - \pi(\mathrm{d}y)||_{\mathrm{TV}} \leqslant Cr^n \tag{B.1}$$

for any $x \in \mathcal{X}$, where $C > 0$ and $r \in [0, 1)$ are constants. The constant r is called the geometric rate of convergence.

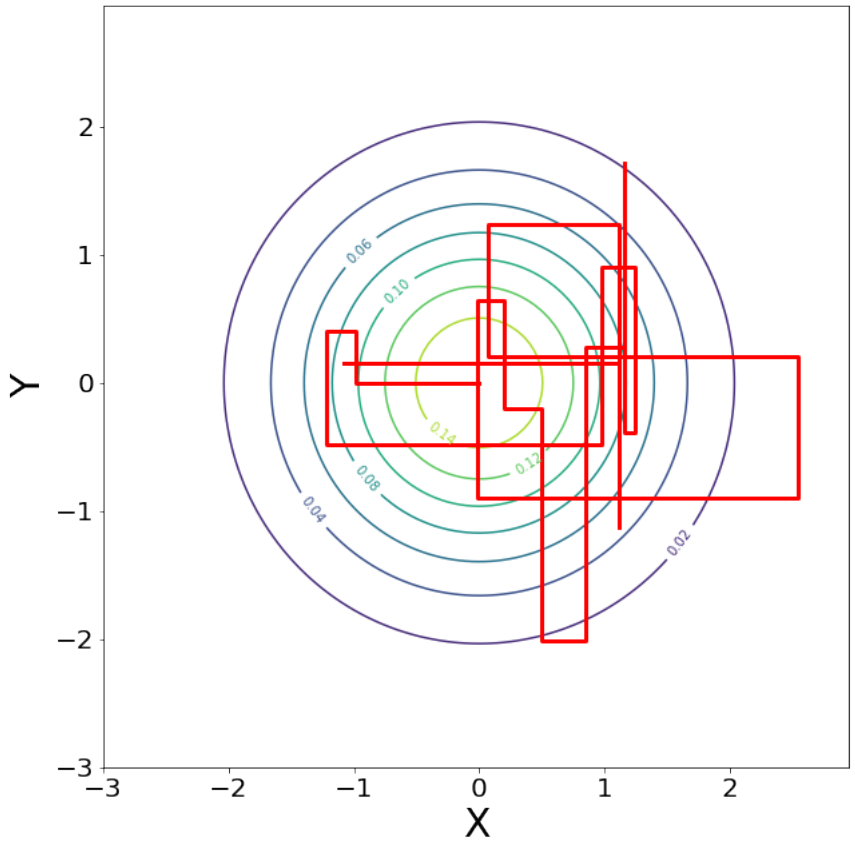

Figure B.1.: Trace plot, in red, of the Gibbs sampler targeting the joint distribution of (X, Y) for $\rho = 0$, described in Appendix B. The circles are the level sets of the normal distribution.

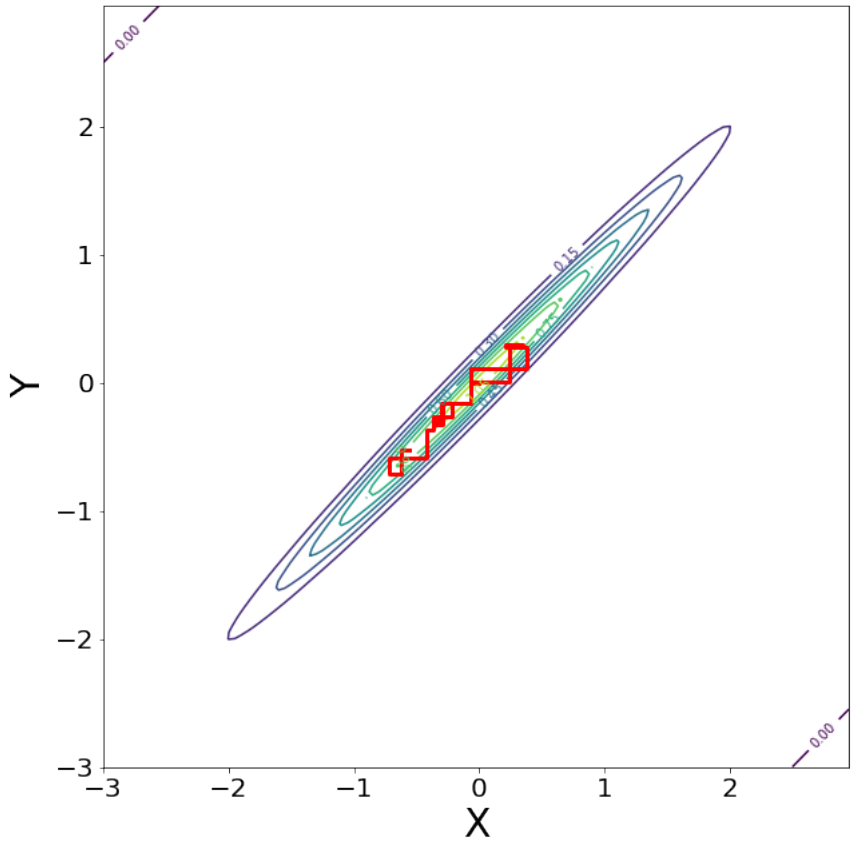

Figure B.2.: Trace plot, in red, of the Gibbs sampler targeting the joint distribution of (X, Y) for $\rho = 0.99$, described in Appendix B. The circles are the level sets of the normal distribution.

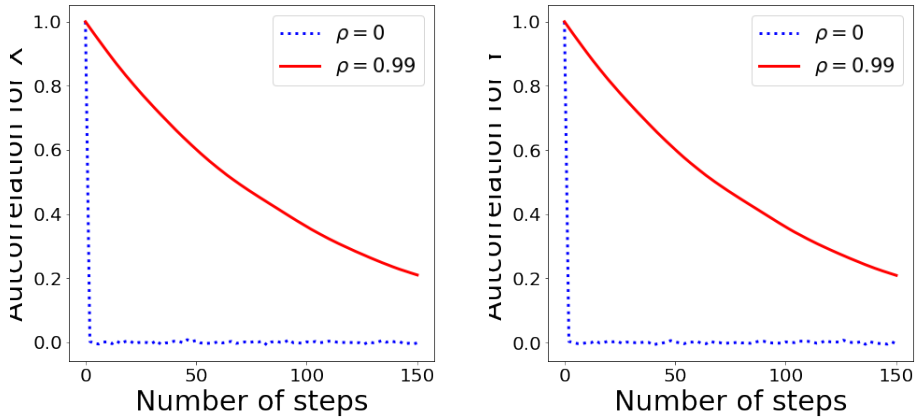

Figure B.3.: Autocorrelation plots of the Gibbs sampler, described in Appendix B, for X and Y and for $\rho = 0$ and $\rho = 0.99$.

Appendix C.

Experiments

C.1. Full-sky polarization experiment

In this appendix we show histograms and autocorrelation plots that we obtained running the full-sky experiment described 3.7.1 on Figures C.1 to C.4. All these figures confirm our analysis of Section 3.5 and the results of Section 3.7.1: the interweaving algorithm performs as good as the centered Gibbs on high SNR components and as good as the non-centered Gibbs on low SNR components. Note also that the kernel density estimation of the histograms of interweaving matches almost perfectly the true posterior marginals for any signal-to-noise ratio.

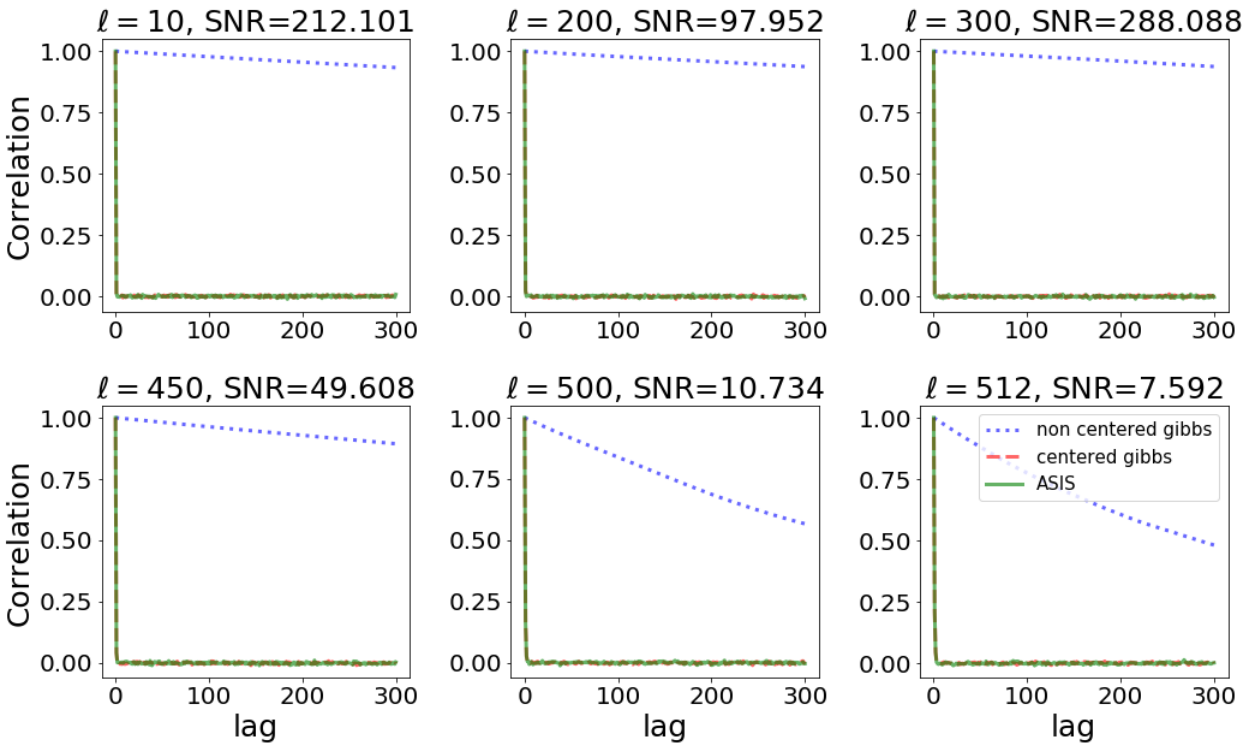

Figure C.1.: Examples of autocorrelations for EE components for full sky experiment, Section 3.7.1.

C.2. Sky masks

In this appendix we provide plots of the two sky maps used in the experiment section.

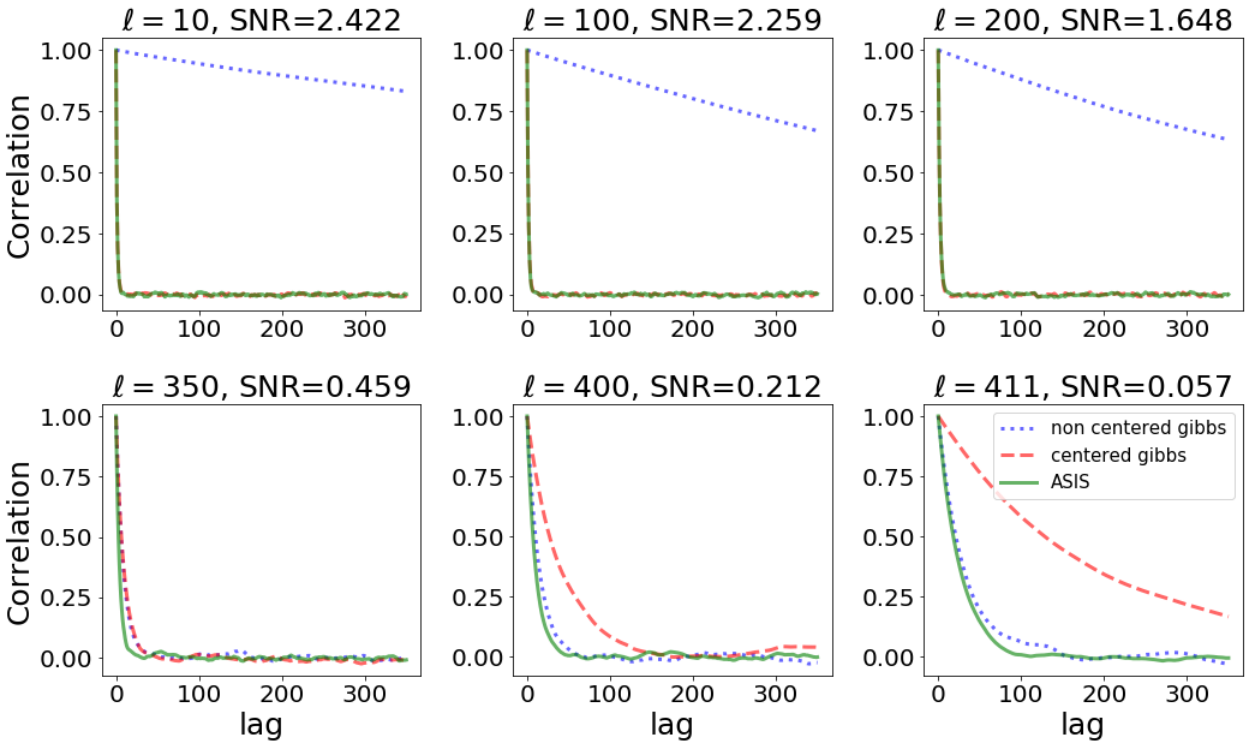

Figure C.2.: Examples of autocorrelations for BB components for full sky experiment, Section 3.7.1.

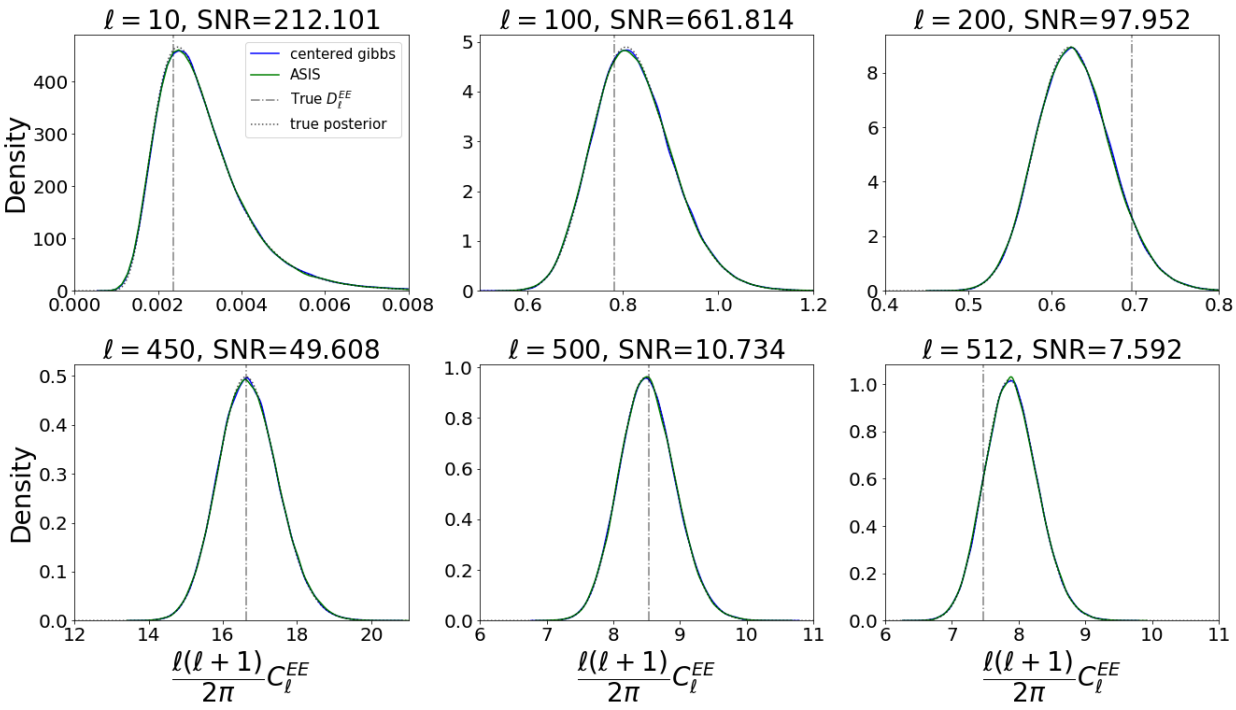

Figure C.3.: Examples of kernel density estimation of histograms for EE components for full sky experiment, Section 3.7.1. For readability and since the mixing of the non centered Gibbs is bad, we don't include its histograms on this figures.

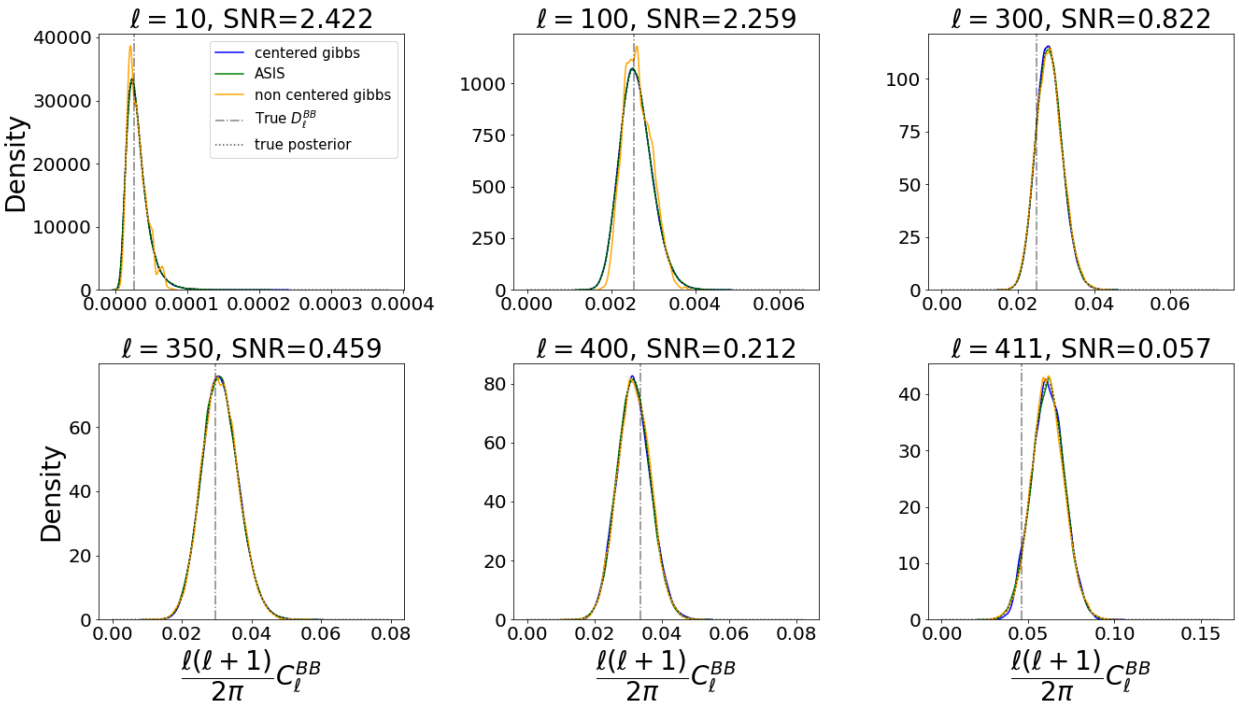

Figure C.4.: Examples of kernel density estimation of histograms for BB components for full sky experiment, Section 3.7.1. For readability and since the mixing of the non centered Gibbs is bad, we don't include its histograms on this figures.

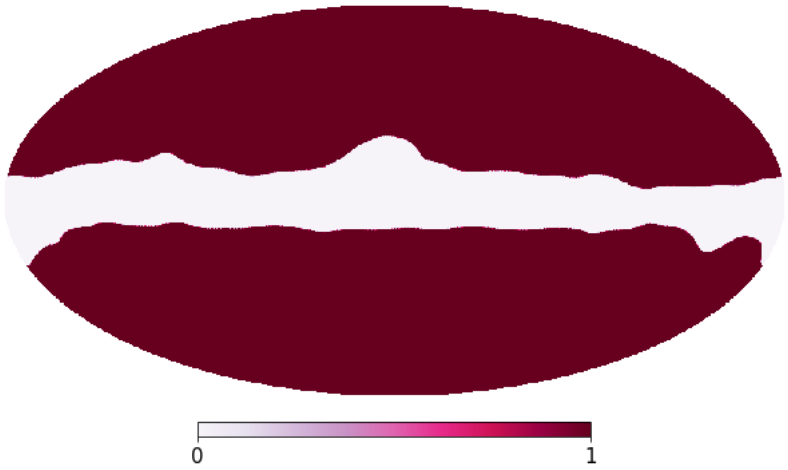

Figure C.5.: Planck sky mask used for the first cut sky experiment described in Section 3.7.2. This mask covers roughly 80% of the sky.

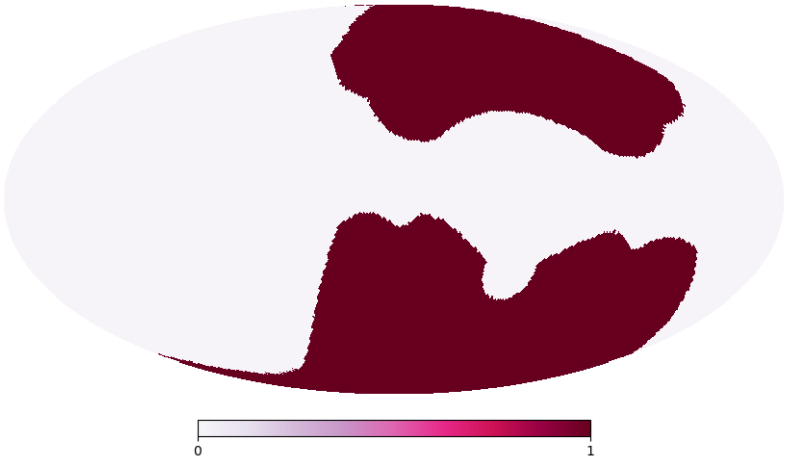

Figure C.6.: Simon sky mask used for the second cut sky experiment described in Section 3.7.3. This mask covers roughly 35% of the sky.

C.3. A first cut-sky polarization experiment

This appendix provides kernel density estimation based on the histograms of the histograms used in Section 3.7.2. See Figures C.7 to C.10.

C.4. A second cut-sky polarization experiment

This appendix provides kernel density estimation based on the histograms obtained in Section 3.7.3. See Figures C.11 and C.12.

Note that for the lowest SNR on BB components, Centered gives an irrelevant estimate of the posterior density while Centered 1 gives a result in agreement with ASIS and ASIS RJPO: even though Centered 1 suffers from the centered parametrization, thanks to its low computational cost, we are able to perform enough iterations to have a reliable estimate. Which is not the case of Centered because of its high computational cost.

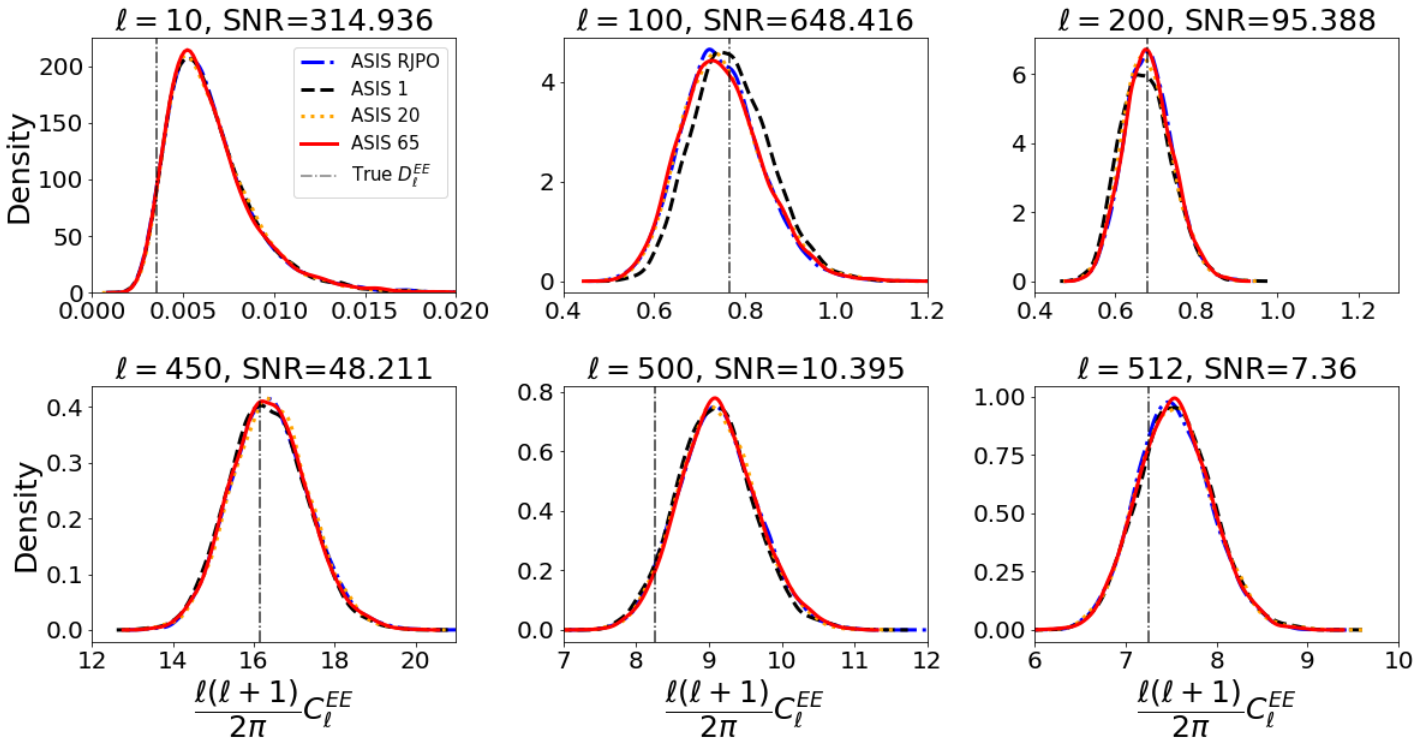

Figure C.7.: Kernel density estimation of marginals for a sample of multipoles for EE components for cut-sky experiment, Section 3.7.2.

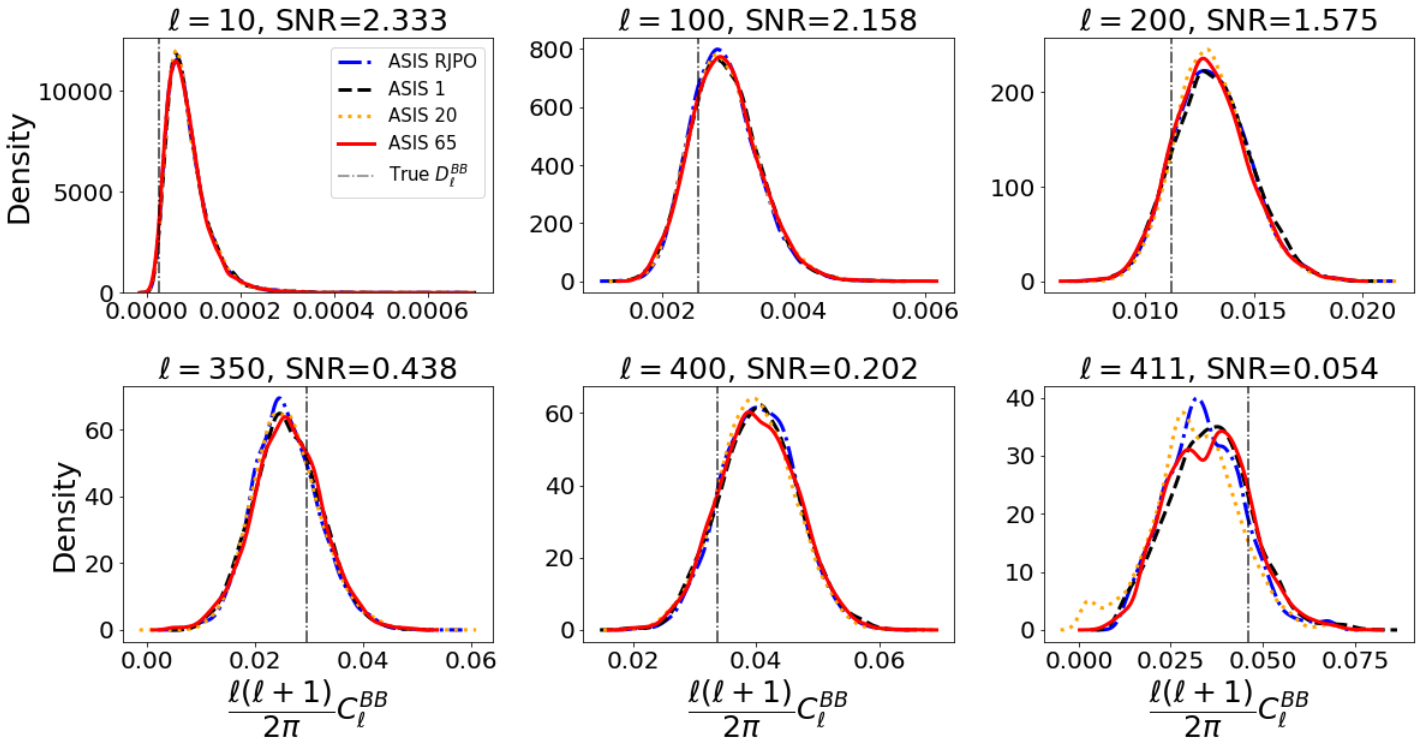

Figure C.8.: Kernel density estimation of marginals for a sample of multipoles for BB components for cut-sky experiment, Section 3.7.2.

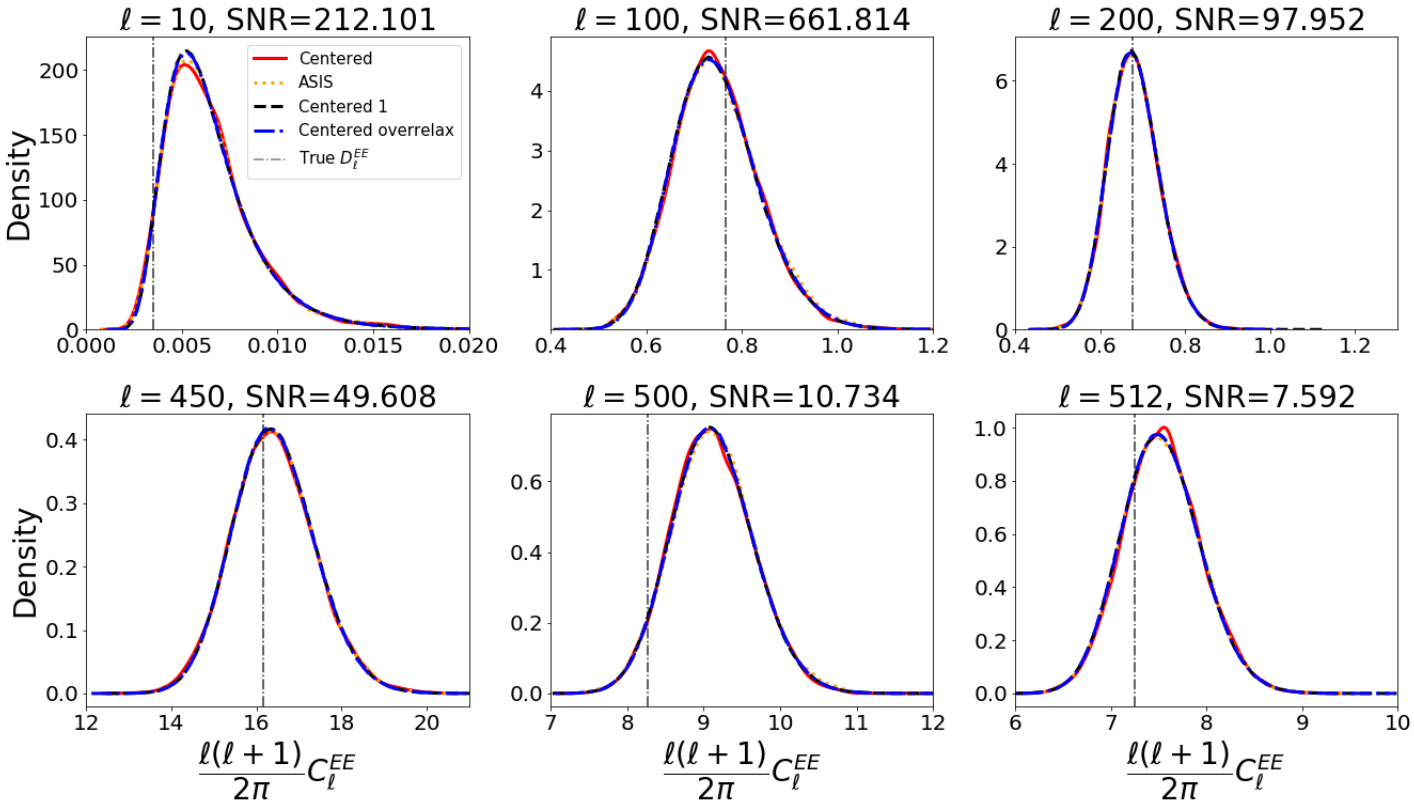

Figure C.9.: Kernel density estimation of marginals for a sample of multipoles for EE components for cut-sky experiment, Section 3.7.2.

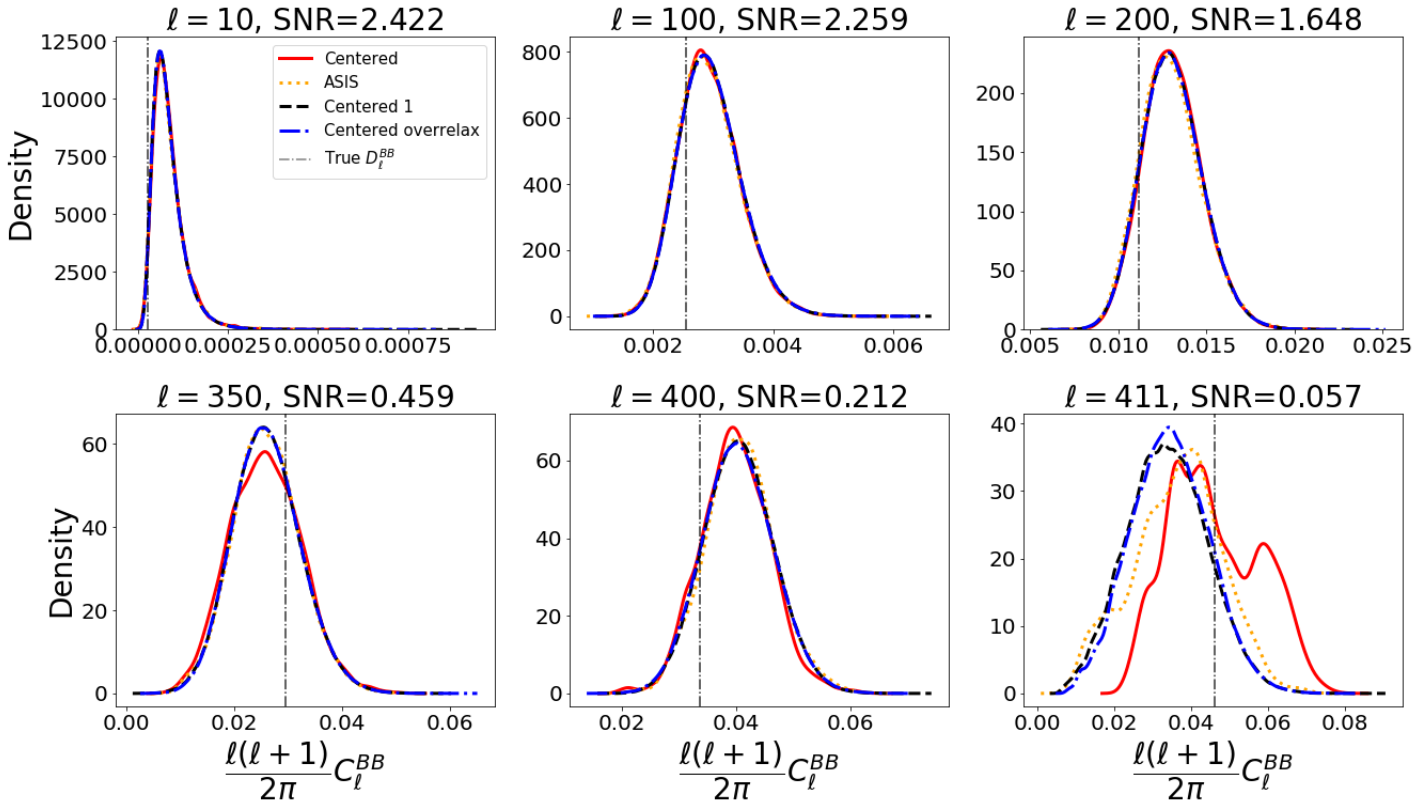

Figure C.10.: Kernel density estimation of marginals for a sample of multipoles for EE components cut-sky experiment, Section 3.7.2.

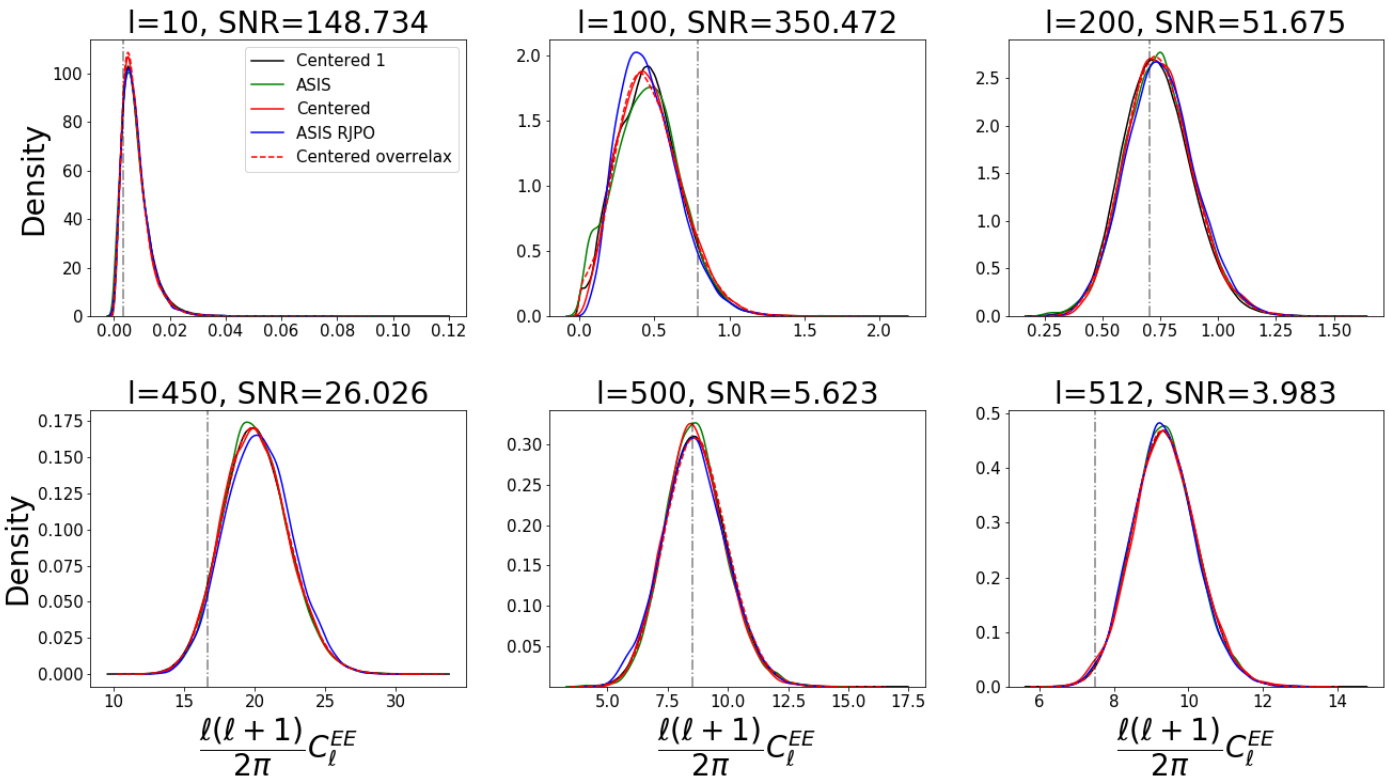

Figure C.11.: Kernel density estimation of the posterior density on EE cut-sky experiment, Section 3.7.3.

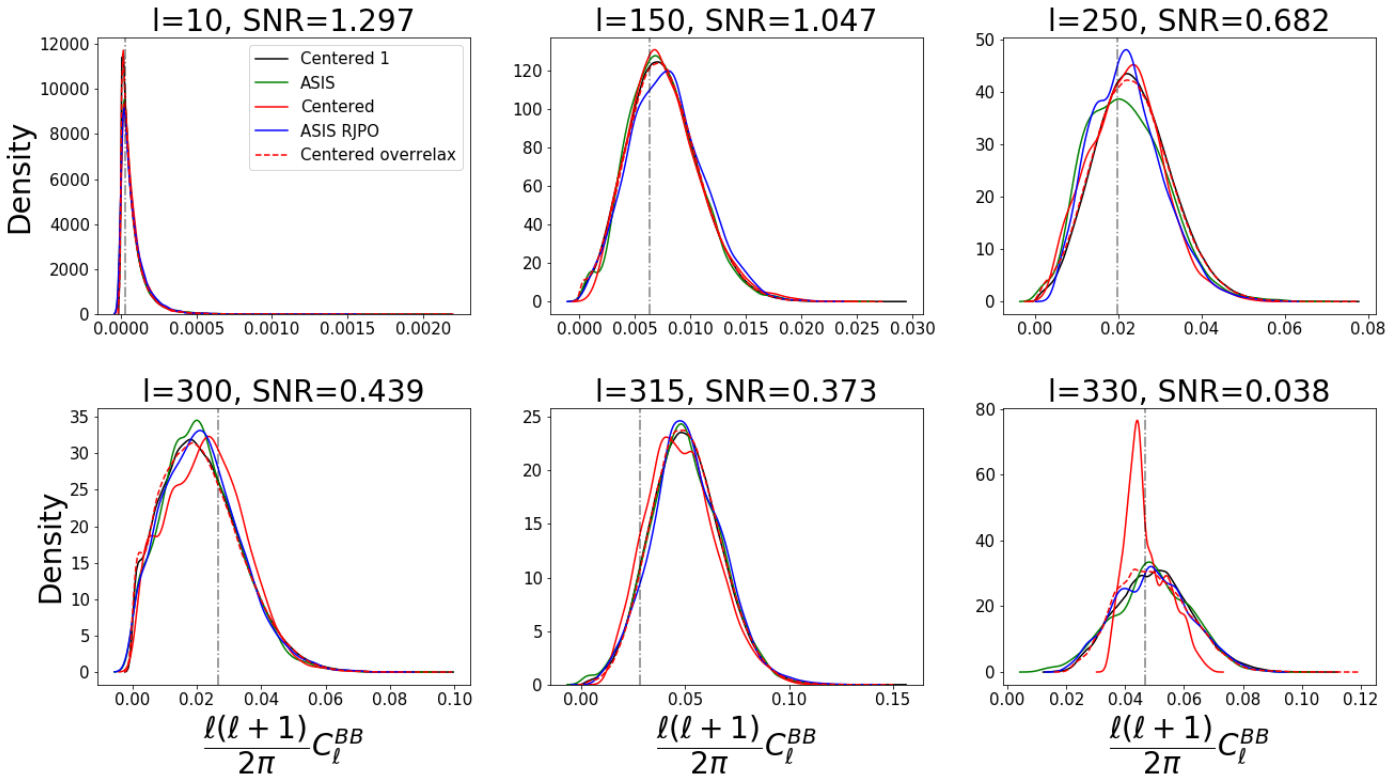

Figure C.12.: Kernel density estimation of the posterior density on BB for cut-sky experiment, Section 3.7.3.

Chapter 4.

Fast compression of MCMC output

Joint work with Nicolas Chopin, appeared in *Entropy* 2021, 23(8), 1017.

We propose cube thinning, a novel method for compressing the output of a MCMC (Markov chain Monte Carlo) algorithm when control variates are available. It amounts to resampling the initial MCMC sample (according to weights derived from control variates), while imposing equality constraints on averages of these control variates, using the cube method of Deville (2004). Its main advantage is that its CPU cost is linear in N, the original sample size, and is constant in M, the required size for the compressed sample. This compares favourably to Stein thinning (Riabiz et al., 2020), which has complexity $\mathcal{O}(NM^2)$, and which requires the availability of the gradient of the target log-density (which automatically implies the availability of control variates). Our numerical experiments suggest that cube thinning is also competitive in terms of statistical error.

4.1. Introduction

MCMC (Markov chain Monte Carlo) remains to this day the most popular approach to sampling from a target distribution p, in particular in Bayesian computation (Robert and Casella, 2004).

Standard practice is to run a single chain, $X_1, \ldots, X_N$ according to a Markov kernel that leaves invariant p. It is also common to discard part of the simulated chain, either to reduce its memory footprint, or to reduce the CPU cost of later post-processing operations, or more generally for the user's convenience. Historically, the two common recipes for compressing MCMC output are:

- burn-in, which amounts to discarding the b first states; and
- thinning, which amounts to retaining only one out of t (post burn-in) states.

The impact of either recipes on the statistical properties of the sub-sampled estimates are markedly different. Burn-in reduces the bias introduced by the discrepancy between p and the distribution of the initial state X_1 (since $X_b \approx p$ for b large enough). On the other hand, thinning always increases the (asymptotic) variance of MCMC estimates (Geyer, 1992).

Practitioners often choose b (the burn-in period) and t (the thinning frequency) separately, in a somewhat ad-hoc fashion (i.e. through visual inspection of the initial chain), or using convergence diagnosis such as e.g. those reviewed in Cowles and Carlin (1996).

Two recent papers (Mak and Joseph, 2018; Riabiz et al., 2020), cast a new light on the problem of compressing a MCMC chain by considering more generally the problem, for a given M, of selecting the subsample of size M that best represents (according to a certain criterion) the target distribution p. We focus for now on Riabiz et al. (2020), for reasons we explain below.

Stein thinning, the method developed in Riabiz et al. (2020), chooses the sub-sample $\mathcal{S}$ of size M which minimises the following criterion:

$$D(\mathcal{S}) := \frac{1}{M^2} \sum_{m,n \in \mathcal{S}} k_p(X_m, X_n), \quad \mathcal{S} \subset \{1, \ldots, N\}, \quad |\mathcal{S}| = M \tag{4.1}$$

where k_p is a $p-$dependent kernel function derived from another kernel function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, as follows:

$$k_p(x, y) = \nabla_x \cdot \nabla_y k(x, y) + \langle \nabla_x k(x, y), s_p(y) \rangle + \langle \nabla_y k(x, y), s_p(x) \rangle + k(x, y) \langle s_p(x), s_p(y) \rangle$$

with $\langle \cdot, \cdot \rangle$ being the Euclidean inner product, $s_p(x) := \nabla \log p(x)$ is the so-called score function (gradient of the log target density), and ∇ the gradient operator.

The rationale behind criterion (4.1) is that it may be interpreted as the KSD (kernel Stein divergence) between the true distribution p and the empirical distribution of sub-sample S. We refer to Riabiz et al. (2020) for more details on the theoretical background of the KSD, and its connection to Stein's method.

Stein thinning is appealing, as it seems to offer a principled, quasi-automatic way to compress MCMC output. However, closer inspection reveals the following three limitations.

First, it requires computing the gradient of the log-target density, $s_p(x) = \nabla \log p(x)$. This restricts the method to problems where this gradient exists and is tractable (and, in particular, to $\mathcal{X} = \mathbb{R}^d$).

Second, its CPU cost is $\mathcal{O}(NM^2)$. This makes it nearly impossible to use Stein thinning for $M \gg 100$. This cost stems from the greedy algorithm proposed in Riabiz et al. (2020), see their Algorithm 1, which adds at iteration t the state X_i which minimises $k_p(X_i, X_i) + \sum_{j \in S_{t-1}} k_p(X_i, X_j)$, where S_{t-1} is the sample obtained from the $t-1$ previous iterations.

Third, its performance seems to depend in a non-trivial way on the original kernel function k; Riabiz et al. (2020) propose several strategies for choosing and scaling k, but none of them seems to perform uniformly well in their numerical experiments.

We propose a different approach in this paper, which we call cube thinning, and which addresses these shortcomings to some extent. Assuming the availability of J control variates (that is, of functions h_j with known expectation under p), we cast the problem of MCMC compression as that of resampling the initial chain under constraints based on these control variates. The main advantage of cube thinning is that its complexity is $\mathcal{O}(NJ^3)$; in particular it does not depend on M. That makes it possible to use it for much larger values of M. (We shall discuss the choice of J, but, by and large, J should be of the same order as d, the dimension of the sampling space). The name stems from the cube method of Deville (2004), which plays a central part in our approach, as we explain in the body of the chapter.

The availability of control variates may seem like a strong requirement. However, if we assume we are able to compute $s_p(x) = \nabla \log p(x)$, then (for a large class of functions $\phi : \mathbb{R}^d \to \mathbb{R}^d$, which we define later)

$$\mathbb{E}_p \left[\phi(x) s_p(x) + \nabla_x \cdot \phi(x) \right] = 0$$

where $\nabla_x \cdot \phi$ denotes the divergence of ϕ. In other words, the availability of the score function implies automatically the availability of control variates. The converse is not true: there exists control variates (e.g. Dellaportas and Kontoyiannis, 2011) that are not gradient-based. One of the examples we consider in our numerical examples features such non gradient-based control variates; as a result, we are able to apply cube thinning, although Stein thinning is not applicable.

The support point methods of Mak and Joseph (2018) does not require control variates. It is thus more generally applicable than either cube thinning or Stein thinning. On the other hand, when gradients (and thus control variates) are available, the numerical experiments of Riabiz et al. (2020) suggest that Stein thinning outperforms support points. From now on, we focus on situations where control variates are available.

The chapter is organised as follows. Section 4.2.3 recalls the concept of control variates, and explains how control variates may be used to reweight a MCMC sample. Section 4.3 describes the cube method of Deville (2004). Section 4.4 explains how to combine control variates and the cube method to perform cube thinning. Section 4.5 assesses the statistical performance of cube thinning through two numerical experiments.

We use the following notations throughout: p denotes both the target distribution and its probability density; $p(f)$ is a short-hand for the expectation of $f(X)$ under p. The gradient of a function f is denoted by $\nabla_x f(x)$, or simply $\nabla f(x)$ when there is no ambiguity. The $i-$th component of a vector $v \in \mathbb{R}^d$ is denoted

by $v[i]$, and its transpose by v^t. The vectors of the canonical basis of $\mathbb{R}^d$ are denoted by e_i, i.e. $e_i[j] = 1$ if $j = i$, 0 otherwise. Matrices are written in upper-case; the kernel (null space) of matrix A is denoted by $\ker A$. The set of functions $f : \Omega \to \mathbb{R}^d$ that are continuously differentiable is denoted by $C^1(\omega, \mathbb{R}^d)$.

4.2. Control variates

4.2.1. Definition

Control variates are a very well known way to reduce the variance of Monte Carlo estimates; see e.g. the books of Robert and Casella (2004), Glasserman (2004) and Owen (2013).

Suppose we want to estimate the quantity $p(f) = \mathbb{E}_p[f(X)]$ for a suitable $f : \mathbb{R}^d \to \mathbb{R}$, based on an IID (independent and identically distributed) sample $\{X_1, \dots, X_N\}$ from distribution p. (The generalisation of control variates to MCMC will be discussed in Section 4.4.)

The usual Monte Carlo estimate of $p(f)$ is

$$\hat{p}(f) = \frac{1}{N} \sum_{n=1}^{N} f(X_n).$$

Assume we know $J \in \mathbb{N}^\star$ functions $h_j : \mathbb{R}^d \to \mathbb{R}$ for $j \in \{1, \dots, J\}$ such that $p(h_j) = 0$. Functions with this property are called control variates. We can use this property to build an estimate with a lower variance: let's denote $h(X) = (h_1(X), \dots, h_J(X))^t$ and write our new estimate:

$$\hat{p}_\beta(f) = \frac{1}{N} \sum_{n=1}^{N} f(X_n) + \beta^t h(X_n) \tag{4.2}$$

with $\beta \in \mathbb{R}^J$. Then it is straightforward to show that $\mathbb{E}[\hat{p}_\beta(f)] = \mathbb{E}[\hat{p}(f)] = p(f)$. Depending on the choice of β we may have $\mathrm{Var}[\hat{p}_\beta(f)] \leqslant \mathrm{Var}[\hat{p}(f)]$. The next section discusses how to choose such a β.

4.2.2. Control variates as a weighting scheme

The standard approach to choose β consists of two steps. First, one shows easily that the value the minimises the variance of estimator (4.2) is:

$$\beta^\star(f) = \mathrm{Var}(h(X))^{-1} \mathrm{Cov}(h(X), f(X))$$

where $\mathrm{Var}(h(X))$ is the $J \times J$ variance matrix of the vector $h(X)$ and $\mathrm{Cov}(h(X), f(X))$ is the $J \times 1$ vector such that $\mathrm{Cov}(h(X), f(X))_{i,1} = \mathrm{Cov}(f(X), h_i(X))$.

Second, one realises that this quantity may be estimated from the sample $X_1, \dots, X_N$ through a simple linear regression model, where the $f(X_n)$'s are the outcome, and the $h_j(X_n)$'s are the predictors:

$$f(X_n) \approx \mu + \beta^t h(X_n) + \epsilon_n, \quad \mathbb{E}[\epsilon_n] = 0.$$

More precisely, let $\gamma \in R^{J+1}$ be the vector such that $\gamma^t = (\mu, \beta^t)$, $H = (H_{ij})$ the design matrix such that $H_{i1} = 1$, $H_{i(j+1)} = h_j(X_i)$, and $F = (f(X_1), \dots, f(X_N))$. Then the OLS (ordinary least squares) estimate of γ is

$$\hat{\gamma}_{\mathrm{OLS}} = (H^t H)^{-1} H^t F. \tag{4.3}$$

Since $\mathbb{E}[f(X_n)] = \mu$ in this artificial regression model, the first component of $\hat{\gamma}_{\mathrm{OLS}}$:

$$\hat{p}_\star(f) := \hat{\gamma}_{\mathrm{OLS}} \times e_1, \tag{4.4}$$

actually corresponds to estimate (4.2) when $\beta = \widehat{\beta}_{\mathrm{OLS}}$.

At first glance, the approach described above seems to require implementing a different linear regression for each function f of interest. Owen (2013) noted however that one may re-express (4.4) as a weighted average:

$$\hat{p}_\star(f) = \sum_{n=1}^{N} w_n f(X_n)$$

where the weights w_n sum to one, and do not depend on f. It is thus possible to compute these weights once from a given sample (given a certain choice of control variates), and then quickly compute $\hat{p}_\star(f)$ for any function f of interest.

The exact expression of the weights are easily deduced from (4.4) and (4.3): $w = (w_n)$ with

$$w = H(H^t H)^{-1} e_1.$$

4.2.3. Gradient-based control variates

In this section and the next, we recall generic methods to construct control variates. This section considers specifically control variates that derive from the score function, $s_p(x) = \nabla \log p(x)$. (We therefore assume that this quantity is tractable.)

Under the following two conditions:

1. the probability density $p \in C^1(\Omega,\ \mathbb{R})$ where $\Omega \subseteq \mathbb{R}^d$ is an open set;
2. Function $\phi \in C^1(\Omega, \mathbb{R}^d)$ is such that $\oint_{\partial\Omega} p(x)\phi(x) \cdot n(x) S(dx) = 0$ where $\oint_{\partial\Omega}$ denotes the integral over the boundary of Ω, and $S(dx)$ is the surface element at $x \in \partial\Omega$;

the following function:

$$h(x) = \nabla_x \cdot \phi(x) + \phi(x) \cdot s_p(x)$$

is a control variate: $p(h) = 0$, see e.g. Mira et al. (2013) or Oates et al. (2016) for further details. To get some intuition, note that in dimension 1 and assuming the domain of integration is an interval $]a, b[\subset \mathbb{R}$, this amounts to an integration by part with the condition that $h(b)p(b) - h(a)p(a) = 0$.

Thus, whenever the score function is available (and the conditions above hold), we are able to construct an infinite number of control variates (one for each function ϕ). For simplicity, we shall focus on the following standard classes of such functions. First, for $i = 1, \ldots, d$,

$$\begin{aligned} \phi_i \colon \mathbb{R}^d &\to \mathbb{R}^d \\ x &\mapsto e_i \end{aligned}$$

which leads to the following d control variates:

$$h_i(x) = s_p(x)[i]. \tag{4.5}$$

For a Gaussian target, $N(\mu, \Sigma)$, the score is $s_p(x) = -\Sigma^{-1}(x - \mu)$, and the control variates above make it possible to reweigh the Monte Carlo sample to make it have the same expectation as the target distribution.

Second, we consider, for $i, j = 1, \ldots, d$:

$$\begin{aligned} \phi_{ij} \colon \mathbb{R}^d &\to \mathbb{R}^d \\ x &\mapsto x[i] e_j \end{aligned}$$

which leads to the following d^2 control variates:

$$h_{ij}(x) = \mathbb{1}\{i = j\} + x[i] s_p(x)[j]. \tag{4.6}$$

Again, for a Gaussian target $N(\mu, \Sigma)$, this makes it possible to fix the empirical covariance matrix to true covariance Σ.

In our simulations, we consider two sets of control variates: the 'full' set, consisting of the d control variates defined by (4.5), and the d^2 control variates defined by (4.6). And a 'diagonal' set of $2d$ control variates, where for (4.6), we only consider the cases where $i = j$. Of course, the former set should lead to better performance (lower variance), but since the complexity of our approach will be $\mathcal{O}(J^3)$, where J is the number of control variates, taking $J = \mathcal{O}(d^2)$ may be too expensive whenever the dimension d is large.

4.2.4. MCMC-based control variates

We mention in passing other ways to construct control variates, in particular in the context of MCMC.

For instance, Dellaportas and Kontoyiannis (2011) noted that, for a Markov chain $\{X_n\}$, the quantity

$$\phi(X_n) - \mathbb{E}\left[\phi(X_n)|X_{n=1}\right]$$

has expectation zero. In particular, if the MCMC kernel is a Gibbs sampler, it is likely that one is able to compute the conditional expectation of each component; i.e. $\phi(x) = x[i]$ for $i = 1, \ldots, d$.

See also Hammer and Tjelmeland (2008) for another way to construct control variates when the X_n's are simulated from a Metropolis kernel.

4.3. The cube method

We review in this section the cube method of Deville (2004). This method originated from survey sampling, and is a way to sample from a finite population under constraints. The first subsection gives some definitions, the second one explains the flight phase of the cube method and the third subsection discusses the landing phase of the method.

4.3.1. Definitions

Suppose we have a finite population $\{1, \ldots, N\}$ of N individuals and that to each individual $n = 1, \ldots, N$ is associated a variable of interest y_n and J auxiliary variables, $v_n = (v_{n1}, \ldots, v_{nJ})$. Without loss of generality, suppose also that the J vectors $(v_{1j}, \ldots, v_{Nj})$ are linearly independent. We are interested in estimating the quantity $Y = \sum_{n=1}^N y_n$ using a subsample of $\{1, \ldots, N\}$. Furthermore, we know the exact value of each sum $V_j = \sum_{n=1}^N v_{nj}$, and we wish to use this auxiliary information to better estimate Y.

We assign, to each individual n, a sampling probability $\pi_n \in [0, 1]$. We consider binary random variables S_n such that, marginally, $\mathbb{P}(S_n = 1) = \pi_n$. We may then define the Horvitz-Thompson estimator of Y:

$$\hat{Y} = \sum_{n=1}^N \frac{S_n y_n}{\pi_n}$$

which is unbiased, and which depends only on selected individuals (i.e $S_n = 1$).

We define similarly the Horvitz-Thompson estimator of V_j:

$$\hat{V}_j = \sum_{n=1}^N \frac{S_n v_{nj}}{\pi_n}.$$

Our objective is to construct a joint distribution ξ for the inclusion variables S_n such that $\mathbb{P}_\xi(S_n = 1) = \pi_n$ for all $n = 1, \ldots, N$, and

$$\hat{V} = V \quad \xi\text{-almost surely.} \tag{4.7}$$

where $V = (V_1, \ldots, V_J)$, $\hat{V} = (\hat{V}_1, \ldots, \hat{V}_J)$. Such a probability distribution is called a balanced sampling design.

4.3.2. Subsamples as vertices

We can view all the possible samples from $\{1, \ldots, N\}$ as the vertices of the hypercube $\mathcal{C} = [0,1]^N$ in $\mathbb{R}^N$. A sampling design with inclusion probabilities $\pi_n = \mathbb{P}_\xi(S_n = 1)$ is then a distribution over the set of these vertices such that $\mathbb{E}[S] = \pi$, where $S = (S_1, \ldots, S_N)^t$, and $\pi = (\pi_1, \ldots, \pi_N)^t$ is the vector of inclusion probabilities. Hence, π is expressed as a convex combination of the vertices of the hypercube.

We can think of a sampling algorithm as finding a way to reach any vertex of the cube, starting at π, while satisfying the balancing equation (4.7). But before we describe such a sampling algorithm, we may wonder if it is possible to find a vertex such that (4.7) is satisfied.

4.3.3. Existence of a solution

The balancing equation (4.7) defines a linear system. Indeed, we can re-express (4.7) as S being a solution to $As = V$, where $A = (A_{jn})$ is of dimension $J \times N$, $A_{jn} = v_{kn}/\pi_n$. This system defines a hyperplane Q of dimension $N - J$ in $\mathbb{R}^N$.

What we want is to find vertices of the hypercube $\mathcal{C}$ that also belong to the hyperplane Q. Unfortunately, it is not necessarily possible, as it depends on how the hyperplane Q intersects the cube $\mathcal{C}$. In addition, there is no way to know beforehand if such a vertex exists. Since $\pi \in Q$, we know that $\mathcal{K} := \mathcal{C} \cap Q \neq \varnothing$ and is of dimension $N - J$. The only thing we can say is stated Proposition 1 in Deville (2004): if r is a vertex of $\mathcal{K}$, then in general $q = \mathrm{card}(\{n : 0 < r[n] < 1\}) \leqslant J$.

The next section describes the flight phase of the cube algorithm, which generates a vertex in $\mathcal{K}$ when such vertices exist, or which, alternatively, returns a point in $\mathcal{K}$ with most (but not all) components set to zero or one. In the latter case, one needs to implement a landing phase, which is discussed in Section 4.3.5.

4.3.4. Flight phase

The flight phases simulates a process $\pi(t)$ which takes values in $\mathcal{K} = \mathcal{C} \cap Q$, and starts at $\pi(0) = \pi$. At every time t, one selects a unit vector $u(t)$, then one chooses randomly between one of the two points that are in the intersection of the hyper-cube $\mathcal{C}$ and the line parallel to $u(t)$ that passes through $\pi(t-1)$. The probability of selecting these two points are set to ensure that $\pi(t)$ is a martingale; in that way, we have $\mathbb{E}[\pi_t] = \pi$ at every time step. The random direction $u(t)$ must be generated to fulfil the following two requirements: (a) that the two points are in Q; i.e. $u(t) \in \ker A$; and (b) whenever $\pi(t)$ has reached one of the faces of the hyper-cube, it must stay within that face; thus, $u(t)[k] = 0$ if $\pi(t-1)[k] = 0$ or 1.

Algorithm 11 describes one step of the flight phase.

Algorithm 11: Flight phase iteration

Input: $\pi(t-1)$
Output: $\pi(t)$

1 Sample $u(t)$ in $\ker A$ with $u_k(t) = 0$ if the k-th component of $\pi(t-1)$ is an integer.
2 Compute $\lambda_1^\star$ and $\lambda_2^\star$, the largest values of $\lambda_1 > 0$ and $\lambda_2 > 0$ such that: $0 \leqslant \pi(t-1) + \lambda_1 u(t) \leqslant 1$ and $0 \leqslant \pi(t-1) - \lambda_2 u(t) \leqslant 1$.
3 With probability $\lambda_2^\star/(\lambda_1^\star + \lambda_2^\star)$, set $\pi(t) \leftarrow \pi(t-1) + \lambda_1 u(t)$; otherwise, set $\pi(t) \leftarrow \pi(t-1) - \lambda_2 u(t)$.

The flight phase stops when Step 1 of Algorithm 11 cannot be performed (i.e. no vector $u(t)$ fulfils these conditions). Until this happens, each iteration increases by at least one the number of components in $\pi(t)$

that are either zero or one. Thus, the flight phases completes at most in N steps.

In practice, to generate $u(t)$, one may proceed as follows: first generate a random vector $v(t) \in \mathbb{R}^N$, then project it in the constraint hyperplane: $u(t) = I(t)v(t) - I(t)A^t(AI(t)A^t)^- AI(t)v(t)$ where $I(t)$ is a diagonal matrix such that $I_{kk}(t)$ is 0 if $\pi_k(t)$ is an integer and 1 otherwise, and M^- denotes the pseudo-inverse of the matrix M.

Chauvet and Tillé (2006) propose a particular method to generate vector $v(t)$ which ensures that the complexity of a single iteration of the flight phase is $\mathcal{O}(J^3)$. This leads to an overall complexity of $\mathcal{O}(NJ^3)$ for the flight phase, since it terminates in at most N iterations.

4.3.5. Landing phase

Denote by $\pi^\star$ the value of process $\pi(t)$ when the flight phase terminates. If $\pi^\star$ is a vertex of $\mathcal{C}$ (i.e. all its components are either zero or one), one may stop and return $\pi^\star$ as the output of the cube algorithm. If $\pi^\star$ is not a vertex, this informs us that no vertex belongs to $\mathcal{K}$. One may implement a landing phase, which aims at choosing randomly a vertex which is close to $\pi^\star$, and such that the variance of the components of $\hat{V}$ is small.

Appendix D gives more details on the landing phase. Note that its worst-case complexity is $\mathcal{O}(2^J)$. However, in practice, it is typically either much faster, or not required (i.e. $\pi^\star$ is already a vertex) as soon as $J \ll N$.

4.4. Cube thinning

We now explain how the previous ingredients (control variates, and the cube method) may be combined in order to thin a Markov chain, $X_1, \ldots, X_N$, into a sub-sample of size M. As before, the invariant distribution of the chain is denoted by p, and we assume we know of J control variates h_j, i.e. $p(h_j) = 0$ for $j = 1, \ldots, J$.

4.4.1. First step: computing the weights

The first step of our method is to use the J control variates to compute the N weights w_n, as defined at the end of Section 4.2.2. Recall that these weights sum to one, that they automatically fulfil the constraints:

$$\sum_{n=1}^{N} w_n h_j(X_n) = 0$$

for $j = 1, \ldots, J$, and that we use them to compute

$$\hat{p}_\star(f) = \sum_{n=1}^{N} w_n f(X_n) \tag{4.8}$$

as a low-variance estimate for $p(f)$ for any f.

Recall that the control variates procedure we described in Section 4.2 assume that the input variables, $X_1, \ldots, X_N$, are IID. This is obviously not the case in a MCMC context; however, we follow the common practice (Mira et al., 2013; Oates et al., 2016) of applying the procedure to MCMC points as if they were IID points. This implies that the weighted estimate above corresponds to a value of β in (4.2) that does not minimise the (asymptotic) variance of estimator (4.2). It is actually possible to estimate the value of β that minimises the asymptotic variance of a MCMC estimate (Dellaportas and Kontoyiannis, 2011; Brosse et al., 2019). However, this type of approach is specific to certain MCMC samplers, and, critically for us, it cannot be cast as a weighting scheme. Thus we stick to this standard approach.

We note in passing that, in our experiments (see Figure 4.1 and the surrounding discussion) the weights w_n makes it easy to assess visually the convergence (and thus the burn-in) of the Markov chain. In fact, since the MCMC points of the burn-in phase are far from the mass of the target distribution, the procedure must assign a small or negative weight to these points in order to respect the constraints based on the control variates. Again, see Section 4.5.2 for more discussion on this issue. The fact that control variates may be used to assess MCMC convergence has been known for a long time (e.g. Brooks and Gelman, 1998), but the visualisation of weights makes this idea more expedient.

4.4.2. Second step: cube resampling

The second step consists in resampling the weighted sample $(w_n, X_n)_{n=1,\dots,N}$, to obtain a sub-sample $\mathcal{S} = \{X_n : S_n = 1\}$ where S_n are random variables such that (a) $\mathbb{E}[S_n] = w_n$; (b) $\sum_{n=1}^N S_n = M$, and (c) for $j = 1, \dots, J$:

$$\sum_{S_n=1} h_j(X_n) = 0.$$

Condition (a) ensures that the procedure does not introduce any bias:

$$\mathbb{E}\left[\frac{1}{M}\sum_{S_n=1} f(X_n)\middle| X_{1:N}\right] = \sum_{n=1}^N w_n f(X_n).$$

Condition (b) ensures that the sub-sample is exactly of size M.

We would like to use the cube method in order to generate the S_n's. Specifically, we would like to assign the inclusion probabilities π_n to w_n, and impose the $(J+1)$ constraints defined above by Conditions (b) and (c). There is one caveat, however: the weights w_n do not necessarily lie in $[0, 1]$.

4.4.3. Dealing with weights outside of $[0, 1]$

We rewrite (4.8) as:

$$\hat{p}_\star(f) = \frac{\Omega}{M} \times \sum_{n=1}^N W_n \times \operatorname{sgn}(w_n) f(X_n)$$

where $\Omega = M^{-1}\sum_{n=1}^N |w_n|$ and $W_n = M|w_n|/\Omega$. We now have $W_n \geqslant 0$, and $\sum_{n=1}^N W_n = M$, which is required for condition (b) in the previous section. We might have a few points such that $W_n > 1$. In that case, we replace them by $\lfloor W_n \rfloor$ copies, with adjusted weights $W_n/\lfloor W_n \rfloor$.

It then becomes possible to implement the cube method, using as inclusion probabilities the W_n's, and as the matrix A that defines the $J+1$ constraints, the matrix $A = (A_{jn})$ such that $A_{1n} = 1$, $A_{(j+1)n} = \operatorname{sgn}(w_n) h_j(X_n)$. The cube method samples variables S_n, which may be used to compute the sub-sampled estimate

$$\hat{\nu}(f) = \frac{\Omega}{M}\sum_{S_n=1} \operatorname{sgn}(w_n) f(X_n).$$

More generally, in our numerical experiments, we shall evaluate to which extent the random signed measure:

$$\hat{\nu} = \frac{\Omega}{M}\sum_{S_n=1} \operatorname{sgn}(w_n)\delta_{X_n}(\mathrm{d}x). \tag{4.9}$$

is a good approximation of the target distribution p.

4.5. Experiments

We consider two examples. The first example is taken from Riabiz et al. (2020), and is used to compare cube thinning with KSD thinning. The second example illustrates cube thinning when used in conjunction with control variates that are not gradient-based. We also include standard thinning in our comparisons.

Note that there is little point in comparing these methods in terms of CPU cost, as KSD thinning is considerably slower than cube thinning and standard thinning whenever $M \gg 100$. (In one of our experiment, for $M = 1000$, KSD took close to 7 hours to run, while cube thinning with all the covariates took about 30 seconds.) Thus, our comparison will be in terms of statistical error, or, more precisely, in terms of how representative of p is the selected sub-sample.

In the following (in particular in the plots), "cubeFull" (resp. "cubeDiagonal") will refer to our approach based on the full (resp. diagonal) set of control variates, as discussed in Section 4.2.3. The mention "NoBurnin" means that burn-in has been discarded manually (hence no burn-in in the inputs). Finally, "thinning" denotes the usual thinning approach, "SMPCOV", "MED" and "SCLMED" are the same names used in Riabiz et al. (2020) for KSD thinning, based on three different kernels.

To implement the cube method, we used R package `BalancedSampling`.

4.5.1. Evaluation criteria

We could compare the three different methods in terms of variance of the estimates of $p(f)$ for certain functions f. However, it is easy to pick functions f that are strongly correlated with the chosen control variates; that would bias the comparison in favour of our approach. In fact, as soon as the target is Gaussian-like, the control variates we chose in Section 4.2.3 should be strongly correlated with the expectation of any polynomial function of order two, as we discussed in that section.

Rather, we consider criteria that are indicative of the performance of the methods for a general class of function. Specifically, we consider three such criteria. The first one is the kernel Stein discrepency (KSD) as defined in Riabiz et al. (2020) and recalled in the introduction, see (4.1). Note that this criterion is particularly favourable to KSD thinning, since this approach specifically minimises this quantity. (We use the particular version based on the median kernel in Riabiz et al. (2020).)

The second criterion is the energy distance (ED) between p and the empirical distribution defined by the thinning method; e.g. (4.9) for cube thinning. Recall that the ED between two distributions F and G is:

$$ED(F, G) = 2\mathbb{E}||Z - X||_2 - \mathbb{E}||Z - Z'||_2 - \mathbb{E}||X - X'||_2 \tag{4.10}$$

where $Z', Z \overset{iid}{\sim} F$ and $X', X \overset{iid}{\sim} G$, and that this quantity is actually a pseudo-distance: $ED(F, G) \geqslant 0$, $ED(F, G) = 0 \Rightarrow F = G$, $ED(F, G) = ED(G, F)$, but ED does not fulfil the triangle inequality (Székely and Rizzo, 2005; Klebanov, 2006).

One technical difficulty is that (4.9) is a signed measure, not a probability measure; see Appendix E on how we dealt with this issue.

Our third criteria is inspired by the star discrepancy, a well-known measure of the uniformity of N points $u_n \in [0, 1]^d$ in the context of quasi-Monte Carlo sampling (Owen, 2013, Chap. 15). Specifically, we consider the quantity

$$d^\star(\hat{P}, \hat{\nu}) = \sup_{B \in \mathcal{B}} \left| \hat{P}_\psi(B) - \hat{\nu}_\psi(B) \right|$$

where $\psi : \mathbb{R}^d \to [0, 1]^d$, $\hat{P}_\psi$ and $\hat{\nu}_\psi$ are the push-forward measures associated to empirical distributions $\hat{P} = (N - b)^{-1} \sum_{n=b+1}^{N} \delta_{X_n}(dx)$, and $\hat{\nu}$ as defined in (4.9), and $\mathcal{B}$ is the set of hyper-rectangles $B = \prod_{i=1}^{d} [0, b_i]$. In practice, we defined function ψ as follows: we apply the linear transform that makes the considered sample to have zero mean and unit variance, and then we applied the inverse CDF (cumulative distribution function) of a unit Gaussian to each component.

Also, since the sup above is not tractable, we replace it by a maximum over a finite number of b_i (simulated uniformly).

4.5.2. Lotka-Volterra model

This example is taken from Riabiz et al. (2020). The Lotka-Volterra model describes the evolution of a prey-predator system in a closed environment. We denote the number of prey by u_1 and the number of predator by u_2. The growth rate of the prey is controlled by a parameter $\theta_1 > 0$ and its death rate - due to the interactions with the predators - is controlled by a parameter $\theta_2 > 0$. In the same way, the predator population has a death rate of $\theta_3 > 0$ and a growth rate of $\theta_4 > 0$. Given these parameters, the evolution of the system is described by a system of ODEs:

$$\begin{aligned} \frac{du_1}{dt} &= \theta_1 u_1 - \theta_2 u_1 u_2 \\ \frac{du_2}{dt} &= \theta_4 u_1 u_2 - \theta_3 u_2 \end{aligned}$$

Riabiz et al. (2020) set $\theta = (\theta_1, \theta_2, \theta_3, \theta_4) = (0.67, 1.33, 1, 1)$, the initial condition $u_0 = (1, 1)$, and simulate synthetic data. They assume they observe the populations of prey and predator at times $t_i, i = 1, \ldots, 2400$ where the t_i are taken uniformly on $[0, 25]$ and that these observations are corrupted with a centered Gaussian noise with a covariance matrix $C = \text{diag}(0.2^2, 0.2^2)$. Finally, the model is parametrized in terms of $x = (\log\theta_1, \log\theta_2, \log\theta_3, \log\theta_4) \in \mathbb{R}^4$ and a standard normal distribution as a prior on x is used.

The authors have provided their code as well as the sampled values they got by running different MCMC chains for a long time. We use the exact same experimental set-up, and we do not run any MCMC chain on our own, but use the ones they provide instead; specifically the simulated chain, of length 2×10^6, from preconditionned-MALA.

We compress this chain into a subsample of size either $M = 100$ or $M = 1000$. For each value of M, we run different variations of our cube method 50 times and make a comparison with the usual thinning method and with the KSD thinning method with different kernels, see Riabiz et al. (2020). In Figure 4.1 we show the first 5000 weights of the cube method. We can see that after 1000 iterations, the weights seem to stabilize. Based on visual examination of these weights, we choose a conservative burnin period of 2000 iterations for the variants where burn-in is removed manually.

We plot the results of the experiment on Figures 4.3, 4.2 and 4.4.

First, we see that regarding the kernel Stein discrepancy metric, Figure 4.2, the KSD method performs better than the standard thinning procedure and the cube method. This is not surprising since even if this method does not properly minimizes the Kernel-Stein Discrepency, this is still its target. We also see that for $M = 1000$, the KSD method performs a bit better than our cube method which in turn performs better than the standard thinning procedure. Note that the relative performance of the KSD method to our cube methods depends on the kernel that is being used and that there is no way to determine which kernel will perform best before running any experiment.

The picture is different for $M = 100$: KSD thinning outperforms standard thinning, which in turn outperforms all of our cube thinning variations. Once again, the fact that the KSD method performs better than any other method seems reasonable: since it is about minimizing the Kernel-Stein Discrepancy, the KSD method is "playing at home" on this metric.

If we look at Figure 4.4, we see that all of our cube methods outperform the KSD method with any kernel. Interestingly, the standard thinning methods has a similar Energy Distance as the cube methods with "diagonal" control variates. These observations are true for both $M = 100$ and $M = 1000$. We can also note that the cube method with the full set of control variates tends to perform much better than its "diagonal" counterpart, whatever the value of M.

Finally, looking at Figure 4.3, it is clear that the KSD method - with any kernel - performs worse than any cube method in terms of star discrepancy.

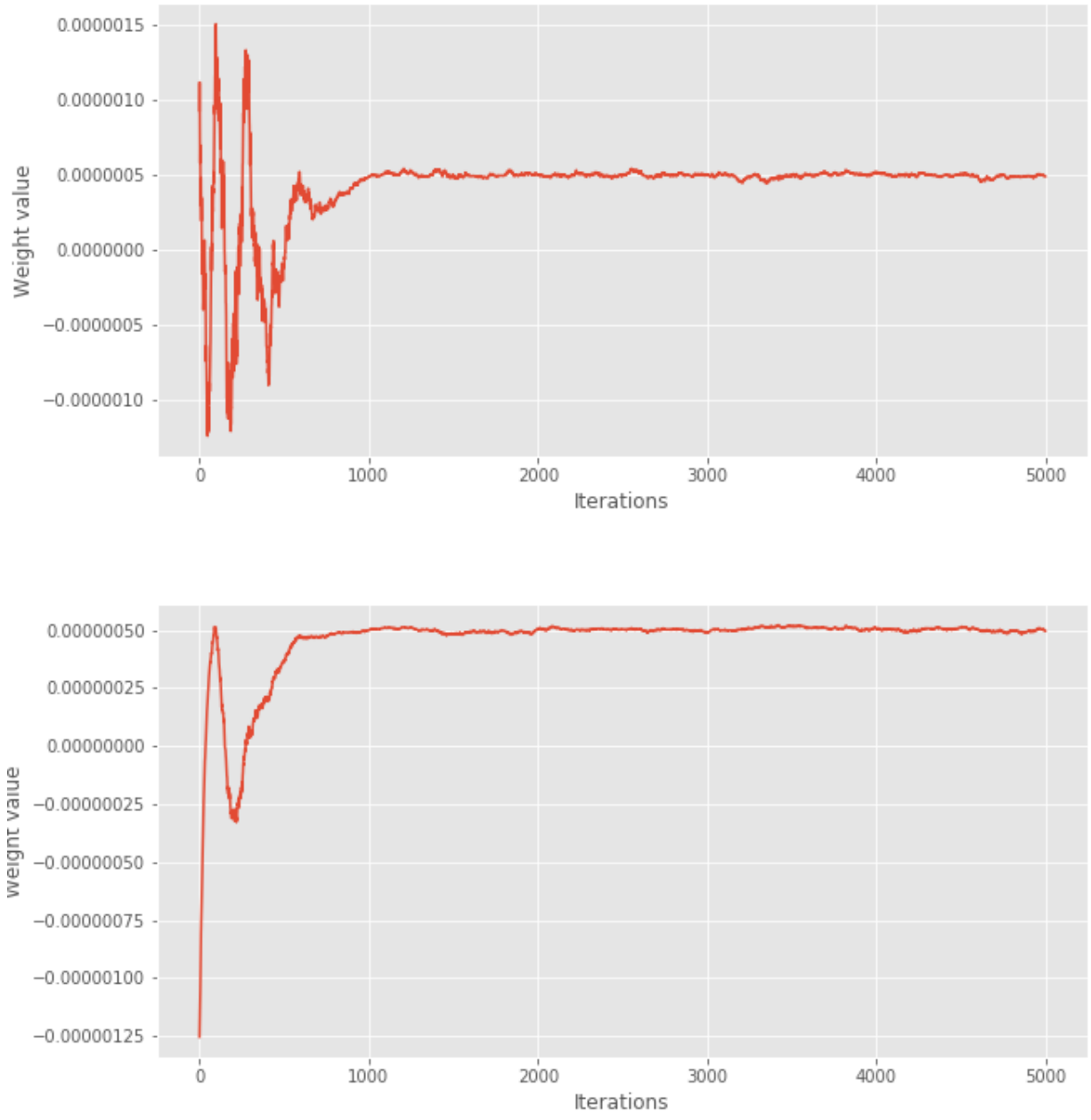

Figure 4.1.: Lotka-Volterra example: first 5000 weights of the cube methods, based on full (top) or diagonal (bottom) set of covariates.

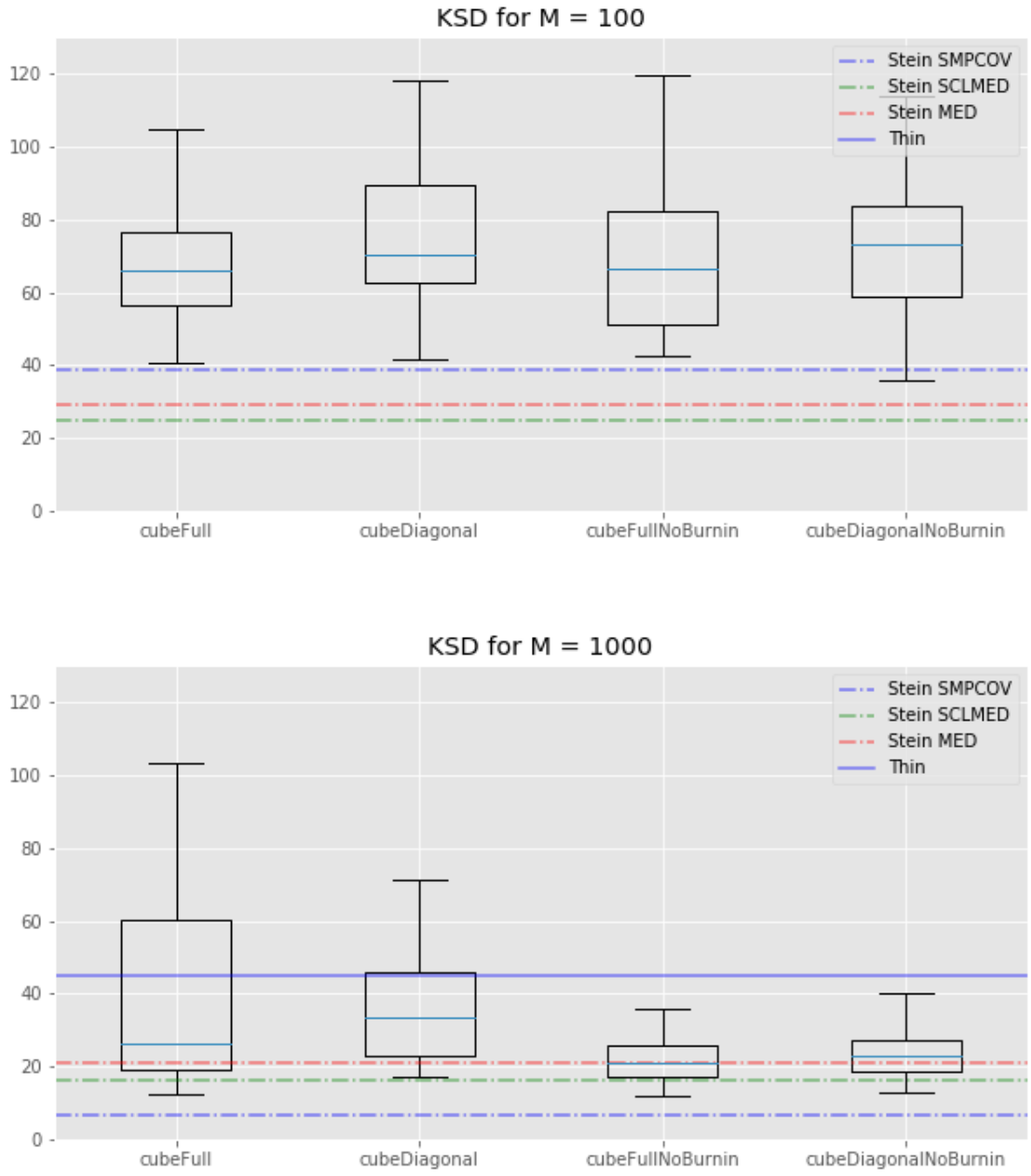

Figure 4.2.: Lotka-Volterra example: box-plots of the kernel Stein discrepency for all the cube method variations, the KSD method for three kernels and the usual thinning method. Top: $M = 100$. Bottom: $M = 1000$. (In the top plot, standard thinning is omitted to improve clarity, as corresponding value is too high.)

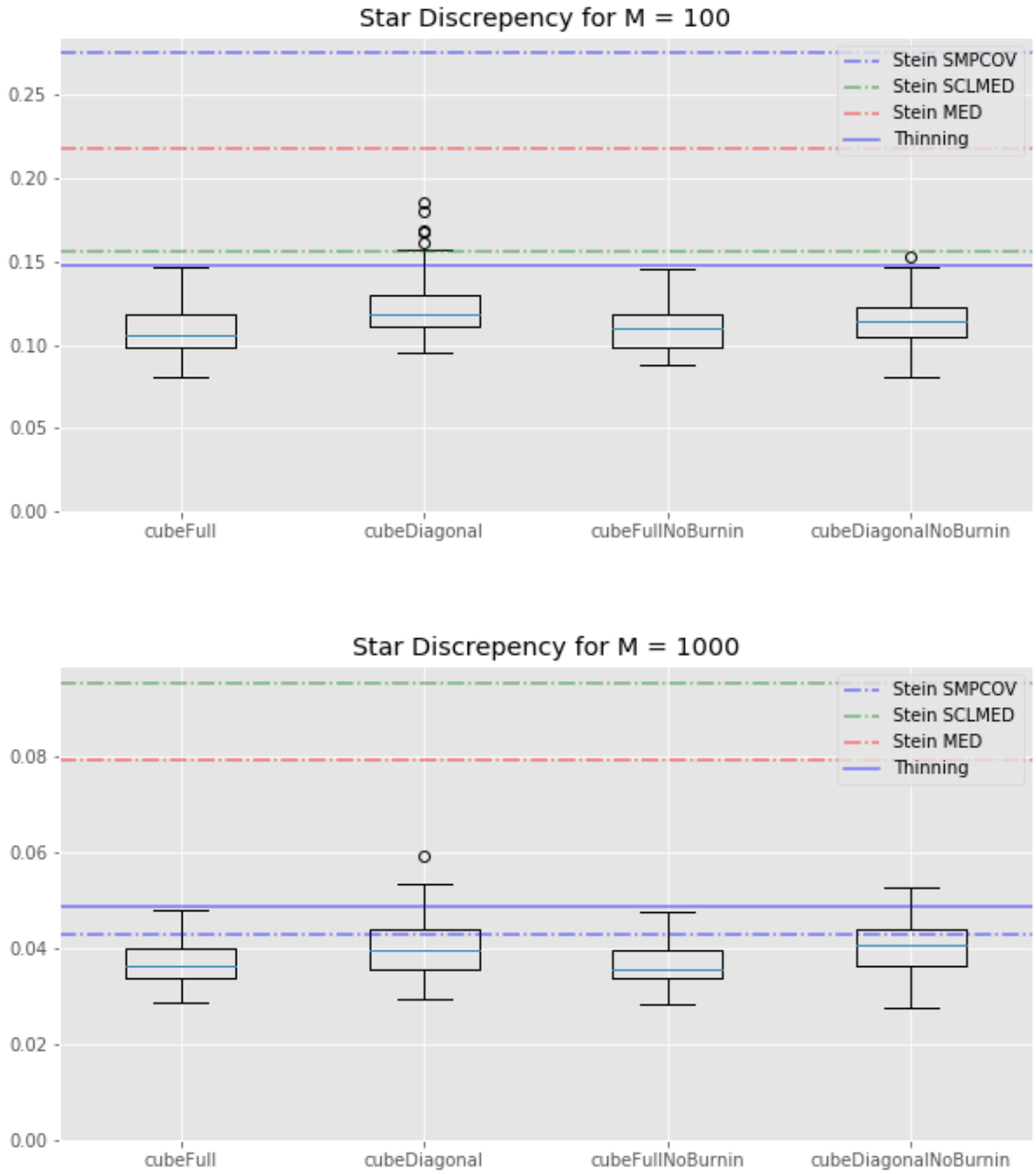

Figure 4.3.: Lotka-Volterra example: box-plots of the star discrepency for all the cube method variations, the KSD method for three kernels and the usual thinning method. Top: $M = 100$. Bottom: $M = 1000$.

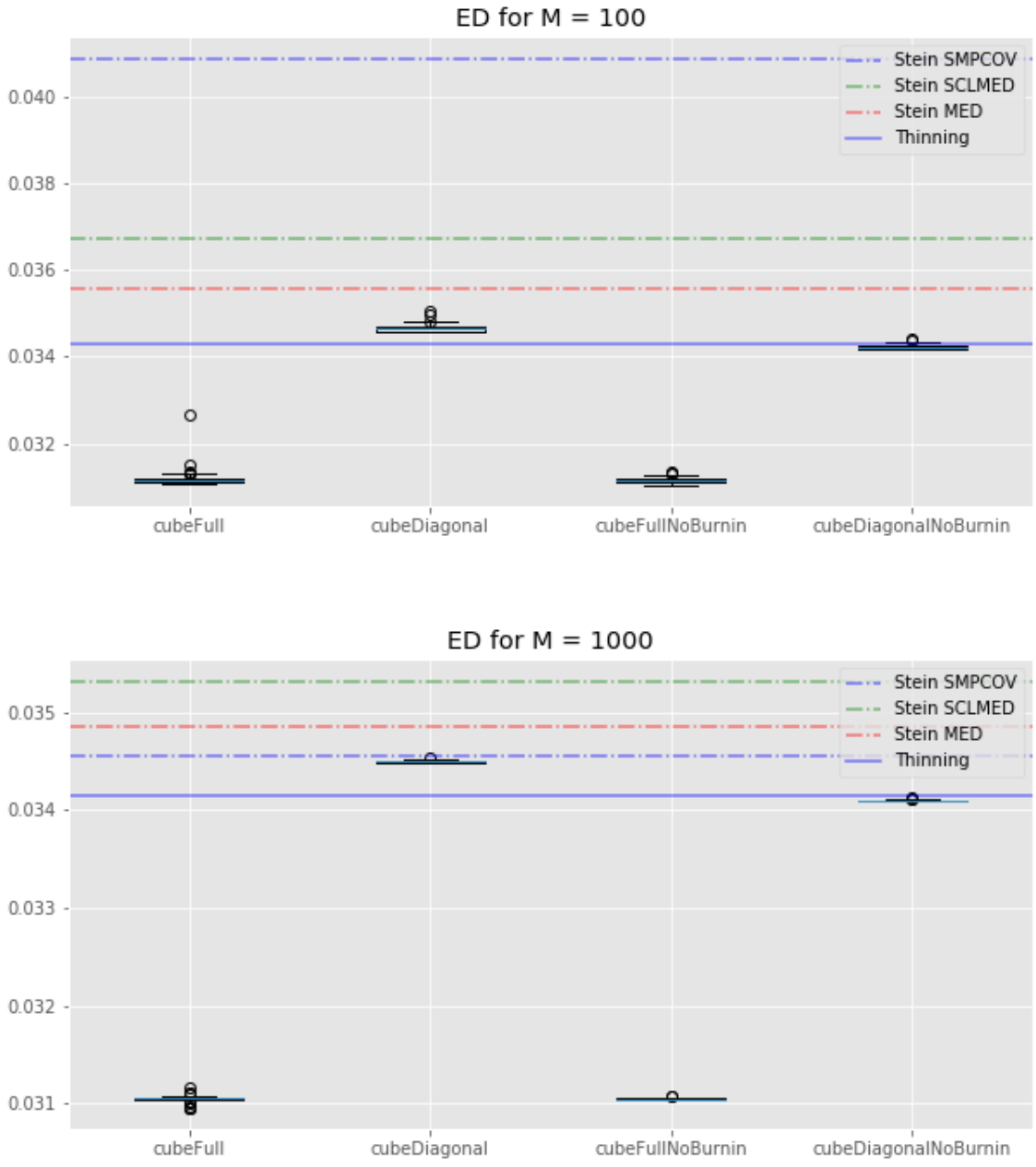

Figure 4.4.: Lotka-Volterra example: boxplots of the energy distance for all the cube method variations, the KSD method for three kernels and the usual thinning method. Top: $M = 100$. Bottom: $M = 1000$.

Overall, the relative performance of the cube methods and KSD methods can change a lot depending on the metric being used and the number of points we keep. In addition, while all the cube methods tend to perform roughly the same, this is not the case of the KSD method, whose performances depend on the kernel we use. Unfortunately, we have no way to determine beforehand which kernel will perform best. This is a problem since the KSD method is computationally expensive for subsamples of cardinal $M \gg 100$.

Thus, by and large, cube thinning seems much more convenient to use (both in terms of CPU time and sensitivity to tuning parameters) while offering, roughly, the same level of statistical performance.

4.5.3. Truncated Normal

In this example, we use the (random-scan version of) the Gibbs sampler of Robert and Casella (2004) to sample from 10-dimensional multivariate normal truncated to $[0, \infty)^{10}$. We generated the parameters of this truncated normal as follows: the mean was set as the realization of a 10-dimensional standard normal distribution, while for the covariance matrix Σ we first generated a matrix $M \in \mathcal{M}_{10,10}(\mathbb{R})$ for which each entry was the realization of a standard normal distribution. Then we set $\Sigma = M^T M$.

Since we are using a Gibbs sampler, we have access to the Gibbs control variates of Dellaportas and Kontoyiannis (2011), based on the expectation of each update (which amounts to simulating from a univariate Gaussian). Thus, we consider 10 control variates.

The Gibbs sampler is run for $N = 10^5$ iterations; no burn-in is performed. We compare the following estimators of the expectation of the target distribution the standard estimator, based on the whole chain ('usualEstim' in the plots), the estimator based on standard thinning ('thinEstim' in the plots), the control variate estimator based on the whole chain, i.e. (4.4) ('regressionEstim' in the plots), and finally our cube estimator described in Section 4.4 ('cubeEstim' in the plots). For standard thinning and cube thinning, the thinning sample size is set to $M = 100$, which corresponds to a compression factor of 10^3.

The results are shown in Figure 4.5. First, we can see that the control variates we chose lead to a substantial decrease in the variance of the estimates for regressionEstim compared to usualEstim. Second, the cube estimator performs worse than the regression estimator in terms of variance, but this was expected, as explained in Section 4.4. More interestingly, if we cannot say that the cube estimator performs better than the usual MCMC estimator in general, we can see that on some components it performs as good or even better, even though the cube estimator uses only $M = 100$ points while the usual estimator uses 10^5 points. This is largely due to the excellent choice of the control variates. Finally, the cube estimator outperforms the regular thinning estimator on every component, sometimes significantly.

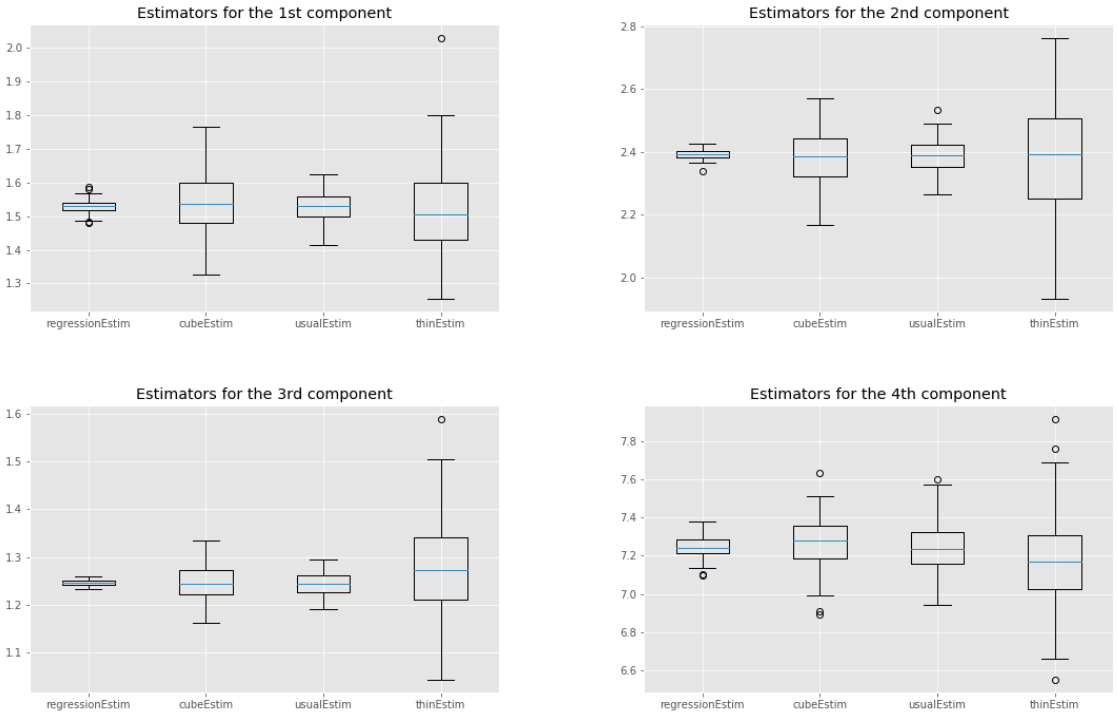

Figure 4.5.: Truncated normal example: box-plots over 100 independent replicates of each estimator; see text for more details.

Appendix D.

Details on the landing phase

The landing phase seeks to generate a random vector S in $\{0,1\}^N$, with expectation $\pi^\star$ (the output of the flight phase), which minimises the criterion $\mathrm{tr}(M\mathrm{Var}(\hat{V}|\pi^\star))$ for a certain matrix M. (The notation $\cdot|\pi^\star$ refers to the distribution of S conditional on $\pi(t)=\pi^\star$ at the end of the flight phase.)

Since $\mathrm{Var}(S)=\mathrm{Var}(\mathbb{E}[S|\pi^\star])+\mathbb{E}[\mathrm{Var}(S|\pi^\star)]$ by the law of total variance, and since the first term is zero (as $\mathbb{E}[S|\pi^\star]=\pi^\star$), we have

$$\mathrm{Var}(\hat{V})=\mathbb{E}[\mathrm{Var}(\hat{V}|\pi^\star)]=\mathbb{E}[A\mathrm{Var}(S|\pi^\star)A^t].$$

and thus:

$$\mathrm{tr}(M\mathrm{Var}(\hat{V}|\pi^\star))=\sum_{s\in\{0,1\}^N} p(s|\pi^\star)(s-\pi^\star)^t A^t M A(s-\pi^\star).$$

Choosing $M=(AA^t)^{-1}$, as recommended by Deville (2004), amounts to minimising the distance to the hyperplane 'on average'. Let

$$C(s)=(s-\pi^\star)^t A^t (AA^t)^{-1} A^t (s-\pi^\star),$$

then the minimisation program is equivalent to the following linear programming problem over q variables only:

$$\min_{\xi^\star(.)} \sum_{s^\star\in\mathcal{S}^\star} C(s^\star)\xi^\star(s^\star)$$

with constraints $\sum_{s^\star\in\mathcal{S}^\star}\xi^\star(s^\star)=1$, $0\leqslant\xi^\star(s^\star)\leqslant 1$, $\sum_{s^\star\in\mathcal{S}^\star|s^\star_k=1}\xi^\star(s^\star)=\pi^\star_k$ for every $k\in U^\star$ and $\mathcal{S}^\star=\{0,1\}^q$ where $q=\mathrm{card}(U^\star)$ and $U^\star=\{k\in U:\,0<\pi^\star[k]<1\}$. Here $\xi^\star$ denotes the marginal distribution of the components $U^\star$ of the sampling design ξ and $C(s^\star)$ must be understood as $C(s)$ with the components of $s\notin U^\star$ being fixed by the result of flight phase, thus in this minimization problem C is in fact depending on the components of s that are in $U^\star$ only.

The constraints define a bounded polyhedron. By the fundamental theorem of linear programming, this optimization problem has at least one solution on a minimal support, see Deville (2004).

The flight phase ends on a vertex of $\mathcal{K}$ and, by Proposition 1 in Deville (2004), $q\leqslant J$; typically $J\ll N$. This means that we are solving a linear programming problem in a dimension q potentially much lower than the population size N, and if we do not have too many auxiliary variables, this optimization problem will not be computationally too expensive. In practice, a simplex algorithm is used to find the solution.

Appendix E.

Estimation of the energy distance

There are two difficulties with computing (4.10). First, it involves intractable expectations. Second, as pointed out at the end of Section 4.4.3, the empirical distribution generated by cube thinning, (4.9), is actually a signed measure.

Regarding the first issue, we can approximate (4.10) from our MCMC sample $X_1, \dots, X_N$. That is, if our subsampled empirical measure writes $\hat{\nu} = \sum_{m=1}^{M} w_m \delta_{Z_m}$ and that we approximate the distribution associated with p by $\hat{P} = (N-b)^{-1} \sum_{n=b+1}^{N} \delta_{X_n}$ where $1 \leqslant b \leqslant N$ is the burn-in of the chain, then, we can estimate $ED(\hat{\mu}, p)$ with $ED(\hat{\mu}, \hat{P})$.

Regarding the second issue, we can generalize the energy distance to finite measures: suppose we have two finite and potentially signed measures ν_1 and ν_2, both defined on the same measurable space $(\Omega, \mathcal{P}(\Omega)\}$ where $\Omega = \{X_1, \dots, X_N\}$ and $\mathcal{P}(\Omega)$ denotes the set of parts of Ω. Suppose in addition that $\nu_1(\Omega) = \alpha_1$ and $\nu_2(\Omega) = \alpha_2$ with $\alpha_1 \neq 0$ and $\alpha_2 \neq 0$. We define the generalized energy distance as:

$$\begin{aligned} ED^{\star}(\nu_1, \nu_2) =& \frac{2}{\alpha_1 \alpha_2} \int_{\Omega} ||x-y||_2 d\nu_1(x) d\nu_2(y) \\ & - \frac{1}{\alpha_1^2} \int_{\Omega} ||x-x'||_2 d\nu_1(x) d\nu_1(x') \\ & - \frac{1}{\alpha_2^2} \int_{\Omega} ||y-y'||_2 d\nu_2(y) d\nu_2(y'). \end{aligned}$$

Then, by negative definiteness of the application $\phi(x,y) = ||x-y||_2$ on $\mathbb{R}^N \times \mathbb{R}^N$, we have that $ED^{\star}(\nu_1, \nu_2) \geqslant 0$ with equality if and only if $\frac{1}{\alpha_1}\nu_1 = \frac{1}{\alpha_2}\nu_2$. Which means that the generalized energy distance is zero if and only if the two measures are equal up to a non-zero multiplicative constant, see Székely and Rizzo (2005) for a demonstration. This generalized energy distance is also symmetric, but the triangle inequality does not hold. It is a pseudo-distance.

Thus we will use the following criterion, which we will abusively call the energy distance in the rest of the paper:

$$\begin{aligned} ED^{\star}(\hat{\nu}, \hat{P}) =& \frac{2}{(N-b)\alpha_1} \sum_{k=1}^{N} \sum_{n=b+1}^{N} \frac{\Omega}{M} sgn(w_k) ||X_k - X_n||_2 \mathbf{1}_{\{S_k=1\}} \\ & - \frac{1}{\alpha_1^2} \sum_{n=1}^{N} \sum_{k=1}^{N} \left(\frac{\Omega}{M}\right)^2 sgn(w_n) sgn(w_k) ||Z_k - Z_n||_2 \mathbf{1}_{\{S_k=1\}} \mathbf{1}_{\{S_n=1\}} \end{aligned}$$

where $\hat{\nu}$ is defined in (4.9) and we dropped the last term because it does not depend on $\hat{\nu}$ and it is a potentially expensive sum of $(N-b)^2$ terms.

Note that the probability of $\hat{\nu}(\Omega)$ being zero is non-null and then there is a non-negligible probability of $ED^{\star}(\hat{\nu}, \hat{P})$ being undefined. However, this event is unlikely to happen.

Chapter 5.

Conclusion

The research presented in this thesis represents only a fraction of the methods that have been tried or could be tried.

For the application to the Cosmic Microwave Background data analysis, we tried to sample from the high dimensional Gaussian distribution using a preconditionned Crank-Nicolson algorithm: since both of the prior term and the likelihood are Gaussians, we could sample from the likelihood on the low signal-to-noise ratio range and correct with a metropolis ratio involving the prior, and sample from the prior on the high signal-to-noise ratio part of the problem, to correct with a metropolis ratio involving the likelihood, as it is usually done. This works well when considering the observed skymap to be zero everywhere. But when it is not the case, this scheme does not work anymore. Other things that have been tried are the Metropolis adjusted Langevin algorithm. But the problem is too high dimensional for a Metropolis ratio to be good and the correlations are too strong on a large set of variables so that fitting a good proposal distribution is very costly. In addition, if the user is willing to spend a long time computing a good approximation of the covariance matrix of the Gaussian target, it is better invested in the computation of a good preconditionner for the PCG resolution, that brings uncorrelated samples, on the contrary of a MALA algorithm. We also tried some piece-wise deterministic Markov processes, which did not bring improvements.

There are several avenues of research. First, using an unadjusted Langevin algorithm within Gibbs would be worth trying. It would bypass the correlation problem we have with the auxiliary variable scheme. It would not suffer from the Metropolis ratio, it would also be very cheap and a Gaussian target has a log-concave distribution, so we can expect the unadjusted Langevin algorithm to ”behave well”. Of course this may introduce a bias in the results of the Gibbs sampler and this bias is hard to quantify theoretically. It would be interesting to observe the behavior of the ULA within Gibbs experimentally.

Another avenue of research would be to integrate the foregrounds to the model. The correlations between the CMB skymap and the foregrounds are strong and a Gibbs sampler targeting their joint distribution while having to make a PCG resolution is inefficient: the correlations are so strong and the number of iterations so low - because of the computing cost - that it cannot explore the target distribution efficiently. Since we presented two very cheap algorithms, namely Centered 1 and Centered overrelax, it may be interesting to try foreground removal. The correlations are still strong, but these algorithms are so cheap that we could do a high number of iterations and thus explore the posterior distribution.

Chapter 6.

List of talks

During these three PhD years, I have been invited to various events to talk about my research. Here is a list of these events.

- ”Fast compression of MCMC output”, talk, Current developments in MCMC methods at IMPAN, Warsaw, December 2021.
- ”Fast compression of MCMC output”, talk and poster, ISBA: Measuring the quality of MCMC output, October 2021
- ”Fast compression of MCMC output”, talk, Journées MAS, August 2021.

Bibliography

N. A. Ade et al. Planck2013 results. x. HFI energetic particle effects: characterization, removal, and simulation. *Astronomy & Astrophysics*, 571:A10, oct 2014a. doi: 10.1051/0004-6361/201321577.

P. Ade, J. Aguirre, and et al. The simons observatory: science goals and forecasts. *Journal of Cosmology and Astroparticle Physics*, 2019(02):056–056, feb 2019. doi: 10.1088/1475-7516/2019/02/056.

P. A. R. Ade, N. Aghanim, et al. Planck2013 results. XII. diffuse component separation. *Astronomy & Astrophysics*, 571:A12, oct 2014b. doi: 10.1051/0004-6361/201321580.

S. Agapiou, J. M. Bardsley, O. Papaspiliopoulos, and A. M. Stuart. Analysis of the Gibbs Sampler for Hierarchical Inverse Problems. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):511–544, Jan. 2014. ISSN 2166-2525. doi: 10.1137/130944229. URL `http://epubs.siam.org/doi/10.1137/130944229`.

S. Brooks and A. Gelman. Some issues for monitoring convergence of iterative simulations. *Computing Science and Statistics*, pages 30–36, 1998.

N. Brosse, A. Durmus, S. Meyn, E. Moulines, and A. Radhakrishnan. Diffusion approximations and control variates for MCMC. *arXiv 1808.01665*, 2019.

G. Chauvet and Y. Tillé. A fast algorithm for balanced sampling. *Computational Statistics*, 21(1):53–62, mar 2006. doi: 10.1007/s00180-006-0250-2.

M. Chu, H. K. Eriksen, L. Knox, K. M. Górski, J. B. Jewell, D. L. Larson, I. J. O'Dwyer, and B. D. Wandelt. Cosmological parameter constraints as derived from the Wilkinson Microwave Anisotropy Probe data via Gibbs sampling and the Blackwell-Rao estimator. *Physical Review D*, 71(10):103002, May 2005. ISSN 1550-7998, 1550-2368. doi: 10.1103/PhysRevD.71.103002. URL `https://link.aps.org/doi/10.1103/PhysRevD.71.103002`.

M. K. Cowles and B. P. Carlin. Markov chain Monte Carlo convergence diagnostics: a comparative review. *J. Amer. Statist. Assoc.*, 91(434):883–904, 1996. ISSN 0162-1459. doi: 10.2307/2291683. URL `https://doi.org/10.2307/2291683`.

P. Dellaportas and I. Kontoyiannis. Control variates for estimation based on reversible Markov chain Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74 (1):133–161, nov 2011. doi: 10.1111/j.1467-9868.2011.01000.x.

J.-C. Deville. Efficient balanced sampling: The cube method. *Biometrika*, 91(4):893–912, dec 2004. doi: 10.1093/biomet/91.4.893.

G. Efstathiou. Myths and truths concerning estimation of power spectra: the case for a hybrid estimator. *Monthly Notices of the Royal Astronomical Society*, 349(2):603–626, apr 2004. doi: 10.1111/j.1365-2966.2004.07530.x.

F. Elsner and B. D. Wandelt. Efficient Wiener filtering without preconditioning. *Astronomy & Astrophysics*, 549:A111, Jan. 2013. doi: 10.1051/0004-6361/201220586.

H. K. Eriksen, I. J. O'Dwyer, J. B. Jewell, B. D. Wandelt, D. L. Larson, K. M. Górski, S. Levin, A. J. Banday, and P. B. Lilje. Power Spectrum Estimation from High-Resolution Maps by Gibbs Sampling. *The Astrophysical Journal Supplement Series*, 155(2):227–241, Dec. 2004. ISSN 0067-0049, 1538-4365. doi: 10.1086/425219. URL https://iopscience.iop.org/article/10.1086/425219.

H. K. Eriksen, J. B. Jewell, C. Dickinson, A. J. Banday, K. M. Górski, and C. R. Lawrence. Joint Bayesian Component Separation and CMB Power Spectrum Estimation. *The Astrophysical Journal*, 676(1):10–32, Mar. 2008. ISSN 0004-637X, 1538-4357. doi: 10.1086/525277. URL https://iopscience.iop.org/article/10.1086/525277.

M. Gerbino, M. Lattanzi, M. Migliaccio, L. Pagano, L. Salvati, L. Colombo, A. Gruppuso, P. Natoli, and G. Polenta. Likelihood Methods for CMB Experiments. *Frontiers in Physics*, 8:15, Feb. 2020. ISSN 2296-424X. doi: 10.3389/fphy.2020.00015. URL https://www.frontiersin.org/article/10.3389/fphy.2020.00015/full.

C. J. Geyer. Practical Markov chain Monte Carlo. *Statistical Science*, 7(4), nov 1992. doi: 10.1214/ss/1177011137.

C. Gilavert, S. Moussaoui, and J. Idier. Efficient Gaussian Sampling for Solving Large-Scale Inverse Problems Using MCMC. *IEEE Transactions on Signal Processing*, 63(1):70–80, Jan. 2015. ISSN 1053-587X, 1941-0476. doi: 10.1109/TSP.2014.2367457. URL http://ieeexplore.ieee.org/document/6945861/.

E. Gjerløw, L. P. L. Colombo, H. K. Eriksen, K. M. Górski, A. Gruppuso, J. B. Jewell, S. Plaszczynski, and I. K. Wehus. Optimized Large-scale CMB Likelihood and Quadratic Maximum Likelihood Power Spectrum Estimation. *The Astrophysical Journal Supp. Series*, 221(1):5, Nov. 2015. doi: 10.1088/0067-0049/221/1/5.

P. Glasserman. *Monte Carlo methods in financial engineering*, volume 53 of *Applications of Mathematics (New York)*. Springer-Verlag, New York, 2004. ISBN 0-387-00451-3. Stochastic Modelling and Applied Probability.

K. M. Gorski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann. HEALPix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759–771, apr 2005. doi: 10.1086/427976.

J. Grain, M. Tristram, and R. Stompor. Polarized CMB power spectrum estimation using the pure pseudo-cross-spectrum approach. *Physical Review D*, 79(12):123515, June 2009. ISSN 1550-7998, 1550-2368. doi: 10.1103/PhysRevD.79.123515. URL https://link.aps.org/doi/10.1103/PhysRevD.79.123515.

A. Hajian. Efficient cosmological parameter estimation with Hamiltonian Monte Carlo technique. *Physical Review D*, 75(8):083525, Apr. 2007. ISSN 1550-7998, 1550-2368. doi: 10.1103/PhysRevD.75.083525. URL https://link.aps.org/doi/10.1103/PhysRevD.75.083525.

S. Hamimeche and A. Lewis. Likelihood analysis of CMB temperature and polarization power spectra. *Physical Review D*, 77(10):103013, May 2008. ISSN 1550-7998, 1550-2368. doi: 10.1103/PhysRevD.77.103013. URL https://link.aps.org/doi/10.1103/PhysRevD.77.103013.

S. Hamimeche and A. Lewis. Properties and use of CMB power spectrum likelihoods. *Physical Review D*, 79(8):083012, Apr. 2009. ISSN 1550-7998, 1550-2368. doi: 10.1103/PhysRevD.79.083012. URL https://link.aps.org/doi/10.1103/PhysRevD.79.083012.

H. Hammer and H. Tjelmeland. Control variates for the Metropolis–Hastings algorithm. *Scandinavian Journal of Statistics*, 35(3):400–414, 2008.

J. Harold. An invariant form for the prior probability in estimation problems. *R. Soc. Lond*, pages 186453–461, 1946. doi: 10.1098/rspa.1946.0056.

C. R. Harris and et al. Array programming with NumPy. *Nature*, 585(7825):357–362, Sept. 2020. doi: 10.1038/s41586-020-2649-2. URL https://doi.org/10.1038/s41586-020-2649-2.

W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, apr 1970. doi: 10.1093/biomet/57.1.97.

M. Hazumi and L. J. S. Group. LiteBIRD satellite: JAXA's new strategic l-class mission for all-sky surveys of cosmic microwave background polarization. In M. Lystrup, N. Batalha, E. C. Tong, N. Siegler, and M. D. Perrin, editors, *Space Telescopes and Instrumentation 2020: Optical, Infrared, and Millimeter Wave*. SPIE, dec 2020. doi: 10.1117/12.2563050.

E. Hivon, K. M. Gorski, C. B. Netterfield, B. P. Crill, S. Prunet, and F. Hansen. MASTER of the cosmic microwave background anisotropy power spectrum: A fast method for statistical analysis of large and complex cosmic microwave background data sets. *The Astrophysical Journal*, 567(1):2–17, mar 2002a. doi: 10.1086/338126.

E. Hivon, K. M. Górski, C. B. Netterfield, B. P. Crill, S. Prunet, and F. Hansen. MASTER of the Cosmic Microwave Background Anisotropy Power Spectrum: A Fast Method for Statistical Analysis of Large and Complex Cosmic Microwave Background Data Sets. *The Astrophysical Journal*, 567(1):2–17, Mar. 2002b. ISSN 0004-637X, 1538-4357. doi: 10.1086/338126. URL https://iopscience.iop.org/article/10.1086/338126.

J. P. Hobert and G. Casella. The Effect of Improper Priors on Gibbs Sampling in Hierarchical Linear Mixed Models. *Journal of the American Statistical Association*, 91(436):1461–1473, Dec. 1996. ISSN 0162-1459, 1537-274X. doi: 10.1080/01621459.1996.10476714. URL http://www.tandfonline.com/doi/abs/10.1080/01621459.1996.10476714.

J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi: 10.1109/MCSE.2007.55.

L. Isserlis. On a formula for the product-moment coefficient of any order of a normal frequency distribution in any number of variables. *Biometrika*, 12(1-2):134–139, nov 1918. doi: 10.1093/biomet/12.1-2.134.

J. Jewell, S. Levin, and C. H. Anderson. Application of Monte Carlo Algorithms to the Bayesian Analysis of the Cosmic Microwave Background. *The Astrophysical Journal*, 609(1):1–14, July 2004. ISSN 0004-637X, 1538-4357. doi: 10.1086/383515. URL https://iopscience.iop.org/article/10.1086/383515.

J. B. Jewell, H. K. Eriksen, B. D. Wandelt, I. J. O'Dwyer, G. Huey, and K. M. Górski. A Markov chain Monte Carlo algorithm for analysis of low signal-to-noise Cosmic Microwave Background data. *The Astrophysical Journal*, 697(1):258–268, May 2009. ISSN 0004-637X, 1538-4357. doi: 10.1088/0004-637X/697/1/258. URL https://iopscience.iop.org/article/10.1088/0004-637X/697/1/258.

L. B. Klebanov. *N-distances and Their Applications*. The Karolinum Press, Charles University, 2006.

D. L. Larson, H. K. Eriksen, B. D. Wandelt, K. M. Górski, G. Huey, J. B. Jewell, and I. J. O'Dwyer. Estimation of Polarized Power Spectra by Gibbs Sampling. *The Astrophysical Journal*, 656(2):653–660, Feb. 2007. ISSN 0004-637X, 1538-4357. doi: 10.1086/509802. URL https://iopscience.iop.org/article/10.1086/509802.

A. Lewis and S. Bridle. Cosmological parameters from CMB and other data: A Monte Carlo approach. *Physical Review D*, 66(10):103511, Nov. 2002. ISSN 0556-2821, 1089-4918. doi: 10.1103/PhysRevD.66.103511. URL https://link.aps.org/doi/10.1103/PhysRevD.66.103511.

C. Liu. Alternating Subspace-Spanning Resampling to Accelerate Markov Chain Monte Carlo Simulation. *Journal of the American Statistical Association*, 98(461):110–117, Mar. 2003. ISSN 0162-1459, 1537-274X. doi: 10.1198/016214503388619148. URL http://www.tandfonline.com/doi/abs/10.1198/016214503388619148.

J. S. Liu. Fraction of Missing Information and Convergence Rate of Data Augmentation. *Computationally Intensive Statistical Methods: Proceedings of the 26th Symposium Interface*, pages 490–497, 1994.

J. S. Liu, W. H. Wong, and A. Kong. Covariance structure of the Gibbs sampler with applications to the comparisons of estimators and augmentation schemes. *Biometrika*, 81(1):27–40, mar 1994. doi: 10.1093/biomet/81.1.27.

J. S. Liu, W. H. Wong, and A. Kong. Covariance Structure and Convergence Rate of the Gibbs Sampler with Various Scans. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):157–169, Jan. 1995. ISSN 00359246. doi: 10.1111/j.2517-6161.1995.tb02021.x. URL http://doi.wiley.com/10.1111/j.2517-6161.1995.tb02021.x.

S. Mak and V. R. Joseph. Support points. *The Annals of Statistics*, 46(6A), dec 2018. doi: 10.1214/17-aos1629.

Y. Marnissi, E. Chouzenoux, A. Benazza-Benyahia, and J.-C. Pesquet. An auxiliary variable method for Markov chain Monte-Carlo algorithms in high dimension. *Entropy*, 20(2):110, feb 2018. doi: 10.3390/e20020110.

M. I. J. Martin J Wainwright. *Graphical Models, Exponential Families, and Variational Inference*. Now Publishers Inc, Dec. 2008. ISBN 1601981848. URL https://www.ebook.de/de/product/8141638/martin_j_wainwright_michael_i_jordan_graphical_models_exponential_families_and_variational_inference.html.

S. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability*. Cambridge University Press, July 2014. ISBN 0521731828. URL https://www.ebook.de/de/product/8049012/sean_meyn_richard_l_tweedie_markov_chains_and_stochastic_stability.html.

A. Mira, R. Solgi, and D. Imparato. Zero variance Markov chain Monte Carlo for Bayesian estimators. *Stat. Comput.*, 23(5):653–662, 2013. ISSN 0960-3174. doi: 10.1007/s11222-012-9344-6. URL https://doi.org/10.1007/s11222-012-9344-6.

D. J. Mortlock, A. D. Challinor, and M. P. Hobson. Analysis of cosmic microwave background data on an incomplete sky. *Monthly Notices of the Royal Astronomical Society*, 330(2):405–420, feb 2002. doi: 10.1046/j.1365-8711.2002.05085.x.

C. Müller. *Spherical harmonics*. Springer-Verlag, Berlin New York, 1966. ISBN 9783540371748.

R. Neal. *Learning in Graphical Models*, chapter Suppressing Random Walks in Markov Chain Monte-Carlo Using Ordered Overrelaxation, pages 205–228. Springer, 1998.

F. Nielsen. Cramér-Rao lower bound and information geometry. In *Texts and Readings in Mathematics*, pages 18–37. Hindustan Book Agency, 2013. doi: 10.1007/978-93-86279-56-9_2.

C. J. Oates, M. Girolami, and N. Chopin. Control functionals for Monte Carlo integration. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(3):695–718, may 2016. doi: 10.1111/rssb.12185.

F. Orieux, O. Feron, and J.-F. Giovannelli. Sampling high-dimensional gaussian distributions for general linear inverse problems. *IEEE Signal Processing Letters*, 19:251–254, 2012. ISSN 1558-2361. doi: 10.1109/LSP.2012.2189104.

A. B. Owen. *Monte Carlo theory, methods and examples*. Work in progress, available on the author's web-site, 2013. URL https://statweb.stanford.edu/~owen/mc/.

O. Papaspiliopoulos and G. Roberts. Non-centered parameterisations for hierarchical models and data augmentation. *Bayesian Statistics*, 7:307–326, 01 2003.

O. Papaspiliopoulos, G. O. Roberts, and M. Sköld. A General Framework for the Parametrization of Hierarchical Models. *Statistical Science*, 22(1):59–73, Feb. 2007. ISSN 0883-4237. doi: 10.1214/088342307000000014. URL http://projecteuclid.org/euclid.ss/1185975637.

J. Papež, L. Grigori, and R. Stompor. Solving linear equations with messenger-field and conjugate gradient techniques: An application to CMB data analysis. *Astronomy and Astrophysics - A&A*, 620:A59, nov 2018. doi: 10.1051/0004-6361/201832987.

A. A. Penzias and R. W. Wilson. A measurement of excess antenna temperature at 4080 mc/s. *The Astrophysical Journal*, 142:419, jul 1965. doi: 10.1086/148307.

W. J. Percival and M. L. Brown. Likelihood techniques for the combined analysis of CMB temperature and polarization power spectra. *Monthly Notices of the Royal Astronomical Society*, 372(3):1104–1116, nov 2006. doi: 10.1111/j.1365-2966.2006.10910.x.

R. A. Polyak. *Introduction to Continuous Optimization*. Springer International Publishing, Apr. 2021. URL https://www.ebook.de/de/product/41244485/roman_a_polyak_introduction_to_continuous_optimization.html.

B. Racine, J. B. Jewell, H. K. Eriksen, and I. K. Wehus. Cosmological Parameters from CMB Maps without Likelihood Approximation. *The Astrophysical Journal*, 820(1):31, Mar. 2016. ISSN 1538-4357. doi: 10.3847/0004-637X/820/1/31. URL https://iopscience.iop.org/article/10.3847/0004-637X/820/1/31.

M. Reinecke. Libpsht – algorithms for efficient spherical harmonic transforms. *Astronomy & Astrophysics*, 526:A108, jan 2011. doi: 10.1051/0004-6361/201015906.

M. Riabiz, W. Chen, J. Cockayne, P. Swietach, S. A. Niederer, L. Mackey, and C. J. Oates. Optimal thinning of MCMC output. *arXiv 2005.03952*, May 2020.

C. Robert. *The Bayesian Choice*. Springer New York, Aug. 2007. ISBN 0387715983. URL https://www.ebook.de/de/product/6531494/christian_robert_the_bayesian_choice.html.

C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer New York, 2004. doi: 10.1007/978-1-4757-4145-2.

G. O. Roberts and J. S. Rosenthal. General state space Markov chains and MCMC algorithms. *Probability Surveys*, 1(none), jan 2004. doi: 10.1214/154957804100000024.

D. S. Seljebotn. Hemispherical power asymmetry in the cosmic microwave background by Gibbs Sampling. Master's thesis, Faculty of Mathematics and Natural Sciences,University of Oslo, 2010.

D. S. Seljebotn, T. Bærland, H. K. Eriksen, K.-A. Mardal, and I. K. Wehus. Multi-resolution bayesian CMB component separation through wiener filtering with a pseudo-inverse preconditioner. *Astronomy & Astrophysics*, 627:A98, jul 2019. doi: 10.1051/0004-6361/201732037.

G. J. Székely and M. L. Rizzo. A new test for multivariate normality. *Journal of Multivariate Analysis*, 93 (1):58–80, mar 2005. doi: 10.1016/j.jmva.2003.12.002.

M. A. Tanner and W. H. Wong. The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association*, 1987.

J. F. Taylor, M. A. J. Ashdown, and M. P. Hobson. Fast optimal CMB power spectrum estimation with Hamiltonian sampling. *Monthly Notices of the Royal Astronomical Society*, 389(3):1284–1292, Sept. 2008. ISSN 00358711, 13652966. doi: 10.1111/j.1365-2966.2008.13630.x. URL https://academic.oup.com/mnras/article-lookup/doi/10.1111/j.1365-2966.2008.13630.x.

M. Tegmark. How to measure CMB power spectra without losing information. *Physical Review D*, 55(10): 5895–5907, may 1997. doi: 10.1103/physrevd.55.5895.

M. Tegmark and A. de Oliveira-Costa. How to measure cmb polarization power spectra without losing information. *Phys.Rev.D64:063001,2001*, Dec. 2000. doi: 10.1103/PhysRevD.64.063001.

L. Tierney. Markov chains for exploring posterior distributions. *The Annals of Statistics*, 22(4), dec 1994. doi: 10.1214/aos/1176325750.

R. E. Upham, L. Whittaker, and M. L. Brown. Exact joint likelihood of pseudo-cℓ estimates from correlated gaussian cosmological fields. *Monthly Notices of the Royal Astronomical Society*, 491(3):3165–3181, nov 2019. doi: 10.1093/mnras/stz3225.

B. D. Wandelt, D. L. Larson, and A. Lakshminarayanan. Global, exact cosmic microwave background data analysis using Gibbs sampling. *Physical Review D*, 70(8):083511, Oct. 2004. ISSN 1550-7998, 1550-2368. doi: 10.1103/PhysRevD.70.083511. URL https://link.aps.org/doi/10.1103/PhysRevD.70.083511.

Whittaker. *Graphical Models in Applied Multi Statis*. John Wiley & Sons, Apr. 1990. ISBN 0471917508. URL https://www.ebook.de/de/product/3056053/whittaker_graphical_models_in_applied_multi_statis.html.

D. Wraith, M. Kilbinger, K. Benabed, O. Cappe, J.-F. Cardoso, G. Fort, S. Prunet, and C. P. Robert. Estimation of cosmological parameters using adaptive importance sampling. *Physical Review D*, 80 (2):023507, July 2009. ISSN 1550-7998, 1550-2368. doi: 10.1103/PhysRevD.80.023507. URL https://link.aps.org/doi/10.1103/PhysRevD.80.023507.

Y. Yu and X.-L. Meng. To Center or Not to Center: That Is Not the Question—An Ancillarity–Sufficiency Interweaving Strategy (ASIS) for Boosting MCMC Efficiency. *Journal of Computational and Graphical Statistics*, 20(3):531–570, Jan. 2011. ISSN 1061-8600, 1537-2715. doi: 10.1198/jcgs.2011.203main. URL http://www.tandfonline.com/doi/abs/10.1198/jcgs.2011.203main.

A. Zonca, L. Singer, D. Lenz, M. Reinecke, C. Rosset, E. Hivon, and K. Górski. healpy: equal area pixelization and spherical harmonics transforms for data on the sphere in python. *Journal of Open Source Software*, 4(35):1298, Mar. 2019. doi: 10.21105/joss.01298. URL https://doi.org/10.21105/joss.01298.

ECOLE
DOCTORALE
DE MATHEMATIQUES
HADAMARD

Titre: Algorithmes bayésiens pour la grande dimension, applications en cosmologie.

Mots clés: Bayésien, Statistiques, MCMC, Cosmologie, CMB, échantillonnage de Gibbs

Résumé: Une très faible lumière nous parvient depuis le ciel. Celle-ci n'est pas uniformément répartie sur la carte du ciel mais présente des anistropies. En analysant ces anistropies, nous pouvons déduire son spectre de puissance, ce qui nous permet de déduire les paramètres de l'univers. En supposant que le modèle statistique de génération de ces anistropies soit un modèle hiéarchique linéaire Gaussien et en ajoutant une distribution a priori sur les paramètres, nous pouvons faire de l'inférence Bayésienne sur ces paramètres. Ceci nous permet d'avoir non pas seulement un estimateur ponctuel des paramètres mais aussi des barres d'erreur sur ces quantités. Afin de mener à bien cette inférence, nous reprenons et développons l'échantillonneur de Gibbs utilisé jusque là dans la littérature sur l'analyse du fond diffus cosmologique. Nous proposons un moyen de raccourcir le temps de résolution d'un système en très grande dimension tout en gardant la distribution cible invariante. Nous proposons également un algorithme basé sur une variable auxiliaire pour contourner cette résolution. Finalement, en présentant les paramétrisations centrée et non centrée, nous utilisons une stratégie d'interweaving afin d'avoir un algorithme mélangeant bien sur l'ensemble du ratio signal sur bruit.

Le second projet concerne la compression des chaînes de MCMC. Sous-échantillonner une chaîne de Markov augmente toujours la variance asymptotique de l'estimateur obtenu. Nous voulons donc garder les points les plus représentatifs afin que cette variance asymptotique n'augmente pas trop. En utilisant une méthode d'échantillonnage pour des sondages et des "control variates", nous proposons une méthode en deux étapes afin de ne garder les points les plus représentatifs de la loi cible parmi une chaîne de MCMC: d'abord, nous utilisons des control-variates afin d'obtenir un estimateur s'écrivant comme une somme pondérée de la chaîne initiale. Ensuite, nous utilisons la méthode du cube afin de sous-échantillonner la chaîne pondérée obtenue à l'étape précédente. Nous proposons une façon de gérer les poids négatifs que la première étape peut donner. Nous proposons également deux façons d'avoir des control-variates: l'une, basée sur le "Stein trick" et la seconde, basée sur les control-variates de Gibbs. Ainsi, notre méthode ne nécessite pas la fonction de score.

Title: Bayesian algorithms for high dimension, application to cosmology.

Keywords: Bayesian, Statistics, MCMC, Cosmologie, CMB, Gibbs sampler

Abstract: We receive a faint light from the sky. This light is not uniform on the map of the sky but presents anisotropies. From these anisotropies, we can deduce its power spectrum, which in turn allows us to determine the cosmological parameters of the universe. Assuming the statistical model generating the sky map is a hierarchical linear Gaussian model and adding a prior distribution on the parameters, we can make Bayesian inference on these parameters. This allows us not only to have point estimates of the parameters, but also error bars on these quantities. In order to make this inference, we further develop the usual Gibbs sampler used in the CMB data analysis literature. We propose a way to shorten the resolution of a very high dimensional system while keeping the target distribution invariant. We also offer an algorithm based on an auxiliary variable to get around this resolution. Finally, using the concepts of centered and non centered parametrization, we use an interweaving strategy to have good mixing properties on the entire signal-to-noise ratio range.

The second project regards the compression of MCMC chains. Subsampling a Markov chain always increases the asymptotic variance of the resulting estimator. Hence we want to keep the points that are the most representative so that this variance does not increase too much. Using a survey sampling method and control variates, we propose a two steps procedure to keep the points that are the most representative of the target distribution out of a MCMC chain: first, we use control-variates in order to get an estimator which writes as a weighted sum of the chain. Then, we use the cube method to subsample the weighted chain we got at the end of the first step. We propose a way to deal with negative weights arising at the first step, which are incompatible with the cube method. We also provide two ways to build control-variates: one based on the Stein trick and the other one based on the Gibbs control variates. Hence, our method does not necessitate the availability of the score function.

Institut Polytechnique de Paris
91120 Palaiseau, France