N° d'ordre: 41447

Universite Lille 1 - Sciences et Technologies

Ecole Doctorale des Sciences pour l'Ingénieur

THÈSE

présentée en vue d'obtenir le grade de

DOCTEUR

Spécialité: Automatique, Génie informatique,
Traitement du signal et Image

Par

Thi Le Thu Nguyen

Doctorat délivré par l'Université Lille 1 - Sciences et Technologies

---

Sequential Monte Carlo Sampler for Bayesian Inference in Complex Systems

---

Soutenue le 3 juillet 2014 devant le jury d'examen

Mme Gersende Fort	Directrice de Recherche CNRS, LTCI	*Présidente du Jury*
M. François Desbouvries	Professeur, Télécom SudParis / SAMOVAR	*Rapporteur*
M. Jean-Yves Tourneret	Professeur, ENSEEIHT / IRIT	*Rapporteur*
Mme Lyudmila Mihaylova	Reader, University of Sheffield (UK)	*Examinateur*
M. Yves Delignon	Professeur, Télécom Lille / LAGIS	*Directeur de thèse*
M. François Septier	Maître de Conférences, Télécom Lille / LAGIS	*Co-encadrant*

Préparée au Laboratoire d'Automatique, Génie Informatique et Signal
LAGIS UMR CNRS 8219 et Télécom Lille
Ecole Doctorale SPI 072
PRES Université Lille - Nord de France

Sequential Monte-Carlo Sampler for Bayesian Inference in Complex Systems

Abstract In many problems, complex non-Gaussian and/or nonlinear models are required to accurately describe a physical system of interest. In such cases, Monte Carlo algorithms are remarkably flexible and extremely powerful to solve such inference problems. However, in the presence of high-dimensional and/or multimodal posterior distribution, standard Monte-Carlo techniques could lead to poor performance. In this thesis, the study is focused on Sequential Monte-Carlo Sampler, a more robust and efficient Monte Carlo algorithm. Although this approach presents many advantages over traditional Monte-Carlo methods, the potential of this emergent technique is however largely underexploited in signal processing. In this thesis, we therefore focus our study on this technique by aiming at proposing some novel strategies that will improve the efficiency and facilitate practical implementation of the SMC sampler. Firstly, we propose an automatic and adaptive strategy that selects the sequence of distributions within the SMC sampler that approximately minimizes the asymptotic variance of the estimator of the posterior normalization constant. Secondly, we present an original contribution in order to improve the global efficiency of the SMC sampler by introducing some correction mechanisms that allow the use of the particles generated through all the iterations of the algorithm (instead of only particles from the last iteration). Finally, to illustrate the usefulness of such approaches, we apply the SMC sampler integrating our proposed improvement strategies to two challenging practical problems: Multiple source localization in wireless sensor networks and Bayesian penalized regression.

Keywords: Statistical signal processing, Bayesian inference, Model Selection, Sequential Monte Carlo methods, Markov Chain Monte Carlo, Source localization, Wireless sensor network, Penalized regression.

Echantillonneur séquentiel de Monte-Carlo pour l'inférence Bayésienne dans des systèmes complexes

Résumé: Dans de nombreux problèmes, des modèles complexes non-Gaussiens et/ou non-linéaires sont nécessaires pour décrire précisément le système physique étudié. Dans ce contexte, les algorithmes de Monte-Carlo sont des outils flexibles et puissants permettant de résoudre de tels problèmes d'inférence. Toutefois, en présence de loi a posteriori multimodale et/ou de grande dimension, les méthodes classiques de Monte-Carlo peuvent conduire à des résultats non satisfaisants. Dans cette thèse, nous étudions une approche plus robuste et efficace: échantillonneur séquentiel de Monte-Carlo. Bien que cette approche présente de nombreux avantages par rapport aux méthodes traditionnelles de Monte-Carlo, le potentiel de cette technique est cependant très largement sous-exploité en traitement du signal. L'objectif de cette thèse est donc de proposer de nouvelles stratégies permettant d'améliorer l'efficacité de cet algorithme et ensuite de faciliter sa mise en œuvre pratique. Pour ce faire, nous proposons une approche adaptive qui sélectionne la séquence de distributions minimisant la variance asymptotique de l'estimateur de la constante de normalisation de la loi a posteriori. Deuxièmement, nous proposons un mécanisme de correction qui permet d'améliorer l'efficacité globale de la méthode en utilisant toutes les particules générées à travers toutes les itérations de l'algorithme (au lieu d'uniquement celles de la dernière itération). Enfin pour illustrer l'utilité de cette approche ainsi que des stratégies proposées, nous utilisons cet algorithme dans deux problèmes complexes: la localisation de sources multiples dans les réseaux de capteurs et la régression Bayésienne pénalisée.

Mots clés: Traitement statistique du signal, inférence bayésienne, Sélection de modèles, méthodes séquentielles de Monte-Carlo, Méthodes de Monte-Carlo par chaînes de Markov, Localisation de sources, Réseau de capteurs, Régression pénalisée.

Acknowledgements

To simply say thank you for the generosity of which I have been the fortunate recipient does not seem like enough. I am grateful for the advisors, teachers, colleagues, family, and friends who have supported me throughout the time that I have been at work on this dissertation.

First, I would like to express my sincere gratitude to my coadvisor, Dr. François Septier, for the guidance, support, and friendship he has offered me in all stages of my work. Without his simultaneous demand for and gentle yet insistent pushing toward intellectual rigor, bravery, and balance, I would not be able to finish this work.

I would like to thank Prof. Yves Delignon, my "directeur de these", who has been helping me with administrative procedures, and encouraged me to push myself even when I thought this process was impossible. I am grateful for his confidence and freedom he has given me to do this work.

Prof. François Desbouvries, Prof. Jean-Yves Tourneret, Prof. Gersende Fort, and Prof. Lyudmila Mihaylova have, as committee members, each contributed time, energy, and attention to this thesis for which I am incredibly grateful. My entire committee has been patient, engaged, and each member has invested tremendous amounts of time and care in my work.

Many other professors also deserve profound thanks for their directly or indirectly contributions to this thesis. At University of science Ho Chi Minh City, I have been fortunate to study with and work under the guidance of an amazing group of academics. I would like to express my deeply grateful to Prof. Nguyen Bac Van who encouraged my intellectual curiosity with a firm and generous hand and encouraged me to pursue an advanced education. I would also like to take this opportunity to express my gratitude to Prof. Richard Emilion and Prof. Pascal Omnes at PUF Master Program for their help, support and encouragement they have given to me. I feel fortunate to be one of their students. I feel lucky, as well, to have had collaborative works with Gareth Peters, who gave me many useful ideas from several discussions and I would like to convey a sincere thanks to him on this occasion.

I would not have survived this process without the friendship of my friends, fellow graduate students, colleagues in Lille and beyond. I would like to express my special thanks of gratitude to chi Nghi who has been constantly offered me help, advice and encouragement since the first day I contacted her. To anh Ha, be Hai, Nguyen, be Trang, be Nga, be Ngoc, Minh, Bao An and all the other friends: Seriously, thank you.

The unfailing and unflinching support and love from my family is most important for me to acknowledge. It is the most important thing. To my brother anh Hai, anh Ba, and Chi Tu - thank you is not enough, but thank you. Last but not least, my parents have shown unending patience, support and generosity. I love

you! Finally, I would like to dedicate this work to my fiancé Hoang Anh Tuan, for his love, encouragement, and for inspiring me to be brave, and to keep being brave. You are my own personal palimpsest. Believe me when I say this I couldn't have made it this far without you. I love you!

Acronyms and notations

Acronyms

AMIS	Adaptive multiple importance sampling
CESS	Conditional Effective sample size
CLT	Central limit theorem
CRB	Cramér-Rao bound
DOA	Direction of arrival
GLM	Generalized linear model
EM	Expectation maximization
EP	Exponential power distribution
ESS	Effective sample size
FIM	Fisher information matrix
IF	Importance function
i.i.d.	independent and identically distributed
IS	Importance sampling
KS	Kolmogorov-Smirnov
LASSO	Least absolute shrinkage and selection operator
MCMC	Markov Chain Monte Carlo
MH	Metropolis Hastings
MWG	Metropolis within Gibbs
MAP	Maximum a posteriori
ML	Maximum likelihood
MC	Monte-Carlo
MMSE	Minimum mean square error
MSE	Mean-squared error
OLS	Ordinary least-square
PCRB	Posterior Cramér-Rao bound
PMC	Population Monte Carlo
RSS	Residual sum of squares
ROI	Region of interest
RWM	Random walk Metropolis
SLLN	Strong law of large numbers
SNIS	Self-normalized importance sampling
SIR	Sampling importance resampling
SMC	Sequential Monte Carlo
TDOA	Time-delay of arrival
WSN	Wireless sensor network

Monte-Carlo algorithm notations

$\eta(\cdot)$	Importance (Proposal) function
w	Incremental importance weight
W	Importance weight
$\widetilde{W}$	Normalized importance weight
$\pi(\cdot)$	Normalized target distribution
$\gamma(\cdot)$	Unnormalized target distribution
Z	Normalizing constant of the target distribution
$\mathcal{K}(\cdot,\cdot)$	Mutation Kernel
$\mathcal{L}(\cdot,\cdot)$	Backward Kernel
N	Number of particles
$\pi^N(\cdot)$	Empirical approximation of the target based on the population of N particles
T	Number of iterations (SMC sampler)

Usual distributions

$\mathcal{U}(a,b)$	Uniform distribution in the interval $[a,b]$
$\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$	Multivariate Gaussian distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$
$\mathcal{IG}(a,b)$	Inverse gamma distribution with shape parameter a and scale parameter b
$\mathcal{S}\alpha\mathcal{S}(\alpha,\gamma)$	Symmetric α-stable distribution with characteristic exponent α and dispersion parameter γ
$\mathcal{P}o(\lambda)$	Poisson distribution with parameter λ

Contents

List of Figures

List of Tables

List of Algorithms

Introduction

The fundamental problem towards which the study of statistics is addressed, is that of inference. Some data are observed and we wish to make statements, *inferences*, about one or more unknown features of the physical system which gave rise to these data. The problem of inference has been the subject of considerable attention since the systematic study of probability theory began in the eighteen century. Many different theories of inference have been proposed, and there has hardly been a time when inference was not a matter of real controversy. The *classical* or *frequentist* was almost uncontested in the statistical community during the middle of twentieth century. Since about 1960 there has been a steady rival of interest in *Bayesian inference*, to the extent that the Bayesian approach is now a well-established alternative to classical inference.

Bayesian inference is a method of inference in which Bayes' rule is used to update the probability estimate for a hypothesis as additional evidence is acquired. Bayesian method briefly compromises the following principle steps.

- *Likelihood.* Obtain likelihood function, i.e., $p(\mathbf{y}|\boldsymbol{\theta})$. This step simply describes the process giving rise to the data $\mathbf{y}$ in terms of the unknown parameter $\boldsymbol{\theta}$.
- *Prior.* Obtain the prior density $p(\boldsymbol{\theta})$. The prior distribution expresses what is known about $\boldsymbol{\theta}$ prior to observing data.
- *Posterior* Apply Bayes's theorem to derive the posterior density $p(\boldsymbol{\theta}|\mathbf{y})$. This will now express what is known about $\boldsymbol{\theta}$ after observing data
- *Inference.* Derive appropriate inference statements from the posterior distribution. These will generally be designed to bring out the information expressed in the posterior distribution, and may include specific inferences such as point estimates, interval estimates or probabilities of hypotheses.

There are several potential difficulties in any practical implementation of Bayesian method. One of them is the issue of specifying the prior distribution. However, extra difficulties arise in actually calculating the various quantities required. First, in applying Bayes's theorem we need to compute the integral in the denominator. Second, the process of inference may require the calculation of further integrals of other operations on the posterior distribution. These calculation may be difficult to perform in practice, especially in complex problems with high-dimensional and/or multimodal posterior distribution. These integrals are typically approximated using Monte Carlo methods, requiring the ability to sample from general probability distributions which can generally be evaluated only up to a normalizing constant.

In many cases, using standard sampling techniques such as inversion or rejection to sample from a target distribution (i.e., posterior distribution) is not possible or

proves too much of a computational burden. This has led to the development in recent years of much more advances algorithms which allow one to obtain the required samples from this target distribution. Standard approaches are mostly based on Markov chain Monte Carlo (MCMC), where the equilibrium distribution of the chain is the target distribution and its ergodic mean converges to the expected value [Robert and Casella, 2004]. MCMC algorithms have been applied with success to many problems, e.g. [Septier and Delignon, 2011, Djuric and Chun, 2002, Dobigeon et al., 2014, Doucet and Wang, 2005]. However, there are two major drawbacks with MCMC methods. Firstly, it is difficult to assess when the Markov chain has reached its stationary regime of interest. Secondly, if the target distribution is highly multi-modal, MCMC algorithms can easily become trapped in local modes.

In recent years, more robust and efficient Monte Carlo algorithms have been established in order to efficiently explore high dimensional and multimodal spaces. Many of them are population based, in that they deal explicitly with a collection of samples at each iteration, including population-based MCMC [Liang and Wong, 2001, Jasra et al., 2007] and sequential Monte-Carlo sampler [Del Moral et al., 2006]. In [Jasra et al., 2007], the authors provide a detailed review as well as several illustrations showing that such population strategies can lead to significant improvement compared to standard MCMC techniques.

Population-based MCMC was originally developed by Geyer [Geyer, 1991]. Further advances came with an evolutionary Monte Carlo algorithm in [Liang and Wong, 2000, Liang and Wong, 2001] who attempted to produce genetic algorithm type moves to improve the mixing of the Markov chain. It works by simulating a population of several Markov chains with different invariant distributions in parallel using MCMC. The population is updated by mutation (Metropolis update in one single chain), crossover (partial states swapping between different chains), and exchange operators (full state swapping between different chains). However, like standard MCMC, this population-based MCMC algorithm still suffers of the difficulty to assess when the Markov chains have reached their stationary regime.

The second population-based simulation approach is the sequential Monte Carlo sampler proposed in [Del Moral et al., 2006]. Sequential Monte Carlo (SMC) methods is a class of sampling algorithms which combine importance sampling and resampling. They have been primarily used as "particle filter" to solve optimal filtering problems; see, for example, [Cappé et al., 2007] and [Doucet and Johansen, 2009] for recent reviews. In this context, SMC methods/particle filters have enjoyed wide-spread use in various applications (tracking, computer vision, digital communications) due to the fact that they provide a simple way of approximating complex filtering distribution sequentially in time. But in [Del Moral et al., 2006], the authors developed a general framework that allows SMC to be used to simulate from a single and static target distribution, thus becoming a promising alternative to standard MCMC methods. The SMC sampler framework involves the construction of a sequence of artificial distributions on spaces of increasing dimensions which admit the distributions of interests as particular marginals. The mecha-

nism is similar to sequential importance sampling (resampling) ([Liu, 2008] and [Doucet and De Freitas, 2001]), with one of the crucial differences being the framework under which the particles are allowed to move, resulting in differences in the calculation of the weights of the particles.

These methods have several advantages over traditional and population-based MCMC methods. Firstly, unlike MCMC, SMC methods do not require any burn-in period and do not face the sometimes contentious issue of diagnosing convergence of a Markov chain. Secondly, as discussed in [Jasra et al., 2007], compared to population-based MCMC, SMC sampler is a richer method since there is substantially more freedom in specifying the mutation kernels in SMC: kernels do not need to be reversible or even Markov (and hence time adaptive). Finally, unlike MCMC, SMC samplers provide an unbiased estimate of the normalizing constant of the posterior distribution which can be one quantity of interest in the inference problem to deal with.

Although this approach presents many advantages over traditional MCMC methods, the potential of these emergent techniques is however largely underexploited in signal processing. In this thesis, we therefore focus our study on this technique by aiming at proposing some novel strategies that will improve the efficiency and facilitate practical implementation of the SMC sampler. More specifically, we firstly derive some convergence results of the SMC sampler for some specific choice of the backward kernel used generally in practice as well as under a perfectly mixing forward kernel. These convergence results, derived for three variants of the SMC sampler (no resampling, resampling after the sampling and resampling before the sampling) facilitate the analysis of the SMC sampler and in particular highlight the impact of the choice of the sequence of target distributions on the algorithm performance. Then, by using these convergence results, we propose an adaptive strategy in order to obtain automatically choose the sequence of intermediate target distributions that optimizes the asymptotic variance of the estimator of the marginal likelihood. Finally, we present an other original contribution in order to improve the global efficiency of the SMC sampler. The idea developed in this thesis is to propose some correction mechanisms that allow the use of the particles generated through all the iterations of the algorithm (instead of only the particles from the last iteration) in order to improve the accuracy of the empirical approximation of the target distribution.

The usefulness of the SMC sampler as well as the improvement gain of the proposed strategies are finally demonstrated in the context of two challenging problems. Firstly, the localization problem of an unknown number of sources in wireless sensor networks with quantized data is tackled. Secondly, we derive an SMC sampler as solution for Bayesian model selection and parameter estimation in penalized regression models.

The dissertation is organized as follows. **Chapter 1** explains the objectives of Bayesian inference and reviews some of the most generic Monte Carlo techniques: importance sampling, Markov chain Monte-Carlo. Then, the SMC sampler is de-

scribed in more details with a presentation of the general principle as well as some discussion regarding the choices of the different quantities required in the practical implementations of the algorithm.

Chapter 2 presents the derivation of proposed variance reduction schemes for SMC sampler. We firstly study the asymptotic variance of the SMC samplers to understand the impact of the choice of the sequence of target distribution on the variance of the estimators obtained from the SMC sampler. From this asymptotic derivation, we propose a novel strategy that is able to automatically choose the sequence of distributions adapted to this specific inference problem to deal with. Finally, we propose original approaches that combine the simulated particles from all the different iterations in order to reduce the variance of the approximation of the target distribution. Performance of both proposed strategies are demonstrated empirically through numerical simulations.

Chapter 3 addresses the localization problem of unknown number of sources in wireless sensor networks with quantized data. After describing the system model, the proposed Bayesian solution based on the SMC sampler which integrates the proposed strategies described in Chapter 2 to enhance its efficiency. Furthermore, we derive the posterior Cramér-Rao bound which provides a theoretical performance limit for the Bayesian estimator of the locations as well as the transmitted powers of the multiple sources given the observations obtained at the fusion center. Performances of the proposed Bayesian solution are finally assessed in different scenarios.

Chapter 4 presents another application of SMC samplers in penalized regression model. We, firstly, presents an introduction to regression and to basic modeling. Secondly, we describe the well-known penalized methods (ridge, LASSO and the bridge regression) and their Bayesian formulation of the regression. Moreover, we propose a new class of priors based on α-stable family distribution in order to act as non-convex penalization for the regularization of the regression coefficients. After describing the class of regression models, the Generalized linear models that remove the Gaussian assumption of the observation noise, we describe our proposed Bayesian solution based on SMC samplers for the challenging problem of the joint model selection and parameter estimation.

We finish this manuscript by drawing some concluding remarks and by providing some interesting lines for further research for both the methodology and the applications addressed in this work.

Introduction en Français

Le problème fondamental dans le domaine des statistiques est celui de l'inférence. Par l'observation de certaines données, nous souhaitons apporter des conclusions, sur un ou plusieurs paramètres inconnus du système physique qui ont donné lieu à ces données. Le problème de l'inférence a fait l'objet d'une attention considérable depuis l'étude de la théorie des probabilités au dix-huitième siècle. Beaucoup de théories différentes d'inférence ont été proposées, et il n'y a guère eu un moment où l'inférence n'a pas été un sujet de réelle controverse. L'approche *classique* ou *fréquentiste* était presque incontestée dans la communauté statistique au milieu du XXe siècle. Cependant, depuis environ 1960, l'inférence bayésienne s'est posée en sérieux rival, dans la mesure où l'approche bayésienne est maintenant une alternative bien établie à l'inférence classique.

L'inférence bayésienne est une méthode d'inférence dans laquelle la règle de Bayes est utilisée afin de mettre à jour l'estimation de la probabilité d'une hypothèse lorsqu'une preuve supplémentaire est acquise. La méthode bayésienne consiste à suivre les principales étapes suivantes.

- *Vraisemblance.* Obtenir la fonction de vraisemblance, i.e., $p(\mathbf{y}|\boldsymbol{\theta})$. Cette étape décrit simplement le processus donnant lieu aux données observées $\mathbf{y}$ en fonction du paramètre inconnu $\boldsymbol{\theta}$.

- *A priori.* Obtenir la loi a priori $p(\boldsymbol{\theta})$. Cette distribution a priori permet d'exprimer ce qui est connu sur $\boldsymbol{\theta}$ avant même l'observation des données.

- *A posteriori* Appliquer le théorème de Bayes afin d'obtenir la loi a posteriori $p(\boldsymbol{\theta}|\mathbf{y})$ exprimant ce qui est connu sur $\boldsymbol{\theta}$ après l'observation des données.

- *Inférence.* Dériver les quantités d'intérêt depuis cette loi a posteriori, telles que les estimations ponctuelles, les estimations d'intervalle ou les probabilités d'hypothèses.

Néanmoins, plusieurs difficultés surviennent généralement lors de la mise en œuvre pratique de cette approche Bayésienne. L'une d'entre elles concerne le choix de la loi a priori des paramètres inconnus. Cependant, les plus grandes difficultés surgissent lors des calculs nécessaires à l'obtention des quantités d'intérêt. Tout d'abord, en appliquant le théorème de Bayes, nous devons obtenir le dénominateur qui est le résultat de l'intégrale du numérateur, soit le produit de la vraisemblance et de la loi a priori. Ensuite, le processus d'inférence peut nécessiter le calcul d'intégrales d'une fonction selon la distribution a posteriori. Tous ces calculs sont malheureusement difficiles à réaliser dans la pratique, en particulier pour des problèmes complexes avec une distribution a posteriori de grande dimension et / ou multimodale. Dans les faits, pour résoudre le problème d'inférence, ces intégrales

sont généralement estimées à l'aide de méthodes de type Monte-Carlo. Ces algorithmes permettent de générer des échantillons suivant une loi quelque conque connue seulement à une constante de proportionnalité près.

Parmi ces méthodes, les techniques simples d'échantillonnage telles que les méthodes d'inversion ou d'acceptation/rejet s'avèrent cependant inefficaces pour l'obtention d'échantillons depuis la distribution cible a posteriori. Cela a conduit le développement durant ces dernières années de techniques plus avancées. Parmi ces approches, les méthodes de Monte-Carlo par chaînes de Markov (MCMC) sont les plus populaires dans la littérature scientifique [Robert and Casella, 2004]. Ces approches MCMC sont des algorithmes produisant une chaîne de Markov ergodique de loi stationnaire la loi cible d'intérêt, soit ici la loi a posteriori. Ces méthodes MCMC ont été appliquées avec succès dans de nombreuses problématiques - par exemple [Septier and Delignon, 2011, Djuric and Chun, 2002, Dobigeon et al., 2014, Doucet and Wang, 2005]. Cependant, ces algorithmes possèdent deux inconvénients majeurs. Tout d'abord il est difficile d'évaluer quand la chaîne de Markov a atteint son régime stationnaire. Ensuite, si la distribution cible est multimodale, ces algorithmes sont facilement piégé dans un des modes de la distribution cible.

Au cours des dernières années, des algorithmes de type Monte Carlo plus robustes et efficaces ont été mis en place afin d'explorer efficacement un espace de dimension élevée et multimodal. La majorité d'entre eux sont basés sur l'utilisation et la propagation d'une population d'échantillons à chaque itération: les méthodes MCMC par population [Liang and Wong, 2001, Jasra et al., 2007] et les échantillonneur séquentiels de Monte-Carlo [Del Moral et al., 2006]. Dans [Jasra et al., 2007], les auteurs proposent une revue détaillée ainsi que plusieurs illustrations montrant que de telles stratégies de type population peuvent conduire à une amélioration significative des performances par rapport aux techniques standard MCMC.

Les méthodes MCMC de type population ont été développées par Geyer [Geyer, 1991]. D'autres propositions sont venues par la suite avec des algorithmes évolutionnaires de type Monte Carlo dans [Liang and Wong, 2000, Liang and Wong, 2001]. Dans ces stratégies, l'idée est de construire un algorithme MCMC basé sur des propositions de type génétique afin d'améliorer le mélange de la chaîne de Markov. Il fonctionne en simulant une population de plusieurs chaînes de Markov avec différentes distributions cibles invariantes en parallèle et en interaction. La population est ainsi mise à jour par mutation (mise à jour classique de type Metropolis d'une seule chaîne), par croisement (permutation partielle entre deux différentes chaînes), ou encore par des opérateurs d'échange (échange complet entre deux différentes chaînes). Cependant, comme les méthodes standard MCMC, il est toujours difficile d'évaluer quand ces différentes chaines de Markov ont atteint leur régime stationnaire.

La deuxième approche de simulation basée sur une population est l'échantillonneur séquentiel de Monte Carlo proposée dans [Del Moral et al., 2006]. Les méthodes séquentielles de Monte Carlo (SMC) forment une classe d'algorithmes d'échantillonnage qui combinent le principe d'échantillonnage par importance et le

rééchantillonnage. Ils ont été principalement utilisés, jusqu'à présent, pour résoudre des problèmes de filtrage optimal et sont appelés dans ce cas "filtre particulaire"; voir, par exemple, [Cappé et al., 2007] et [Doucet and Johansen, 2009] pour un panorama détaillé de ces méthodes. Dans ce contexte, les méthodes SMC / filtres particulaires ont été énormément utilisées dans diverses applications (suivi, vision par ordinateur, communications numériques) puisqu'elles offrent un moyen simple d'approcher une distribution complexe de filtrage séquentiellement. Mais dans [Del Moral et al., 2006], les auteurs ont développé un nouveau cadre général qui permet d'utiliser l'idée de ces méthodes SMC pour simuler des échantillons d'une distribution cible unique et statique, devenant ainsi une alternative prometteuse aux méthodes MCMC standard. Plus précisément, partant d'une loi cible d'intérêt, cette méthode consiste à créer artificiellement une suite de distributions cibles intermédiaires grâce à l'emploi d'un noyau markovien rétrograde; cette séquence de distributions intermédiaires est construite selon un principe de correction progressive: deux distributions consécutives ne diffèrent que de peu, et la complexité du problème est ainsi prise en compte de façon graduée, depuis la distribution cible initiale qui est est généralement très simple, jusqu'à la distribution finale d'intérêt qui elle concentre toute la complexité du système étudié. En définitive, la méthode peut être vue comme une façon de faire évoluer progressivement une population de particules qui tout d'abord suivent une loi simple, puis évoluent progressivement suivant des distributions intermédiaires de plus en plus complexes, jusqu'à finalement imiter au mieux la distribution finale d'intérêt.

Cette méthode présente plusieurs avantages par rapport aux méthodes traditionnelles MCMC. Tout d'abord, elle ne nécessite pas de période de "chauffe" (*burn-in*) et n'est pas confrontée à la question délicate du diagnostic de la convergence de la chaîne de Markov. Deuxièmement, cet échantillonneur SMC fournit une estimation non biaisée de la constante de normalisation de la distribution a posteriori qui peut être une quantité d'intérêt dans le problème d'inférence considéré, notamment dans un contexte de sélection de modèles. Enfin, la méthode est plus flexible car une grande liberté est offerte lors de la définition des deux noyaux de mutation: les hypothèses de réversibilité et même de markovianité peuvent être levées, et des noyaux adaptatifs peuvent plus facilement être employés.

Bien que cette approche présente de nombreux avantages par rapport aux méthodes MCMC, le potentiel de cette technique émergente est cependant largement sous-exploité en traitement du signal. Dans cette thèse, notre étude s'est concentrée sur cette technique en cherchant à proposer des stratégies novatrices qui permettent d'améliorer l'efficacité et de faciliter la mise en œuvre pratique de cet échantillonneur SMC. Plus précisément, nous étudions premièrement les résultats de convergence de l'échantillonneur SMC pour certains choix spécifiques du noyau rétrograde généralement utilisé en pratique ainsi que sous l'hypothèse d'un noyau avant idéal. Ces résultats de convergence, dérivés pour trois variantes de l'échantillonneur SMC (pas de rééchantillonnage, rééchantillonnage après l'échantillonnage et le rééchantillonnage avant l'échantillonnage) facilitent l'analyse de l'échantillonneur SMC et en particulier mettent en évidence l'impact du choix de la séquence de distribu-

tions cibles intermédiaires sur les performances finales de l'algorithme. En utilisant ces résultats de convergence, nous proposons ainsi une stratégie d'adaptation dans le but d'obtenir automatiquement la séquence de distributions cibles intermédiaires qui optimise la variance asymptotique de l'estimateur de la vraisemblance marginale. Enfin, nous présentons une autre contribution originale dans le but d'améliorer l'efficacité globale de l'échantillonneur SMC. L'idée développée dans cette thèse est de proposer des mécanismes de correction qui permettent d'utiliser l'ensemble des particules générées à travers toutes les itérations de l'algorithme (au lieu d'uniquement celles obtenues à la dernière itération) afin d'améliorer la précision de l'approximation empirique de la distribution cible.

L'utilité de l'échantillonneur SMC ainsi que le gain obtenu par les les différentes stratégies proposées dans cette thèse sont finalement illustrés dans le cadre de deux problèmes difficiles. Tout d'abord, nous abordons le problème de la localisation d'un nombre inconnu de sources dans un réseaux de capteurs sans fil avec des données quantifiées. Deuxièmement, nous utilisons un échantillonneur SMC comme solution Bayésienne pour la sélection de modèles et l'estimation des paramètres dans des problèmes de régression pénalisée.

Le manuscrit est organisé comme suit. Le **chapitre 1** explique les objectifs de l'inférence Bayésienne et décrit les principales techniques de type Monte-Carlo: échantillonnage d'importance, méthodes de Monte-Carlo par chaînes de Markov. Ensuite, l'échantillonneur SMC est décrit plus en détail avec une présentation du principe général suivie d'une discussion sur les choix des différentes quantités requises dans la mise en œuvre pratique de cet algorithme.

Le **chapitre 2** présente les différentes stratégies de réduction de variance pour l'échantillonneur SMC proposées dans le cadre de cette thèse. Afin de comprendre clairement l'impact du choix de la séquence de distributions intermédiaires sur les performances de l'algorithme, nous étudions les variances asymptotiques des estimateurs basés sur cette approche. De cette étude, nous proposons alors une nouvelle stratégie capable de choisir automatiquement la séquence de distributions intermédiaires en fonction du problème considéré. Enfin, nous proposons une approche originale qui combinent l'ensemble des particules simulées à travers les différentes itérations de l'algorithme de façon à réduire la variance de l'approximation empirique de la distribution cible. Les performances de ces deux stratégies proposées sont finalement démontrées empiriquement par plusieurs simulations numériques.

Le **chapitre 3** aborde le problème de la localisation d'un nombre inconnu de sources dans les réseaux de capteurs sans fil avec des données quantifiées. Après avoir décrit le modèle statistique approprié à ce système, la solution Bayésienne proposée est décrite. Celle-ci est basée sur l'emploi d'un échantillonneur SMC intégrant les stratégies proposées dans le chapitre 2 afin d'améliorer son efficacité. De plus, nous dérivons la borne de Cramér-Rao a posteriori qui fournit la limite inférieure de performance d'un estimateur Bayésien sur la position et la puissance émise de chaque source, compte-tenu des observations obtenues au centre de fusion. Les performances de la solution bayésienne proposée sont finalement évaluées dans

différents scénarios.

Le **chapitre 4** présente une autre application des échantillonneurs SMC pour l'inférence dans des modèles de régression pénalisée. Tout d'abord, nous décrivons le problème général de regression et sa modélisation généralement employée pour le résoudre. Ensuite, nous discutons des méthodes de résolution les plus connues (ridge, LASSO et bridge) et leur formulation Bayésienne équivalente. Une nouvelle classe de pénalisation non-convexe pour les coefficients de regression est introduite par la proposition d'une loi a priori α-stable. Enfin, nous décrivons la solution d'inférence basée sur l'utilisation d'échantillonneurs SMC proposée dans le cadre de cette thèse pour résoudre ce problème conjoint difficile de sélection de modèles et d'estimation des paramètres dans les problèmes de régression pénalisée.

Finalement, nous concluons ce manuscrit et nous discutons de quelques perspectives pour de futures directions de recherches.

Chapter 1

Bayesian inference and Monte Carlo Methods

Contents

This chapter explains the objectives of Bayesian inference and reviews some of the most generic Monte Carlo techniques: importance sampling, Markov chain Monte-Carlo. Then, we will describe population-based simulation techniques that have been established in order to obtain more robust and efficient Monte Carlo algorithms for efficiently exploring high dimensional and multimodal spaces. Population-based MCMC algorithm is firstly presented. Finally, the SMC sampler is described in more details with a presentation of the general principle as well as some discussion regarding the choices of the different quantities required in the practical implementations of the algorithm.

1.1 Bayesian analysis

The main purpose of statistical theory is to derive from observations of a random phenomenon an inference about the probability distribution underlying this phenomenon. So inference provides either an analysis of a past phenomenon, or some predictions about a future phenomenon of a similar nature. In other words, the goal in statistical inference is to make conclusion about a phenomenon based on observed

data. There are two major approaches to performing statistical inference, namely the Frequentist and Bayesian approach. In this thesis, we are only interested in the later approach and therefore will discuss in detail only the later one.

In Bayesian analysis, the main ingredients are prior, and likelihood distributions. The prior distribution represents our beliefs about the problem before obtaining the observations. On the other hand, the likelihood distribution gives us the probabilities of obtaining the data given a certain set of parameter values. The posterior distribution is calculated from the prior and the likelihood by using the Bayesian rule (Section 1.1.1) [Gelman et al., 2003], [Neal, 1993]. This posterior distribution represents our belief updated with the observations, in other words, the distribution of parameters conditioned on the observed data. As opposed to Frequentist approach, Bayesian statistics is concerned with generating the posterior distribution of the unknown parameters given both the data and some prior density for these parameters. As such, Bayesian statistics provides a complete picture of the uncertainty in the estimation of the unknown parameters.

1.1.1 Bayes's rule

As stated above, we are interested in the posterior distribution of the parameters, $p(\boldsymbol{\theta}|\mathbf{y})$, which is the conditional probability distribution of the unknown parameters $\boldsymbol{\theta} \in E$ given the observed data $\mathbf{y}$.

Assume that the vector random variable $(\boldsymbol{\theta}, \mathbf{y})$ has *joint probability density* function $p(\boldsymbol{\theta}, \mathbf{y})$, we can write

$$p(\boldsymbol{\theta}, \mathbf{y}) = p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}) \tag{1.1}$$

where $p(\boldsymbol{\theta})$ is the *prior distribution* and $p(\mathbf{y}|\boldsymbol{\theta})$ is the *likelihood function*. We here want to emphasize that the likelihood function, $p(\mathbf{y}|\boldsymbol{\theta})$, plays a very important role in the Bayes's formula. It is the function through which the data $\mathbf{y}$ modifies the prior knowledge of $\boldsymbol{\theta}$. The Bayesian rule is the expression which defines the posterior density as the normalized product of the prior density and the likelihood

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\boldsymbol{\theta}, \mathbf{y})}{p(\mathbf{y})} = \frac{p(\boldsymbol{\theta})p(\mathbf{y}|\boldsymbol{\theta})}{p(\mathbf{y})} \tag{1.2}$$

in which $p(\mathbf{y})$ is the *normalizing constant* of $p(\boldsymbol{\theta}|\mathbf{y})$. This quantity is often called the *marginal likelihood* or the *Bayesian Evidence* and is given by

$$p(\mathbf{y}) = \int_E p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta} \tag{1.3}$$

The calculation of $p(\mathbf{y})$ is crucial in Bayesian statistics because it can be used to statistically compare different models, specifically by the computation of Bayes factors or posterior model probabilities (see Section 1.1.3). However, in high dimensional parameter space, its computation is generally challenging as the posterior distribution may be multimodal and/or shows strong non-linear parameter-dependencies.

Combining equation (1.2) and (1.3), the Bayesian formula can be expressed as

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\boldsymbol{\theta})p(\mathbf{y}|\boldsymbol{\theta})}{\int p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}} \tag{1.4}$$

An equivalent form of equation (1.2) omits the factor $p(\mathbf{y})$, which does not depend on $\boldsymbol{\theta}$ and, with fixed $\mathbf{y}$, can thus be considered as a constant, yielding the *unnormalized posterior density*, which is the right side of (1.5)

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto p(\boldsymbol{\theta})p(\mathbf{y}|\boldsymbol{\theta}) \tag{1.5}$$

This simple expression encapsulates the technical core of Bayesian inference: the primary task of any specific application is to develop the model $p(\boldsymbol{\theta}, \mathbf{y})$ and perform the necessary computations to summarize $p(\boldsymbol{\theta}|\mathbf{y})$ in appropriate way.

From equation (1.5) it is clear that $p(\boldsymbol{\theta}|\mathbf{y})$ involves a contribution from the observed data through $p(\mathbf{y}|\boldsymbol{\theta})$, and a contribution from prior information quantified through $p(\boldsymbol{\theta})$. The posterior $p(\boldsymbol{\theta}|\mathbf{y})$ contains all relevant information on the unknown parameters $\boldsymbol{\theta}$ given the observed data $\mathbf{y}$. All statistical inference can be deduced from posterior distribution by reducing to the evaluation of the following integral

$$\mathbb{E}_{p(\boldsymbol{\theta}|\mathbf{y})}[\varphi(\boldsymbol{\theta})] = \int \varphi(\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y})d\boldsymbol{\theta} \tag{1.6}$$

of some function $\varphi(\boldsymbol{\theta})$ with respect to the posterior distribution. For example, point estimates for unknown parameters are given by the posterior mean, i.e., $\varphi(\boldsymbol{\theta}) = \boldsymbol{\theta}$; second moment matrix $\varphi(\boldsymbol{\theta}) = \boldsymbol{\theta}\boldsymbol{\theta}^T$, from which the posterior covariance matrix and posterior standard deviations may be computed, etc.

One of the main challenges in the practical implementation of Bayesian inference is the computation of $p(\mathbf{y})$, as we mentioned before. As we can see from equation (1.3), to evaluate $p(\mathbf{y})$ we have to perform the integration on a multidimensional region. This task is, in general, impossible to be done analytically and we, in most of the cases, have to numerically approximate it. However, classical numerical integration methods often fail. The most common approach to tackle this problem is to use Monte-Carlo methods for which the computation of $p(\mathbf{y})$ is not necessary in order to obtain samples from the distribution of interest, the posterior distribution (see section 1.2 in this chapter).

Let us now introduce some problems of interest that are directly related to our work in this thesis.

1.1.2 Bayesian Parameter Estimation

In many practical problems, we are generally interested in performing parameter estimation of the unknown parameter $\boldsymbol{\theta}$. The two most common criterion used to obtain a parameter estimate from the posterior distribution of interest are the Maximum a Posterior (MAP) criterion and the Minimum Mean Square Error (MMSE) or minimum variance estimator [O'Ruanaidh and Gerald, 1996].

The use of the MAP criterion consists in setting the mode of the posterior distribution, $p(\boldsymbol{\theta}|\mathbf{y})$, as the estimate of the unknown parameter $\boldsymbol{\theta}$:

$$\hat{\boldsymbol{\theta}}_{MAP} = \arg\max_{\boldsymbol{\theta}} \quad p(\boldsymbol{\theta}|\mathbf{y}) \tag{1.7}$$

For the MAP estimator, only the unnormalized posterior in equation (1.5) is required since $p(\mathbf{y})$ does not depend on $\boldsymbol{\theta}$.

The MMSE estimate which consists in finding the value of the unknown parameter that minimizes the mean squared error (MSE) is defined in terms of the trace of state error covariance as following

$$\begin{aligned} \hat{\boldsymbol{\theta}}_{MMSE} &= \arg\min_{\hat{\boldsymbol{\theta}}} \quad \mathbb{E}_{p(\boldsymbol{\theta}|\mathbf{y})}\left(\text{trace}\left[(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^T\right]\right) \\ &= \mathbb{E}_{p(\boldsymbol{\theta}|\mathbf{y})}[\boldsymbol{\theta}] \\ &= \int \boldsymbol{\theta} p(\boldsymbol{\theta}|\mathbf{y}) d\boldsymbol{\theta} \end{aligned} \tag{1.8}$$

Both MMSE and MAP methods required the estimation of the posterior distribution, $p(\boldsymbol{\theta}|\mathbf{y})$, but MAP does not require the calculation of the denominator (integration) and thereby less computational expensive; whereas the former requires full knowledge of the prior, likelihood and marginal likelihood. Note that however, MAP estimator has drawback especially in a high-dimension space. A narrow spike with very small width can have a very high density, but the actual probability of estimated parameter belonging to it is small. Hence, the width of the mode is as important as its height in the high-dimensional case. Moreover, finding the mode in such a case could be very challenging and clearly more difficult than computing the posterior mean.

1.1.3 Bayesian Model Selection

Consider the following general setting. Suppose there is a set of L models $\mathcal{M} = \{\mathcal{M}_1, \cdots, \mathcal{M}_L\}$ under consideration for data $\mathbf{y}$, and that under $\mathcal{M}_l$, $\mathbf{y}$ has density $p(\mathbf{y}|\boldsymbol{\theta}_l, \mathcal{M}_l)$ where parameters $\boldsymbol{\theta}_l$ is a vector of unknown parameters that indexes the members of $\mathcal{M}_l$. The Bayesian approach proceeds by assigning a prior probability distribution $p(\boldsymbol{\theta}_l|\mathcal{M}_l)$ to the parameters of each model, and a prior probability $p(\mathcal{M}_l)$ to each model.

The challenging interest is thus to find the model in $\mathcal{M}$ that most accurately represents the data according to some criterion of interest. In a Bayesian framework, we are particularly interested in the model posterior given by:

$$p(\mathcal{M}_l|\mathbf{y}) = \frac{p(\mathbf{y}|\mathcal{M}_l)p(\mathcal{M}_l)}{\sum_l p(\mathbf{y}|\mathcal{M}_l)p(\mathcal{M}_l)} \tag{1.9}$$

where

$$p(\mathbf{y}|\mathcal{M}_l) = \int p(\mathbf{y}|\boldsymbol{\theta}_l, \mathcal{M}_l)p(\boldsymbol{\theta}_l|\mathcal{M}_l)d\boldsymbol{\theta}_l \tag{1.10}$$

is the marginal likelihood of $\mathcal{M}_l$, also named the evidence of model $\mathcal{M}_l$. Based on these posterior probabilities, pairwise comparison of models, say $\mathcal{M}_1$ and $\mathcal{M}_2$, is summarized by the posterior odds

$$\frac{p(\mathcal{M}_1|\mathbf{y})}{p(\mathcal{M}_2|\mathbf{y})} = \frac{p(\mathbf{y}|\mathcal{M}_1)}{p(\mathbf{y}|\mathcal{M}_2)} \times \frac{p(\mathcal{M}_1)}{p(\mathcal{M}_2)} \tag{1.11}$$

This expression reveals how the data, through the Bayes factor $\frac{p(\mathbf{y}|\mathcal{M}_1)}{p(\mathbf{y}|\mathcal{M}_2)}$, updates the prior odds $\frac{p(\mathcal{M}_1)}{p(\mathcal{M}_2)}$ to yield the posterior odds.

The model posterior distribution $p(\mathcal{M}_l|\mathbf{y})$ is the fundamental object of interest for model selection in a Bayesian setting. Insofar as the priors $p(\boldsymbol{\theta}_l|\mathcal{M}_l)$ and $p(\mathcal{M}_l)$ provide an initial representation of model uncertainty, the model posterior summarizes all the relevant information in the data $\mathbf{y}$ and provides a complete post-data representation of model uncertainty. By treating $p(\mathcal{M}_l|\mathbf{y})$ as a measure of the "truth" of model $\mathcal{M}_l$, a natural and simple strategy for model selection is to choose the most probable $\mathcal{M}_l$, the one for which $p(\mathcal{M}_l|\mathbf{y})$ is the largest.

As stated above, all the needed information for Bayesian inference and decision is implicitly contained in the posterior. In large problems, where exact calculation of (1.9) and (1.10) is not feasible, Monte Carlo methods can often be used to extract such information by simulating an approximate sample from the posterior. Such samples can be used to estimate posterior characteristics or to explore the posterior, searching for models with high posterior probability.

1.2 Classical Monte Carlo Methods

1.2.1 Introduction

As mentioned in the previous section, in Bayesian analysis, we have generally to compute multidimensional integrals in the space of parameters to determine some quantities of interest such as model evidence, $p(\mathbf{y}|\mathcal{M}_k)$, or the expectation with respect to some known functions of the unknown parameters. These integrals, in general, does not have analytical forms. Therefore, it is crucial to have efficient numerical techniques to approximate them. Deterministic methods based on grid such as Gaussian quadrature or Simpson rule are potential solutions. However, all these deterministic numerical integration methods only work well for low dimensional spaces as their computational cost increases dramatically with the dimension of the problem [Ruanaidh et al., 1996]. As a consequence, for most applications, these techniques are inappropriate. Alternative solutions are Monte-Carlo methods which unlike previous solutions do not have this dimensional constraint [Doucet and De Freitas, 2001].

The term *Monte Carlo* (MC) method is normally expressed in a very general way- MC methods are stochastic methods; methods that involve sampling random numbers from probability distributions to investigate a certain problem. Monte

Carlo methods are mainly for solving two kinds of problems that often arise in statistical analysis: MC methods provide a way to generate samples from a given probability distribution. On the other hand they give a solution to the problem of estimating expectations of functions under some distribution and thus calculating numerical approximations for integrals. The first problem is a design problem, and the second one is an inference problem invoking integration. Several central issues are concerned in the Monte Carlo sampling:

1. **Consistency**: An estimator is *consistent* if the estimator converges to the true value almost surely as the number of samples approaches infinity.

2. **Unbiasedness**: An estimator is *unbiased* if its expected value is equal to the true value.

3. **Efficiency**: An estimator is *efficient* if it produces the smallest error covariance matrix among all unbiased estimator.

4. **Robustness**: An estimator is *robust* if it is insensitive to the gross measurement errors and the uncertainties of the model.

5. **Minimal variance**: Variance reduction is the central issue of Monte Carlo approximation methods, most improvement techniques are variance-reduction oriented.

The power of Monte Carlo techniques to solve high dimensional integrals has been utilized extensively throughout many fields. The reason why these techniques have been so successful is that they are not subject to any constraints on linearity or Gaussianity and hence prove to be very general in nature. The importance of the methods lies in the fact that one may consider difficult integrals as expectations since if we can decompose the integrand, say $h(\boldsymbol{\theta})$, into a product of a function $\varphi(\boldsymbol{\theta})$ and a probability density function $\pi(\boldsymbol{\theta})$ (which corresponds to the posterior distribution in Bayesian inference), the definite integral can be written as

$$J = \int h(\boldsymbol{\theta}) d\boldsymbol{\theta} = \int \varphi(\boldsymbol{\theta}) \pi(\boldsymbol{\theta}) d\boldsymbol{\theta} = \mathbb{E}_{\pi(\boldsymbol{\theta})} \left[\varphi(\boldsymbol{\theta}) \right] \tag{1.12}$$

and thus samples from the target distribution, $\pi(\boldsymbol{\theta})$, with respect to which the expectation is defined can be used to compute an approximation of the integral as a sample average. Therefore, Monte Carlo algorithms are remarkably flexible and extremely powerful to solve such integration approximation problems. Furthermore, convergence results for several key classes of Monte Carlo approximation techniques have been studied and are now well understood. This allows one to optimize Monte Carlo techniques and places them on a sound mathematical footing, which enables practitioners to be confident that the results obtained through application are mathematically consistent, logical and reproducible.

In this section we consider the mathematical background of Monte Carlo methods, explaining why Monte Carlo methods work in the problems stated above.

The objective of the Monte Carlo approach is to draw an i.i.d sample from the distribution of interest π which can then be used to compute sample averages as approximations to population averages. The basic idea behind Monte Carlo methods is that any probability measure, π, defined with respect to a measure space, E, can be approximated using the following empirical measure:

$$\pi^N(d\boldsymbol{\theta}) = \frac{1}{N}\sum_{m=1}^{N} \delta_{\boldsymbol{\theta}^{(m)}}(d\boldsymbol{\theta}) \tag{1.13}$$

where, $\left\{\boldsymbol{\theta}^{(m)}\right\}_{m=1}^{N}$, is a sequence of N i.i.d samples of law, π, and one assumes $\pi(d\boldsymbol{\theta})$ admits a density with respect measure denoted $\pi(\boldsymbol{\theta})$.

This approximation has led to wide-spread use of Monte Carlo techniques, specifically with respect to approximating difficult integrals. In what is known as "Perfect Monte Carlo Sampling", one can generate samples, $\left(\boldsymbol{\theta}^{(1)}, \cdots, \boldsymbol{\theta}^{(N)}\right)$, from the density, $\pi(\boldsymbol{\theta})$. Then these samples may be used to obtain an empirical average, which can be used as an approximation to the solution of the integral in equation (1.12),

$$\hat{J} = \mathbb{E}_{\pi^N}\left[\varphi(\boldsymbol{\theta})\right] = \frac{1}{N}\sum_{m=1}^{N} \varphi(\boldsymbol{\theta}^{(i)}) \tag{1.14}$$

Then, applying the Strong Law of Large Numbers (SLLN), it can be seen that $\hat{J}$ converges almost surely (π-a.s.) to $\mathbb{E}_{\pi}\left[\varphi(\boldsymbol{\theta})\right]$, for a suitable class of functions. The second thing to note is that when the second moment is finite, then not only it is known that a.s convergence applies but one can also obtain a rate of convergence of $\hat{J}$ to $\mathbb{E}_{\pi}\left[\varphi(\boldsymbol{\theta})\right]$, assuming that φ is an element of class of square integrable functions. The rate of convergence, in this case, is defined as the variance of $\varphi(\boldsymbol{\theta})$ which is empirically computed by

$$V_N = \mathbb{V}\mathrm{ar}_{\pi^N}(\varphi(\boldsymbol{\theta})) = \frac{1}{N}\sum_{m=1}^{N}\left[\varphi(\boldsymbol{\theta}^{(m)}) - \mathbb{E}_{\pi^N}\left[\varphi(\boldsymbol{\theta})\right]\right]^2 \tag{1.15}$$

Using this approximation and apply the Central Limit Theorem (CLT) we can prove the following convergence

$$\sqrt{N}\frac{\mathbb{E}_{\pi^N}\left[\varphi(\boldsymbol{\theta})\right] - \mathbb{E}_{\pi}\left[\varphi(\boldsymbol{\theta})\right]}{\sqrt{V_N}} \Rightarrow \mathcal{N}(0,1) \tag{1.16}$$

in which "$\Rightarrow$" denotes the convergence in distribution.

From the above convergence, one can obtain the confidence interval or confidence bounds for the estimator $\mathbb{E}_{\pi^N}$. However, we have to emphasize that the practicability of all these results relies heavily on the assumption that we can easily obtain the samples from the distribution of interest, $\pi(\boldsymbol{\theta})$. Unfortunately, drawing independent samples directly from the distribution of interest is generally not possible, so researchers have developed other techniques to utilize the framework of

Monte Carlo simulation, which only require the knowledge of the functional form of the density of interest, up to a normalizing constant. We assume that we do not know the density $\pi(\boldsymbol{\theta})$ exactly, but that we can evaluate $\pi(\boldsymbol{\theta})$ up to a normalizing constant. That is,

$$\pi(\boldsymbol{\theta}) = \frac{\gamma(\boldsymbol{\theta})}{Z} \propto \gamma(\boldsymbol{\theta}) \tag{1.17}$$

where $\gamma(\boldsymbol{\theta})$ is known or easy to compute but Z is unknown. Since in this thesis, we are interested in Bayesian inference, we thus have:

$$\gamma(\boldsymbol{\theta}) = p(\boldsymbol{\theta})p(\mathbf{y}|\boldsymbol{\theta}) \tag{1.18}$$

and

$$Z = \int_E \gamma(\boldsymbol{\theta})d\boldsymbol{\theta} = \int_E p(\boldsymbol{\theta})p(\mathbf{y}|\boldsymbol{\theta})d\boldsymbol{\theta} = p(\mathbf{y}) \tag{1.19}$$

The main challenge in Monte Carlo approximation is that direct sampling from the posterior distribution is generally not possible because these distributions are complex (multimodal and/or high-dimensional) and cannot be written in an analytic form. Fortunately, a variety of sampling methods have been developed to help in drawing the samples from complicated distributions. Instead of using direct i.i.d sampling, one typically draws sample from a "similar" distribution and utilizes a correction step in order to obtain samples from the target distribution. Let us now describe the classical Monte-Carlo techniques.

1.2.2 Rejection Sampling

The basic idea of rejection sampling, also called Accept-Reject, is to sample from a proposal distribution, $\eta(\cdot)$, and reject samples that are "unlikely" under the target distribution. This technique only requires the knowledge of the functional form of the target distribution up to a normalization. The first requirement is to determine a constant M such as

$$\gamma(\boldsymbol{\theta}) \leq M\eta(\boldsymbol{\theta}) \tag{1.20}$$

is true on the support of $\gamma(\boldsymbol{\theta})$ [Robert, 2004]. The Rejection sampling proceeds as shown in Algorithm 1.1.

Algorithm 1.1 Rejection sampling algorithm

1: Draw $\boldsymbol{\theta} \sim \eta(\cdot)$
2: Accept $\boldsymbol{\theta}$ as sample from $\pi(\cdot)$ with probability

$$\frac{\gamma(\boldsymbol{\theta})}{M\eta(\boldsymbol{\theta})}$$

3: otherwise go back to step 1.

The proof of this procedure to obtain sample from the target distribution $\pi(\cdot)$

is very simple. The distribution of the accepted samples is

$$\begin{aligned}\mathbb{P}(\boldsymbol{\theta} \in \Theta | \boldsymbol{\theta} \text{ is accepted}) &= \frac{\mathbb{P}(\boldsymbol{\theta} \in \Theta \text{ and } \boldsymbol{\theta} \text{ is accepted})}{\mathbb{P}(\boldsymbol{\theta} \text{ is accepted})} \\ &= \frac{\frac{1}{M}\int_{\Theta} \eta(\boldsymbol{\theta}) \frac{\gamma(\boldsymbol{\theta})}{M\eta(\boldsymbol{\theta})} d\theta}{\frac{1}{M}\int_{E} \eta(\boldsymbol{\theta}) \frac{\gamma(\boldsymbol{\theta})}{M\eta(\boldsymbol{\theta})} d\theta} \\ &= \frac{\frac{Z}{M}\int_{\Theta} \pi(\boldsymbol{\theta}) d\theta}{\frac{Z}{M}\int_{E} \pi(\boldsymbol{\theta}) d\theta} \\ &= \int_{\Theta} \pi(\boldsymbol{\theta}) d\theta \end{aligned} \tag{1.21}$$

which proves the required result.

The main limitation of this technique is that we might be unable to bound the density of interest $\gamma(\boldsymbol{\theta})$ by $M\eta(\boldsymbol{\theta})$, where $\eta(\boldsymbol{\theta})$ is a density that we can easily sample from. As a consequence, researchers have developed other techniques to utilize in the framework of Monte-Carlo algorithms. The first of these techniques to be discussed is *importance sampling* (IS).

1.2.3 Importance Sampling

In rejection sampling we have compensated for the fact that we sampled from the proposal distribution $\eta(\boldsymbol{\theta})$ instead of $\pi(\boldsymbol{\theta})$ by rejecting some of the values proposed by $\eta(\boldsymbol{\theta})$. Importance sampling is based on the idea of using weights to correct for the fact that we sample from the instrumental distribution $\eta(\boldsymbol{\theta})$ instead of the target distribution $\pi(\boldsymbol{\theta})$.

As in rejection sampling, Importance Sampling also requires to choose a *proposal distribution* or *importance distribution* $\eta(\boldsymbol{\theta})$ that we can easily sample from. The condition for $\eta(\boldsymbol{\theta})$ is: wherever $\pi(\boldsymbol{\theta}) > 0, \eta(\boldsymbol{\theta})$ is also greater than zero. It implies that the support of $\eta(\boldsymbol{\theta})$ must contain the support $\pi(\boldsymbol{\theta})$. This is a weaker condition compared with the condition of rejection sampling. In importance sampling, we do not need to reject the samples from $\eta(\boldsymbol{\theta})$ with some probability, instead, we give them corresponding weights, hence, it is easy to implement. Equation (1.22) shows the principle of importance sampling.

$$\mathbb{E}_{\pi}\left[\varphi(\boldsymbol{\theta})\right] = \int \varphi(\boldsymbol{\theta})\pi(\boldsymbol{\theta})d\boldsymbol{\theta} = \int \varphi(\boldsymbol{\theta})\frac{\pi(\boldsymbol{\theta})}{\eta(\boldsymbol{\theta})}\eta(\boldsymbol{\theta})d\boldsymbol{\theta} = \mathbb{E}_{\eta}\left[\varphi(\boldsymbol{\theta})\frac{\pi(\boldsymbol{\theta})}{\eta(\boldsymbol{\theta})}\right] \tag{1.22}$$

$\frac{\pi(\boldsymbol{\theta})}{\eta(\boldsymbol{\theta})}$ is the *importance weights* function in order to correct for the fact that these samples were not taken from the distribution of interest, $\pi(\boldsymbol{\theta})$, but instead from the importance distribution, $\eta(\boldsymbol{\theta})$. Let $W^{(c)} = \frac{\pi(\boldsymbol{\theta}^{(c)})}{\eta(\boldsymbol{\theta}^{(c)})}$, then

$$\mathbb{E}_{\pi_{IS}^{N}}\left[(\varphi(\boldsymbol{\theta}))\right] = \frac{1}{N}\sum_{c=1}^{N} \varphi(\boldsymbol{\theta}^{(c)})W^{(c)}(\boldsymbol{\theta}^{(c)}) \tag{1.23}$$

and the target distribution, $\pi(\boldsymbol{\theta})$, can be approximated using the following empirical measure:

$$\pi_{IS}^{N}(d\boldsymbol{\theta}) = \frac{1}{N}\sum_{c=1}^{N} W^{(c)}\delta_{d\boldsymbol{\theta}^{(c)}}(\boldsymbol{\theta}) \tag{1.24}$$

where $(\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \ldots, \boldsymbol{\theta}^{(N)})$ are i.i.d. samples from $\eta(\boldsymbol{\theta})$. To get better approximation (i.e. reducing the variance of the importance weights), the proposal distribution $\eta(\boldsymbol{\theta})$ should be close to $\pi(\boldsymbol{\theta})$ as possible, when they are equal, the weights are always 1 and it is the same as to draw samples directly from $\pi(\boldsymbol{\theta})$.

If the normalizing constant of either the target distribution π or the importance function η are unknown, an alternative to $\mathbb{E}_{\pi_{IS}^{N}}$ is the self-normalized importance sampling (SNIS) estimator, that is

$$\mathbb{E}_{\pi_{SNIS}^{N}(\boldsymbol{\theta})}\left[\varphi(\boldsymbol{\theta})\right] = \sum_{c=1}^{N}\varphi(\boldsymbol{\theta}^{(c)})\widetilde{W}^{(c)} \tag{1.25}$$

and the approximation of the posterior is as followed

$$\pi_{SNIS}^{N}(d\boldsymbol{\theta}) = \sum_{c=1}^{N}\widetilde{W}^{(c)}\delta_{\boldsymbol{\theta}^{(c)}}(d\boldsymbol{\theta}) \tag{1.26}$$

where $\widetilde{W}^{(c)} = \frac{W^{(c)}}{\sum_{j=1}^{N} W^{(j)}}$ is the normalized weight in which $W^{(c)} = \frac{\gamma(\boldsymbol{\theta}^{(c)})}{\eta(\boldsymbol{\theta}^{(c)})}$ is the unnormalized weight. It has been shown that when the sample size N increases, the estimate of $\mathbb{E}_{\pi}(\varphi(\boldsymbol{\theta}))$ obtained by the above IS algorithm converges to the mean of the posterior [Robert, 2004]. Algorithm 1.2 shows the algorithm of importance sampling

We can use the same importance distribution, η, to estimate the normalizing constant, Z, in equation (1.19) for different values of $\boldsymbol{\theta}$. The importance sampling estimator of Z is based on the identity

$$Z = \mathbb{E}_{\eta}\left[\frac{\gamma(\boldsymbol{\theta})}{\eta(\boldsymbol{\theta})}\right] \tag{1.27}$$

and the corresponding unbiased Monte Carlo estimator is

$$\hat{Z}_{IS}^{N} = \frac{1}{N}\sum_{c=1}^{N}\frac{\gamma(\boldsymbol{\theta}^{(c)})}{\eta(\boldsymbol{\theta}^{(c)})} \tag{1.28}$$

Importance sampling can be used to draw samples from both uni-variate and multivariate distributions, also because of its simplicity. However, the variability of the importance weights may strongly affect the accuracy of the estimate. In the case that these weights vary widely, only few points with the highest weights effectively contribute to the estimate. This problem causes the subsequent estimations to be inaccurate. This situation may arise when the importance function generates samples that are concentrated on regions with low probability mass under the target

Algorithm 1.2 (Self-normalized) Importance sampling algorithm

1: **for** $c = 1, \ldots, N$ **do** `# IS Iterations`
2: Generate $\boldsymbol{\theta}^{(c)} \sim \eta(\boldsymbol{\theta})$.
3: Set $W^{(c)} = \dfrac{\gamma(\boldsymbol{\theta}^{(c)})}{\eta(\boldsymbol{\theta}^{(c)})}$
4: **end for**
5: Compute normalized weights
6: **for** $c = 1, \ldots, N$ **do**
7: $\widetilde{W}^{(c)} = \dfrac{W^{(c)}}{\sum_{j=1}^{N} W^{(j)}}$
8: **end for**

posterior. In such cases, the IS approximation is not very robust owing to the high discrepancy between the target density and the importance density. Therefore, for importance sampling to work well, the *importance distribution* (or *proposal distribution*) defined by $\eta(\boldsymbol{\theta})$ must really be a fairly good approximation to that defined by $\pi(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\mathbf{y})$, so that the ratio $\frac{\pi(\boldsymbol{\theta})}{\eta(\boldsymbol{\theta})}$ does not vary widely. When $\boldsymbol{\theta}$ is high - dimensional, and $\pi(\boldsymbol{\theta})$ is complex, and perhaps multimodal, finding a good importance sampling distribution can be very difficult and could be challenging since the certain required information about the target distribution is usually not available, limiting the applicability of the methods.

1.2.4 Sampling Importance Resampling

The Sampling Importance Resampling (SIR), in [Rubin, 1987] and [Rubin, 1988], is an extension of the IS method that achieves simulation from π by resampling rather than by simple reweighting. The resampling step is aimed to eliminate the samples with small importance weights and duplicate the samples with big weights. More precisely, the SIR algorithm is held in two steps: the first step is similar to IS and consists in generating i.i.d samples $\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(N)}$ from η, the second step builds samples from π, $\tilde{\boldsymbol{\theta}}^{(1)}, \ldots, \tilde{\boldsymbol{\theta}}^{(N)}$ obtained by using a resampling procedure of the instrumental samples $\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(N)}$.

In order to quantify the performance of importance distribution η, the *effective sample size* $\mathbb{ESS}$ (in [Liu and Chen, 1998]) was designed to provide a measure of how much the importance distribution η differs from the target distribution π and is given by

$$\mathbb{ESS} = \left[\sum_{c=1}^{N} (\widetilde{W}^{(c)})^2\right]^{-1} = \frac{\left(\sum_{j=1}^{N} W^{(j)}\right)^2}{\sum_{c=1}^{N} (W^{(c)})^2} \tag{1.29}$$

The ESS is N if all weights are equal and is 1 if all mass is concentrated in a single particle. A small effective sample size indicates severe degeneracy of the algorithm. To reduce the degeneracy present in this algorithm, the resampling criterion that is commonly used, is to resample only when the effective sample size drops below some threshold. The particles are resampled according to their importance weights from which the particles with larger weights become better represented in the resampled

population than those with smaller weights and those particles with sufficient small weights, which poorly approximate π, may be eliminated.

There exist several efficient recipes to implement the resampling step like residual sampling ([Liu and Chen, 1998]) and stratified resampling ([Kitagawa, 1996]) that reduce the variance of the resulting estimators. In this thesis, we use the multinomial resampling ([Gordon et al., 1993]) since this is the simplest approach in which the normalized weights of the particles are used as probabilities of a multinomial distribution. More precisely, a (non-i.i.d) sample from π, $\tilde{\boldsymbol{\theta}}^{(1)}, \cdots, \tilde{\boldsymbol{\theta}}^{(M)}$, can be asymptotically derived from the instrumental sample $\boldsymbol{\theta}^{(1)}, \cdots, \boldsymbol{\theta}^{(N)}$ by resampling using the normalized weights $\widetilde{W}^{(1)}, \cdots, \widetilde{W}^{(N)}$, that is,

$$\tilde{\boldsymbol{\theta}}^{(i)} = \boldsymbol{\theta}^{(J_i)}, \qquad 1 \leq i \leq M,$$

where the random variables $(J_1, \ldots, J_M)$ are i.i.d conditionally on $\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(N)}$ and distributed as

$$\mathbb{P}\left[J_l = c | \boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(N)}\right] = \widetilde{W}^{(c)} = \left(\sum_{j=1}^{N} \frac{\pi(\boldsymbol{\theta}^{(j)})}{\eta(\boldsymbol{\theta}^{(j)})}\right)^{-1} \frac{\pi(\boldsymbol{\theta}^{(c)})}{\eta(\boldsymbol{\theta}^{(c)})}$$

By using SIR, any given sample from an importance distribution η can be asymptotically transformed into a sample of points marginally distributed from the target distribution π, that is

$$\pi_{SIR}^{M}(d\boldsymbol{\theta}) = \frac{1}{M} \sum_{j=1}^{M} \delta_{\tilde{\boldsymbol{\theta}}^{(j)}}(d\boldsymbol{\theta}) \tag{1.30}$$

The approximation of $\mathbb{E}_{\pi}[\varphi(\boldsymbol{\theta})]$ is then

$$\mathbb{E}_{\pi_{SIR}^{M}}[\varphi(\boldsymbol{\theta})] = \frac{1}{M} \sum_{j=1}^{M} h(\tilde{\boldsymbol{\theta}}^{(j)})$$

which almost surely converges to $\mathbb{E}_{\pi}[\varphi(\boldsymbol{\theta})]$ since each $\tilde{\boldsymbol{\theta}}^{(i)}$ is (marginally and asymptotically) distributed from π.

Although resampling reduces the effects of degeneracy on the sample approximation, by the construction, the variance of the SIR estimator is greater than the variance of the SNIS estimator. Additionally, an asymptotic analysis of $\mathbb{E}_{\pi_{SIR}^{M}}(\varphi(\boldsymbol{\theta}))$ is quite delicate because of the dependencies in the SIR algorithm introduced by the resampling step. In addition, using resampling step may lead to a poor estimate in some cases. For example, when only few particles have non-zero weights, the resulting set of resampled particles will contain many repeated samples, thus loosing the diversity of the set of particles. This phenomenon is known as *sample impoverishment*.

If the dimension of $\boldsymbol{\theta}$ is large, importance sampling and this SIR variant are typically inefficient since the design a good proposal distribution becomes really

challenging. Thus, it is necessary to use alternative methods such as Markov chain Monte-Carlo algorithms (MCMC).

1.2.5 Markov Chain Monte Carlo

MCMC algorithms [Gamerman and Lopes, 2006], [Neal, 1993] are used to solve problems in many scientific fields, including for examples physics (where many MCMC algorithms originated) and signal processing [Septier and Delignon, 2011, Djuric and Chun, 2002, Dobigeon et al., 2014, Doucet and Wang, 2005]. The widespread popularity of MCMC samplers is largely due to their impact on solving complex statistical computation problems related to Bayesian inference [Gelfand and Smith, 1990], [Gamerman and Lopes, 2006]. MCMC provides the ability of getting Bayesian estimates for analytically intractable posterior distributions, only known up to a normalization.

MCMC is a general method based on drawing values of $\boldsymbol{\theta}$ from some proposal distributions and then correcting those draws to approximate the target posterior distribution, π. The samples are drawn sequentially. The distribution of the current sample depends only on the value of samples from the previous drawn; hence, the samples form a Markov chain. Therefore, those samples can be used to approximate the posterior distribution and obtain highly accurate approximations of Bayesian estimates. To understand how Markov chain Monte Carlo works, there are some basic concepts need to be known.

Definition 1.1 *A sequence of random variables* $\boldsymbol{\theta}^{(0)}, \boldsymbol{\theta}^{(1)}, \cdots, \boldsymbol{\theta}^{(i-1)}, \boldsymbol{\theta}^{(i)}, \cdots$ *forms a Markov chain if* $p(\boldsymbol{\theta}^{(i)}|\boldsymbol{\theta}^{(i-1)}, \boldsymbol{\theta}^{(i-2)}, ..., \boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(0)}) = p(\boldsymbol{\theta}^{(i)}|\boldsymbol{\theta}^{(i-1)}), \forall i = 1, 2, \cdot$, *i.e., the probability density function of one random variable only depends on the previous variable in the sequence.*

A Markov chain can be specified by giving the marginal distribution, $p_0(\boldsymbol{\theta})$, for $\boldsymbol{\theta}^{(0)}$ – the initial probability density function – and the conditional distribution for $\boldsymbol{\theta}^{(i+1)}$ given $\boldsymbol{\theta}^{(i)}$ – the *transition distribution* , $\mathcal{K}(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i+1)})$, for one state to follow another state. A Markov chain has a *stationary* transition probability when transition probability distributions are the same for all time steps.

Definition 1.2 *For a Markov chain with a stationary transition probability distribution* $\mathcal{K}(\boldsymbol{\theta}, \boldsymbol{\theta}')$, *if there exists a distribution* $\pi(\boldsymbol{\theta}')$ *such that* $\pi(\boldsymbol{\theta}') = \int \mathcal{K}(\boldsymbol{\theta}, \boldsymbol{\theta}')\pi(\boldsymbol{\theta})d\boldsymbol{\theta}$, *then* $\pi(\boldsymbol{\theta}')$ *is a stationary distribution of the Markov chain.*

The transition probability distributions must be constructed so that the Markov chain converges to the unique stationary distribution that is,$\pi(\boldsymbol{\theta})$. If the stationary distribution $\pi(\boldsymbol{\theta})$ is unique, then,

$$\mathbb{E}_{\pi^N_{MCMC}}[\varphi(\boldsymbol{\theta})] = \frac{1}{N}\sum_{i=1}^{N}\varphi(\boldsymbol{\theta}^{(i)}) \tag{1.31}$$

where $(\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \cdots, \boldsymbol{\theta}^{(N)})$ forms the Markov chain with $\pi(\boldsymbol{\theta})$ as the stationary distribution. This estimate in Eq. (1.31) can be regarded as an ergodic average and convergence to the required expectation is ensured by the ergodic theorem.

The key of MCMC algorithms is thus to create a Markov process whose stationary distribution is the specified π and run the simulation long enough that the distribution of current draws is close enough to this stationary distribution. Since the marginal distribution of a Markov chain starting from an arbitrary initial state, takes time to converge to the stationary distribution, initial MCMC draws are discarded: this is called the burn-in period. For example, if there are N samples of $\boldsymbol{\theta}$ in the Markov chain, which has a stationary distribution $\pi(\boldsymbol{\theta})$, and the number of burn-in samples is k, then the effective MCMC sample is the last $N-k$ draws, and the Monte Carlo approximation in Equation (1.12) should be changed to:

$$\mathbb{E}_{\pi_{MCMC}^{N-k}}[\varphi(\boldsymbol{\theta})] = \frac{1}{N-k}\sum_{i=k+1}^{N}\varphi(\boldsymbol{\theta}^{(i)}) \tag{1.32}$$

There exists several methods of constructing a Markov chain which has as its stationary distribution the required target distribution. However, all of them can be viewed as special cases of the general framework established by Metropolis and Hastings [Metropolis et al., 1953, Hastings, 1970].

1.2.5.1 Metropolis– Hastings

The Metropolis–Hastings (MH) algorithm was first developed by [Metropolis et al., 1953] and then later extended by [Hastings, 1970]. The *Metropolis–Hasting* is a term for a family of Markov chain simulation methods that are useful for drawing samples from general distributions, such as the posterior distribution.

In Metropolis-Hasting sampling, we need to choose a proposal distribution $\eta(\boldsymbol{\theta}'|\boldsymbol{\theta})$ for creating the Markov chain. This is a conditional distribution which depends on the last element in the Markov chain. At the i-th iteration of the algorithm, we draw a sample $\boldsymbol{\theta}^*$ given $\boldsymbol{\theta}^{(i-1)}$ from the conditional distribution $\eta(\boldsymbol{\theta}|\boldsymbol{\theta}^{(i-1)})$, and similar to rejection sampling [Robert and Casella, 2004], the sample will be accepted, i.e. $\boldsymbol{\theta}^{(i)} = \boldsymbol{\theta}^*$ with the acceptance probability $\alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}^{(i-1)})$ given by:

$$\alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}^{(i-1)}) = \min\left\{1, \frac{\pi(\boldsymbol{\theta}^*)\eta(\boldsymbol{\theta}^{(i-1)}|\boldsymbol{\theta}^*)}{\pi(\boldsymbol{\theta}^{(i-1)})\eta(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(i-1)})}\right\} \tag{1.33}$$

Algorithm 1.3 summarizes the Metropolis-Hastings algorithm. But unlike acceptance rejection sampling, if the sample is rejected, we do not draw a new sample, instead, we let $\boldsymbol{\theta}^{(i)} = \boldsymbol{\theta}^{(i-1)}$ and move to the next time step. In other words, in rejection sampling, rejected points are discarded and have no influence on the list of samples $\left\{\boldsymbol{\theta}^{(i)}\right\}$ that we collected, whereas here, a rejection causes the current state to be kept into the final collection of samples that approximates the target distribution.

Note here that the acceptance ratio (1.33) is independent of the normalizing constant for π. This makes the approach applicable to problems in which the target density (or also the proposal density) is known only up to a constant of normalization.

Algorithm 1.3 Metropolis-Hasting algorithm

1: Initialization: set $\boldsymbol{\theta}^{(0)}$
2: **for** $i = 1, \ldots, N$ **do** `# MCMC Iterations`
3: Draw $\boldsymbol{\theta}^*$ from $\eta(\boldsymbol{\theta}|\boldsymbol{\theta}^{(i-1)})$
4: Compute the acceptance ratio:

$$\alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}^{(i-1)}) = \min\left\{1, \frac{\pi(\boldsymbol{\theta}^*)\eta(\boldsymbol{\theta}^{(i-1)}|\boldsymbol{\theta}^*)}{\pi(\boldsymbol{\theta}^{(i-1)})\eta(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(i-1)})}\right\}$$

5: Sample random variate $U^{(i)}$ from $\mathcal{U}(0,1)$
6: **if** $U^{(i)} \leq \alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}^{(i-1)})$ **then**
7: $\boldsymbol{\theta}^{(i)} = \boldsymbol{\theta}^*$
8: **else**
9: $\boldsymbol{\theta}^{(i)} = \boldsymbol{\theta}^{(i-1)}$
10: **end if**
11: **end for**

The algorithm produces a reversible Markov chain which has the required target distribution, $\pi(\boldsymbol{\theta})$. Using the Metropolis-Hastings algorithm, the transition kernel of the Markov chain is

$$\begin{aligned}\mathcal{K}(\boldsymbol{\theta}^{(i-1)}, \boldsymbol{\theta}^{(i)}) &= \eta(\boldsymbol{\theta}^{(i)}|\boldsymbol{\theta}^{(i-1)})\alpha(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i-1)}) \\ &\quad + \left[1 - \int_E \eta(\boldsymbol{z}|\boldsymbol{\theta}^{(i-1)})\alpha(\boldsymbol{z}, \boldsymbol{\theta}^{(i-1)})d\boldsymbol{z}\right] \mathbb{1}(\boldsymbol{\theta}^{(i-1)} = \boldsymbol{\theta}^{(i)})\end{aligned} \tag{1.34}$$

where $\mathbb{1}(\cdot)$ is the indicator function, the first term represents the acceptance probability of the proposed state and the second term corresponds to the rejection probability of the proposed state which cannot typically be computed analytically. Many proposal distribution can be chosen but bad choice can lead to slow mixing of the chain and long burn-in times. Many versions of the algorithm have been developed - each having different strategies to explore the state's space. Among them, *random walk Metropolis*(RWM) chain, in which $\boldsymbol{\theta}^* = \boldsymbol{\theta}^{(i)} + \boldsymbol{\varepsilon}^{(i)}$, where $\boldsymbol{\varepsilon}^{(i)}$ is a random variable generated from a multivariate symmetric distribution $h(\boldsymbol{\varepsilon}^{(i)})$. Thus, the symmetric proposal distribution gives $\eta(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(i-1)}) = h(\boldsymbol{\theta}^* - \boldsymbol{\theta}^{(i-1)}) = h(\boldsymbol{\theta}^{(i-1)} - \boldsymbol{\theta}^*)$ leading to the following acceptance ratio:

$$\alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}^{(i-1)}) = \min\left\{1, \frac{\pi(\boldsymbol{\theta}^*)}{\pi(\boldsymbol{\theta}^{(i-1)})}\right\} \tag{1.35}$$

The distribution $h(\boldsymbol{\varepsilon}^{(i)})$ could be multivariate *normal*, *student's t* or other distributions.

Another common choice is the independence sampler in which $\eta(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(i)}) =$

$\eta(\boldsymbol{\theta}^*)$, i.e., the sample drawn from this distribution does not depend on the sample in the last time step. However, the output samples still form a Markov chain due to the acceptance/rejection step. See [Robert, 2004] for a detailed review of the different variants of the Metropolis-Hastings algorithm.

1.2.5.2 Gibbs Sampling

The Gibbs sampler is one of the most famous MCMC samplers proposed in [Geman and Geman, 1984] and further developed in [Gelfand and Smith, 1990]. Because of its simplicity, it has been used in many Bayesian data analysis problems, specifically, when conditionally conjugate prior distributions are used.

The main idea behind the Gibbs sampler is that it is generally more convenient and computationally efficient (especially in high-dimensional problems) to divide the state of interest $\boldsymbol{\theta}$ into sub-component, of possibly different dimension and then update them one by one successively.

Assumed that the vector of parameters $\boldsymbol{\theta} \in E$ is partitioned into B sub-blocks so that $\boldsymbol{\theta} = [\boldsymbol{\varrho}_1, \boldsymbol{\varrho}_2, \cdots, \boldsymbol{\varrho}_B]$.The proposal function for the Gibbs chain requires updating, *in turn*, each sub-block by sampling it from its conditional distribution given all the other sub-blocks, $\pi(\boldsymbol{\varrho}_b|\boldsymbol{\varrho}_{-b}^{(i-1)})$, where $\boldsymbol{\varrho}_{-b}^{(i-1)}$ represents all the sub-blocks of $\boldsymbol{\theta}$, except for, $\boldsymbol{\varrho}_b$, at their current values: $\boldsymbol{\varrho}_{-b}^{(i-1)} = [\boldsymbol{\varrho}_1^{(i)}, \cdots, \boldsymbol{\varrho}_{b-1}^{(i)}, \boldsymbol{\varrho}_{b+1}^{(i-1)}, \cdots, \boldsymbol{\varrho}_B^{(i-1)}]$. Thus, each sub-block $\boldsymbol{\varrho}_b$ is updated conditional on the latest values of the other components of $\boldsymbol{\theta}$. Algorithm 1.4 summarizes the Gibbs sampling algorithm. This is useful when the full posterior $p(\boldsymbol{\theta}|\mathbf{y})$ is difficult to sample from directly but the posterior marginal distributions of each sub-blocks are easy to sample from.

Algorithm 1.4 Gibb Sampling algorithm

1: Initialization: $i = 0, \boldsymbol{\theta}^{(0)} = \boldsymbol{\theta}^{(0)} = [\boldsymbol{\varrho}_1^{(0)}, \boldsymbol{\varrho}_2^{(0)}, \cdots, \boldsymbol{\varrho}_B^{(0)}]$
2: **for** $i = 1, \ldots, N$ **do** `# MCMC Iterations`
3: **for** $b = 1, \ldots, B$ **do** `# Block Sampling`
4: Draw $\boldsymbol{\varrho}_b^{(i)}$ from $\pi(\boldsymbol{\varrho}_b|\boldsymbol{\varrho}_{-b}^{(i-1)})$
5: **end for**
6: **end for**

The Gibbs sampler is a special case of MH algorithm in which the proposal distribution used to move each sub-block is the conditional posterior distribution leading to an acceptance ratio equal to one. The transition kernel of the Markov chain created by the Gibbs sampler is

$$\mathcal{K}(\boldsymbol{\theta}^{(i-1)}, \boldsymbol{\theta}^{(i)}) = \prod_{b=1}^{B} \pi(\boldsymbol{\varrho}_b|\boldsymbol{\varrho}_{-b}^{(i-1)}) \tag{1.36}$$

Unfortunately, in practice, not all problems can reasonably be tackled using Gibbs sampling since it is usually not easy to draw samples from the full posterior conditional distribution of each sub-block.

1.2.5.3 Metropolis–Hastings within Gibbs

This subsection talks about combining the Gibbs sampler and the Metropolis–Hastings algorithm, which is often helpful in practical problems when sampling from the full posterior conditional distribution of each sub-blocks of the state is impossible. Metropolis-within-Gibbs (MWG) was proposed as a hybrid algorithm that combines Metropolis-Hastings and Gibbs sampling, and was suggested in [Tierney, 1994]. The idea was to substitute a Metropolis step when Gibbs sampling (from conditional distribution) is impossible. The algorithm is described in Algorithm 1.5

Algorithm 1.5 Metropolis-Hasting within Gibbs algorithm

Initialisation: $i = 0, \boldsymbol{\theta}^{(0)} = \boldsymbol{\theta}^{(0)} = [\boldsymbol{\varrho}_1^{(0)}, \boldsymbol{\varrho}_2^{(0)}, \cdots, \boldsymbol{\varrho}_B^{(0)}]$
for $i = 1, \ldots, N$ **do** `# MCMC Iterations`
  for $b = 1, \ldots, B$ **do** `# Block Sampling`
    if $\pi(\boldsymbol{\varrho}_b|\boldsymbol{\varrho}_{-b}^{(i-1)})$ is possible to sample directly **then**
      Draw $\boldsymbol{\varrho}_b^{(i)}$ from $\pi(\boldsymbol{\varrho}_b|\boldsymbol{\varrho}_{-b}^{(i-1)})$
    else
      Generate $\boldsymbol{\varrho}_b^*$ from $\eta_b(\boldsymbol{\varrho}_b|\boldsymbol{\varrho}_1^{(i)}, \cdots, \boldsymbol{\varrho}_{b-1}^{(i)}, \boldsymbol{\varrho}_b^{(i-1)}, \boldsymbol{\varrho}_{b+1}^{(i-1)}, \cdots, \boldsymbol{\varrho}_B^{(i-1)})$
      Compute the acceptance ratio:

$$\alpha(\boldsymbol{\varrho}_b^*, \boldsymbol{\varrho}_b^{(i-1)}) = \min\left\{1, \frac{\pi(\boldsymbol{\varrho}_b^{(*)}|\boldsymbol{\varrho}_{-b}^{(i-1)})\eta_b(\boldsymbol{\varrho}_b^{(i-1)}|\boldsymbol{\varrho}_1^{(i)}, \cdots, \boldsymbol{\varrho}_{b-1}^{(i)}, \boldsymbol{\varrho}_b^*, \boldsymbol{\varrho}_{b+1}^{(i-1)}, \cdots, \boldsymbol{\varrho}_B^{(i-1)})}{\pi(\boldsymbol{\varrho}_b^{(i-1)}|\boldsymbol{\varrho}_{-b}^{(i-1)})\eta_b(\boldsymbol{\varrho}_b^*|\boldsymbol{\varrho}_1^{(i)}, \cdots, \boldsymbol{\varrho}_{b-1}^{(i)}, \boldsymbol{\varrho}_b^{(i-1)}, \boldsymbol{\varrho}_{b+1}^{(i-1)}, \cdots, \boldsymbol{\varrho}_B^{(i-1)})}\right\}$$

      Sample random variate $U^{(i)}$ from $\mathcal{U}(0, 1)$
      if $U^{(i)} \leq \alpha(\boldsymbol{\varrho}_b^*, \boldsymbol{\varrho}_b^{(i-1)})$ **then**
        $\boldsymbol{\varrho}_b^{(i)} = \boldsymbol{\varrho}_b^*$
      else
        $\boldsymbol{\varrho}_b^{(i)} = \boldsymbol{\varrho}_b^{(i-1)}$
      end if
    end if
  end for
end for

Please note that the algorithm 1.5 is the *deterministic scan Metropolis–Hastings within Gibbs sampling* version since we sample sequentially each sub-block of state $\boldsymbol{\theta}$. Another common scheme is *Random scan Metropolis–Hastings within Gibbs Sampling* where block b can be chosen randomly in $\{1, \cdots, B\}$.

1.2.5.4 Discussion on MCMC methods

MCMC algorithms have been applied with success to many problems as Bayesian inference solution. These algorithms have thus played a significant role in statistics, econometrics, physics, signal processing and computing science over the last three decades. Unlike importance sampling technique, the possibility in MCMC methods of using local moves for all the elements or sub-blocks of the state is really

an appealing feature that could lead to significant improvement of the resulting estimators, especially in high-dimensional problems.

However, there are two major drawbacks with MCMC methods. Firstly, it is difficult to assess when the Markov chain has reached its stationary regime of interest. As a consequence, we do not know at what point to start collecting the generated samples. Secondly, if the target distribution is highly multi-modal, MCMC algorithms can easily become trapped in local modes.

Furthermore, if we are interested in estimating the normalizing constant of the target given in Eq. (1.19) (e.g., for model selection - see Section 1.1.3), unlike importance sampler for which an unbiased estimate is easily obtained by Eq. (1.28), special methods have to be used in order to obtain such estimate with samples obtained from the output samples of an MCMC algorithm (see [Robert, 2007, Friel and Wyse, 2012] for a review). As a consequence, extra complexity cost will be added to obtain such estimate of the normalization constant and more importantly the unbiasedness of this estimate will not be preserved for finite number of samples (unlike IS-based estimator).

In recent years, more robust and efficient Monte Carlo algorithms have been established in order to efficiently explore high dimensional and multimodal spaces. Many of them are population based, in that they deal explicitly with a collection of samples at each iteration, including population-based MCMC [Liang and Wong, 2001, Jasra et al., 2007] and sequential Monte-Carlo sampler [Del Moral et al., 2006].

1.3 Population-based simulation algorithms

1.3.1 Population-based MCMC

Population-based MCMC was originally developed by Geyer [Geyer, 1991]. Further advances came with an evolutionary Monte Carlo algorithm in [Liang and Wong, 2000, Liang and Wong, 2001] who attempted to produce genetic algorithm type moves to improve the mixing of the Markov chain.

The new target distribution used in the population-based MCMC is defined as:

$$\pi^*(\boldsymbol{\theta}_{1:T}) = \prod_{k=1}^{T} \pi_k(\boldsymbol{\theta}_k) \tag{1.37}$$

where it is assumed that the true target of interest (the posterior distribution in Bayesian inference) $\pi = \pi_k$ for at least one $k = 1, \ldots, T$. A (time homogeneous) Markov kernel that is π^*-irreducible, aperiodic and admits π^* as its invariant distribution is needed in order to contruct a valid MCMC algorithm by simply considering the target distribution in Eq. 1.37 on an extended space $\boldsymbol{\theta}_{1:T} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_T)$ as in hybrid MCMC such as Metropolis-within-Gibbs updates of $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$, etc. - See Section 1.2.5.3. Thus, approximation of integrals with respect to the target distribution is

defined as in classical MCMC - Eq. (1.31) - by using samples from the chain with target of interest, i.e., $\cup k$ such that $\pi = \pi_k$.

Concerning the choice of the target distribution to be used for each chain, a well known choice, especially for multimodal distribution, is to temper the target distribution of interest [Geyer, 1991], i.e.,

$$\pi_k(\boldsymbol{\theta}) \propto \pi(\boldsymbol{\theta})^{\phi_k} \tag{1.38}$$

or only the likelihood in the Bayesian setting

$$\pi_k(\boldsymbol{\theta}) \propto p(\mathbf{y}|\boldsymbol{\theta})^{\phi_k} p(\boldsymbol{\theta}) \tag{1.39}$$

with $\forall k$, $\phi_k \in (0, 1]$ and for at least one $k = 1, \ldots, T$, $\phi_k = 1$. The idea of tempering the target distribution is that the distributions at low temperatures, that is ϕ_k close to zero, are easily sampled and can improve the mixing of the entire algorithm, especially when the target is highly multimodal [Liang and Wong, 2001, Geyer, 1991]. The use of tempered target in population-based MCMC could indeed be beneficial in order to escape from a local mode which could be very unlikely to happen in classical MCMC.

This population-based algorithm, summarized in Algorithm 1.6, works by simulating a population of T Markov chains with different invariant distributions in parallel using MCMC. The population is updated by mutation (Metropolis update in one single chain), crossover (partial states swapping between different chains), and exchange operators (full state swapping between different chains).

1.3.1.1 Mutation

In this type of move, the population is updated via a Markov kernel, $\boldsymbol{\theta}^*_{1:T} \sim \mathcal{K}(\boldsymbol{\theta}^{(i-1)}_{1:T}, \cdot)$. All T chains could be updated using for example the following product mutation kernel:

$$\mathcal{K}(\boldsymbol{\theta}^{(i-1)}_{1:T}, \boldsymbol{\theta}^*_{1:T}) = \prod_{k=1}^{T} \mathcal{K}_k(\boldsymbol{\theta}^{(i-1)}_k, \boldsymbol{\theta}^*_k) \tag{1.40}$$

where $\mathcal{K}_k$ is a Markov kernel that is π_k-invariant. Let us remark that the transition kernels $\{\mathcal{K}_k\}_{k=1}^T$ can be the ones of a Metropolis-Hastings algorithm, Gibbs sampler or Metropolis-within-Gibbs sampler.

1.3.1.2 Exchange

The main idea of this move is to exchange information between different chains. This is a Metropolis-Hastings type mechanism which will accept the move with probability:

$$\min\left\{1, \frac{\pi_k(\boldsymbol{\theta}_j)\pi_j(\boldsymbol{\theta}_k)}{\pi_k(\boldsymbol{\theta}_k)\pi_j(\boldsymbol{\theta}_j)}\right\} \tag{1.41}$$

if we select both chains (j, k) to be swapped with an equal probability. This move is one of the key ingredient that will allow the chains that target π to escape from a local mode.

1.3.1.3 Crossover

The crossover works by switching some information between two parent samples from two different chains for producing two novel offsprings. If the state can be decomposed as $\boldsymbol{\theta}_k = (\theta_{1k}, \ldots, \theta_{pk}) \forall k$, then the crossover selects a position in the vector to be swapped. That is, if we propose to crossover the m-th position in the vector for chains j and k we have as proposal samples:

$$\begin{aligned} \boldsymbol{\theta}_k^* &= (\theta_{1k}, \ldots, \theta_{(m-1)k}, \theta_{mj}, \theta_{(m+1)k}, \ldots, \theta_{pk}) \\ \boldsymbol{\theta}_j^* &= (\theta_{1j}, \ldots, \theta_{(m-1)j}, \theta_{mk}, \theta_{(m+1)j}, \ldots, \theta_{pj}) \end{aligned} \tag{1.42}$$

If we choose the two chains as well as the crossover position with uniform probability, this moves is accepted with the same probability as for the exchange move, i.e.:

$$\min \left\{ 1, \frac{\pi_k(\boldsymbol{\theta}_k^*)\pi_j(\boldsymbol{\theta}_j^*)}{\pi_k(\boldsymbol{\theta}_k)\pi_j(\boldsymbol{\theta}_j)} \right\} \tag{1.43}$$

Algorithm 1.6 Population-based MCMC algorithm

1: Initialization: Sample $\boldsymbol{\theta}_{1:T}^{(0)}$ from some initial pdf
2: **for** $i = 1, \ldots, N$ **do**
3: **for** $k = 1, \ldots, T$ **do**
4: Perform the mutation : Sample $\boldsymbol{\theta}_k^{(j)}$ using the Markov kernel $\mathcal{K}_k(\boldsymbol{\theta}_k^{(i-1)}, \cdot)$ which is π_k-invariant
5: **end for**
6: Sample random variate u from $\mathcal{U}(0, 1)$
7: **if** $u \leq P_e$ **then**
8: Perform the exchange move described in Section 1.3.1.2
9: **else**
10: Perform the crossover move described in Section 1.3.1.3
11: **end if**
12: **end for**

However, like standard MCMC, this population-based MCMC algorithm still suffer of the difficulty to assess when the Markov chains have reached their stationary regime. Furthermore, as discussed in Section 1.2.5.4 for classical MCMC methods, special techniques have to be used in order to obtain an estimate of the normalization constant of the target distribution.

1.3.2 Sequential Monte Carlo (SMC) Samplers

Sequential Monte Carlo (SMC) methods is a class of sampling algorithms which combine importance sampling and resampling. They have been primarily used as "particle filter" to solve optimal filtering problems; see, for example, [Cappé et al., 2007] and [Doucet and Johansen, 2009] for recent reviews. In this context, SMC methods/particle filters have enjoyed wide-spread use in various applications (tracking, computer vision, digital communications) due to the fact that they provide a simple way of approximating complex filtering distribution sequentially in time. But in [Del Moral et al., 2006], the authors developed a general framework that allows SMC to be used to simulate from a single and static target distribution, thus becoming an interesting alternative to standard MCMC methods as well as to population-based MCMC algorithms.

1.3.2.1 General Idea

Standard SMC techniques have been developed to deal with "on-line" applications which involve sampling form a sequence of distributions sequentially in time [Doucet and De Freitas, 2001]. Until the development of SMC samplers, SMC techniques have been solely confined to situations which involve a sequence of probability distributions π_t whose dimension is increasing over time, optimal Bayesian filtering. Indeed, a distribution, at time t in the sequence, is defined on a measurable product space of the form $E_t = E \times \cdots \times E = E^t$ which means that $\dim(E_{t-1}) < \dim(E_t)$. In [Del Moral et al., 2006], the authors propose the SMC samplers that generalizes the methodology of SMC in order to sample sequentially from a sequence of probability distribution $\{\pi_t\}_{t=1}^{T}$ where now each distribution in the sequence is defined on a common measurable space E.

This later sampler enables one to sample sequentially from a sequence of probability distributions which are known up to a normalizing constant and defined on a common space E. It will be the aim to approximate these probability distributions by a cloud of weighted random samples which are propagated over time using SMC methods. The SMC spirit is to start with a tractable and easy to sample distribution π_1, then increase the complexity of the problem to finish by the distribution we are interested in, π_T (i.e., $p(\boldsymbol{\theta}|\mathbf{y})$ the posterior in the Bayesian setting), through a sequence of artificial intermediate distributions.

The generality of this methodology, allows one not only to derive simple algorithms to make parallel Markov Chain Monte Carlo runs interact in a principled manner, but also to obtain new methods for global optimization and sequential Bayesian estimation. SMC methods possess several interesting advantages compared to traditional MCMC methods. First of all, SMC samplers do not suffer of the difficulty to assess the convergence of a Markov chain unlike MCMC methods. Secondly, as discussed in [Sisson et al., 2007], in some challenging problems for which the mixing of the Markov chain is very slow, the SMC sampler appears as an efficient alternative solution. Finally, since the SMC sampler is based on the importance sampling principle and does not rely on ergodic properties of a Markov

chain, adaptive proposal distributions can be easily used, thus giving a lot more of opportunities to improve its efficiency. It is important to note that SMC samplers should be viewed as a complementary approach to MCMC, and that MCMC kernels will in most cases be ingredients of the SMC method, thus allowing for example some local moves on sub-block of the particles (instead of sampling all the state on one step as in classical importance sampling). SMC methods, in addition, allow for computation of all kinds of moments, quantiles and highest posterior density regions. Moreover, it is worth noting that SMC based approach in which particles are carried forward over time using a combination of Sequential Importance Sampling (SIS) and resampling ideas is completely different from population-based MCMC algorithm described previously, where one runs an MCMC chain on an extended space E^t. Additionally, it is well known that SMC is well suited for the computation of Bayesian Evidence using the product of estimates of ratio between two successive normalizing constants. It is clearly evident that SMC samplers offer a number of significant advantages compared with other Bayesian techniques currently available, thus being a promising solution to deal with Bayesian Inference.

Finally, let us note that there exists few other SMC methods appropriate for static inference such as annealed importance sampling [Neal, 2001], the sequential particle filter of [Chopin, 2002] and population Monte Carlo [Cappé et al., 2004] but since SMC sampler approach contains all of these methods as a special case we concentrate upon this only.

1.3.2.2 SMC Sampler Methodology

The SMC Sampler methodology is a generic approach to approximate a sequence of probability distribution $\{\pi_t\}_{t=1}^{T}$ defined upon a common measurable space E ([Del Moral et al., 2006]), where the final distribution π_T is the posterior distribution of interest. As all through this chapter, we consider that the target distribution π_t is only known up to a normalizing constant, i.e.,

$$\pi_t(\boldsymbol{\theta}_t) = \frac{\gamma_t(\boldsymbol{\theta}_t)}{Z_t} \tag{1.44}$$

where $Z_t = \int_E \gamma_t(\boldsymbol{\theta}) d\boldsymbol{\theta}$ is normalizing constant of target distribution π_t.

The method begins at time $t = 1$ start with a target π_1 which is assumed to be easy to approximate efficiently by using IS, i.e., η_1 can be selected such that the variance of importance weights is small (simplest case is to have $\eta_1 = \pi_1$) . The samples $\left\{\boldsymbol{\theta}_1^{(m)}\right\}_{m=1}^{N}$ are generated from an initial proposal distribution η_1 then the importance weights are computed using the classical IS identity (Section 1.2.3)

$$W_1^{(m)} = \frac{\pi_1(\boldsymbol{\theta}^{(m)})}{\eta_1(\boldsymbol{\theta}^{(m)})} \tag{1.45}$$

Then, at time $t = 2$, we consider the new target distribution π_2. To build the associated IS distribution η_2, we use the particles sampled at time $t = 1$, say $\left\{\boldsymbol{\theta}_1^{(m)}\right\}$.

The rationale is that, if π_1 and π_2 are not too different from one another, then it should be possible to move the particles $\boldsymbol{\theta}_1^{(m)}$ in the regions of high probability density of π_2 in a sensible way.

At time $t-1$, the particles $\left\{\boldsymbol{\theta}_{t-1}^{(m)}\right\}_{m=1}^{N}$ which are distributed according to the *importance distribution* η_{t-1} are then moved, from time $t-1$ to t, by using a *mutation kernel* $\mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)$ which denotes the probability density of moving from $\boldsymbol{\theta}_{t-1}$ to $\boldsymbol{\theta}_t$. In this way, the particles $\left\{\boldsymbol{\theta}_t^{(m)}\right\}_{m=1}^{N}$ are marginally distributed according to

$$\eta_t(\boldsymbol{\theta}_t) = \int_E \eta_{t-1}(\boldsymbol{\theta}_{t-1})\mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)d\boldsymbol{\theta}_{t-1} \tag{1.46}$$

If η_t can be computed pointwise, then it is possible to use the standard IS estimate of π_t and Z_t. However, in most cases, it is impossible to compute the importance distribution $\eta_t(\boldsymbol{\theta}_t)$ that is given by

$$\eta_t(\boldsymbol{\theta}_t) = \eta_1\mathcal{K}_{2:t}(\boldsymbol{\theta}_t) = \int \eta_1(\boldsymbol{\theta}_1)\prod_{k=2}^{t}\mathcal{K}_k(\boldsymbol{\theta}_{k-1}, \boldsymbol{\theta}_k)d\boldsymbol{\theta}_{1:t-1} \tag{1.47}$$

hence impossible to compute the importance weights

$$W_t^{(m)} = \frac{\gamma_t(\boldsymbol{\theta}_t^{(m)})}{\eta_t(\boldsymbol{\theta}_t^{(m)})} \tag{1.48}$$

A potential solution is to attempt to approximate η_t pointwise by

$$\hat{\eta}_{t-1}\mathcal{K}_t(\boldsymbol{\theta}_t) = \frac{1}{N}\sum_{m=1}^{N}\mathcal{K}_t(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t) \tag{1.49}$$

However, as discussed in [Del Moral et al., 2006], this approach has two major drawbacks. Firstly, the computational complexity of this algorithm would be $O(N^2)$ which is too costly. Secondly, in order to perform the algorithm we need to be able to compute $\mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)$ which is impossible in many important scenarios, e.g., when $\mathcal{K}_t$ is a Metropolis-Hasting (MH) kernel - defined in Eq. (1.34) - of invariant distribution π_t.

To overcome this limitation, the idea developed in [Del Moral et al., 2006] is to introduce a sequence of extended probability distributions $\{\tilde{\pi}_t\}_{t=1}^{T}$ on state-spaces of increasing dimension E^{t+1} which admits the distribution of interest $\{\pi_t\}_{t=1}^{T}$ as marginals and $\{Z_t\}_{t=1}^{T}$ as normalizing constants. They defined this novel sequence of target distributions $\tilde{\pi}_t$ as follows:

$$\tilde{\pi}_t(\boldsymbol{\theta}_{1:t}) = \frac{\tilde{\gamma}_t(\boldsymbol{\theta}_{1:t})}{Z_t} \tag{1.50}$$

where

$$\tilde{\gamma}_t(\boldsymbol{\theta}_{1:t}) = \gamma_t(\boldsymbol{\theta}_t)\prod_{k=1}^{t-1}\mathcal{L}_k(\boldsymbol{\theta}_{k+1}, \boldsymbol{\theta}_k) \tag{1.51}$$

in which the artificial kernels introduced $\{\mathcal{L}_k\}_{k=1}^{t-1}$ are called *backward* Markov kernels since $\mathcal{L}_t(\boldsymbol{\theta}_{t+1}, \boldsymbol{\theta}_t)$ denotes the probability density of moving back from $\boldsymbol{\theta}_{t+1}$ to $\boldsymbol{\theta}_t$. By using such a sequence of extended target distributions $\{\tilde{\pi}_t\}_{t=1}^{T}$ based on the introduction of backward kernels $\{\mathcal{L}_k\}_{k=1}^{t-1}$, IS can be used without having to compute the marginal distributions η_t explicitly.

Within this framework, one may then work with the constructed sequence of distributions, $\widetilde{\pi}_t$, under the standard SMC algorithm [Doucet and De Freitas, 2001]. In summary, the SMC sampler algorithm involves three stages:

1. *mutation*, where the particles are moved from $\boldsymbol{\theta}_{t-1}$ to $\boldsymbol{\theta}_t$ via a mutation kernel $\mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)$;

2. *correction*, where the particles are reweighted with respect to π_t via the incremental importance weight (Equation (1.55)); and

3. *selection*, where according to some measure of particle diversity, such as effective sample size, the weighted particles may be resampled in order to reduce the variability of the importance weights.

In more detail, suppose that at time $t-1$, we have a set of weighted particles $\left\{\boldsymbol{\theta}_{1:t-1}^{(m)}, \widetilde{W}_{t-1}^{(m)}\right\}_{m=1}^{N}$ that approximates $\tilde{\pi}_{t-1}$ via the empirical measure

$$\tilde{\pi}_{t-1}^{N}(d\boldsymbol{\theta}_{1:t-1}) = \sum_{m=1}^{N} \widetilde{W}_{t-1}^{(m)} \delta_{\boldsymbol{\theta}_{1:t-1}^{(m)}}(d\boldsymbol{\theta}_{1:t-1}) \tag{1.52}$$

These particles are first propagated to the next distribution $\tilde{\pi}_t$ using a Markov kernel $\mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)$ to obtain the set of particles $\left\{\boldsymbol{\theta}_{1:t}^{(m)}\right\}_{m=1}^{N}$. IS is then used to correct for the discrepancy between the sampling distribution $\eta t(\boldsymbol{\theta}_{1:t})$ defined as

$$\eta_t(\boldsymbol{\theta}_{1:t}^{(m)}) = \eta_1(\boldsymbol{\theta}_1^{(m)}) \prod_{k=2}^{t} \mathcal{K}_k(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t^{(m)}) \tag{1.53}$$

and $\tilde{\pi}_t(\boldsymbol{\theta}_{1:t})$. In this case the new expression for the unnormalized importance weights is given by

$$W_t^{(m)} \propto \frac{\tilde{\pi}_t(\boldsymbol{\theta}_{1:t}^{(m)})}{\eta_t(\boldsymbol{\theta}_{1:t}^{(m)})} = \frac{\pi_t(\boldsymbol{\theta}_t^{(m)}) \prod_{s=1}^{t-1} \mathcal{L}_s(\boldsymbol{\theta}_{s+1}^{(m)}, \boldsymbol{\theta}_s^{(m)})}{\eta_1(\boldsymbol{\theta}_1^{(m)}) \prod_{k=2}^{t} \mathcal{K}_k(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t^{(m)})} \propto w_t(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t^{(m)}) W_{t-1}^{(m)} \tag{1.54}$$

where w_t, termed the (unnormalized) *incremental weights*, are calculated as,

$$w_t(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t^{(m)}) = \frac{\gamma_t(\boldsymbol{\theta}_t^{(m)}) \mathcal{L}_{t-1}(\boldsymbol{\theta}_t^{(m)}, \boldsymbol{\theta}_{t-1}^{(m)})}{\gamma_{t-1}(\boldsymbol{\theta}_{t-1}^{(m)}) \mathcal{K}_t(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t^{(m)})} \tag{1.55}$$

However, as in the particle filter, since the discrepancy between the target distribution $\tilde{\pi}_t$ and the proposal η_t increases with t, the variance of the unnormalized importance weights tends therefore to increase as well, leading to a degeneracy of

the particle approximation. A common criterion used in practice to check this problem is the effective sample size $\mathbb{ESS}$, which was introduced in equation (1.29). For the SMC sampler, the $\mathbb{ESS}$ can be computed by:

$$\mathbb{ESS}_t = \left[\sum_{m=1}^{N} (\widetilde{W}_t^{(m)})^2\right]^{-1} = \frac{\left(\sum_{m=1}^{N} W_{t-1}^{(m)} w_t(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t^{(m)})\right)^2}{\sum_{j=1}^{N} \left(W_{t-1}^{(j)}\right)^2 \left(w_t(\boldsymbol{\theta}_{t-1}^{(j)}, \boldsymbol{\theta}_t^{(j)})\right)^2} \tag{1.56}$$

If the degeneracy is too high, i.e., the $\mathbb{ESS}_t$ is below a prespecified threshold, $\overline{\mathbb{ESS}}$, then a resampling step is performed. The particles with low weights are discarded whereas particles with high weights are duplicated. After resampling, the particles are equally weighted.

To sum up the algorithm proceeds as shown in Algorithm 1.7.

Algorithm 1.7 Generic SMC Sampler Algorithm

1: Initialize particle system

2: Sample $\left\{\boldsymbol{\theta}_1^{(m)}\right\}_{m=1}^{N} \sim \eta_1(\cdot)$ and compute $\widetilde{W}_1^{(m)} = \left(\frac{\gamma_1(\boldsymbol{\theta}_1^{(m)})}{\eta_1(\boldsymbol{\theta}_1^{(m)})}\right)\left[\sum_{j=1}^{N} \frac{\gamma_1(\boldsymbol{\theta}_1^{(j)})}{\eta_1(\boldsymbol{\theta}_1^{(j)})}\right]^{-1}$ and do resampling if $\mathbb{ESS} < \overline{\mathbb{ESS}}$

3: **for** $t = 2, \ldots, T$ **do**

4: Mutation: for each $m = 1, \ldots, N$: Sample $\boldsymbol{\theta}_t^m \sim \mathcal{K}_t(\boldsymbol{\theta}_{t-1}^{(m)}; \cdot)$ where $\mathcal{K}_t(\cdot;\cdot)$ is a $\pi_t(\cdot)$ invariant Markov kernel.

5: Computation of the weights: for each $m = 1, \ldots, N$

$$W_t^{(m)} = \widetilde{W}_{t-1}^{(m)} \frac{\gamma_t(\boldsymbol{\theta}_t^{(m)}) \mathcal{L}_{t-1}(\boldsymbol{\theta}_t^{(m)}, \boldsymbol{\theta}_{t-1}^{(m)})}{\gamma_{t-1}(\boldsymbol{\theta}_{t-1}^{(m)}) \mathcal{K}_t(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t^{(m)})}$$

Normalization of the weights : $\widetilde{W}_t^{(m)} = W_t^{(m)} \left[\sum_{j=1}^{N} W_t^{(j)}\right]^{-1}$

6: Selection: if $ESS < \overline{\mathbb{ESS}}$ then Resample

7: **end for**

The final weighted particles at distribution π_T are considered weighted samples from the target distribution π of interest.

Let us mention two interesting estimates from SMC samplers. Firstly, since $\tilde{\pi}_t$ admits π_t as marginals by construction for any $1 \leq t \leq T$, the SMC sampler provides an estimate of this distribution

$$\pi_t^N(d\boldsymbol{\theta}) = \sum_{m=1}^{N} \widetilde{W}_t^{(m)} \delta_{\boldsymbol{\theta}_t^{(m)}}(d\boldsymbol{\theta}) \tag{1.57}$$

and the estimation of expectation in equation (1.12) is given by

$$\mathbb{E}_{\pi_t^N}[\varphi(\boldsymbol{\theta})] = \sum_{m=1}^{N} \widetilde{W}_t^{(m)} \varphi(\boldsymbol{\theta}_t^{(m)}) \tag{1.58}$$

and secondly, the estimated ratio of normalizing constants $\frac{Z_t}{Z_{t-1}} = \frac{\int \gamma_t(\boldsymbol{\theta})d\boldsymbol{\theta}}{\int \gamma_{t-1}(\boldsymbol{\theta})d\boldsymbol{\theta}}$ is given by

$$\widehat{\frac{Z_t}{Z_{t-1}}} = \sum_{m=1}^{N} \widetilde{W}_{t-1}^{(m)} w_t(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t^{(m)}) \tag{1.59}$$

Consequently, the estimate of $\frac{Z_t}{Z_1}$ is

$$\widehat{\frac{Z_t}{Z_1}} = \prod_{k=2}^{t} \widehat{\frac{Z_k}{Z_{k-1}}} = \prod_{k=2}^{t} \sum_{m=1}^{N} \widetilde{W}_{k-1}^{(m)} w_k(\boldsymbol{\theta}_{k-1}^{(m)}, \boldsymbol{\theta}_k^{(m)}) \tag{1.60}$$

If the resampling scheme used is unbiased, then (1.60) is also unbiased ([Del Moral and Miclo, 2000]). Moreover, the complexity of this algorithm is in $O(N)$ and it can be easily parallelized.

1.3.2.3 Algorithm settings

The algorithm presented in the previous subsection is very general. There is a wide range of possible choices to consider when designing an SMC sampler algorithm, the appropriate sequence of distributions $\{\pi_t\}_{1\leq t\leq T}$, the choice of both the mutation kernel $\{\mathcal{K}_t\}_{2\leq t\leq T}$ and the backward mutation kernel $\{\mathcal{L}_{t-1}\}_{t=2}^{T}$ (for a given mutation kernels), see details in [Del Moral et al., 2006]. In this subsection, we provide a discussion on how to choose these parameters of the algorithm in practice.

a) Sequence of distributions π_t

There are many potential choices for $\{\pi_t\}$ leading to various integration and optimization algorithms. As a special case, we can set $\pi_t = \pi$ for all $t \in \mathcal{N}$. Alternatively, to maximize $\pi(\boldsymbol{\theta})$, we could consider $\pi_t(\boldsymbol{\theta}_t) = [\pi(\boldsymbol{\theta}_t)]^{\xi_t}$ for an increasing schedule $\{\xi_t\}_{t\in\mathcal{N}}$ to ensure $\pi_T(\boldsymbol{\theta})$ is concentrated around the set of global maxima of $\pi(\boldsymbol{\theta})$. In the context of Bayesian inference for static parameters which is the main focus of this thesis, one can consider $\pi_t(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|y_1, \cdots, y_t)$, which corresponds to *data tempered* schedule.

In this thesis, we are interested in the *likelihood tempered* target sequence ([Neal, 2001])

$$\pi_t(\boldsymbol{\theta}) \propto p(\boldsymbol{\theta})p(\mathbf{y}|\boldsymbol{\theta})^{\phi_t} \tag{1.61}$$

where $\{\phi_t\}$ is a non-decreasing temperature schedule with $\phi_0 = 0$ and $\phi_T = 1$. We thus sample initially from the prior distribution $\pi_0 = p(\boldsymbol{\theta})$ directly and introduce the effect of the likelihood gradually in order to obtain at the end $t = T$ an approximation of the posterior distribution $p(\boldsymbol{\theta}|\mathbf{y})$. As discussed for the population-based MCMC, tempering the likelihood could significantly improve the exploration of the state space in complex multimodal posterior distribution. From equation (1.60), the *marginal likelihood* of interest, $p(\mathbf{y})$, can be approximated with SMC samplers

as:

$$Z_T = Z_1 \prod_{t=2}^{T} \frac{Z_t}{Z_{t-1}} \approx \prod_{t=2}^{T} \sum_{m=1}^{N} \widetilde{W}_{t-1}^{(m)} w_t(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t^{(m)}) \tag{1.62}$$

where $Z_t = \int p(\mathbf{y}|\boldsymbol{\theta})^{\phi_t} p(\boldsymbol{\theta}) d\boldsymbol{\theta}$ corresponds to the normalizing constant of the target distribution at iteration t (thus $Z_1 = \int p(\boldsymbol{\theta}) d\boldsymbol{\theta} = 1$). The approximation of an expectation with respect to the posterior is given by:

$$\mathbb{E}_{\pi^N}[\varphi(\boldsymbol{\theta})] = \sum_{i=1}^{N} \widetilde{W}_T^{(i)} \varphi(\boldsymbol{\theta}_T^{(i)}) \tag{1.63}$$

b) Sequence of mutation kernels $\mathcal{K}_t$

The performance of SMC samplers depends heavily upon the selection of the transition kernels $\{\mathcal{K}_t\}_{t=2}^T$ and the auxiliary backward kernels $\{\mathcal{L}_{t-1}\}_{t=2}^T$. There are many possible choices for $\mathcal{K}_t$ which have been discussed in [Del Moral et al., 2006]. In this study, we propose to employ MCMC kernels of invariant distribution π_t for $\mathcal{K}_t$. This is an attractive strategy since we can use the vast literature on the design of efficient MCMC algorithms to build a good importance distributions (See section 1.2.5 and [Robert and Casella, 2004]).

More precisely, since we are interested in complex models with potentially high-dimensional and multimodal posterior distribution, a series Metropolis-within-Gibbs kernels with local moves, as described in Section 1.2.5.3, will be employed in order to successively move the B sub-blocks of the state of interest, $\boldsymbol{\theta} = [\boldsymbol{\varrho}_1, \boldsymbol{\varrho}_2, \cdots, \boldsymbol{\varrho}_B]$. A random walk proposal distribution is used for each sub-block with a multivariate Gaussian distribution as proposal:

$$\boldsymbol{\varrho}_{b,t}^* = \boldsymbol{\varrho}_{b,t-1} + \boldsymbol{\varepsilon}_{b,t} \tag{1.64}$$

in which $\boldsymbol{\varepsilon}_{b,t}$ is a Gaussian random variable with zero mean and covariance matrix $\boldsymbol{\Sigma}_{b,t}$. As with any sampling algorithm, faster mixing does not harm performance and in some cases will considerably improve it. In the particular case of Metropolis-Hastings kernels, the mixing speed relies on adequate proposal scales. As a consequence, we adopt the strategy proposed in [Jasra et al., 2011]. The authors applied an idea used within adaptive MCMC methods [Andrieu and Moulines, 2006] to SMC samplers by using variance of parameters estimated from its particle system approximation as the proposal scale for the next iteration, i.e., the covariance matrix of the random-walk move for the b-th sub-block at time t is given by:

$$\begin{aligned} \boldsymbol{\Sigma}_{b,t} &= \sum_{m=1}^{N} \widetilde{W}_{t-1}^{(m)} \left(\boldsymbol{\varrho}_{b,t-1}^{(m)} - \boldsymbol{\mu}_{b,t-1}\right) \left(\boldsymbol{\varrho}_{b,t-1}^{(m)} - \boldsymbol{\mu}_{b,t-1}\right)^T \\ \text{with} \quad \boldsymbol{\mu}_{b,t-1} &= \sum_{m=1}^{N} \widetilde{W}_{t-1}^{(m)} \boldsymbol{\varrho}_{b,t-1}^{(m)} \end{aligned} \tag{1.65}$$

The motivation is that if π_{t-1} is close to π_t (which is recommended for efficient algorithm), then the variance estimated at iteration $t-1$ will provide a sensible scaling at time t. This adaptive Metropolis within Gibbs used in the implementation of the SMC sampler through this thesis is summarized in Algorithm 1.9.

In difficult problems, other approaches could be added in order to have appropriate scaling adaptation; one approach demonstrated in [Jasra et al., 2011] is to simply employ a pair of acceptance rate thresholds and to alter the proposal scale from the simply estimated value whenever the acceptance rate falls outside those threshold values. This scheme is to ensure that the acceptance rates in the Metropolis-Hastings steps did not get too large or small. Through all this thesis, we use this procedure which consists for example to multiply the covariance matrix by 5 (resp. 1/5) if the rate exceeded 0.7 (resp. fell below 0.2).

c) Sequence of backward kernels $\mathcal{L}_t$

The backward kernel $\mathcal{L}_t$ is arbitrary, however as discussed in [Del Moral et al., 2006], it should be optimized with respect to mutation kernel $\mathcal{K}_t$ to obtain good performance. [Del Moral et al., 2006] establish that the backward kernel which minimize the variance of the unnormalized importance weights, W_t, are given by

$$\mathcal{L}_t^{\text{opt}}(\boldsymbol{\theta}_{t+1}, \boldsymbol{\theta}_t) = \frac{\eta_t(\boldsymbol{\theta}_t)\mathcal{K}_{t+1}(\boldsymbol{\theta}_t, \boldsymbol{\theta}_{t+1})}{\eta_{t+1}(\boldsymbol{\theta}_{t+1})} \tag{1.66}$$

However, as discussed in Section 1.3.2.2, it is typically impossible to use these optimal kernels as they rely on marginal distributions defined in Eq. (1.46) which do not admit any closed form expression, especially if an MCMC kernel is used as $\mathcal{K}_t$ which is π_t-invariant distribution. Thus we can either choose to approximate $\mathcal{L}_t^{\text{opt}}$ or choose kernels $\mathcal{L}_t$ so that the importance weights are easily calculated or have a familiar form. As discussed in [Del Moral et al., 2006], if an MCMC kernel is used as forward kernel, the following $\mathcal{L}_t$ is employed

$$\mathcal{L}_{t-1}(\boldsymbol{\theta}_t, \boldsymbol{\theta}_{t-1}) = \frac{\pi_t(\boldsymbol{\theta}_{t-1})\mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)}{\pi_t(\boldsymbol{\theta}_t)} \tag{1.67}$$

which is a good approximation of the optimal backward if the discrepancy between π_t and π_{t-1} is small; note that (1.67) is the reversal Markov kernel associated with $\mathcal{K}_t$. In this case, the unnormalized incremental weights becomes

$$w_t^{(m)}(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t^{(m)}) = \frac{\gamma_t(\boldsymbol{\theta}_{t-1}^{(m)})}{\gamma_{t-1}(\boldsymbol{\theta}_{t-1}^{(m)})} = p(\mathbf{y}|\boldsymbol{\theta}_{t-1}^{(m)})^{(\phi_t - \phi_{t-1})} \tag{1.68}$$

This expression (1.68) is remarkably easy to compute and valid regardless of the MCMC kernel adopted. Note that $\phi_t - \phi_{t-1}$ is the step length of the cooling schedule of the likelihood at time t. As we choose this step larger, the discrepancy between π_t and π_{t-1} increases, leading to increase the variance of the importance approximation

when it deteriorates. Thus, it is important to construct a smooth sequence of distributions $\{\pi_t\}_{0\leq t\leq T}$ by judicious choice of an associated real sequence $\{\phi_t\}_{t=0}^{T}$.

Let us remark that when such backward kernel is used, the unnormalized incremental weights in Eq. (1.68) at time t does not depend on the particle value at time t but just on the previous particle set. As suggested in [Del Moral et al., 2006], in such case, the particles $\left\{\boldsymbol{\theta}_t^{(m)}\right\}$ should be sampled after the weights $\left\{W_t^{(m)}\right\}$ have been computed and after the particle approximation $\left\{W_t^{(m)}, \boldsymbol{\theta}_{t-1}^{(m)}\right\}$ has possibly been resampled. The benefit of using such strategy in that case will be demonstrated in the next chapter by studying the asymptotic variances of the SMC estimates.

Based on this discussion regarding the different choices, the SMC sampler that will be used for Bayesian inference in the following chapters is summarized in Algorithm 1.8.

Algorithm 1.8 SMC Sampler Algorithm

1: Initialize particle system

2: Sample $\left\{\boldsymbol{\theta}_1^{(m)}\right\}_{m=1}^{N} \sim \eta_1(\cdot)$ and compute $\widetilde{W}_1^{(m)} = \left(\frac{\gamma_1(\boldsymbol{\theta}_1^{(m)})}{\eta_1(\boldsymbol{\theta}_1^{(m)})}\right)\left[\sum_{j=1}^{N}\frac{\gamma_1(\boldsymbol{\theta}_1^{(j)})}{\eta_1(\boldsymbol{\theta}_1^{(j)})}\right]^{-1}$ and do resampling if $\mathbb{ESS} < \overline{\mathbb{ESS}}$

3: **for** $t = 2, \ldots, T$ **do**

4: Computation of the weights: for each $m = 1, \ldots, N$

$$W_t^{(m)} = \widetilde{W}_{t-1}^{(m)} p(\mathbf{y}|\boldsymbol{\theta}_{t-1}^{(m)})^{(\phi_t - \phi_{t-1})}$$

Normalization of the weights : $\widetilde{W}_t^{(m)} = W_t^{(m)}\left[\sum_{j=1}^{N} W_t^{(j)}\right]^{-1}$

5: Selection: if $ESS < \overline{\mathbb{ESS}}$ then Resample

6: Mutation: for each $m = 1, \ldots, N$: Sample $\boldsymbol{\theta}_t^m \sim \mathcal{K}_t(\boldsymbol{\theta}_{t-1}^{(m)}; \cdot)$ where $\mathcal{K}_t(\cdot;\cdot)$ is a $\pi_t(\cdot)$ invariant Markov kernel described in more details in Algo. 1.9.

7: **end for**

1.4 Conclusion

In this chapter, the objectives of Bayesian inference are firstly presented. Then, classical Monte-Carlo techniques like rejection sampling, importance sampling and MCMC methods are reviewed. Since we are interested in this study on Bayesian inference solutions for complex systems with high dimensional and/or multimodal posterior distribution, we discuss the two major population-based simulation techniques that have been established in order to obtain more robust and efficient Monte Carlo algorithms for efficiently exploring such distributions: the population-based MCMC and the SMC sampler.

Although this SMC sampler approach presents many advantages over traditional MCMC methods, the potential of these emergent techniques is however largely underexploited in signal processing. In this chapter, we therefore give a larger

Algorithm 1.9 Adaptive Metropolis-within-Gibbs Kernel $\mathcal{K}_t(\cdot;\cdot)$ for the m-th particle

Initialization Set $\boldsymbol{\theta}^0 = [\boldsymbol{\varrho}_1^0, \ldots, \boldsymbol{\varrho}_B^0] = \boldsymbol{\theta}_{t-1}^{(m)} = [\boldsymbol{\varrho}_{1,t-1}^{(m)}, \ldots, \boldsymbol{\varrho}_{B,t-1}^{(m)}]$
for $i = 1, \ldots, N_{\text{MCMC}}$ **do**
  for $b = 1, \ldots, B$ **do**
    Sample $\boldsymbol{\varrho}_b^* \sim \mathcal{N}\left(\boldsymbol{\varrho}_b^{i-1}, \boldsymbol{\Sigma}_{b,t}\right)$ with $\boldsymbol{\Sigma}_{b,t}$ defined in Eq. 1.65
    Compute the Acceptance ratio:

$$\alpha(\boldsymbol{\varrho}_b^*, \boldsymbol{\varrho}_b^{i-1}) = \min\left\{1, \frac{p(\mathbf{y}|\boldsymbol{\theta}^*)^{\phi_t} p(\boldsymbol{\theta}^*)}{p(\mathbf{y}|\boldsymbol{\theta}^{i-1})^{\phi_t} p(\boldsymbol{\theta}^{i-1})}\right\}$$

    with $\boldsymbol{\theta}^* = [\boldsymbol{\varrho}_1^i, \ldots, \boldsymbol{\varrho}_{b-1}^i, \boldsymbol{\varrho}_b^*, \boldsymbol{\varrho}_{b+1}^{i-1}, \ldots, \boldsymbol{\varrho}_{B,t-1}^0]$ and $\boldsymbol{\theta}^{i-1} = [\boldsymbol{\varrho}_1^i, \ldots, \boldsymbol{\varrho}_{b-1}^i, \boldsymbol{\varrho}_b^{i-1}, \boldsymbol{\varrho}_{b+1}^{i-1}, \ldots, \boldsymbol{\varrho}_{B,t-1}^0]$
    Sample random variate u from $\mathcal{U}(0,1)$
    if $u \leq \alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}^{i-1})$ **then**
      $\boldsymbol{\varrho}_b^i = \boldsymbol{\varrho}_b^*$
    else
      $\boldsymbol{\varrho}_b^i = \boldsymbol{\varrho}_b^{i-1}$
    end if
  end for
end for
Set the new particle value at time t as $\boldsymbol{\theta}_t^{(m)} = [\boldsymbol{\varrho}_1^{N_{\text{MCMC}}}, \ldots, \boldsymbol{\varrho}_B^{N_{\text{MCMC}}}]$

focus on SMC sampler and explain how the main parameters of the algorithm can be chosen for having an efficient Bayesian solution. In the next chapter, some novel strategies in order to improve and to facilitate the use of SMC sampler in practice will be derived.

Chapter 2

Variance Reduction Schemes for SMC Samplers

Contents

As discussed in the previous chapter, the SMC sampler is a promising Bayesian inference technique that possesses good convergence properties. In this chapter, we will focus on the derivation of variance reduction schemes for SMC samplers. Firstly, we will study the asymptotic variance of the SMC samplers in order to understand the impact on the choice of the sequence of target distribution. From these derivations, an automatic scheme for choosing this sequence will be derived. Finally, recycling approaches of all simulated particles in the SMC sampler will be proposed in order to reduce the variance of the approximation of the target distribution. Performance of the different propositions will be illustrated through numerical simulations with different models.

2.1 Theoretical Analysis of SMC Samplers

2.1.1 General convergence results

In this section, we present the convergence results of SMC samplers derived in [Del Moral et al., 2006]. More specifically, the authors derive the convergence results of the following two estimates obtained using an SMC sampler:

1.
$$\mathbb{E}_{\pi_t^N}(\varphi) = \int_E \varphi(\boldsymbol{\theta})\pi_t^N(d\boldsymbol{\theta}) \tag{2.1}$$
as approximate of
$$\mathbb{E}_{\pi_t}(\varphi) = \int_E \varphi(\boldsymbol{\theta})\pi_t(\boldsymbol{\theta})d\boldsymbol{\theta} \tag{2.2}$$

2.
$$\log\left(\widehat{\frac{Z_t}{Z_1}}\right) = \sum_{k=2}^{t} \log\left(\widehat{\frac{Z_k}{Z_{k-1}}}\right) \tag{2.3}$$
as approximate of the normalizing constant of the target distribution at time t.

In particular, a central limit theorem is presented which gives the asymptotic variance of these estimators in two "extreme" cases: when we never resample and when we resample using multinomial resampling at each iteration.

When no resampling is performed, the following convergence results are obtained [Del Moral et al., 2006]:

1. For the expectation estimator:
$$N^{\frac{1}{2}}\left\{\mathbb{E}_{\pi_t^N}(\varphi) - \mathbb{E}_{\pi_t}(\varphi)\right\} \Rightarrow \mathcal{N}(0, \sigma_{IS,t}^2(\varphi)) \tag{2.4}$$
with
$$\sigma_{IS,t}^2(\varphi) = \int \frac{\tilde{\pi}_t^2(\boldsymbol{\theta}_{1:t})}{\eta_t(\boldsymbol{\theta}_{1:t})}\left\{\varphi(\boldsymbol{\theta}_t) - \mathbb{E}_{\pi_t}(\varphi)\right\}^2 d\boldsymbol{\theta}_{1:t} \tag{2.5}$$

2. For the normalizing constant estimator:
$$N^{\frac{1}{2}}\left\{\log\left(\widehat{\frac{Z_t}{Z_1}}\right) - \log\left(\frac{Z_t}{Z_1}\right)\right\} \Rightarrow \mathcal{N}(0, \sigma_{IS,t}^2) \tag{2.6}$$
with
$$\sigma_{IS,t}^2 = \int \frac{\tilde{\pi}_t(\boldsymbol{\theta}_{1:t})^2}{\eta_t(\boldsymbol{\theta}_{1:t})} d\boldsymbol{\theta}_{1:t} - 1 \tag{2.7}$$

where $\Rightarrow$ denotes the convergence in distribution. When multinomial resampling is performed at each iteration of the SMC sampler, the following convergence results are obtained:

1. For the expectation estimator:
$$N^{\frac{1}{2}}\left\{\mathbb{E}_{\pi_t^N}(\varphi) - \mathbb{E}_{\pi_t}(\varphi)\right\} \Rightarrow \mathcal{N}(0, \sigma_{SMC,t}^2(\varphi)) \tag{2.8}$$

with

$$\begin{aligned}\sigma^2_{SMC,t}(\varphi) = &\int \frac{\tilde{\pi}_t^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)} \left\{ \int \varphi(\boldsymbol{\theta}_t) \tilde{\pi}_t(\boldsymbol{\theta}_t|\boldsymbol{\theta}_1) d\boldsymbol{\theta}_t - \mathbb{E}_{\pi_t}(\varphi) \right\}^2 d\boldsymbol{\theta}_1 \\ &+ \sum_{k=2}^{t-1} \int \frac{\tilde{\pi}_t^2(\boldsymbol{\theta}_k) \mathcal{L}_{k-1}^2(\boldsymbol{\theta}_k, \boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1}) \mathcal{K}_k(\boldsymbol{\theta}_{k-1}, \boldsymbol{\theta}_k)} \left\{ \int \varphi(\boldsymbol{\theta}_t) \tilde{\pi}_t(\boldsymbol{\theta}_t|\boldsymbol{\theta}_k) d\boldsymbol{\theta}_t - \mathbb{E}_{\pi_t}(\varphi) \right\}^2 d\boldsymbol{\theta}_{k-1:k} \\ &+ \int \frac{\pi_t^2(\boldsymbol{\theta}_t) \mathcal{L}_{t-1}^2(\boldsymbol{\theta}_t, \boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1}) \mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)} \left\{ \varphi(\boldsymbol{\theta}_t) - \mathbb{E}_{\pi_t}(\varphi) \right\}^2 d\boldsymbol{\theta}_{t-1:t}\end{aligned} \tag{2.9}$$

2. For the normalizing constant estimator:

$$N^{\frac{1}{2}} \left\{ \log\left(\widehat{\frac{Z_t}{Z_1}}\right) - \log\left(\frac{Z_t}{Z_1}\right) \right\} \Rightarrow \mathcal{N}(0, \sigma^2_{SMC,t}) \tag{2.10}$$

with

$$\begin{aligned}\sigma^2_{SMC,t} = &\int \frac{\tilde{\pi}_t^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)} d\boldsymbol{\theta}_1 - 1 + \sum_{k=2}^{t-1} \left[\int \frac{\tilde{\pi}_t^2(\boldsymbol{\theta}_k) \mathcal{L}_{k-1}^2(\boldsymbol{\theta}_k, \boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1}) \mathcal{K}_k(\boldsymbol{\theta}_{k-1}, \boldsymbol{\theta}_k)} d\boldsymbol{\theta}_{k-1:k} - 1 \right] \\ &+ \int \frac{\pi_t^2(\boldsymbol{\theta}_t) \mathcal{L}_{t-1}^2(\boldsymbol{\theta}_t, \boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1}) \mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)} d\boldsymbol{\theta}_{t-1:t} - 1\end{aligned} \tag{2.11}$$

where the notations used in the results are:

$$\tilde{\pi}_t(\boldsymbol{\theta}_k) = \int \tilde{\pi}_t(\boldsymbol{\theta}_{1:t}) d\boldsymbol{\theta}_{1:k-1} d\boldsymbol{\theta}_{k+1:t} \tag{2.12}$$

$$\tilde{\pi}_t(\boldsymbol{\theta}_t|\boldsymbol{\theta}_k) = \frac{\int \tilde{\pi}_t(\boldsymbol{\theta}_{1:t}) d\boldsymbol{\theta}_{1:k-1} d\boldsymbol{\theta}_{k+1:t-1}}{\tilde{\pi}_t(\boldsymbol{\theta}_k)} \tag{2.13}$$

These results are very general since no assumption has been made on the various choices required in the design of the SMC sampler. As a consequence, it is particularly difficult to analyze these results from a practical point of view. In the next section, we will thus derive these results for some specific choices of the backward kernel which is the one used in practice when an MCMC kernel is used as forward kernel.

2.1.2 Specific convergence results

As discussed in the previous chapter, one of the main attractive properties of the SMC sampler is to be able to use some local moves (using an MCMC kernel) in order to draw the particles at the next iteration. Such local moves are particularly interesting when the state of interest is high-dimensional. As discussed in Section 1.3.2.2, when such an MCMC kernel is used as forward kernel in the SMC sampler, $\mathcal{K}_t$, the backward kernel used in order to be able to compute the incremental weight

is generally:

$$\mathcal{L}_{t-1}(\boldsymbol{\theta}_t, \boldsymbol{\theta}_{t-1}) = \frac{\pi_t(\boldsymbol{\theta}_{t-1})\mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)}{\pi_t(\boldsymbol{\theta}_t)} \tag{2.14}$$

Let us remark once again that if such forward kernel is used, since the incremental weights given in Eq. (1.68) at iteration t are independent of the particles $\{\boldsymbol{\theta}_t\}$, the resampling step could be performed before the sampling.

As a consequence, we derive the asymptotic variance of the SMC sampler estimators with this specific choice of backward kernel defined in Eq. (2.14. Moreover, in order to obtain convergence results that are easy to analyze and utilize, we assume that the MCMC kernel used is perfectly mixing, i.e.,:

$$\mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t) = \pi_t(\boldsymbol{\theta}_t) \tag{2.15}$$

Under these two assumptions, we derive the asymptotic variance of the SMC sampler estimates in the two extreme cases studied in [Del Moral et al., 2006]: never resampling and resampling after the sampling step at each iteration. Moreover, we derive the asymptotic variance when resampling is performed before the sampling stage at each iteration.

2.1.2.1 Case 1: Never resampling

Proposition 2.1.1 *Under perfect mixing assumption and if the backward kernel given in Eq. (2.14) is used, we obtain the following results:*

1. *For the expectation estimator:*

$$N^{\frac{1}{2}}\left\{\mathbb{E}_{\pi_t^N}(\varphi) - \mathbb{E}_{\pi_t}(\varphi)\right\} \Rightarrow \mathcal{N}(0, \sigma^2_{IS,t}(\varphi)) \tag{2.16}$$

with

$$\sigma^2_{IS,t}(\varphi) = \int \frac{\pi_2^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)} d\boldsymbol{\theta}_1 \prod_{k=3}^{t} \int \frac{\pi_k^2(\boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1})} d\boldsymbol{\theta}_{k-1} \underbrace{\left\{\mathbb{E}_{\pi_t}(\varphi^2(\boldsymbol{\theta})) - \mathbb{E}^2_{\pi_t}(\varphi(\boldsymbol{\theta}))\right\}}_{\mathbb{V}ar_{\pi_t}(\varphi(\boldsymbol{\theta}))} \tag{2.17}$$

2. *For the normalizing constant estimator:*

$$N^{\frac{1}{2}}\left\{\log\left(\frac{\widehat{Z_t}}{Z_1}\right) - \log\left(\frac{Z_t}{Z_1}\right)\right\} \Rightarrow \mathcal{N}(0, \sigma^2_{IS,t}) \tag{2.18}$$

with

$$\sigma^2_{IS,t} = \left(\int \frac{\pi_2^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)} d\boldsymbol{\theta}_1\right) \prod_{k=3}^{t} \int \frac{\pi_k^2(\boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1})} d\boldsymbol{\theta}_{k-1} - 1 \tag{2.19}$$

The proof of Proposition 2.1.1 is given in Appendix A.

2.1.2.2 Case 2: Resampling after the sampling at each iteration

Proposition 2.1.2 *Under perfect mixing assumption and if the backward kernel given in Eq. (2.14) is used, we obtain the following results:*

1. *For the expectation estimator:*

$$N^{\frac{1}{2}}\left\{\mathbb{E}_{\pi_t^N}(\varphi)-\mathbb{E}_{\pi_t}(\varphi)\right\}\Rightarrow\mathcal{N}(0,\sigma^2_{SMC,t}(\varphi)) \tag{2.20}$$

with

$$\sigma^2_{SMC,t}(\varphi)=\int\frac{\pi_t^2(\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})}d\boldsymbol{\theta}_{t-1}\underbrace{\left\{\mathbb{E}_{\pi_t}(\varphi^2(\boldsymbol{\theta}))-\mathbb{E}^2_{\pi_t}(\varphi(\boldsymbol{\theta}))\right\}}_{\mathbb{V}ar_{\pi_t}(\varphi(\boldsymbol{\theta}))} \tag{2.21}$$

2. *For the normalizing constant estimator:*

$$N^{\frac{1}{2}}\left\{\log\left(\frac{\widehat{Z_t}}{Z_1}\right)-\log\left(\frac{Z_t}{Z_1}\right)\right\}\Rightarrow\mathcal{N}(0,\sigma^2_{SMC,t}) \tag{2.22}$$

with

$$\begin{aligned}\sigma^2_{SMC,t}=\int\frac{\pi_2^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)}d\boldsymbol{\theta}_1 \quad &+\sum_{k=2}^{t-1}\int\frac{\pi_{k+1}^2(\boldsymbol{\theta}_k)}{\pi_k(\boldsymbol{\theta}_k)}d\boldsymbol{\theta}_k\int\frac{\pi_k^2(\boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1})}d\boldsymbol{\theta}_{k-1}\\ &+\int\frac{\pi_t^2(\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})}d\boldsymbol{\theta}_{t-1}-t\end{aligned} \tag{2.23}$$

The proof of Proposition 2.1.2 is given in Appendix B.

2.1.2.3 Case 3: Resampling before the sampling at each iteration

Proposition 2.1.3 *Under perfect mixing assumption and if the backward kernel given in Eq. (2.14) is used, we obtain the following results:*

1. *For the expectation estimator:*

$$N^{\frac{1}{2}}\left\{\mathbb{E}_{\pi_t^N}(\varphi)-\mathbb{E}_{\pi_t}(\varphi)\right\}\Rightarrow\mathcal{N}(0,\sigma^2_{SMC_2,t}(\varphi)) \tag{2.24}$$

with

$$\sigma^2_{SMC_2,t}(\varphi)=\left\{\mathbb{E}_{\pi_t}(\varphi^2(\boldsymbol{\theta}))-\mathbb{E}^2_{\pi_t}(\varphi(\boldsymbol{\theta}))\right\}=\mathbb{V}ar_{\pi_t}(\varphi(\boldsymbol{\theta})) \tag{2.25}$$

2. *For the normalizing constant estimator:*

$$N^{\frac{1}{2}}\left\{\log\left(\frac{\widehat{Z_t}}{Z_1}\right)-\log\left(\frac{Z_t}{Z_1}\right)\right\}\Rightarrow\mathcal{N}(0,\sigma^2_{SMC_2,t}) \tag{2.26}$$

with

$$\sigma^2_{SMC_2,t} = \int \frac{\pi_2^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)} d\boldsymbol{\theta}_1 \quad + \sum_{k=2}^{t-1} \int \frac{\pi_{k+1}^2(\boldsymbol{\theta}_k)}{\pi_k(\boldsymbol{\theta}_k)} d\boldsymbol{\theta}_k - (t-1) \tag{2.27}$$

The proof of Proposition 2.1.3 is given in Appendix C.

2.1.2.4 Discussion on the convergence results

We can firstly remark that in all the three cases, all the asymptotic variances of the SMC sampler depend on a measure of difference between two probability density functions of the form:

$$I_k = \int \frac{\pi_k^2(\boldsymbol{\theta})}{\pi_{k-1}(\boldsymbol{\theta})} d\boldsymbol{\theta} \tag{2.28}$$

This dissimilarity between two distributions can be related to the Rényi divergence, defined as [Gil et al., 2013]:

$$D_\alpha(f_1||f_2) = \frac{1}{\alpha - 1} \log \int f_1^\alpha(x) f_2^{1-\alpha}(x) dx \geq 0 \tag{2.29}$$

We have indeed,

$$I_k = \exp\left(D_2(\pi_k||\pi_{k-1})\right) \geq 1 \tag{2.30}$$

Since $I_k \geq 1$, we can therefore conclude that under the assumption of perfect mixing and the choice of the backward kernel given in Eq. (2.14), we have the following inequalities between the variances of the different variants of the SMC sampler:

1. For the expectation estimator:

$$\sigma^2_{SMC_2,t}(\varphi) \leq \sigma^2_{SMC,t}(\varphi) \leq \sigma^2_{IS,t}(\varphi)$$

2. For the normalizing constant estimator:

$$\sigma^2_{SMC_2,t} \leq \sigma^2_{SMC,t}$$

From these expressions, we can clearly see that it is advisable, as expected, to do the resampling before sampling the new particles as the associated asymptotic variance is smaller than the SMC sampler with a resampling performed after. From now, we will thus focus on the scheme in which the resampling step is performed before the sampling stage. For this approach, we can conclude that even if a perfect mixing MCMC kernel is used, the variance of the estimator associated with the normalizing constant in Eq. (2.27) still depends on all the sequence of target distributions as a cumulative sum of the discrepancy between two consecutive target distributions. In the next section, we will use this result in order to design an automatic procedure for the selection of the sequence of target distributions.

2.2 Adaptive Sequence of Target Distributions

2.2.1 Existing approaches

Several statistical approaches have been proposed in order to improve the choice of the sequence of target distributions via some criteria which are known as *on-line* schemes. [Jasra et al., 2011] proposed adaptive selection methods based on controlling the rate of the effective sample size ($\mathbb{ESS}_t$), defined in Eq. (1.56). This scheme thus provides an automatic method to obtain the tempering schedule such that the $\mathbb{ESS}$ decays in a regular predefined way. However, the $\mathbb{ESS}_t$ of the current sample weights corresponds to some empirical measure of the accumulated discrepancy between the proposal and the target distribution since the last resampling time. As a consequence, it does not really represent the dissimilarity between each pair of successive distributions unless resampling is conducted after every iteration.

In order to handle this problem, [Zhou et al., 2013] proposed a slight modification of the $\mathbb{ESS}$, named the *conditional* $\mathbb{ESS}$ ($\mathbb{CESS}$), by considering how good an importance sampling proposal π_{t-1} would be for the estimation of expectation under π_t. At the t-th iteration, this quantity is defined as follows:

$$\mathbb{CESS}_t = \left[\sum_{i=1}^{N} N\widetilde{W}_{t-1}^{(i)} \left(\frac{w_t^{(i)}}{\sum_{j=1}^{N} N\widetilde{W}_{t-1}^{(j)} w_t^{(j)}}\right)^2\right]^{-1} = \frac{\left(\sum_{i=1}^{N} \widetilde{W}_{t-1}^{(i)} w_t^{(i)}\right)^2}{\sum_{j=1}^{N} \frac{1}{N}\widetilde{W}_{t-1}^{(j)} (w_t^{(j)})^2} \tag{2.31}$$

Nevertheless, by using either $\mathbb{ESS}$ or $\mathbb{CESS}$ criterion, the number of steps T of the SMC Samplers completely depends on the complexity of the integration problem at hand and could not be known in advance. In other words, for either fixed $\mathbb{ESS}^\star$ or fixed $\mathbb{CESS}^\star$, the associated sequence $\{\phi_t\}_{t=1}^T$ is a on-line self-tuning parameter. Smaller values significantly speed up the Sequential Monte Carlo algorithm but lead to a higher variation in the results. Consequently, we are not able to control the total complexity of the algorithm, and it is typically impossible to obtain the comprehensive view of the behavior of the cooling schedule $\{\phi_t\}$ before the algorithm is conducted.

2.2.2 Proposed Approach

In this work, we propose an alternative strategy to choose the sequence of target distributions adaptively to the current problem to deal with. In particular, we propose to consider the sequence of distributions which minimizes the variance of the particle approximation of the normalizing constant derived previously in Eq. (2.27). This strategy is thus based on a global optimization of cooling schedule $\{\phi_t\}$ which enable us to control the complexity of the algorithm by determining before any simulation the number of SMC iterations T. In this way we obtain what will be referred to as *off-line* scheme, and we will obtain the complete view of cooling schedule performance before starting the SMC sampler.

By carrying out our criterion, we have to find $T-1$ positive step lengths $\boldsymbol{\varrho} = \{\varrho_t\}_{t=2}^T$, defined as $\phi_t - \phi_{t-1}$ such that $\sum_{t=2}^T \varrho_t = 1$, which minimize the asymptotic variance given in Eq. (2.27). Here, we are aiming at finding

$$\begin{aligned} \widehat{\boldsymbol{\varrho}} = \{\widehat{\varrho}_2, \ldots, \widehat{\varrho}_T\} &= \underset{\{\varrho_2,\ldots,\varrho_T\}}{\arg\min} \sum_{t=1}^{T-1} \int \frac{\pi_{t+1}^2(\boldsymbol{\theta}_t)}{\pi_t(\boldsymbol{\theta}_t)} d\boldsymbol{\theta}_t - (T-1) \\ &\text{subject to } \sum_{t=2}^{T} \varrho_t = 1 \text{ and } \forall m = 2, \ldots, T : \varrho_m \geq 0 \end{aligned} \tag{2.32}$$

where

$$\pi_t(\theta) = \frac{p(y|\theta)^{\phi_t} p(\theta)}{\int p(y|\theta)^{\phi_t} p(\theta) d\theta} = \frac{p(y|\theta)^{\phi_t} p(\theta)}{Z_t} \text{ with } \phi_t = \sum_{m=2}^{t} \varrho_m \tag{2.33}$$

Equation (2.32) involves $T-1$ integrals and each integral represents, as discussed in Section 2.1.2.4, a dissimilarity measure between each pair of successive distributions. The main difficulty in carrying out this construction is that these integrals are generally intractable, so numerical methods are required.

In order to avoid the use of numerical methods to approximate the $T-1$ integrals which could be very challenging to do if $\boldsymbol{\theta}$ is high-dimensional, we propose instead to approximate each target distribution $\pi_t(\boldsymbol{\theta})$ by a multivariate normal distribution. By doing that, an analytical expression for the asymptotic variance to minimizes can be obtained and thus evaluated for a specific set of values for the step lengths. Indeed, from the connection between these integrals and the Rényi divergence, we have [Gil et al., 2013]:

For Gaussian multivariate distribution $f_1 = \mathcal{N}(\mu_1, \Sigma_1)$ and $f_2 = \mathcal{N}(\mu_2, \Sigma_2)$ we have

$$\begin{aligned} \int f_1^{\alpha}(x) f_2^{1-\alpha}(x) dx = &\frac{\det\left(\alpha\Sigma_2 + (1-\alpha)\Sigma_1\right)^{-\frac{1}{2}}}{\det(\Sigma_1)^{\frac{\alpha-1}{2}} \det(\Sigma_2)^{-\frac{\alpha}{2}}} \\ &\times \exp\left\{\frac{\alpha(\alpha-1)}{2}(\mu_1-\mu_2)^T(\alpha\Sigma_2 + (1-\alpha)\Sigma_1)^{-1}(\mu_1-\mu_2)\right\} \end{aligned} \tag{2.34}$$

which is finite iff $\alpha\Sigma_1^{-1} + (1-\alpha)\Sigma_2^{-1}$ is positive definite.

Finally, a nonlinear optimization technique, such as for example the Nelder-Mead algorithm [Nelder and Mead, 1965], can be used to solve this optimization in order to obtain the value $\widehat{\boldsymbol{\varrho}}$. Let us now describe how the multivariate normal approximation of each target could be done in an efficient way.

Normal approximation of each target distribution

In order to find the value $\widehat{\boldsymbol{\varrho}}$ that minimizes the asymptotic variance of the estimate of the normalizing constant, we need to approximate the T intermediate

target distributions, π_t for $t = 1, \cdots, T$ by multivariate normal distributions, i.e.,:

$$\begin{aligned} \pi_t(\boldsymbol{\theta}) &\propto p(\mathbf{y}|\boldsymbol{\theta})^{\phi_t} p(\boldsymbol{\theta}) \\ &\approx \mathcal{N}(\boldsymbol{\theta}|\mu_t, \Sigma_t) \end{aligned} \tag{2.35}$$

In order to reduce the complexity associated with these T different normal approximations of the intermediate target distributions (which consists in finding both T mean vectors $\{\mu_t\}_{t=1}^T$ and covariance matrices $\{\Sigma_t\}_{t=1}^T$), we propose to only approximate the prior and the posterior $p(\boldsymbol{\theta}|\mathbf{y})$ distribution and thus deduce all normal approximation required by using the convenient properties of the normal distribution.

Indeed, approximating both the prior and the posterior by normal distributions with parameters $(\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)$ and $(\boldsymbol{\mu}_T, \boldsymbol{\Sigma}_T)$ respectively, leads to a normal likelihood approximation with

$$\begin{aligned} \boldsymbol{\Sigma}_l &= \left(\boldsymbol{\Sigma}_T^{-1} - \boldsymbol{\Sigma}_p^{-1}\right)^{-1} \\ \boldsymbol{\mu}_l &= \boldsymbol{\Sigma}_l \left(\boldsymbol{\Sigma}_T^{-1}\boldsymbol{\mu}_T - \boldsymbol{\Sigma}_p^{-1}\boldsymbol{\mu}_p\right) \end{aligned} \tag{2.36}$$

Moreover, since a tempered normal is proportional to a normal with modification of the covariance and that the product of 2 multivariate normals is a multivariate normal distribution, the t-th target distribution is therefore approximated by :

$$\pi_t(\boldsymbol{\theta}) \approx \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t) \tag{2.37}$$

with

$$\begin{aligned} \boldsymbol{\Sigma}_t &= \left(\boldsymbol{\Sigma}_p^{-1} + \phi_t \boldsymbol{\Sigma}_l^{-1}\right)^{-1} \\ \boldsymbol{\mu}_t &= \boldsymbol{\Sigma}_t \left(\boldsymbol{\Sigma}_p^{-1}\boldsymbol{\mu}_p + \phi_t \boldsymbol{\Sigma}_l^{-1}\boldsymbol{\mu}_l\right) \end{aligned} \tag{2.38}$$

Only the prior and the posterior require normal approximations which can be performed using either the Laplace's method (which requires to be able to compute the first and second derivatives) or a simulation-based moment matching technique (e.g., using random draws from a simple importance sampler).

2.3 Scheme for Recycling all past simulated particles

In the previous section, we propose a strategy in order to automatically specify the sequence of target distributions in order to obtain an estimator of the normalizing constant with the smallest variance. Now, in this section, we focus on the design of a strategy to reduce the variance of the SMC expectation estimator with respect to $\pi(\cdot)$. With SMC samplers, this quantity is approximated, Eq. (1.63), as:

$$J = \mathbb{E}_{\pi}\left[\varphi(\boldsymbol{\theta})\right] = \int \pi(\boldsymbol{\theta})\varphi(\boldsymbol{\theta})d\boldsymbol{\theta} \approx \mathbb{E}_{\pi^N}\left[\varphi(\boldsymbol{\theta})\right] = \sum_{i=1}^{N} \widetilde{W}_T^{(i)} \varphi(\boldsymbol{\theta}_T^{(i)}) \tag{2.39}$$

since $\pi_T(\cdot) = \pi(\cdot)$. Only the samples from the iterations targeting the true posterior (generally only the last one) are taking into account for the approximation of the expectation. In this thesis, in order to reduce the variance associated with this estimator in Eq. (2.39), we propose two different strategies that will use particles drawn at the previous iterations of the sampler by a recycling principle. Let us remark that these two recycling schemes are performed after the SMC sampler is finished.

2.3.1 Recycling based on Effective Sample Size

As discussed above, the SMC approximation of posterior expectation is based on only the samples from the last SMC iteration. In order to have more efficient estimator in the sense of minimizing the variance of the estimator in Eq. (2.39), the idea we propose to explore in this work is to recycle all the particles that have been generated through the T iterations of the SMC sampler. In [Gramacy et al., 2010], a strategy has been proposed in order to recycle all the elements of the Markov chain obtained from a simulated tempering based MCMC algorithm. In this paper, we propose to adapt this approach to the T collections of weighted samples given at each iteration of the SMC sampler. The main idea is to correct each of these T set of weighted random samples by using an importance sampling identity because these samples are not drawn from the distribution of interest $\pi(\cdot)$.

More specifically, at the end of the t-th iteration of the SMC sampler, the weighted particle system approximates the target distribution $\pi_t(\cdot)$ as follows:

$$\pi_t^N(d\boldsymbol{\theta}) \approx \sum_{i=1}^{N} \widetilde{W}_t^{(i)} \delta_{\boldsymbol{\theta}_t^{(i)}}(d\boldsymbol{\theta}) \tag{2.40}$$

However, in order to be able to use importance sampling identity, we need to have a set of unweighted samples from $\pi_t(\boldsymbol{\theta})$. For this purpose, an unbiased resampling step that consists in selecting particles according to their importance weights can be used [Kunsch, 2005]. With a multinomial resampling scheme, we obtain a new collection

$$\left\{\widetilde{\boldsymbol{\theta}}_t^{(i)}\right\}_{i=1}^{N} \sim \pi_t(\boldsymbol{\theta}) \tag{2.41}$$

where for $i = 1, \ldots, N$

$$\widetilde{\boldsymbol{\theta}}_t^{(i)} = \boldsymbol{\theta}_t^{(J_t^i)} \quad \text{with} \quad J_t^i \overset{\text{iid}}{\sim} \mathcal{M}\left(\widetilde{W}_t^{(1)}, \ldots, \widetilde{W}_t^{(N)}\right) \tag{2.42}$$

Let us remark that if the resampling stage has been already performed at a specific iteration of the SMC sampler, the previous described steps are not necessary since

the obtained samples are already asymptotically drawn from the target distribution $\pi_t(\cdot)$ (in this case, we set directly $\widetilde{\boldsymbol{\theta}}_t^{(i)} = \boldsymbol{\theta}_t^{(i)}$ for $i = 1, \ldots, N$). At the end of the SMC sampler, we have T collections of random samples drawn from each distribution of the targeted sequence. Since we know the distribution from which these random samples $\left\{\widetilde{\boldsymbol{\theta}}_t^{(i)}\right\}_{i=1}^{N}$ are sampled, an estimate of the expectation in (2.39) can be obtained by using importance sampling identity:

$$\widehat{h}_t = \sum_{j=1}^{N} \frac{w_{\text{ESS},t}(\widetilde{\boldsymbol{\theta}}_t^{(j)})}{\sum_{i=1}^{N} w_{\text{ESS},t}(\widetilde{\boldsymbol{\theta}}_t^{(i)})} \varphi(\widetilde{\boldsymbol{\theta}}_t^{(i)}) \tag{2.43}$$

with

$$w_{\text{ESS},t}(\widetilde{\boldsymbol{\theta}}_t^{(j)}) = \frac{\gamma(\widetilde{\boldsymbol{\theta}}_t^{(j)})}{\gamma_t(\widetilde{\boldsymbol{\theta}}_t^{(j)})} \tag{2.44}$$

with $\gamma(\cdot)$ and$\gamma_t(\cdot)$ being the unnormalized target distribution at the final iteration (i.e., the posterior) and at the t-iteration, respectively.

Finally, an overall estimator that will take into account all these estimators (or potentially a subset Ω among these T estimates) can be obtained as follows:

$$\widehat{h} = \sum_{t \in \Omega} \lambda_t \widehat{h}_t \tag{2.45}$$

where $0 \leq \lambda_t \leq \sum_{t \in \Omega} \lambda_t = 1$.

As discussed in [Gramacy et al., 2010], the combination coefficients λ_t have to be chosen carefully if we do not want to have the variance of the estimator (2.45) larger than the one without recycling given in Eq. (2.39). For example, a tempting solution is to take for $t = 1, \ldots, T$:

$$\lambda_t = \frac{W_{\text{ESS},t}}{W_{\text{ESS}}} \tag{2.46}$$

with $W_{\text{ESS},t} = \sum_{j=1}^{N} w_{\text{ESS},t}(\widetilde{\boldsymbol{\theta}}_t^{(j)})$ and $W_{\text{ESS}} = \sum_{t=1}^{T} W_{\text{ESS},t}$ but this can lead to very poor estimator as illustrated empirically in the numerical simulation section as the "naive" recycling scheme. The solution proposed by [Gramacy et al., 2010] is thus to find all the λ_t that maximizes the effective sample size of the weights of the entire population of particles. By using Lagrangian multipliers, the optimal λ_t^* developed in [Gramacy et al., 2010] are defined by:

$$\lambda_t^* = \frac{l_t}{\sum_{n=1}^{T} l_n} \qquad \text{with} \quad l_t = \frac{W_{\text{ESS},t}^2}{\sum_{j=1}^{N} w_{\text{ESS},t}(\widetilde{\boldsymbol{\theta}}_t^{(j)})^2} \tag{2.47}$$

Let us remark that the value l_t involved in this optimal coefficients λ_t^* corresponds to the effective sample size of the t-th collection of importance weights given in (2.44) and as a consequence $1 \leq l_t \leq N_t$.

2.3.2 Recycling based on Deterministic Mixture Weights

The second solution we propose in this work is to use the principle of the technique of the *deterministic mixture* weight estimator proposed in [Veach and Guibas, 1995] and discussed by Owen and Zhou in [Owen and Zhou, 2000]. This approach has been derived in order to combine weighted samples obtained from different proposal distributions in the importance sampler framework. More recently, this technique has been used in the Adaptive Multiple Importance Sampling (AMIS) of [Cornuet et al., 2012] in order to recycle all past simulated particles in order to improve the adaptivity and variance of the Population Monte Carlo algorithm. We propose to adapt here this technique to the framework of the SMC sampler.

As discussed at large in [Owen and Zhou, 2000], using a deterministic mixture as a representation of the production of the simulated samples has the potential to exploit the most efficient proposals in the sequence $\eta_1(\boldsymbol{\theta}), \ldots, \eta_T(\boldsymbol{\theta})$ without rejecting any simulated value nor sample, while reducing the variance of the corresponding estimators. The poorly performing proposal functions are simply eliminated through the erosion of their weights:

$$\frac{\pi(\boldsymbol{\theta}_t^{(i)})}{\sum_{n=1}^{T} c_n \eta_n(\boldsymbol{\theta}_t^{(i)})} \tag{2.48}$$

as T increases (with $c_n = N_n / \sum_{t=1}^{T} N_t$ is the proportion of particles drawn from the proposal η_n)[1]. Indeed, if η_1 is the poorly performing proposal, while the η_n's ($n > 1$) are good approximations of the target π, for a value $\boldsymbol{\theta}_1^{(i)}$ such that $\pi(\boldsymbol{\theta}_1^{(i)})/\eta_1(\boldsymbol{\theta}_1^{(i)})$ is large, because $\eta_1(\boldsymbol{\theta}_1^{(i)})$ is small (and not because it is a sample with high posterior value), $\pi(\boldsymbol{\theta}_t^{(i)}) \big/ \{c_1\eta_1(\boldsymbol{\theta}_1^{(i)}) + \ldots + c_T\eta_T(\boldsymbol{\theta}_1^{(i)})\}$ will behave like $\pi(\boldsymbol{\theta}_t^{(i)}) \big/ \{c_2\eta_2(\boldsymbol{\theta}_1^{(i)}) + \ldots + c_T\eta_T(\boldsymbol{\theta}_1^{(i)})\}$ and will decrease to zero as T increases.

In our case, since we are not in the importance sampling framework with well defined proposal distribution but instead with T collections of samples from the intermediate target distribution $\left(\left\{\widetilde{\boldsymbol{\theta}}_1^{(i)}\right\}_{i=1}^{N_1}, \ldots, \left\{\widetilde{\boldsymbol{\theta}}_T^{(i)}\right\}_{i=1}^{N_T}\right)$ by following the same resampling step as described in the previous section in Eq. (2.42), the estimator of an expectation using this proposed deterministic mixture will be given by:

$$\mathbb{E}_\pi[\varphi] \approx \sum_{t=1}^{T}\sum_{i=1}^{N_t} \frac{w_{\text{DeMix},t}^{(i)}}{\sum_{k=1}^{T}\sum_{j=1}^{N_k} \hat{w}_{\text{DeMix},k}^{(j)}} \varphi(\widetilde{\boldsymbol{\theta}}_t^{(i)}) \tag{2.49}$$

with

$$w_{\text{DeMix},t}^{(i)} = \frac{\pi(\widetilde{\boldsymbol{\theta}}_t^{(i)})}{\sum_{n=1}^{T} c_n \pi_n(\widetilde{\boldsymbol{\theta}}_t^{(i)})} \tag{2.50}$$

where $c_n = N_n / \sum_{t=1}^{T} N_t$ is the proportion of particles drawn from π_n amongst all the simulated particles. The problem with this strategy is we need to evaluate

[1] Here we assume the general case in which a different number of particles could be drawn at each iteration of the SMC sampler.

the target $\pi_t(\cdot)$ exactly (not up to a constant) and thus we need to know the normalizing constant Z_t involved in all the intermediate target distributions $\pi_t(\cdot) = \gamma_t(\cdot)/Z_t$. The idea we propose is to use the (unbiased) SMC approximation of each normalizing constant given by Eq. (1.60). As a consequence, the weights of this proposed recycling scheme, defined originally in Eq (2.50), is thus approximated by:

$$w_{\text{DeMix},t}^{(i)} \approx \frac{\gamma(\widetilde{\boldsymbol{\theta}}_t^{(i)})}{\sum_{n=1}^{T} c_n \gamma_n(\widetilde{\boldsymbol{\theta}}_t^{(i)}) \hat{Z}_n^{-1}} \tag{2.51}$$

2.4 Numerical Simulations

In this section, we will assess the performance of the proposed strategies used to improve the estimation of SMC samplers through different models. In the rest of this work, even if the proposed approach to adaptively choose the sequence of target distribution could be performed on the $T-1$ step lengths $\{\varrho_t\}_{t=2}^{T}$, we will simplify this problem by assuming a parametric form: $\phi_t = h(t;\gamma,T)$, which satisfies the following conditions: $\{\phi_t\}$ is non-decreasing function, $\phi_0 = 0$ and $\phi_T = 1$. By doing that, the goal now is to find the optimal value for an unique parameter γ instead of $T-1$ parameters $\{\varrho_t\}_{t=2}^{T}$. The parametric function used for the proposed adaptive cooling schedule strategy in this section is defined as:

$$\phi_t = \frac{\exp(\gamma t/T) - 1}{\exp(\gamma) - 1} \tag{2.52}$$

and is depicted for different value of the parameter in Fig. 2.1.

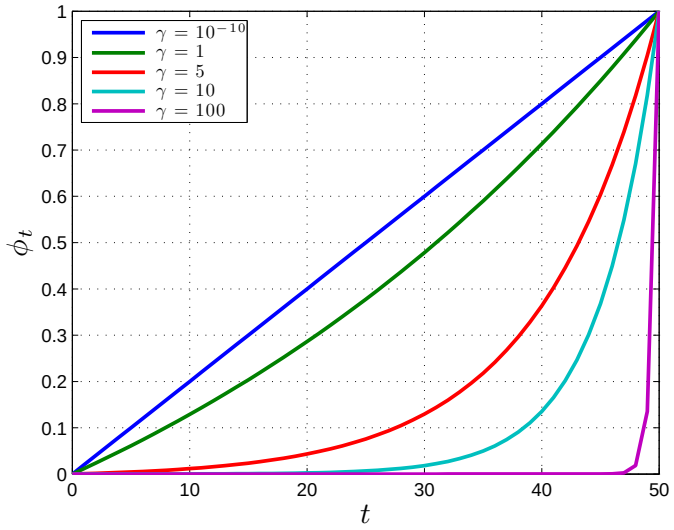


Figure 2.1: Evolution of the parametric function ϕ_t in Eq. 2.52 chosen for the cooling schedule for different values of γ with $T = 50$

Performances of the proposed adaptive cooling strategy and the recycling schemes are now assessed through two different statistical models.

2.4.1 Model 1: Linear and Gaussian Model

Let us firstly consider a linear and Gaussian model for which the a posteriori distribution as well as the marginal likelihood can be derived analytically. The comparison of our proposed strategies to improve SMC samplers can thus be compared with the optimal Bayesian inference method. More precisely, we assume

$$\begin{aligned} p(\boldsymbol{\theta}) &= \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \\ p(\mathbf{y}|\boldsymbol{\theta}) &= \mathcal{N}(\mathbf{y}|\boldsymbol{H}\boldsymbol{\theta}, \boldsymbol{\Sigma}_y) \end{aligned} \tag{2.53}$$

For this model, the posterior distribution is given by :

$$p(\boldsymbol{\theta}|\mathbf{y}) = \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p) \tag{2.54}$$

with

$$\boldsymbol{\mu}_p = \boldsymbol{\mu} + \boldsymbol{\Sigma}\boldsymbol{H}^T \left(\boldsymbol{H}\boldsymbol{\Sigma}\boldsymbol{H}^T + \boldsymbol{\Sigma}_y\right)^{-1} [\mathbf{y} - \boldsymbol{H}\boldsymbol{\mu}] \tag{2.55}$$

$$\boldsymbol{\Sigma}_p = \left(\boldsymbol{I}_{n_{\boldsymbol{\theta}}} - \boldsymbol{\Sigma}\boldsymbol{H}^T \left(\boldsymbol{H}\boldsymbol{\Sigma}\boldsymbol{H}^T + \boldsymbol{\Sigma}_y\right)^{-1} \boldsymbol{H}\right) \boldsymbol{\Sigma} \tag{2.56}$$

In addition, the marginal likelihood (i.e. the normalizing constant) is:

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|\boldsymbol{H}\boldsymbol{\mu}, \boldsymbol{H}\boldsymbol{\Sigma}\boldsymbol{H}^T + \boldsymbol{\Sigma}_y) \tag{2.57}$$

In all the numerical results, we have chosen $\boldsymbol{\Sigma} = 10\boldsymbol{I}_{10}$, $\boldsymbol{\mu} = \mathbf{0}_{10\times 1}$ for the prior distribution. Concerning the likelihood parameters, all the elements of the transition matrix have been randomly generated using a standard normal distribution and $\boldsymbol{\Sigma}_y = \boldsymbol{I}_{n_{\mathbf{y}}}$ with a varying number of observations $n_{\mathbf{y}}$ depending on the figure or table. Regarding to the SMC sampler, and in particular on the adaptive MWG (summarized in Algo. 1.9) used as forward kernel, we have chosen $N_{\text{MCMC}} = 5$ and $B = 5$.

2.4.1.1 Analysis of the proposed adaptive cooling schedule

In this model, the proposed approach is optimal (in the sense of minimizing the asymptotic variance of the normalizing constant) since each intermediate target distribution is a multivariate normal distribution. In Fig. 2.2, the evolution of the theoretical asymptotic variance of the normalizing constant estimator with the parameter value γ is depicted for the 3 different variants of the SMC sampler (Never resampling - Eq. (2.19), Resampling After - Eq. (2.23) , Resampling before - Eq. (2.27)). As discussed in Section 2.1.2.4, the asymptotic variance from the SMC sampler when resampling is performed before is lower than the one in which

resampling is done after. From this figure, we can clearly see that there exists an optimal value of the parametric function of the cooling schedule that will minimize the asymptotic variances.

In Fig. 2.3, we compare the theoretical asymptotic variances of the normalizing constant with the ones obtained by simulation. In order to obtain these results, we have run 500 times an SMC sampler that utilizes a perfect mixing forward kernel which can be straightforwardly obtained analytically for this specific model, i.e.:

$$\begin{aligned} K_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t) &= \pi_t(\boldsymbol{\theta}_t) \propto p(\mathbf{y}|\boldsymbol{\theta})^{\phi_t} p(\boldsymbol{\theta}) \\ &= \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t) \end{aligned} \tag{2.58}$$

with

$$\boldsymbol{\mu}_t = \boldsymbol{\mu} + \boldsymbol{\Sigma}\boldsymbol{H}^T \left(\boldsymbol{H}\boldsymbol{\Sigma}\boldsymbol{H}^T + \frac{1}{\phi_t}\boldsymbol{\Sigma}_y\right)^{-1} [\mathbf{y} - \boldsymbol{H}\boldsymbol{\mu}] \tag{2.59}$$

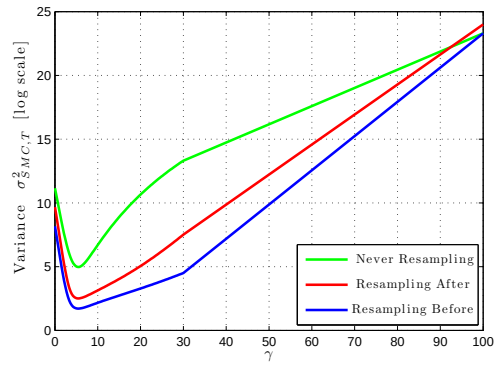
$$\boldsymbol{\Sigma}_t = \left(\boldsymbol{I}_{n_{\boldsymbol{\theta}}} - \boldsymbol{\Sigma}\boldsymbol{H}^T \left(\boldsymbol{H}\boldsymbol{\Sigma}\boldsymbol{H}^T + \frac{1}{\phi_t}\boldsymbol{\Sigma}_y\right)^{-1} \boldsymbol{H}\right) \boldsymbol{\Sigma} \tag{2.60}$$

From Fig. 2.3, we can see that the variance of the normalizing constant estimator for the SMC sampler with a finite number of particles is very close to the theoretical asymptotic value. Especially for the resampling after and resampling before, a quite few number of particles is thus required to reach these asymptotic variances under this model.

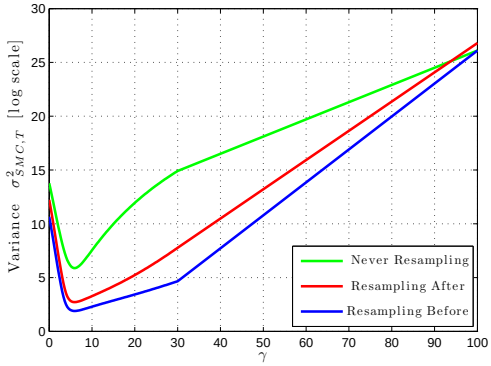
Then, in Fig. 2.4, we compare the proposed approach for adaptive cooling schedule versus the one based on the $\mathbb{CESS}$ and also if the linear cooling schedule is used. Performances of the SMC samplers are illustrated with the use of either the perfect mixing kernel (Eq. 2.58) or when a random walk Metropolis Hastings kernel is employed as forward kernel. These results clearly show the benefit of using such adaptive cooling schedule - a bad choice can lead to a very poor estimate in term of the variance only because the estimate of the normalizing constant obtained from an SMC sampler (Eq. 1.60) is always unbiased (if the resampling procedure is also unbiased). The variance obtained from the proposed approach and the $\mathbb{CESS}$-based one are very close. The main advantage of our proposed approach is that we totally control the global complexity of the SMC sampler since we set the number of iterations of the SMC sampler whereas in the $\mathbb{CESS}$-based strategy, the number of iterations of the SMC samplers will depend on the problem to deal with as well as the predefined value of $\mathbb{CESS}$. In order to be able to compare both approaches with the same complexity, several runs of the SMC sampler with different values of the $\mathbb{CESS}$ have been required to obtain the $\mathbb{CESS}$ value that roughly leads to a specific number of iterations T (25, 50 and 100).

2.4.1.2 Analysis of the proposed recycling schemes

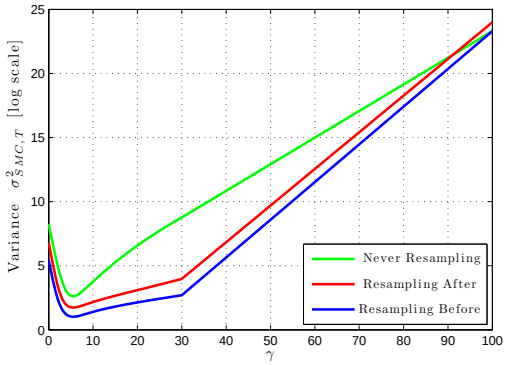
We finally assess the performance of the two proposed recycling schemes. In order to analyze the potential gain of recycling past simulated particles, four different estimators based on the output of the SMC sampler are compared: *"no recycling"*



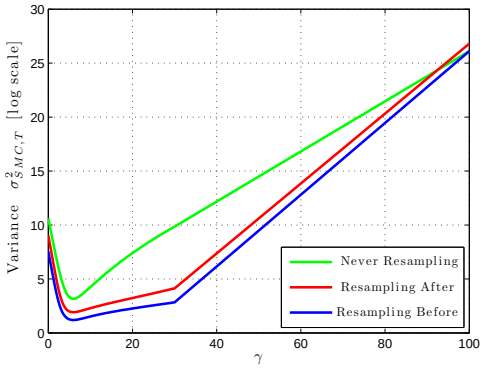
(a) 20 Obs. - 25 Iterations



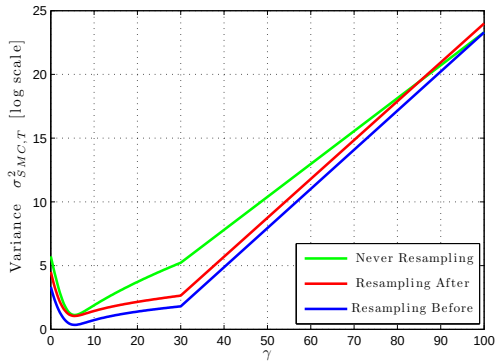
(b) 30 Obs. - 25 Iterations



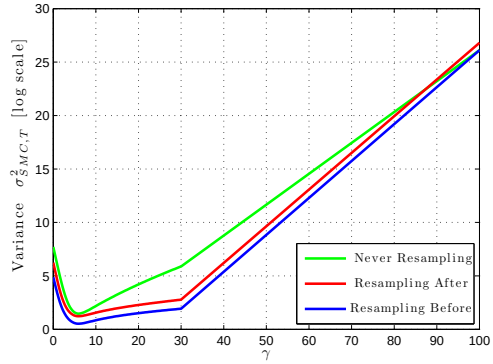
(c) 20 Obs. - 50 Iterations



(d) 30 Obs. - 50 Iterations



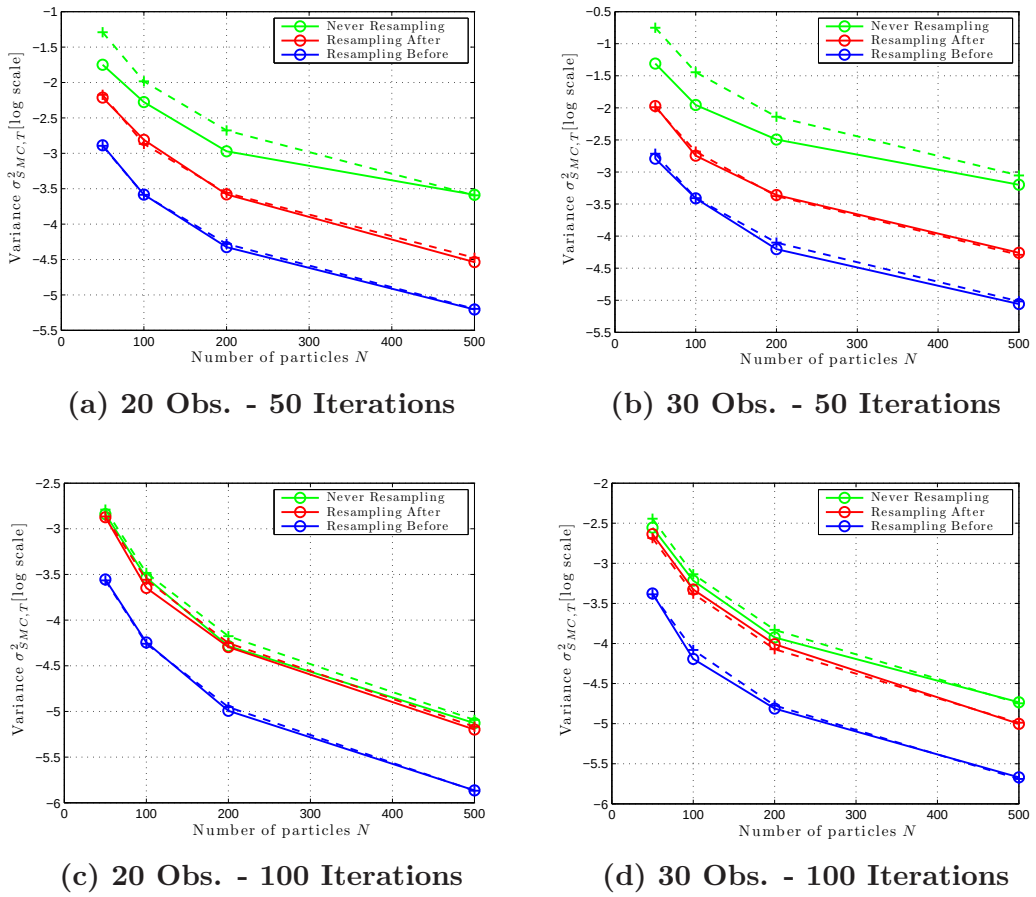
(e) 20 Obs. - 100 Iterations



(f) 30 Obs. - 100 Iterations

Figure 2.2: Evolution of the theoretical asymptotic variance of the SMC sampler estimate of the normalizing constant versus the value of γ in the cooling schedule for 3 different numbers of iterations

given in Eq. (2.39), *"Naïve recycling"* and *"ESS-based recycling"* given in Eq. (2.45)



(a) 20 Obs. - 50 Iterations

(b) 30 Obs. - 50 Iterations

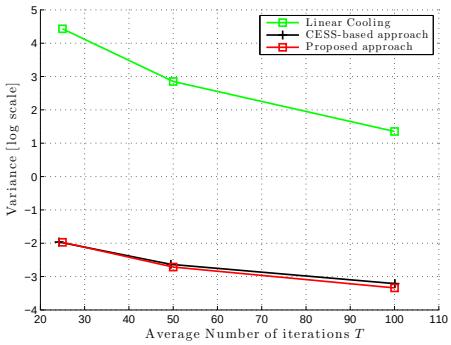
(c) 20 Obs. - 100 Iterations

(d) 30 Obs. - 100 Iterations

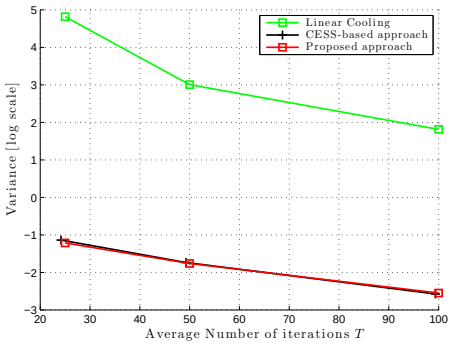
Figure 2.3: Comparison of the theoretical asymptotic variances (dashed lines with cross) and the empirical ones from SMC sampler using perfect mixing Markov Kernel (solid lines with circle) by using the optimal value of the parameter $\hat{\gamma}$.

with λ_t defined respectively in Eq (2.46) and(2.47) and the *"DeMix-based recycling"* described in Section 2.3.2.

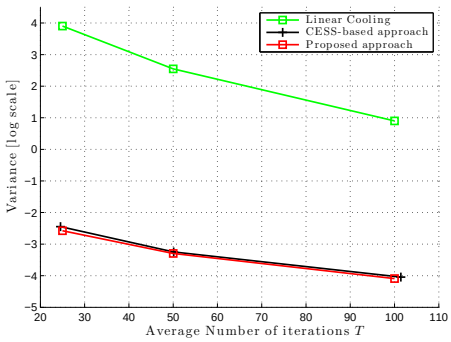
Fig. 2.5 show the mean squared error (MSE) between the estimated posterior mean and the true one given by $\boldsymbol{\mu}_p$ in Eq. (2.55). We can firstly remark from these results that the naïve recycling scheme does not really improve the performance of the estimator of this posterior mean. On the contrary, both proposed schemes outperform significantly this naïve recycling and the classical estimator that uses only the final population of particles (No recycling scheme). The improvement gap increases with the number of iterations used in the SMC sampler, as expected since more collection of particles can be recycled in the estimator. These results demonstrate also empirically for this model the superiority of the DeMix recycling approach.



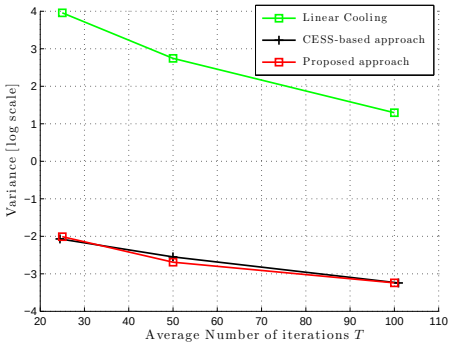
(a) **50 Particles - Perfect Mixing Kernel**



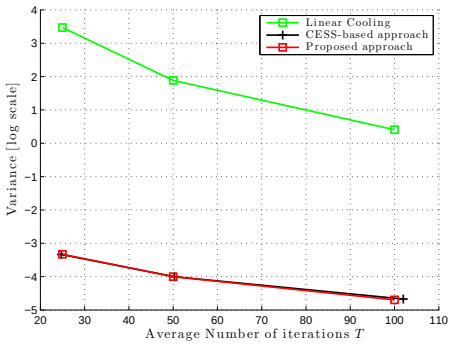
(b) **50 Particles - Adaptive MWG kernel**



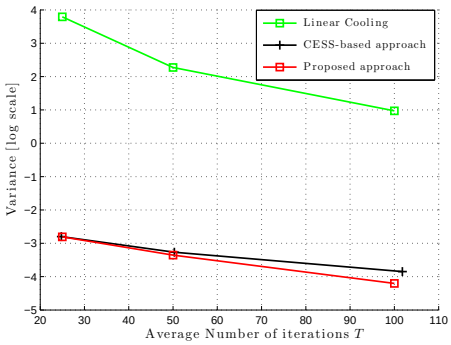
(c) **100 Particles - Perfect Mixing Kernel**



(d) **100 Particles - Adaptive MWG kernel**

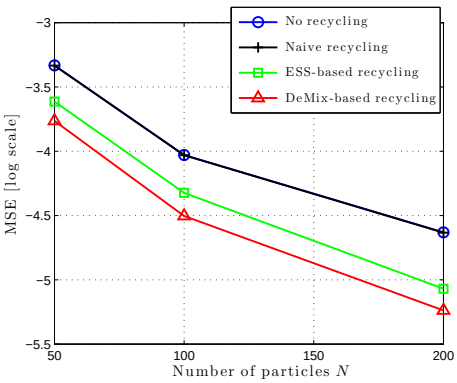


(e) **200 Particles - Perfect Mixing Kernel**

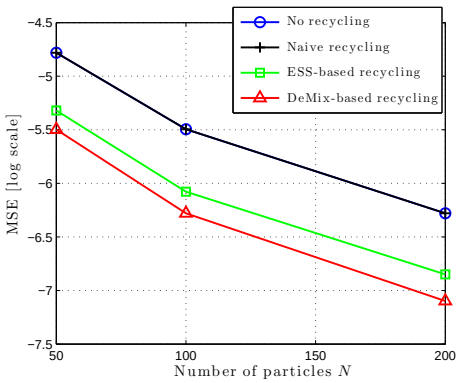


(f) **200 Particles - Adaptive MWG kernel**

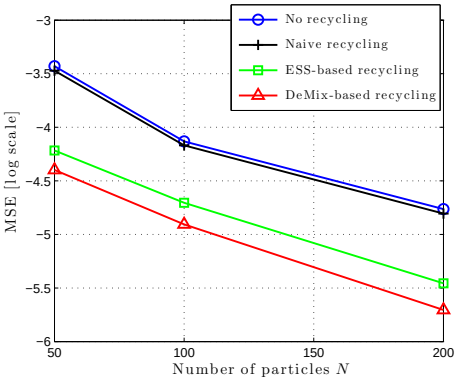
Figure 2.4: Comparison of the different cooling schedule strategies in terms of the variance of the normalizing constant estimate for different number of particles. Results are obtained with the use of either the perfect mixing Kernel (left) or the adaptive MWG kernel (right).



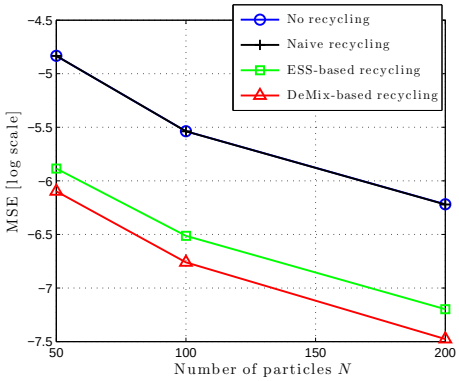
(a) 20 Obs. - 25 Iterations



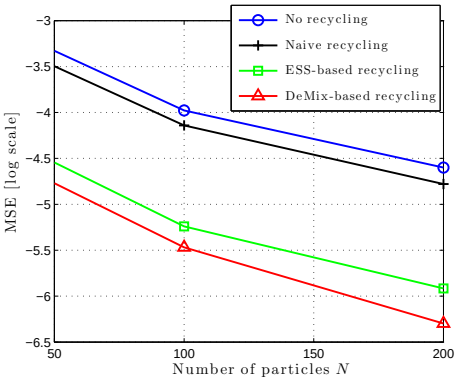
(b) 40 Obs. - 25 Iterations



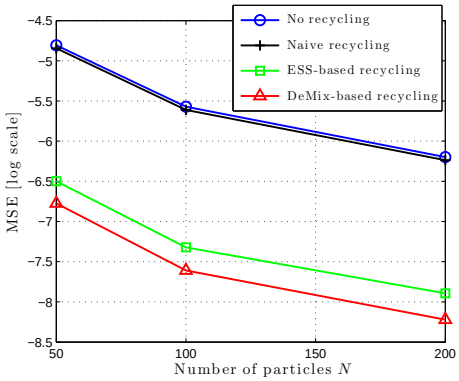
(c) 20 Obs. - 50 Iterations



(d) 40 Obs. - 50 Iterations



(e) 20 Obs. - 100 Iterations



(f) 40 Obs. - 100 Iterations

Figure 2.5: Mean square error between the estimated and the true posterior mean for Model 1 using the different recycling schemes

2.4.2 Model 2: Multivariate Student's t Likelihood

Let us now consider the following model, which is composed of a multivariate normal prior and a multivariate Student's t distribution as likelihood:

$$\begin{aligned} p(\boldsymbol{\theta}) &= \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \\ p(\mathbf{y}|\boldsymbol{\theta}) &= \frac{\Gamma\left(\frac{\nu+n_{\mathbf{y}}}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)(\nu\pi)^{n_{\mathbf{y}}/2}}|\boldsymbol{\Sigma}_l|^{-\frac{1}{2}}\left[1+\frac{[\boldsymbol{y}-\boldsymbol{H\theta}]^T\boldsymbol{\Sigma}_l^{-1}[\boldsymbol{y}-\boldsymbol{H\theta}]}{\nu}\right]^{-\frac{(\nu+n_{\mathbf{y}})}{2}} \end{aligned} \quad (2.61)$$

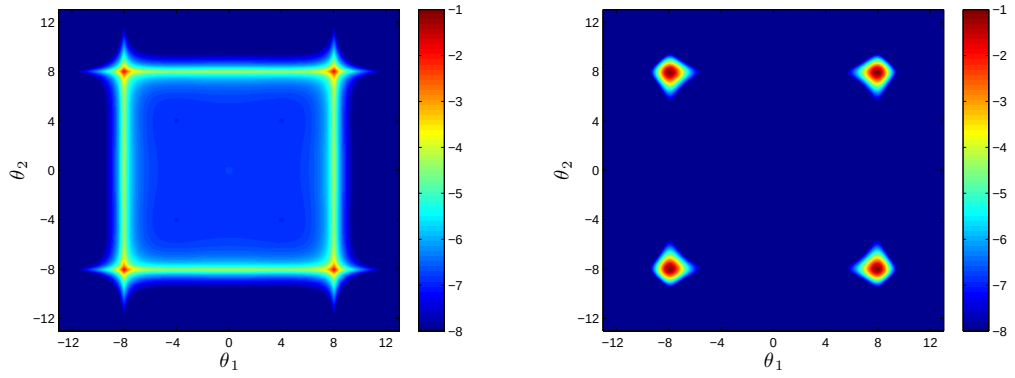
This model could be particularly challenging due to possible multimodality of the target posterior when contradictory observations are used. To analyze the performance of the proposed scheme in complex situation, we use

$$\boldsymbol{H} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}^T$$

$\boldsymbol{\Sigma} = 20\boldsymbol{I}_2$, $\boldsymbol{\mu} = \mathbf{0}_{2\times 1}$, $\boldsymbol{\Sigma}_l = 0.1\boldsymbol{I}_4$ and observations $\mathbf{y} = \begin{bmatrix} y_1 & y_2 & y_3 & y_4 \end{bmatrix}^T = \begin{bmatrix} 8 & -8 & 8 & -8 \end{bmatrix}^T$. These particular choices lead to an highly multimodal posterior distribution as illustrated in Fig. 2.6 for two different values of the degree of freedom of the multivariate Student's t likelihood. From this model and the parameters used, we have for both values of the degree of freedom:

$$\mathbb{E}_\pi \left[\boldsymbol{\theta}\right] = \left[0\ 0\right]^T \tag{2.62}$$

which is confirmed by the numerical evaluation of the posterior shown in Fig. 2.6. For this model, we will follow the same procedure as in the previous Model: analysis of proposed adaptive cooling schedule and then of the proposed recycling schemes. In all the numerical simulations presented in this section, we have chosen $N_{\text{MCMC}} = 10$ and $B = 2$ as parameters of the adaptive MWG kernel within the SMC sampler.



(a) $\nu = 0.2$ **(b)** $\nu = 7$

Figure 2.6: Target posterior distribution $p(\boldsymbol{\theta}|\mathbf{y})$ in log scale evaluated on a grid with 2 different values for the degree of freedom of the Student's t likelihood

Tables 2.1 and 2.2 show the variance of the normalizing constant (i.e., $p(\mathbf{y})$) estimator when the degree of freedom of the multivariate Student's t distribution is $\nu = 0.2$ and $\nu = 7$, respectively. We compare the results obtained using different cooling schedules. The proposed adaptive approach, the $\mathbb{CESS}$-based one as well as the linear cooling schedule yield similar results. From our simulation results, we can see that the proposed adaptive procedure takes a very small value (close to 0) as optimal value for γ which thus leads to the linear cooling schedule. Same

remark when we analyze the evolution of the temperature given by the "on-line" $\mathbb{CESS}$-based strategy. Nevertheless, we can see that these variances can degrade very significantly if another value of γ is chosen (here we take $\gamma = 6$). This clearly demonstrates the impact of this temperature schedule in term of the variance of the normalizing constant. The proposed procedure is thus of great interest in order to automatically decide what should be the evolution of this cooling schedule for a given number of SMC iterations.

		Linear Cooling $\gamma \to 0$	Cooling $\gamma = 6$	CESS Approach	Proposed Approach
T =25 Iter.	$N = 50$	0.0026	0.0110	0.0028	0.0030
	$N = 100$	0.0013	0.0055	0.0012	0.0015
	$N = 200$	0.0006	0.0024	0.0007	0.0008
T =50 Iter.	$N = 50$	0.0011	0.0046	0.0016	0.0013
	$N = 100$	0.0007	0.0029	0.0006	0.0006
	$N = 200$	0.0003	0.0012	0.0004	0.0004
T =100 Iter.	$N = 50$	0.0006	0.0026	0.0006	0.0009
	$N = 100$	0.0003	0.0013	0.0003	0.0004
	$N = 200$	0.0002	0.0005	0.0002	0.0002

Table 2.1: Comparison of the variance of the normalizing constant estimator obtained by using different cooling schedules for Model 2 with $\nu = 0.2$.

		Linear Cooling $\gamma \to 0$	Cooling $\gamma = 6$	CESS Approach	Proposed Approach
T =25 Iter.	$N = 50$	0.0146	0.0375	0.0177	0.0209
	$N = 100$	0.0086	0.0152	0.0079	0.0088
	$N = 200$	0.0050	0.0090	0.0041	0.0042
T =50 Iter.	$N = 50$	0.0105	0.0160	0.0078	0.0072
	$N = 100$	0.0050	0.0102	0.0037	0.0039
	$N = 200$	0.0028	0.0043	0.0025	0.0017
T =100 Iter.	$N = 50$	0.0047	0.0078	0.0051	0.0037
	$N = 100$	0.0022	0.0044	0.0022	0.0023
	$N = 200$	0.0016	0.0026	0.0010	0.0013

Table 2.2: Comparison of the variance of the normalizing constant estimator obtained by using different cooling schedules for Model 2 with $\nu = 7$.

Fig. 2.7 shows the mean squared error between the estimated posterior mean from the proposed recycling scheme and the true one. Unlike the previous model (linear and Gaussian one) for $\nu = 0.2$, the naïve recycling outperforms the classical estimator of the SMC sampler when only the last collection of particles is used. This could be explained by the shape of the target posterior (Fig. 2.6). Indeed,

in such a case, the posterior has a large region with a non-zero probability in the middle of the "square". As a consequence, the particles of the first iteration of the SMC sampler that target the prior can be very useful. However, when the degree of freedom of the likelihood is high ($\nu = 7$), this remark does not hold since the posterior is really concentrated on 4 modes. From this case, it is also interesting to see that the MSE increases with the number of iterations used in the SMC sampler when either no recycling or naïve recycling is performed. Indeed, by increasing the number of iterations, we increase also the number of potential resampling steps and we know that during the resampling procedure, some particles which are currently located in one of the 4 modes can be discarded. Therefore it becomes very difficult for the SMC sampler to jump between two well separated modes, thus leading to an unexplored mode by the SMC sampler for the next iteration. This effect does not appear with the proposed recycling scheme since we recycle all the past simulated particles.

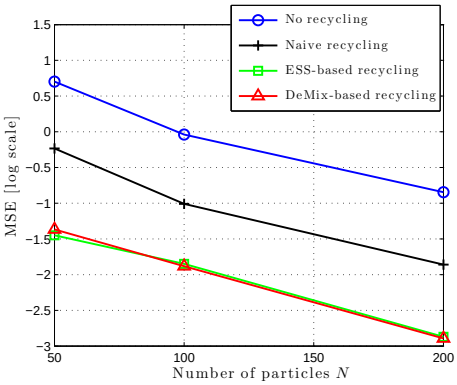
Finally, in order to emphasize the significant gain that could be obtained using our proposed recycling schemes, Tables 2.3 and 2.4 show the mean and standard deviation of the Kolmogorov-Smirnov distance defined as

$$D = \sup_{\theta_1} \left| F^N(\theta_1) - F(\theta_1) \right| \tag{2.63}$$

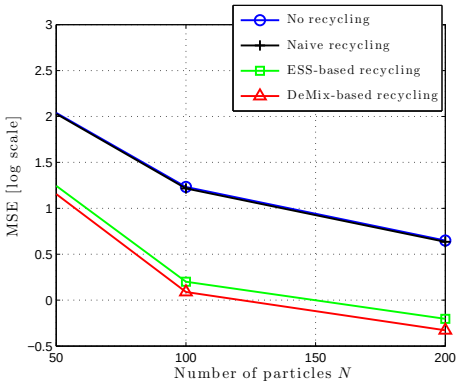
where F^N and F are the empirical cumulative distribution obtained from the SMC sampler and the true posterior cumulative distribution, respectively. This distance D is obtained through 100 runs of the SMC samplers. Compared to the previous comparisons related to the MSE of the posterior mean, this measure give us some information about the quality of the approximation of the whole target distribution. In order to obtain these results, the true target cumulative distribution $F(\theta_1)$ has been obtained numerically by using a very fine grid. In both cases ($\nu = 0.2$ and $\nu = 7$), these results empirically demonstrate the significant gain obtained by using the proposed recycling schemes with a slight advantage to the DeMix-based approach. The average and the standard deviation of this Kolmogorov-Smirnov distance are divided by a factor of 2-3 compared to the case in which we use only the collection of particles from the last iteration of the SMC sampler.

2.5 Conclusion

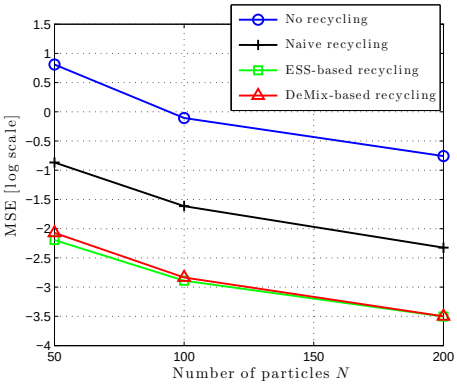
In this chapter, simple forms of the asymptotic variances for the SMC sampler estimator are derived under some assumptions. From these expressions, a novel criterion to optimize is described in order to automatically and adaptively decides the cooling schedule of the algorithm. The proposed strategy is thus to find the evolution of the temperature along the SMC iterations that will optimize the (asymptotic) variance of the estimator of the normalizing constant of the target distribution. Furthermore, we propose two different approaches (ESS and DeMix) that recycle all past simulated particles for the final approximation of the posterior distribution.



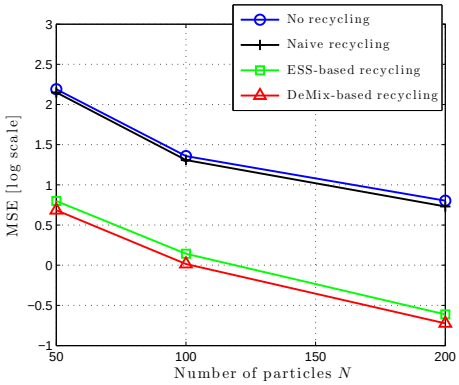
(a) $\nu = 0.2$ - T =**25 Iterations**



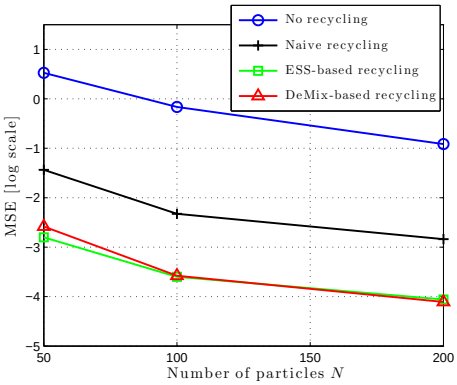
(b) $\nu = 7$ - T =**25 Iterations**



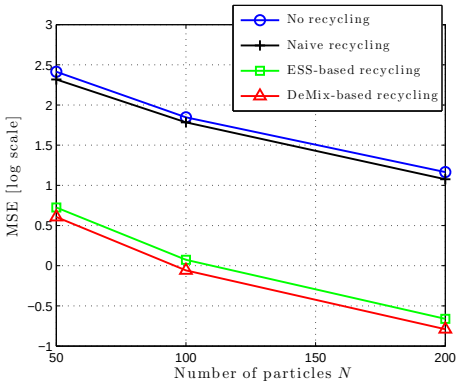
(c) $\nu = 0.2$ - T =**50 Iterations**



(d) $\nu = 7$ - T =**50 Iterations**



(e) $\nu = 0.2$ - T =**100 Iterations**



(f) $\nu = 7$ - T =**100 Iterations**

Figure 2.7: Mean squared error between the estimated and the true posterior mean for Model 2 using the different recycling schemes.

Numerical simulations clearly show that significant improvement can be obtained by using these different propositions. In the next chapters, we will apply the SMC

		No Recycling	Naive Recycling	ESS-based Recycling	DeMix Recycling
T =25 Iter.	$N = 50$	0.1276 (0.0460)	0.0727 (0.0234)	0.0458 (0.0121)	0.0407 (0.0123)
	$N = 100$	0.0835 (0.0224)	0.0488 (0.0164)	0.0366 (0.0094)	0.0315 (0.0089)
	$N = 200$	0.0615 (0.0182)	0.0353 (0.0103)	0.0254 (0.0055)	0.0237 (0.0053)
T =50 Iter.	$N = 50$	0.1274 (0.0424)	0.0514 (0.0167)	0.0357 (0.0096)	0.0311 (0.0097)
	$N = 100$	0.0898 (0.0251)	0.0379 (0.0117)	0.0268 (0.0067)	0.0230 (0.0055)
	$N = 200$	0.0627 (0.0188)	0.0267 (0.0068)	0.0201 (0.0045)	0.0185 (0.0037)
T =100 Iter.	$N = 50$	0.1186 (0.0352)	0.0391 (0.0117)	0.0315 (0.0066)	0.0243 (0.0060)
	$N = 100$	0.0846 (0.0231)	0.0288 (0.0079)	0.0226 (0.0054)	0.0187 (0.0038)
	$N = 200$	0.0599 (0.0188)	0.0216 (0.0054)	0.0177 (0.0033)	0.0159 (0.0031)

Table 2.3: Comparison of recycling schemes for the accuracy to approximate the posterior distribution $p(\theta_1|\mathbf{y})$ in terms of the Kolmogorov-Smirnov distance (mean and standard deviation in parentheses) for Model 2 with $\nu = 0.2$.

		No Recycling	Naive Recycling	ESS-based Recycling	DeMix Recycling
T =25 Iter.	$N = 50$	0.1607 (0.0615)	0.1571 (0.0612)	0.0861 (0.0417)	0.0839 (0.0390)
	$N = 100$	0.1048 (0.0331)	0.1026 (0.0325)	0.0596 (0.0216)	0.0578 (0.0203)
	$N = 200$	0.0825 (0.0299)	0.0809 (0.0296)	0.0494 (0.0201)	0.0476 (0.0188)
T =50 Iter.	$N = 50$	0.1641 (0.0651)	0.1499 (0.0649)	0.0678 (0.0289)	0.0655 (0.0274)
	$N = 100$	0.1126 (0.0392)	0.1020 (0.0385)	0.0517 (0.0215)	0.0500 (0.0204)
	$N = 200$	0.0878 (0.0378)	0.0803 (0.0369)	0.0404 (0.0147)	0.0396 (0.0139)
T =100 Iter.	$N = 50$	0.1795 (0.0883)	0.1528 (0.0845)	0.0623 (0.0420)	0.0604 (0.0393)
	$N = 100$	0.1261 (0.0580)	0.1092 (0.0570)	0.0475 (0.0229)	0.0459 (0.0214)
	$N = 200$	0.0901 (0.0329)	0.0761 (0.0326)	0.0352 (0.0141)	0.0342 (0.0135)

Table 2.4: Comparison of recycling schemes for the accuracy to approximate the posterior distribution $p(\theta_1|\mathbf{y})$ in terms of the Kolmogorov-Smirnov distance (mean and standard deviation in parentheses) for Model 2 with $\nu = 7$.

sampler, with the proposed approaches derived in this chapter, to two challenging practical problems: Multiple source localization in wireless sensor networks and Bayesian penalized regression.

Chapter 3

Multiple Source Localization in Wireless Sensor Networks

Contents

Wireless sensor networks (WSNs) are composed of a large numbers of low-cost, low-power, densely distributed, and possibly heterogeneous sensors. WSNs have a wide range of application areas such as battlefield surveillance, environment or health monitoring, and disaster relief operations. In these applications, WSNs are used for a variety of tasks such as detection, recognition, localization and tracking of objects or events of interest. In this chapter, we study the source localization problem where the aim is to estimate the coordinates of an energy emitting source (e.g., acoustic source). The idea is to use the SMC sampler with the proposed technique described in Chapter 2 in order to efficiently infer this quantity of interest.

3.1 Introduction and Problem Formulation

3.1.1 Existing works

In a WSN, there is typically a large number of inexpensive sensors that are densely deployed in a region of interest (ROI). This makes possible therefore accurate energy based target localization. Signal intensity measurements are very convenient and economical to localize a target, since no additional sensor functionalities and measurement features, such as direction of arrival (DOA) or time-delay of arrival (TDOA), are needed.

Energy-based methods have been proposed and developed in [Li et al., 2002, Li and Hu, 2003, Sheng and Hu, 2005]. Such methods are very suitable for WSNs because they only require the energy readings of the sensors. Energy-based methods are based on the fact that the intensity (energy) of the acoustic signal attenuates as a function of distance from the source. In [Li and Hu, 2003], a least-square method is proposed to localize a single source based on the energy ratios between sensors. In [Sheng and Hu, 2005], a maximum-likelihood (ML) acoustic source localization method has been presented. However, in all these papers, analog measurements from sensors are required to estimate the source location. For a typical WSN with limited resources (energy and bandwidth), it is important to limit the communication with the network. Therefore, it is often desirable that only binary or multiple bit quantized data be transmitted from local sensors to the fusion center (processing node).

Motivated by such constraints, several papers have more recently proposed source localization techniques using only quantized data [Niu and Varshney, 2006, Ozdemir et al., 2009, Masazade et al., 2010]. In [Niu and Varshney, 2006], a maximum likelihood (ML) based approach has been proposed by using multi-bit (M-bit) sensor measurements transmitted to the fusion center. In [Masazade et al., 2010], the authors developed on the same problem an importance sampler in order to approximate the posterior distribution of the single source given the quantized data. However, in both works, perfect communication channels between sensors and the fusion center are assumed. Usually, in a target localization scenario, a large number of sensors is deployed in a particular area where a line-of-sight between sensors and the fusion center is not always guaranteed. In [Ozdemir et al., 2009], a maximum likelihood estimator is designed in order to have a localization algorithm that incorporates the imperfect nature of communication channels as well as based on the constraints of limited resources in a WSN with quantized data. Unfortunately, these approaches have been developed for the localization on a single source. To our best knowledge, the main paper that deals with multiple source localization is [Sheng and Hu, 2005] which, as discussed before, propose an ML estimator for multiple source localization but with perfect channel and analog sensor measurements. Moreover, they assume that the number of sources is perfectly known.

In this chapter, we thus propose to derive a localization algorithm for an unknown number of sources given some quantized data obtained at the fusion center from different sensors with imperfect wireless channels. As a consequence, the problem we propose to address is a generalization of existing ones. The proposed Bayesian algorithm will be derived by using the different propositions that have been developed in Chapter 2.

3.1.2 System Model

As illustrated in Fig. 3.1, we are interested in localizing an unknown number of targets in a wireless sensor environment where deployed homogeneous and low-cost wireless sensors are employed. All the sensors report to a fusion center which

estimates the target locations based on local sensor observations. Sensors can be deployed in any manner since our approach is capable of handling any kind of deployment as long as the location information for each sensor is available at the fusion center.

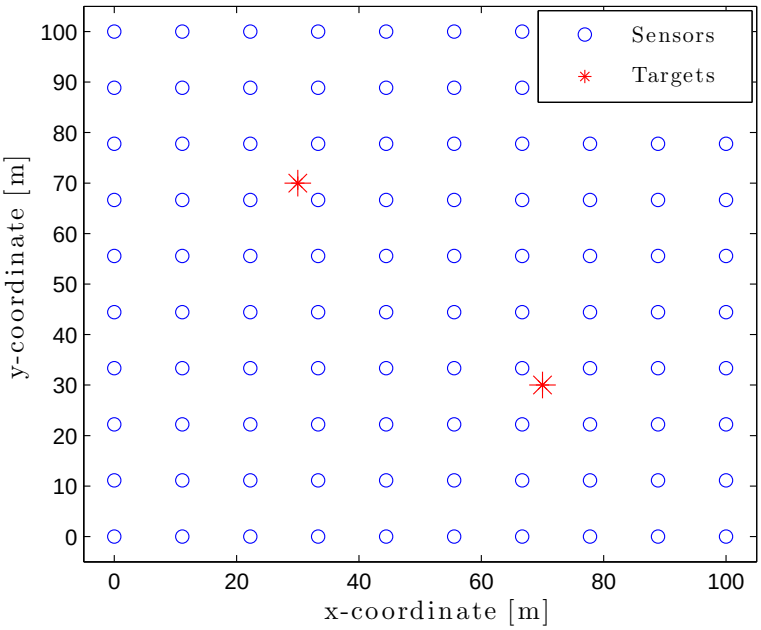


Figure 3.1: Example of two targets in a grid deployed sensor field.

Each target is assumed to be a source that follows the power attenuation model, such as an acoustic source for example. We thus use a signal attenuation model to represent the observed power that is emitted by each target [Niu and Varshney, 2006]. This signal attenuation model is based on the fact that an acoustic omnidirectional point source emits signals that attenuate at a rate inversely proportional to the distance from the source if the propagation is through ground surface. In this work as in [Sheng and Hu, 2005], we will further assume that the intensities of the K sources will be linearly superimposed without any interaction between them. The received signal amplitude at the i-th sensor ($i = 1, \ldots, N$) is thus given by

$$s_i = a_i + n_i \tag{3.1}$$

where the measurement noise term, n_i, is modeled as an additive white Gaussian noise (AWGN), i.e., $n_i \sim \mathcal{N}(0, \sigma^2)$ which represents the cumulative effects of sensor background noise and the modeling error of acoustic signal parameters (the Gaussian assumption is generally admitted since the central limit theorem could be applied on a processed signal resulting on the average of the samples received during a time period). The true signal amplitude a_i from all the targets is defined

as [Sheng and Hu, 2005]:

$$a_i = \sum_{k=1}^{K} P_k^{1/2} \left(\frac{d_0}{d_{i,k}} \right)^{\frac{n}{2}} \tag{3.2}$$

where P_k denotes the k-th source signal power at a reference distance d_0. The signal decay n is approximately 2 when the detection distance is less than 1km [Li and Hu, 2003]. Finally $d_{i,k}$ corresponds to the distance between the i-th sensor and the k-th target:

$$d_{i,k} = \sqrt{(x_k - p_{x,i})^2 + (y_k - p_{y,i})^2} \tag{3.3}$$

where $(p_{x,i}, p_{y,i})$ and (x_k, y_k) are the coordinates of the i-th sensor and the k-th target, respectively. In this work, we assume that sensor noises as well as wireless links between the sensors and the fusion center are independent across sensors, and that σ^2 is known (although it is not required for our proposed approach to work - this could be indeed embedded in the parameters to infer).

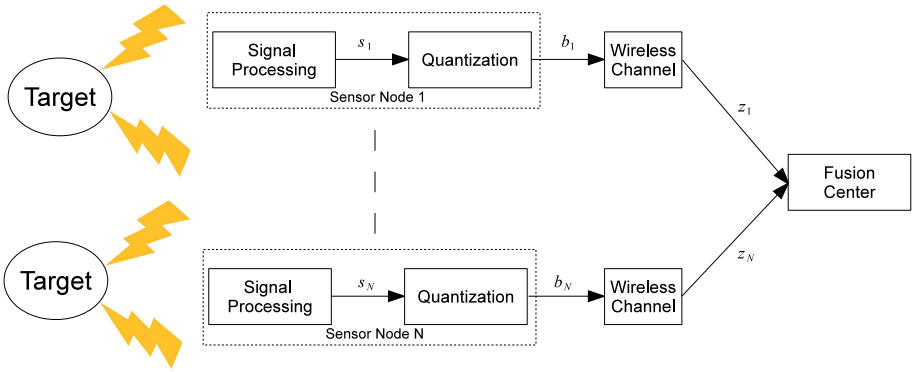


Figure 3.2: Illustration of the system model.

As illustrated in Fig. 3.2, at each sensor, the received signal is quantized before being sent to the fusion center. Quantization is done locally at the sensors in order to decrease the communication bandwidth on the sensors thereby reducing energy consumption. The data is quantized using an M-bit quantizer ($M \geq 1$) which takes values from 0 to $2^M - 1$ where $L = 2^M$ is the number of quantization levels. The quantizer of the i-th sensor transforms its input s_i to its output b_i through a mapping: $\mathbb{R} \mapsto \{0, \ldots, L-1\}$ such that

$$b_i = \begin{cases} 0 & \lambda_{i,0} \leq s_i < \lambda_{i,1} \\ 1 & \lambda_{i,1} \leq s_i < \lambda_{i,2} \\ \vdots & \vdots \\ L-1 & \lambda_{i,L-1} \leq s_i < \lambda_{i,L} \end{cases} \tag{3.4}$$

with $\lambda_{i,0} = -\infty$ and $\lambda_{i,L} = +\infty$. Let $\boldsymbol{\theta} = \left[P_1, x_1, y_1, \ldots, P_K, x_K, y_K\right]^T$ be all the K source locations and their associated transmitted powers. Under Gaussian assumption of the measurement noise, the probability that b_i takes a specific value $l \in \{0, \ldots, L-1\}$ is:

$$p(b_i = l|\boldsymbol{\theta}) = Q\left(\frac{\lambda_{i,l} - a_i}{\sigma}\right) - Q\left(\frac{\lambda_{i,l+1} - a_i}{\sigma}\right) \tag{3.5}$$

where $Q(\cdot)$ is the complementary distribution function of the Gaussian distribution defined as:

$$Q(x) = \int_x^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt \tag{3.6}$$

Finally, the quantized observation are transmitted to the fusion center through an imperfect channel which may introduce transmission errors. Let $\boldsymbol{z} = \left[z_1, \ldots, z_N\right]$ denote the observations collected at the fusion center via independent channels from the N sensors. As in [Ozdemir et al., 2009, Nevat et al., 2014], the probability of a received observation z_i taking a specific value j, given the targets' parameters, $\boldsymbol{\theta}$, can be written as:

$$p(z_i = j|\boldsymbol{\theta}) = \sum_{m=0}^{L-1} p(z_i = j|b_i = m) p(b_i = m|\boldsymbol{\theta}) \tag{3.7}$$

The channel statistics can thus be represented in the following matrix form:

$$\begin{bmatrix} p_{0,0} & p_{0,1} & \cdots & p_{0,L-1} \\ p_{1,0} & p_{1,1} & \cdots & p_{0,L-1} \\ \vdots & \vdots & \ddots & \vdots \\ p_{L-1,0} & p_{L-1,1} & \cdots & p_{L-1,L-1} \end{bmatrix} \tag{3.8}$$

where

$$\begin{aligned} & p_{m,j} := p(z_i = j|b_i = m) \quad \forall m, j \in \{0, \ldots, L-1\} \\ & \sum_{j=0}^{L-1} p_{m,j} = 1 \quad \forall j \in \{0, \ldots, L-1\} \end{aligned}$$

Since sensor noises and wireless links are assumed to be independent, the likelihood function at the fusion center can be written as:

$$\begin{aligned} p(\boldsymbol{z}|\boldsymbol{\theta}) &= \prod_{i=1}^{N} p(z_i|\boldsymbol{\theta}) \\ &= \prod_{i=1}^{N} \left[\sum_{m=0}^{L-1} p(z_i|b_i = m) p(b_i = m|\boldsymbol{\theta})\right] \end{aligned} \tag{3.9}$$

Concerning the prior information related to the parameters of interest $\boldsymbol{\theta}$, we use

in this work:

$$p(\boldsymbol{\theta}) = \prod_{k=1}^{K} p(x_k, y_k)p(P_k) \tag{3.10}$$

where

$$p(x_k, y_k) = \mathcal{N}(\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p) \tag{3.11}$$
$$p(P_k) = \mathcal{IG}(a, b) \tag{3.12}$$

with $\boldsymbol{\mu}_p$ is the center of the ROI and $\boldsymbol{\Sigma}_p = \begin{bmatrix} \sigma_{p,x}^2 & 0 \\ 0 & \sigma_{p,y}^2 \end{bmatrix}$ is the covariance matrix which is very coarse so that its 99% confidence region covers the entire ROI. $\mathcal{IG}(a, b)$ corresponds to the inverse gamma distribution with a and b being the shape and the scale parameter, respectively. Note that the proposed inference algorithm does not require the prior distributions to be Gaussian and inverse-gamma and will work with other prior distribution also.

3.2 Proposed Bayesian Solution

3.2.1 Bayesian modeling

In this work, we are interested in estimating the unknown number of sources as well as their parameters (locations and transmitted powers). This problem can therefore be seen as a joint model selection and parameter estimation task. We have a collection of K competing models $\{\mathcal{M}_k\}_{k\in\{1,\dots,K\}}$ (corresponding in our case to the number of sources in the ROI) and one of them generates the observations obtained at the fusion center. Associated with each model, there is a vector of parameters $\boldsymbol{\theta}_k \in \Theta_k$, where Θ_k denotes the parameter space of the model $\mathcal{M}_k$. The objective is to identify the true model as well as to estimate the parameters, $\boldsymbol{\theta}_k = \Big[P_1, x_1, y_1, \dots, P_k, x_k, y_k\Big]^T$, associated with this model.

As discussed in Chapter 1 - Section 1.1.3, Bayesian inference proceeds from a prior distribution over the collection of models, $p(\mathcal{M}_k)$, a prior distribution for the parameters of each model, $p(\boldsymbol{\theta}_k|\mathcal{M}_k)$ and the likelihood under each model $p(\boldsymbol{z}|\boldsymbol{\theta}_k, \mathcal{M}_k)$. In order to perform model comparison, one requires the posterior model probability,

$$p(\mathcal{M}_k|\boldsymbol{z}) = \frac{p(\boldsymbol{z}|\mathcal{M}_k)p(\mathcal{M}_k)}{p(\boldsymbol{z})} \tag{3.13}$$

where

$$p(\boldsymbol{z}|\mathcal{M}_k) = \int_{\Theta_k} p(\boldsymbol{z}|\boldsymbol{\theta}_k, \mathcal{M}_k)p(\boldsymbol{\theta}_k|\mathcal{M}_k)d\boldsymbol{\theta}_k \tag{3.14}$$

is termed the evidence for model $\mathcal{M}_k$ and the $p(\boldsymbol{z}) = \sum_{k=1}^{K} p(\boldsymbol{z}|\mathcal{M}_k)p(\mathcal{M}_k)$ is easily calculated for finite number of competing models (as in our case) and if the evidence for each model is available.

To solve this problem in the Bayesian context, one typically employs the maximum a posteriori (MAP) rule for the model selection, which can be expressed as:

$$\begin{aligned} k^* &= \arg\max_k \{p(\mathcal{M}_k|\boldsymbol{z})\} \\ &= \arg\max_k \{p(\boldsymbol{z}|\mathcal{M}_k)p(\mathcal{M}_k)\} \end{aligned} \tag{3.15}$$

The estimate of the parameters can then be deduced from the posterior distribution associated with the model $\mathcal{M}_{k^*}$, i.e. $p(\boldsymbol{\theta}_{k^*}|\boldsymbol{z}, \mathcal{M}_{k^*})$. Unfortunately, $\forall k \in \{1, \ldots, K\}$, both the evidence $p(\boldsymbol{z}|\mathcal{M}_k)$ and the posterior distribution of the parameters $p(\boldsymbol{\theta}_k|\boldsymbol{z}, \mathcal{M}_k)$ are intractable. In this work, we propose to use SMC sampler in order to have an accurate approximation of both quantities.

3.2.2 Proposed SMC sampler algorithm

Let us firstly remark that the evidence of the model $\mathcal{M}_k$ corresponds to the normalizing constant of the posterior distribution of the parameters associated with this model, i.e.:

$$p(\boldsymbol{\theta}_k|\boldsymbol{z}, \mathcal{M}_k) = \frac{p(\boldsymbol{z}|\boldsymbol{\theta}_k, \mathcal{M}_k)p(\boldsymbol{\theta}_k|\mathcal{M}_k)}{p(\boldsymbol{z}|\mathcal{M}_k)} = \frac{p(\boldsymbol{z}|\boldsymbol{\theta}_k, \mathcal{M}_k)p(\boldsymbol{\theta}_k|\mathcal{M}_k)}{\int_{\Theta_k} p(\boldsymbol{z}|\boldsymbol{\theta}_k, \mathcal{M}_k)p(\boldsymbol{\theta}_k|\mathcal{M}_k)d\boldsymbol{\theta}_k} \tag{3.16}$$

As a consequence, we propose to use the following procedure:

1. For each model $\mathcal{M}_k$, $k \in 1, \ldots, K$: approximate the conditional parameter posterior distribution $p(\boldsymbol{\theta}_k|\boldsymbol{z}, \mathcal{M}_k)$ as well as the marginal likelihood $p(\boldsymbol{z}|\mathcal{M}_k)$ using an SMC sampler algorithm.

2. Approximate the model posterior $p(\mathcal{M}_k|\boldsymbol{z})$, via the approximation of $p(\boldsymbol{z}|\mathcal{M}_k)$ and model prior $p(\mathcal{M}_k)$ - Eq. (3.13).

As summarized in Algo 3.1, we propose to use the strategies proposed in Chapter 2 in order to improve:

- the variance of the estimator of the normalizing constant (i.e., the evidence of the model) by using the adaptive cooling schedule of the SMC sampler,

- the variance of the final approximation of the parameter posterior distribution by using the proposed recycling schemes.

3.2.3 Point estimate for the state of interest

Owing to the non-identifiability of the target label in the likelihood and to the same prior for each target, the posterior distribution will be multimodal (as it will

Algorithm 3.1 SMC Sampler Algorithm for Model $\mathcal{M}_k$ of the multiple source localization problem

1: Find the optimal parameter value γ^* of the parametric cooling schedule using the strategy described in Section 2.2.2
2: Initialize particle system from the prior
3: $\left\{\boldsymbol{\theta}_1^{(i)}\right\}_{i=1}^{N} \sim p(\boldsymbol{\theta}|\mathcal{M}_k)$ and set $\left\{\widetilde{W}_1^{(i)}\right\}_{i=1}^{N} = 1/N$
4: **for** $t = 2, \ldots, T$ **do**
5: Computation of the weights: for each $i = 1, \ldots, N$

$$W_t^{(i)} = \widetilde{W}_{t-1}^{(i)} \frac{\pi_t(\boldsymbol{\theta}_{t-1}^{(i)})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1}^{(i)})} = \widetilde{W}_{t-1}^{(i)} \frac{p(\boldsymbol{z}|\boldsymbol{\theta}_{t-1}, \mathcal{M}_k)^{\phi_t}}{p(\boldsymbol{z}|\boldsymbol{\theta}_{t-1}, \mathcal{M}_k)^{\phi_{t-1}}}$$

Normalization of the weights : $\widetilde{W}_t^{(i)} = W_t^{(i)} \left[\sum_{j=1}^{N} W_t^{(j)}\right]^{-1}$
6: Selection: if $ESS < N/2$ then Resample
7: Mutation: for each $i = 1, \ldots, N_p$: Sample $\theta_t^{(i)} \sim \mathcal{K}_t(\boldsymbol{\theta}_{t-1}^{(i)}; \cdot)$ where $\mathcal{K}_t(\cdot;\cdot)$ is a $\pi_t(\cdot)$ invariant Markov kernel using a series of Adaptive Metropolis within Gibbs algorithms for each of the K sources successively - see details in Algo. 1.9
8: **end for**
9: Approximate the model evidence, $p(\boldsymbol{z}|\mathcal{M}_k)$, using Eq. (1.62)
10: Use the proposed recycling schemes described in Section 2.3 in order to combine all simulated particles from iteration 1 to T in order to obtain an approximation of $p(\boldsymbol{\theta}_k|\boldsymbol{z}, \mathcal{M}_k)$

be illustrated in Fig 3.7). The posterior is invariant under the permutations of source parameters, i.e.,

$$p(\boldsymbol{\theta}_k|\boldsymbol{z}, \mathcal{M}_k) = p(\vartheta(\boldsymbol{\theta}_k)|\boldsymbol{z}, \mathcal{M}_k) \tag{3.17}$$

where $\vartheta(\cdot) \in \mathcal{P}$ denotes any the permutation for which the posterior is invariant and $\mathcal{P}$ is the set of these permutations.

In that case, the MMSE estimate (i.e., posterior mean) would lead to very poor performance as point estimate of the source parameters. The problem of having a Monte-Carlo algorithm that approximates such multimodal posterior invariant under permutation is known in the literature as the *label switching problem* [Stephens, 2000].

There exists many algorithms that have been proposed in order to deal with this label switching problem in Monte-Carlo algorithms. A recent and detailed review of these techniques can be found in [Bardenet, 2012]. Here, we are interested in only post-processing technique in order to extract an accurate point-estimate of the state interest from our particle approximation of the posterior distribution. One of the most commonly used relabelling algorithm is the one proposed in [Stephens, 2000].

Let us denote the unweighted set of particles[1] that target the posterior distribution by $\underset{\sim}{\boldsymbol{\theta}} = \left\{\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(N)}\right\}$. With the algorithm in [Stephens, 2000], one performs

[1] obtained by performing a resampling step on all the simulated particles that have been potentially recycled using the proposed strategy described in Section 2.3

inference tasks (e.g., point estimate) as usual but with the relabelled samples, defined as:

$$\boldsymbol{\vartheta}(\underset{\sim}{\boldsymbol{\theta}}) = \left(\vartheta_1(\boldsymbol{\theta}^{(1)}), \ldots, \vartheta_N(\boldsymbol{\theta}^{(N)})\right) \tag{3.18}$$

where

$$\boldsymbol{\vartheta} = (\vartheta_1, \ldots, \vartheta_N) = \underset{\mathcal{P}\times\cdots\times\mathcal{P}}{\arg\min}\, L(\underset{\sim}{\boldsymbol{\theta}}, \boldsymbol{\vartheta}) \tag{3.19}$$

and $L(\cdot)$ is a user-defined cost-function. Explicit choices for $L(\cdot)$, among which

$$L(\underset{\sim}{\boldsymbol{\theta}}, \boldsymbol{\vartheta}) = \prod_{i=1}^{N} \mathcal{N}\left(\vartheta_i(\boldsymbol{\theta}^{(i)}) | \boldsymbol{\mu}_N^{\boldsymbol{\vartheta}}, \boldsymbol{\Sigma}_N^{\boldsymbol{\vartheta}}\right) \tag{3.20}$$

with

$$\boldsymbol{\mu}_N^{\boldsymbol{\vartheta}} = \frac{1}{N}\sum_{i=1}^{N} \vartheta_i(\boldsymbol{\theta}^{(i)}) \tag{3.21}$$

$$\boldsymbol{\Sigma}_N^{\boldsymbol{\vartheta}} = \frac{1}{N}\sum_{i=1}^{N} (\vartheta_i(\boldsymbol{\theta}^{(i)}) - \boldsymbol{\mu}_N^{\boldsymbol{\vartheta}})(\vartheta_i(\boldsymbol{\theta}^{(i)}) - \boldsymbol{\mu}_N^{\boldsymbol{\vartheta}})^T \tag{3.22}$$

The Gaussian cost function in Eq. (3.20) translates the idea that one wants a relabelled sample to be the most Gaussian possible among its permutations $\boldsymbol{\vartheta}(\underset{\sim}{\boldsymbol{\theta}})$, $\boldsymbol{\vartheta} \in \mathcal{P}^N$, in order for $\boldsymbol{\vartheta}(\underset{\sim}{\boldsymbol{\theta}})$ to look as unimodal as possible.

However, this technique is particularly costly since it involves a combinatorial optimization over $\mathcal{P}^N$, which is unfeasible in practice: here the posterior is defined on $\mathbb{R}^{3K}$ and $\mathcal{P}$ is the group formed by the permutations of K elements, $\mathcal{P}^N$ has cardinal $(K!)^N$. As a consequence, in this work, we use the online version of this algorithm proposed in [Celeux, 1998] and having a final cost of $N(K!)$. This approach adapted to our algorithm is described in Algorithm 3.2.

3.3 Derivation of the Posterior Cramér-Rao bound

In this section, we will derive the posterior Cramér-Rao bound (PCRB) as an estimation benchmark of the parameters only. We will thus assume here that the number of sources is known. This PCRB will thus provides a theoretical performance limit for the Bayesian estimator of the locations as well as the transmitted powers of the K sources given the observations, $\boldsymbol{z}$, obtained at the fusion center. Let us remark that in [Ozdemir et al., 2009], the authors have derived the Cramér-Rao bound for the single source problem with quantized data and imperfect channel between the sensors and the fusion center. Here, we propose to generalize this result by considering $\boldsymbol{\theta}$ as a random variable (Bayesian framework which leads to the posterior CRB) and $\boldsymbol{\theta}$ composed of multiple sources.

Indeed, the PCRB gives a lower bound for the error covariance matrix

Algorithm 3.2 Online post-processing relabeling algorithm

1: Set $\boldsymbol{\mu}_2 = \boldsymbol{\theta}^{(1)}$ and $\boldsymbol{\theta}^{(1)}_{relabel} = \boldsymbol{\theta}^{(1)}$
2: **for** $n = 2, \ldots, N$ **do**
3: Find

$$\vartheta_n = \arg\min_{\mathcal{P}} \mathcal{N}\left(\vartheta(\boldsymbol{\theta}^{(n)})|\boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n\right)$$

where by denoting $\boldsymbol{\theta}^{(n)}_{relabel} = \vartheta(\boldsymbol{\theta}^{(n)})$:

$$\boldsymbol{\mu}_n = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\theta}^{(i)}_{relabel} \tag{3.23}$$

$$\boldsymbol{\Sigma}_n = \frac{1}{n}\sum_{i=1}^{n} (\boldsymbol{\theta}^{(i)}_{relabel} - \boldsymbol{\mu}_{n+1})(\boldsymbol{\theta}^{(i)}_{relabel} - \boldsymbol{\mu}_{n+1})^T \tag{3.24}$$

4: Set $\boldsymbol{\theta}^{(n)}_{relabel} = \vartheta_n(\boldsymbol{\theta}^{(n)})$
5: **end for**
6: Use the relabelled collection of unweighted particles $\left\{\boldsymbol{\theta}^{(1)}_{relabel}, \ldots, \boldsymbol{\theta}^{(N)}_{relabel}\right\}$ to compute point estimate.

[Van Trees, 1968]:

$$\mathbb{E}\left[\left(\hat{\boldsymbol{\theta}}(\boldsymbol{z}) - \boldsymbol{\theta}\right)\left(\hat{\boldsymbol{\theta}}(\boldsymbol{z}) - \boldsymbol{\theta}\right)^T\right] \geq \boldsymbol{J}^{-1} \tag{3.25}$$

where $\boldsymbol{J}$ is the $3K \times 3K$ Fisher information matrix (FIM)

$$\begin{aligned} \boldsymbol{J} &= \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{z}, \boldsymbol{\theta}_K|\mathcal{M}_k = K) \nabla^T_{\boldsymbol{\theta}} \log p(\boldsymbol{z}, \boldsymbol{\theta}_K|\mathcal{M}_k = K)\right] \\ &= -\mathbb{E}\left[\Delta^{\boldsymbol{\theta}}_{\boldsymbol{\theta}} \log p(\boldsymbol{z}, \boldsymbol{\theta}_K|\mathcal{M}_k = K)\right] \end{aligned} \tag{3.26}$$

where $\Delta^{\boldsymbol{\theta}}_{\boldsymbol{\theta}} := \nabla_{\boldsymbol{\theta}} \nabla^T_{\boldsymbol{\theta}}$ is the second derivative operator and $\nabla_{\boldsymbol{\theta}}$ is the gradient operator with respect to $\boldsymbol{\theta}$.

Using the fact that $p(\boldsymbol{z}, \boldsymbol{\theta}_K|\mathcal{M}_k = K) = p(\boldsymbol{z}|\boldsymbol{\theta}_K, \mathcal{M}_k = K) p(\boldsymbol{\theta}_K|\mathcal{M}_k = K)$, the expression of the FIM in Eq. (3.26) can be expressed as:

$$\boldsymbol{J} = -\mathbb{E}\left[\Delta^{\boldsymbol{\theta}}_{\boldsymbol{\theta}} \log p(\boldsymbol{z}|\boldsymbol{\theta}_K, \mathcal{M}_k = K)\right] - \mathbb{E}\left[\Delta^{\boldsymbol{\theta}}_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}_K|\mathcal{M}_k = K)\right] = \boldsymbol{J}_d + \boldsymbol{J}_p \tag{3.27}$$

where $\boldsymbol{J}_p$ represents the *a priori* information and $\boldsymbol{J}_d$ is the "standard" FIM (used in the derivation of the CRB) averaged over the prior of the different location and power of the K sources:

$$\boldsymbol{J}_d = \int_{\Theta_k} \boldsymbol{J}_d(\boldsymbol{\theta}_K) p(\boldsymbol{\theta}_K|\mathcal{M}_k = K) d\boldsymbol{\theta}_K \tag{3.28}$$

As demonstrated in Appendix D, this standard FIM is defined for this problem

as follows:

$$\boldsymbol{J}_d(\boldsymbol{\theta}_K) = \sum_{i=1}^{N}\sum_{j=0}^{L-1} \frac{\nabla_{\boldsymbol{\theta}} p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)\nabla_{\boldsymbol{\theta}}^T p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)} \tag{3.29}$$

with the gradient operator given by:

$$\nabla_{\boldsymbol{\theta}} = \begin{bmatrix} \frac{\partial}{\partial P_1} & \frac{\partial}{\partial x_1} & \frac{\partial}{\partial y_1} & \cdots & \frac{\partial}{\partial P_K} & \frac{\partial}{\partial x_K} & \frac{\partial}{\partial y_K} \end{bmatrix}^T \tag{3.30}$$

Using Eq. (3.7), the gradient term in Eq. (3.29) is expressed as:

$$\nabla_{\boldsymbol{\theta}} p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K) = \sum_{l=0}^{L-1} p(z_i = j|b_i = l)\nabla_{\boldsymbol{\theta}} p(b_i = l|\boldsymbol{\theta}_K, \mathcal{M}_k = K) \tag{3.31}$$

in which for $k = 1, \ldots, K$:

$$\begin{aligned} \frac{\partial p(b_i = l|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{\partial P_k} &= \left(\frac{d_0}{d_{i,k}}\right)^{n/2} \frac{\rho_{i,l}}{2\sqrt{2\pi\sigma^2 P_k}} \\ \frac{\partial p(b_i = l|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{\partial x_k} &= \left(\frac{d_0}{d_{i,k}}\right)^{n/2} \frac{nP_k^{1/2} d_{i,k}^{-2}\rho_{i,l}(p_{x,i} - x_k)}{2\sqrt{2\pi\sigma^2}} \\ \frac{\partial p(b_i = l|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{\partial y_k} &= \left(\frac{d_0}{d_{i,k}}\right)^{n/2} \frac{nP_k^{1/2} d_{i,k}^{-2}\rho_{i,l}(p_{y,i} - y_k)}{2\sqrt{2\pi\sigma^2}} \end{aligned} \tag{3.32}$$

and

$$\rho_{i,l} = \left(e^{-\frac{(\lambda_{i,l} - a_i)^2}{2\sigma^2}} - e^{-\frac{(\lambda_{i,l+1} - a_i)^2}{2\sigma^2}}\right) \tag{3.33}$$

Although an analytical expression for $\boldsymbol{J}_d(\boldsymbol{\theta}_K)$ has been derived, in order to obtain $\boldsymbol{J}_d$ involved in the computation of the FIM defined in Eq. (3.27), we need to resort to some numerical techniques for the approximation of the integral that defines this quantity in Eq. (3.28). The procedure we use is a simple Monte-Carlo integration:

1. Draw N_{MC} realization of the state from the prior: $\{\boldsymbol{\theta}_K^i\}_{i=1}^{N_{MC}} \sim p(\boldsymbol{\theta}_k)$

2. Approximate the quantity of interest by:

$$\boldsymbol{J}_d \approx \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} \boldsymbol{J}_d(\boldsymbol{\theta}_K^i) \tag{3.34}$$

Finally, the second term representing the *a priori* information in Eq. (3.27) is

a $3K \times 3K$ matrix defined as:

$$\boldsymbol{J}_p = \begin{bmatrix} \xi & & & \\ & \boldsymbol{\Sigma}_p^{-1} & & 0 \\ & & \ddots & \\ & 0 & & \xi \\ & & & \boldsymbol{\Sigma}_p^{-1} \end{bmatrix} \tag{3.35}$$

with $\xi = \frac{a(a+1)(a+3)}{b^2}$ - *Proof:* See Appendix D.

3.4 Numerical Simulations

In all the experiments, we consider a signal decay exponent and a reference distance as $n = 2$ and $d_0 = 1$ respectively. The ROI is a $100 \times 100m$ field in which the sensors are deployed in a grid where the location of each sensor is assumed to be known. The thresholds of the M-bit quantizer defined in Eq. (3.4) are the same for each sensor and are obtained by following the procedure described in [Niu and Varshney, 2006]. All the results have been obtained by using $N_{\text{MCMC}} = 5$ and $B = K$ (K corresponds to the number of of sources) for the adaptive MWG (summarized in Algo. 1.9) used in the SMC sampler as forward kernel. The forward kernel of the SMC sampler devoted to model $\mathcal{M}_i$ (i.e., i sources) updates successively each source parameters (transmitted power and location).

3.4.1 Case 1: Single Source scenario

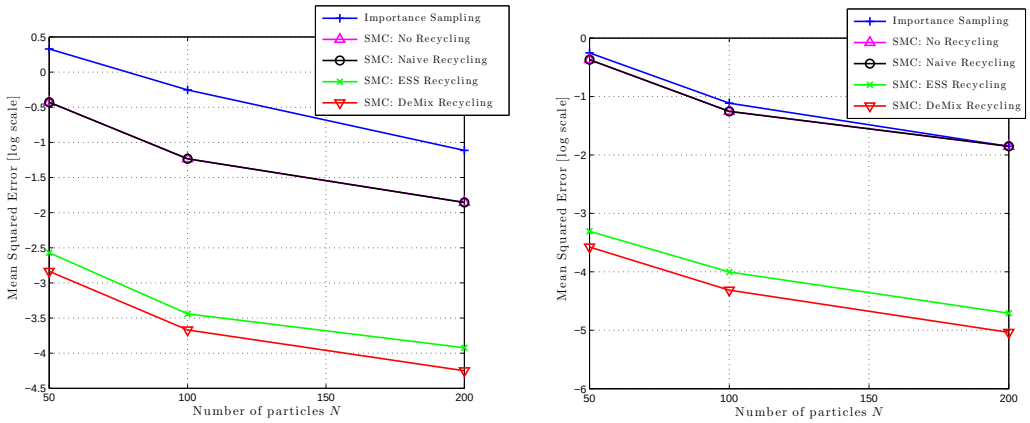
In this first scenario, we study the performance of the SMC sampler for the localization of a unique source with fixed and known transmitted power ($P_1 = 5000$). For this task, 49 sensors are deployed uniformly in a grid with a measurement noise variance of $\sigma^2 = 1$.

Since the posterior distribution of interest is a distribution of only 2 dimension (x_1 and y_1, the two coordinates of the source), we have computed numerically on a fine grid an approximation of the "true" posterior distribution in order to clearly understand the accuracy of the SMC sampler to approximate this posterior distribution of interest. In this section, we also compare the performance of an importance sampler in which the prior distribution is used as proposal for the target location. This scheme has been proposed for this localization problem in [Masazade et al., 2010]. In order to be fair in comparing SMC samplers versus Importance sampler, we decide to set the number of particles used in the importance sampler to be $N \times T$ which corresponds to the total number of particles that have been generated in the SMC sampler through the T iterations of the algorithm.

Firstly, Fig. 3.3 illustrates the performance of the proposed recycling schemes in the estimation of the posterior mean[2]. From these results, it is interesting to remark

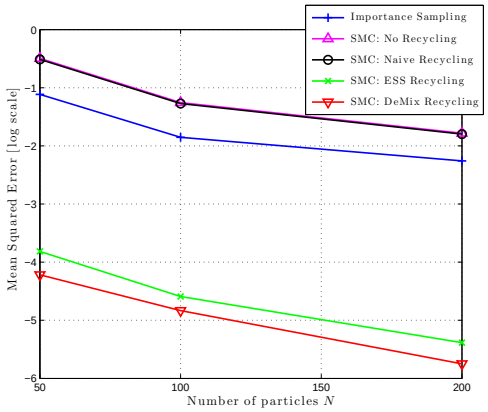
[2]This mean squared error is obtained using the SMC sampler estimate of the posterior mean and the posterior mean obtained from the grid approximation of the posterior

that the benefit of using SMC sampler compared toimportance sampler. Indeed, a bad choice of proposal (like the prior) in the importance sampler generally leads to poor performance. The proposed recycling schemes, and more especially the DeMix-based strategy, outperform significantly the traditional importance sampler.



(a) 25 Iterations

(b) 50 Iterations



(c) 100 Iterations

Figure 3.3: Evolution of the mean squared error between the posterior mean and the "true" one as a function of the number of particles for the different recycling schemes as well as a simple importance sampler in which the number of particles is set to $N \times T$ - 49 sensors with a number of quantization levels $L = 16$ and $\sigma^2 = 1$.

This analysis is confirmed by the study of the Kolmogorov-Smirnov distance in Table 3.1 between the approximation of the posterior from the Monte-Carlo algorithms and the "true" posterior obtained using a fine grid numerical evaluation. It can be remarked that the standard deviation of the KS-distance from the SMC sampler without recycling is smaller than the one from the Importance sampler (the average KS-distance are quite similar). That means that SMC sampler is really more stable in the approximation of the posterior distribution between multiple

runs of the algorithms. Let us point out that the SMC sampler without recycling approximates the posterior with only N particles whereas the importance sampler uses $N \times T$ particles, which clearly shows the benefit of SMC sampler techniques. We can once again see from these results that the proposed DeMix algorithm outperforms all the other techniques both in terms of mean and standard deviation.

		No Recycling	Naive Recycling	ESS-based Recycling	DeMix Recycling	Importance Sampler
25 Iter.	$N = 50$	0.1353 (0.0415)	0.1353 (0.0415)	0.0672 (0.0168)	0.0647 (0.0160)	0.1563 (0.1026)
	$N = 100$	0.0975 (0.0254)	0.0975 (0.0254)	0.0540 (0.0119)	0.0527 (0.0112)	0.1181 (0.0870)
	$N = 200$	0.0822 (0.0214)	0.0822 (0.0214)	0.0468 (0.0091)	0.0456 (0.0082)	0.0943 (0.0715)
50 Iter.	$N = 50$	0.1375 (0.0392)	0.1374 (0.0392)	0.0557 (0.0135)	0.0543 (0.0131)	0.1159 (0.0796)
	$N = 100$	0.0988 (0.0266)	0.0988 (0.0266)	0.0459 (0.0089)	0.0449 (0.0084)	0.0908 (0.0541)
	$N = 200$	0.0790 (0.0200)	0.0790 (0.0200)	0.0406 (0.0067)	0.0399 (0.0064)	0.0737 (0.0601)
100 Iter.	$N = 50$	0.1308 (0.0398)	0.1297 (0.0393)	0.0470 (0.0083)	0.0456 (0.0077)	0.0900 (0.0589)
	$N = 100$	0.0996 (0.0272)	0.0988 (0.0270)	0.0413 (0.0075)	0.0406 (0.0073)	0.0735 (0.0413)
	$N = 200$	0.0797 (0.0194)	0.0791 (0.0053)	0.0371 (0.0054)	0.0367 (0.0053)	0.0611 (0.0427)

Table 3.1: Comparison of recycling schemes for the accuracy to approximate the posterior distribution $p(x_1|\boldsymbol{z})$ in terms of the Kolmogorov-Smirnov distance (mean and standard deviation in parentheses).

Finally, Fig. 3.4 shows the mean squared error between the estimate of the target location and the true one for different numbers of quantization levels. Only the performance when no recycling, DeMix recycling and importance have been depicted since the naïve recycling scheme and the ESS-based strategy give similar results as the no recycling and the DeMix approach, respectively. We can remark that the results are quite close from the posterior Cramér-Rao lower bound and there is not significant difference in term of MSE. The DeMix approach slightly outperforms the other strategies.

3.4.2 Case 2: Multiple source scenario

In this second scenario, we are interested in jointly estimating the number of sources in the ROI as well as their characteristics (transmitted power and location). For this task, the SMC sampler described in Section 3.2.2 is employed with observations given from 100 sensors uniformly deployed in a grid.

To assess the performance of the proposed Bayesian solution, we analyse results obtained from 100 realizations of the observations given that there are 2 sources in the ROI with parameters $\boldsymbol{\theta} = \begin{bmatrix} 3000 & 30 & 70 & 5000 & 70 & 30 \end{bmatrix}^T$.

Fig. 3.5 shows the estimated posterior probability of each model using SMC sampler (with the proposed adaptive cooling schedule). The proposed algorithm is clearly able to detect that there are two targets in the ROI whatever the different values of the sensor noise variance as well as the number of quantization levels.

Then, we compare the benefit of using the proposed adaptive cooling schedule strategy described in Section 2.2.2 in term of the variance of the estimator of the model evidence $p(\boldsymbol{z}|\mathcal{M}_k)$ $\forall k \in \{1, 2, 3, 4\}$. From Fig. 3.6, we can remark that the

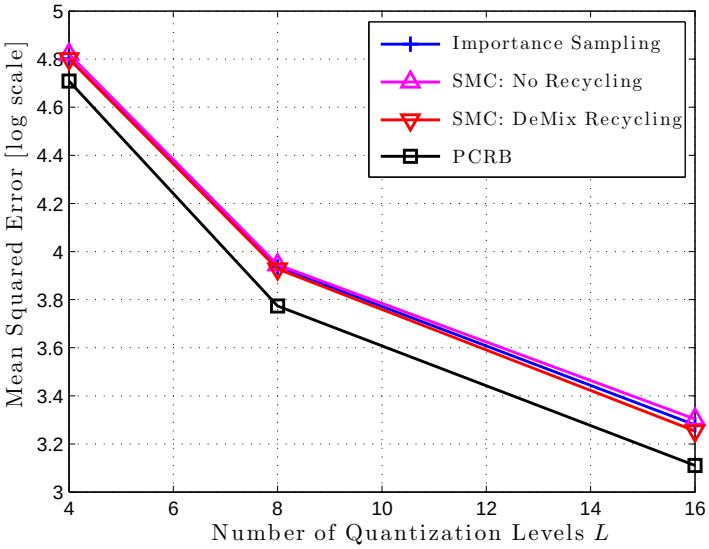
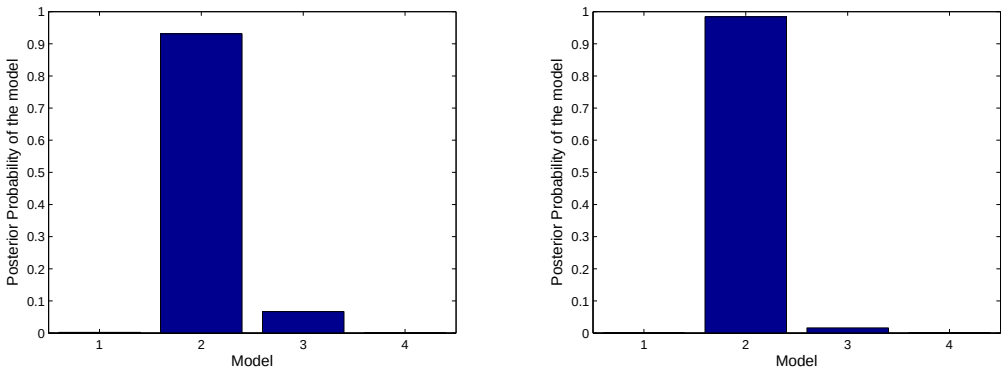


Figure 3.4: Evolution of the mean squared error as a function of the number of quantization levels L for the SMC sampler (with either no recycling or DeMix based strategy - $N = 50$ particles and $T = 100$ Iterations) as well as a simple importance sampler in which the number of particles is set to 5000 - 49 sensors with $\sigma^2 = 1$.

variance of the model evidence estimator clearly decreases when using the proposed adaptive cooling schedule is used compared to a linear cooling schedule. Moreover, from the 100 simulations runs, when the proposed adaptive cooling strategy, the algorithm always detects that there are 2 targets whereas when the linear cooling schedule is used, the algorithm detects the presence of 3 targets for 3 runs out of the 100.

Now, we compare the stability of the posterior mean estimator of the marginal distribution $p(x_1|\boldsymbol{z})$ using the different proposed recycling schemes. Table 3.2 shows the mean and standard deviation of this estimator. Owing to the non-identifiability of the target label in the likelihood in Eq. 3.2 (i.e., the likelihood is the same for any permutation of the target label), we expect the estimated means (for each source coordinates) to be all equal and approximately 50. We can see that the use of the proposed recycling scheme (ESS and DeMix) allows the variance of this estimator to be decreased. The DeMix recycling scheme slightly outperforms the ESS-based strategy in terms of variance.

Fig. 3.7 illustrates the approximated marginals distribution of $p(x_1|\boldsymbol{z})$, $p(y_1|\boldsymbol{z})$ and $p(P_1|\boldsymbol{z})$ using the DeMix recycling scheme for different values of the sensor noise variance. From these results, we can firstly remark, that the algorithm is clearly able to capture the multimodality of each marginals. This multimodality is due to the non-identifiability of the target label in the likelihood in Eq. (3.2). These results also indicate that the posterior distribution becomes peakier around each mode as the variance of the sensor decreases - the proposed algorithm estimates are



(a) $L = 20$ **and** $\sigma^2 = 1$ **(b)** $L = 20$ **and** $\sigma^2 = 10^{-3}$

(c) $L = 40$ **and** $\sigma^2 = 1$ **(d)** $L = 40$ **and** $\sigma^2 = 10^{-3}$

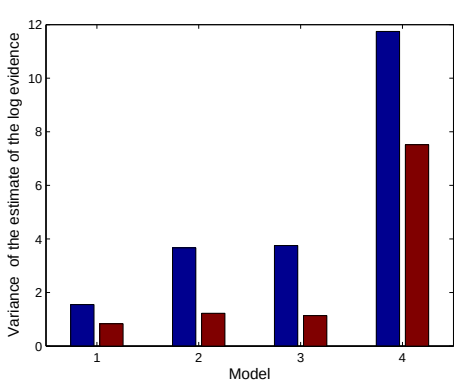
Figure 3.5: Comparison of the posterior probability of each model with different numbers of quantization levels, L, and different values for the measurement noise variance (results are obtained by averaging the posterior obtained from 100 Monte runs of the SMC samplers (100 particles and 50 iterations) with the proposed adaptive cooling schedule).

		No Recycling	Naive Recycling	ESS-based Recycling	DeMix Recycling
25 Iter	$N = 50$	48.4528 ± 8.0339	48.4528 ± 8.0339	48.2660 ± 7.3589	48.1991 ± 7.1917
	$N = 100$	48.0451 ± 5.5036	48.0451 ± 5.5036	48.2098 ± 5.0768	48.1178 ± 4.8631
50 Iter	$N = 50$	48.2521 ± 6.9984	48.2521 ± 6.9984	48.1172 ± 5.7118	48.0864 ± 5.4831
	$N = 100$	47.1779 ± 5.5427	47.1779 ± 5.5427	47.2778 ± 4.6058	47.2352 ± 4.3480

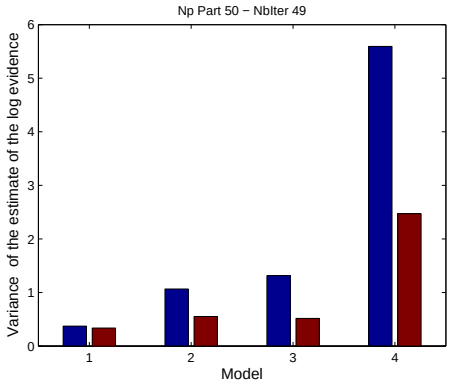
Table 3.2: Comparison of recycling schemes for the stability to approximate the posterior mean $p(x_1|\boldsymbol{z})$ (x-coordinate of the first target) in term of the mean and the standard deviation obtained from 100 Monte-Carlo runs ($L = 40$ and $\sigma^2 = 1$)

consequently converging (in the mean square sense).

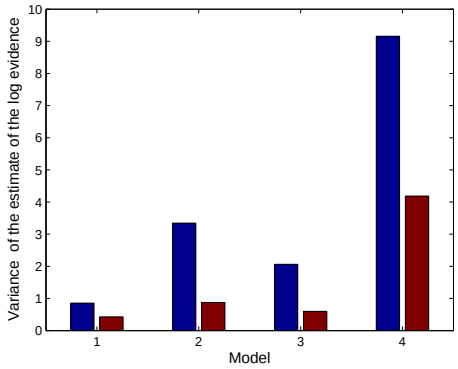
Let us now illustrate with Fig. 3.8, the challenging problem of having two targets of interest that are placed very close to each other. This case is very difficult to



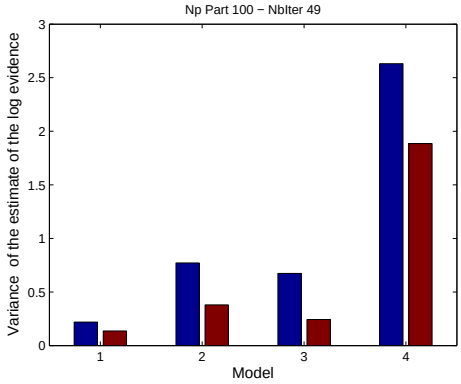
(a) 50 Particles - 25 Iterations



(b) 50 Particles - 50 Iterations



(c) 100 Particles - 25 Iterations

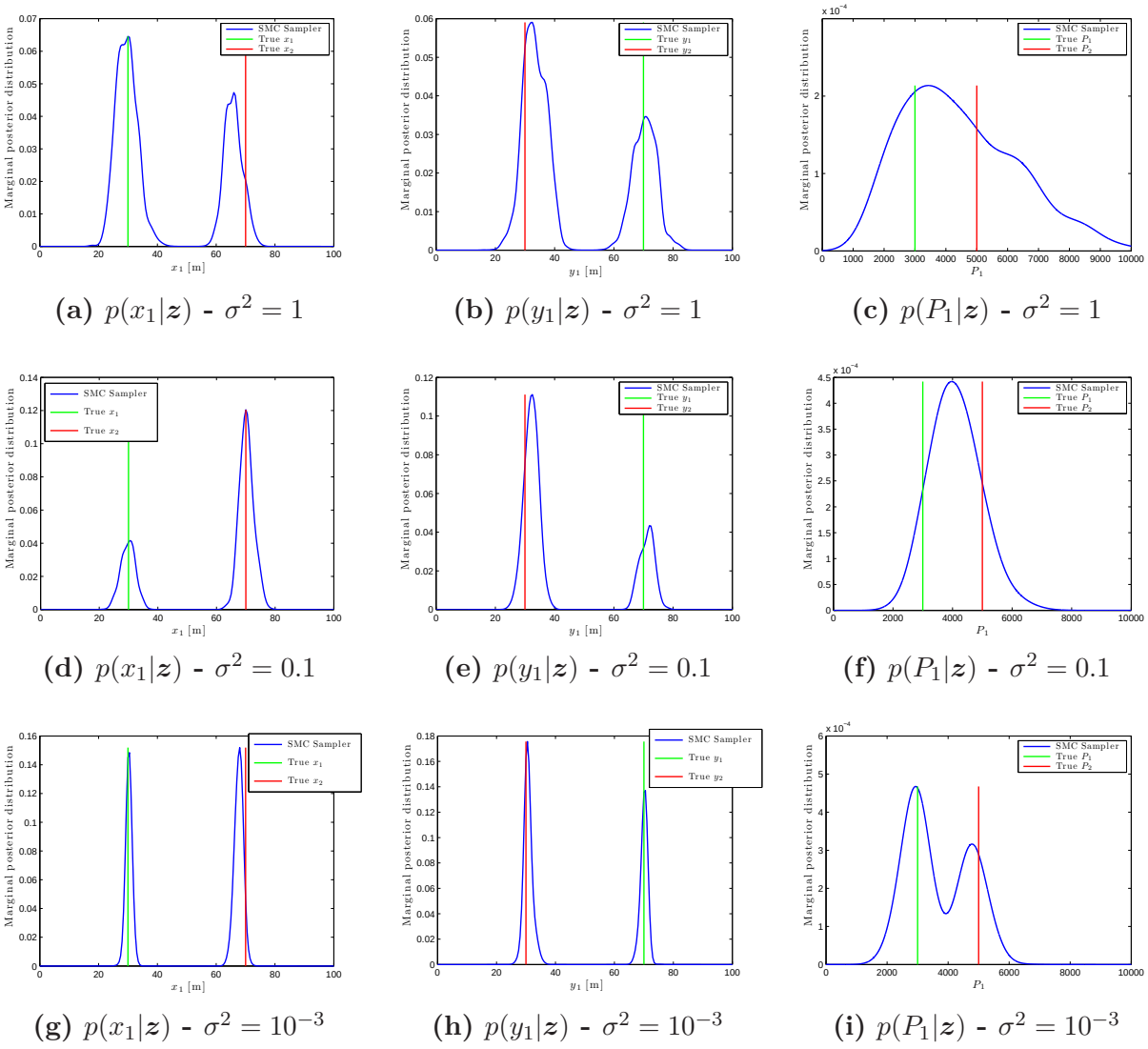


(d) 100 Particles - 50 Iterations

Figure 3.6: Comparison between the variances of the evidence estimate for each model when either the linear cooling schedule (blue) or the proposed adaptive cooling strategy (red) is used - (Number of quantization levels: $L = 40$ and $\sigma^2 = 10^{-3}$.

deal with owing to the label switching problem when one is interested in point estimate of the sources' location as discussed in Section 3.2.3. From the figures of the particles before the relabeling, we can firstly see (as illustrated in Fig. 3.7) that the SMC sampler is able to capture the multimodality of the marginal of each target, thus clearly showing its ability to efficiently explore the space by not being trapped in local modes (i.e. some specific configuration of the labels). Secondly, the relabeling algorithm shows its limitation when the sources of interest are very close to each other (Cases a and b in Fig. 3.8). In that case, it becomes difficult to isolate correctly the targets. However, the relabeling algorithm is working well when the modes are more separated as the noise variance decreases and/or as the distance between the sources increases.

Finally, we compare the performances of the proposed algorithm when 4 sources are in the ROI. In order to obtain the following results, 100 realizations of the four sources and associated observations have been drawn from the prior and likelihood



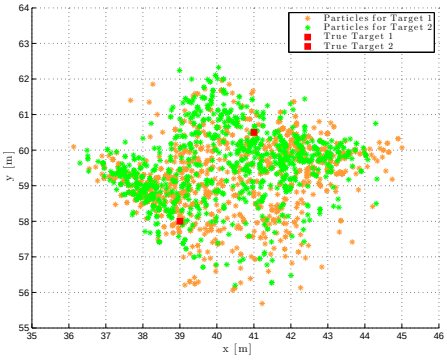
(a) $p(x_1|\boldsymbol{z})$ - $\sigma^2 = 1$ **(b)** $p(y_1|\boldsymbol{z})$ - $\sigma^2 = 1$ **(c)** $p(P_1|\boldsymbol{z})$ - $\sigma^2 = 1$

(d) $p(x_1|\boldsymbol{z})$ - $\sigma^2 = 0.1$ **(e)** $p(y_1|\boldsymbol{z})$ - $\sigma^2 = 0.1$ **(f)** $p(P_1|\boldsymbol{z})$ - $\sigma^2 = 0.1$

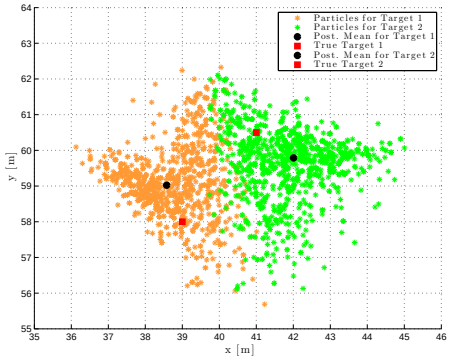
(g) $p(x_1|\boldsymbol{z})$ - $\sigma^2 = 10^{-3}$ **(h)** $p(y_1|\boldsymbol{z})$ - $\sigma^2 = 10^{-3}$ **(i)** $p(P_1|\boldsymbol{z})$ - $\sigma^2 = 10^{-3}$

Figure 3.7: Approximation of the posterior marginal distributions using the SMC sampler with the DeMix recycling scheme (100 Particles - 100 iterations) - Number of quantization levels : $L = 40$.

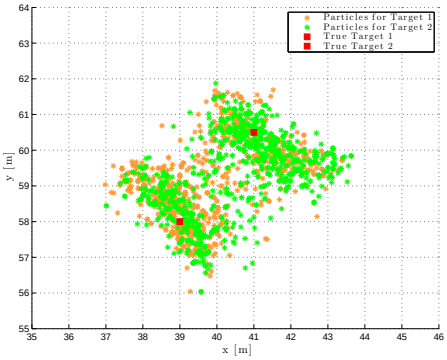
defined in Sections 3.1 and 3.2 ($a = 50$ and $b = 250000$ have been used for the prior of transmitted power of each source). Fig. 3.9 illustrates the ability of the proposed to detect the correct number of targets. The correct number (Model 4) of target is chosen for each of the 100 Monte-Carlo runs of the algorithm when the noise measurement decreases and the number of quantization levels increases. Even for a small number of quantization levels, the algorithm detects correctly that there are 4 sources 95 times over 100.



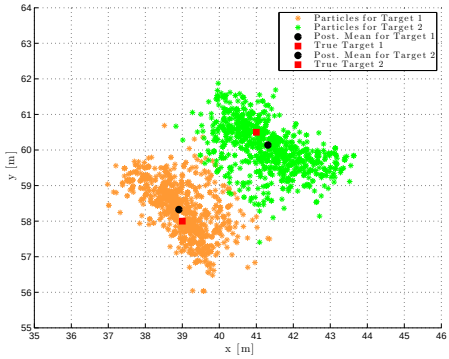
(a) **Before Relabeling -** $\sigma^2 = 0.05$



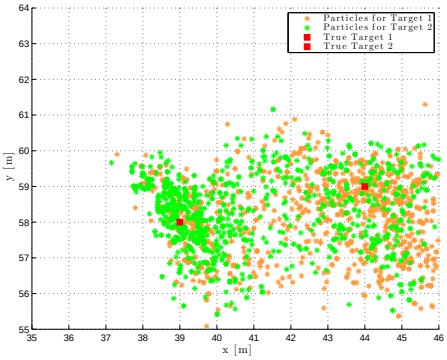
(b) **After Relabeling -** $\sigma^2 = 0.05$



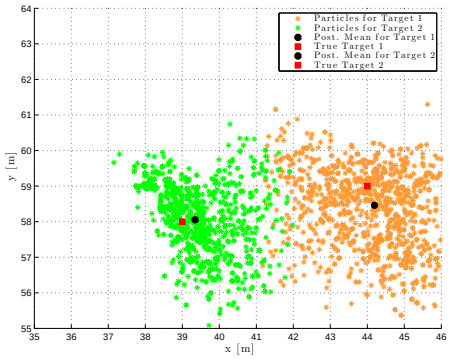
(c) **Before Relabeling -** $\sigma^2 = 0.01$



(d) **After Relabeling -** $\sigma^2 = 0.01$

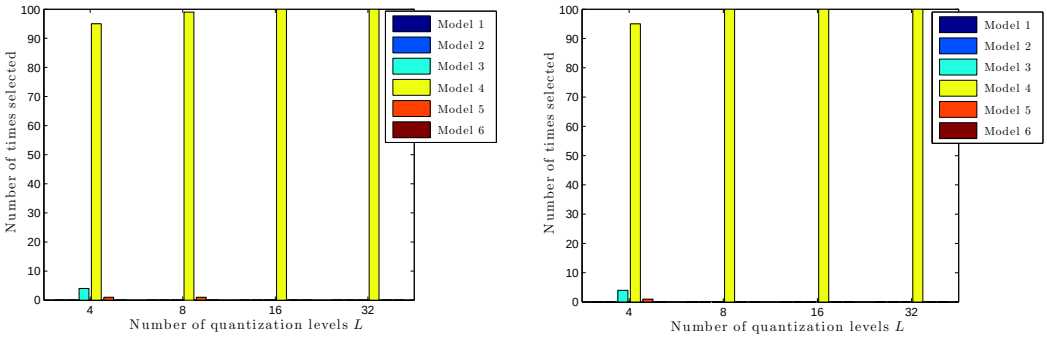


(e) **Before Relabeling -** $\sigma^2 = 0.05$



(f) **After Relabeling -** $\sigma^2 = 0.05$

Figure 3.8: Illustration of the effect of having 2 targets that are very close to each other with the representation of the particles before and after the relabeling algorithm (The sources' coordinates are (39,58),(41,59) [and (39,58),(44,59)] in cases a, b, c and d [in e, f respectively]).



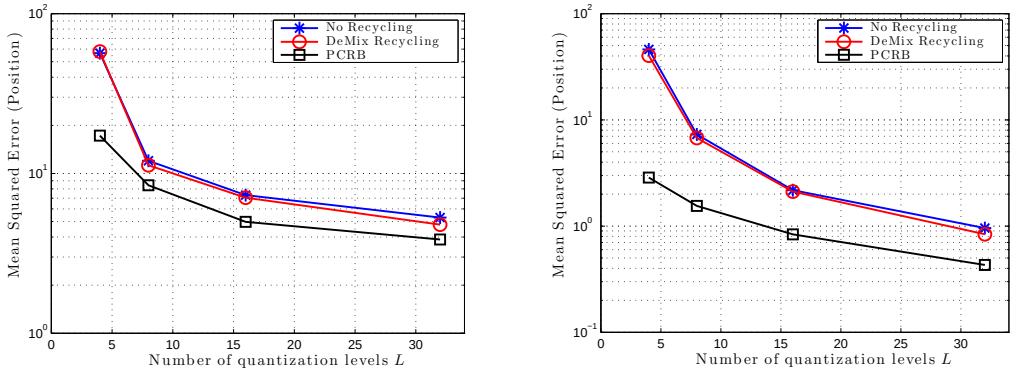
(a) $\sigma^2 = 1$ **(b)** $\sigma^2 = 0.05$

Figure 3.9: Number of times that each model has been selected with the approximated model posterior from the SMC sampler using the proposed adaptive cooling schedule ($N = 50$ particles and $T = 100$ Iterations) with different number of quantization levels (over 100 realizations of the scenario)

In Fig. 3.10 the performance of the proposed SMC sampler in term of the mean squared error between point estimate $\hat{\boldsymbol{\theta}}_p$ of the algorithm and the true location $\boldsymbol{\theta}_p$ of the four sources:

$$MSE = \text{trace}\left\{\mathbb{E}\left[(\hat{\boldsymbol{\theta}}_p - \boldsymbol{\theta}_p)(\hat{\boldsymbol{\theta}}_p - \boldsymbol{\theta}_p)^T\right]\right\} \tag{3.36}$$

with $\hat{\boldsymbol{\theta}}_p = \begin{bmatrix}\hat{x}_1 & \hat{y}_1 & \cdots & \hat{x}_4 & \hat{y}_4\end{bmatrix}^T$ and $\boldsymbol{\theta}_p = \begin{bmatrix}x_1 & y_1 & \cdots & x_4 & y_4\end{bmatrix}^T$ represents the estimated and the true location of the two targets, respectively. We also plot the associated PCRB that we have derived in Section 3.3. Owing to the non-identifiability of the target label in the likelihood, the posterior distribution will be multimodal as illustrated in Fig 3.7. As a consequence, in order to obtain the point estimate for the state of interest, we use the procedure described in Section 3.2.3 since the MMSE estimate (i.e., posterior mean) would lead to very poor performance. In order to obtain the results we use 100 realizations (of the different source characteristics and associated observations by avoiding the case in which two targets are very close). The results depicted in Fig. 3.10 clearly demonstrate the good localization performance of the proposed algorithm. As expected, the accuracy on the localization improves as the number of quantization levels of all the sensors in the ROI increases and the measurement noise variance decreases. Furthermore, the DeMix recycling strategy slightly improves the mean squared error on the location of the four sources.



(a) $\sigma^2 = 1$ **(b)** $\sigma^2 = 0.05$

Figure 3.10: Evolution of the mean squared error for the source locations as a function of the number of quantization levels L for the SMC sampler (with either no recycling or DeMix based strategy - $N = 50$ particles and $T = 100$ Iterations) with two different values of the measurement noise σ^2

3.5 Conclusion

In this chapter, we addressed the problem of localizing an unknown number of energy emitting sources in wireless sensor networks with quantized data. We provided a generalization of recent existing works considering a single source. We firstly proposed a Bayesian solution for the joint estimation of the unknown number of sources as well as their associated parameters. Then, we derived the posterior Cramér-Rao bound for the estimation of the characteristics of these multiple energy emitting sources. Numerical simulations clearly illustrated the ability of the proposed SMC sampler to perform this challenging joint estimation. Moreover, the different experiments showed that the proposed adaptive cooling schedule as well as the proposed recycling schemes for SMC sampler improve quite significantly the accuracy of the estimators that are required for model selection (i.e., the number of sources) and the estimation of the source characteristics, respectively.

Chapter 4

Bayesian Solution for Penalized Regression

Contents

Penalized regression methods have received a great deal of attention in recent years, mostly through frequentist models using ℓ_1-regularization. However, all existing works assume that the design matrix, that links the explanatory variables to the observed response, is known a priori. Unfortunately, this is often not the case and thus solving this challenging problem is of considerable interest. In this chapter, we look at a fully Bayesian formulation of this problem. We propose the use of Sequential Monte Carlo samplers for joint model selection and parameters estimation. Furthermore, a new class of priors based on α-stable family distribution is proposed as non-convex penalty for regularization of the regression coefficients. The performance of the proposed methodology is demonstrated in two different settings.

4.1 Regression Analysis

4.1.1 Introduction

One of the most fundamental problems appearing in a wide range of applications is to quantity the relationship between a response (output) variable of interest and some input variable (predictors). The aim of regression is to approximate this relationship between output and input variable which is continuous as opposed to classification analysis in which the response is categorical.

Let us illustrate with an example, depicted in Fig. 4.1, for which regression is appropriate. The interest lies in determining the relationship between the input variables, x and the response, y. The aim of the regression is to provide a good approximation to the true relationship between these two variables. The simplest approximation widely used in statistics assumes that the relationship is linear such that :

$$y = \beta_0 + \beta_1 x \tag{4.1}$$

where β_0 and β_1 give respectively the intercept and slope of the line. The line with two parameters set so that they best fit the data, in the least square sense, is shown in Fig. 4.1. However, the linear functional seems to oversimplify the true relationship between the variables, and fails to model accurately the data.

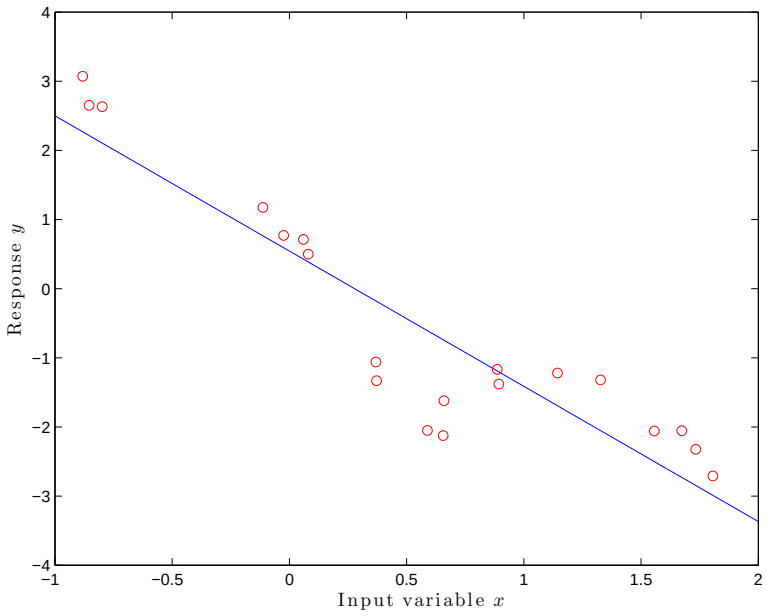


Figure 4.1: Illustration of the regression with linear relationship.

Despite their widespread use and popularity, we see from this simple example that the linear models can be too restrictive to accurately capture the actual underlying relationship. As a consequence, more sophisticated models using nonlinear

functional for the relationship, are required to model accurately a wide range of datasets (e.g., Fig. 4.2). Let us now describe the basics of regression modeling.

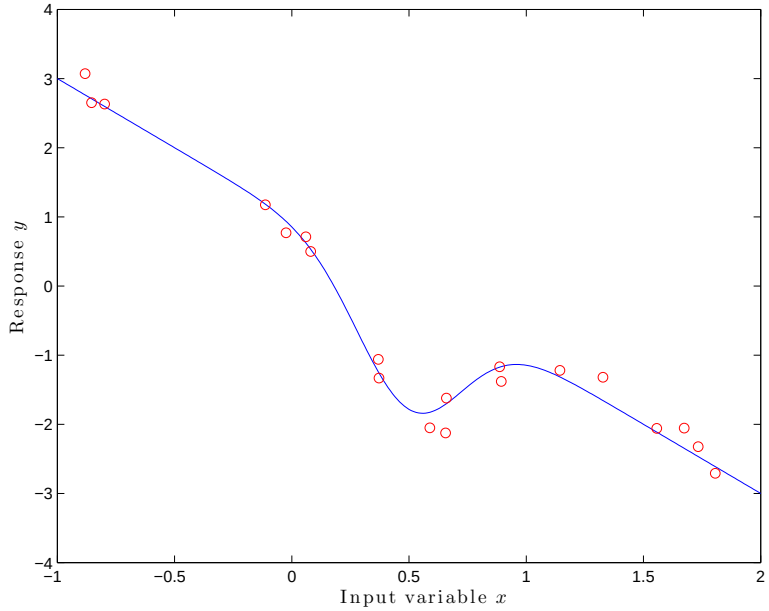


Figure 4.2: Illustration of the regression with nonlinear functional.

4.1.2 Basics of Regression Modeling

As discussed previously, the regression problem consists in determining the relationship between some response variable y_i and a set of k predictor variables (also called covariates) $\mathbf{x}_i = \left[x_{i,1} \dots x_{i,k}\right]$. The most common structural assumption is that the responses are linked to predictors via some deterministic function f and some additive random error component $\boldsymbol{\varepsilon}_i$, so that $\forall i = 1, \dots, n_{\mathbf{y}}$

$$y_i = f(\mathbf{x}_i) + \boldsymbol{\varepsilon}_i \tag{4.2}$$

where $\boldsymbol{\varepsilon}_i$ is a zero-mean error random variable.

In most situations the predictor variables, $\mathbf{x}_i$ are assumed to be observed without error so they are not considered as unknown random variables. The aim is thus to determine f so we can uncover the true relationship between the response and the predictors in order to do for example some prediction of the response for some specific value of the predictor $\mathbf{x}_*$.

However, the true regression function f is unknown. Therefore, we must find approximation to it as close as possible to the truth. To do this, we must make use of the observed dataset, which consists of $n_{\mathbf{y}}$ observed responses at some known predictor locations $\{y_i, \mathbf{x}_i\}_{i=1}^{n_{\mathbf{y}}}$. A simple solution to approximate the true function

f is to make direct linear assumptions about the estimating function:

$$f(\mathbf{x}_i) \approx \beta_0 + \sum_{n=1}^{k} \beta_n x_{i,n} \tag{4.3}$$

However, as discussed previously and as it is expected, this linear relationship does not have the flexibility to model general dataset adequately. Instead, a more general model based on *basis functions* is generally used. This model assumes that f can be better approximated by a linear combination of basis functions and corresponding coefficients:

$$f(\mathbf{x}_i) \approx \sum_{j=1}^{p} \beta_j \Phi_{i,j}\left(\mathbf{x}_i\right) \tag{4.4}$$

where $\boldsymbol{\beta} = \begin{bmatrix}\beta_1 \dots \beta_p\end{bmatrix}^T$ is the set of coefficients corresponding to basis functions for the i- observations $(\Phi_{i,1}, \dots, \Phi_{i,p})$. Let us remark that the linear model in (4.3) is just a special case of (4.4) where $p = k+1$, $\Phi_{i,1}\left(\mathbf{x}_i\right) = 1$ and $\Phi_{i,j}\left(\mathbf{x}_i\right) = x_{i,j-1}$ for $j = 2, \dots, k+1$.

The basis functions used in (4.4) are nonlinear transformations of the input variables $\mathbf{x}$. Table 4.1 lists linear, polynomial and some common basis functions, some of them will be used later in the numerical simulation section of this chapter.

Table 4.1: Some common basis functions.

Name	Definition
Polynomial	$\Phi_{i,j}(\mathbf{x}_i) = x_{i,j}^j$
Gaussian	$\Phi_{i,j}(\mathbf{x}_i) = \exp\left\{\frac{\|(\mathbf{x}_i - \mathbf{c}_j\|_2^2}{r_j^2}\right\}$
Multiquadric	$\Phi_{i,j}(\mathbf{x}_i) = \frac{\sqrt{r_j^2 + \|(\mathbf{x}_i - \mathbf{c}_j\|_2^2}}{r_j}$
Inverse quadratic	$\Phi_{i,j}(\mathbf{x}_i) = \frac{1}{1+\left(\frac{\|(\mathbf{x}_i - \mathbf{c}_j\|_2}{r_j}\right)^2}$
Sigmoidal	$\Phi_{i,j}(\mathbf{x}_i) = \frac{1}{1+\exp\left\{-\frac{\|(\mathbf{x}_i - \mathbf{c}_j\|_2}{r_j}\right\}}$

Note: $\mathbf{c}_1, \dots, \mathbf{c}_p$ and $r_1, \dots, r_p$ the set of p centers and radius of the basis functions function, respectively.

To summarize, the aim of regression is to make inference on the unknown parameters which is typically the coefficients $\boldsymbol{\beta} = \begin{bmatrix}\beta_1 \dots \beta_p\end{bmatrix}^T$ given the observation of the response variables at some known predictor locations by using the functional relationship given in (4.4):

$$\mathbf{y} = \boldsymbol{\Phi}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \tag{4.5}$$

where $\mathbf{y} = \begin{bmatrix}y_1 \dots y_{n_\mathbf{y}}\end{bmatrix}^T$, $\boldsymbol{\varepsilon} = \begin{bmatrix}\varepsilon_1 \dots \varepsilon_{n_\mathbf{y}}\end{bmatrix}^T$ and

$$\boldsymbol{\Phi} = \begin{bmatrix} \Phi_{1,1}\left(\mathbf{x}_1\right) & \cdots & \Phi_{1,p}\left(\mathbf{x}_1\right) \\ \vdots & \ddots & \vdots \\ \Phi_{n_\mathbf{y},1}\left(\mathbf{x}_{n_\mathbf{y}}\right) & \cdots & \Phi_{n_\mathbf{y},p}\left(\mathbf{x}_{n_\mathbf{y}}\right) \end{bmatrix} \tag{4.6}$$

known as the *design matrix* of the regression.

4.1.3 Ordinary Least Square Solution

In order to estimate the regression coefficients $\boldsymbol{\beta}$ from the response vector $\mathbf{y}$, the most common method is the Ordinary Least Squares (OLS) which minimizes the residual sum of squares (RSS) with respect to $\boldsymbol{\beta}$:

$$\hat{\boldsymbol{\beta}}_{OLS} = \arg\min_{\boldsymbol{\beta}} (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta})^T (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta}) \tag{4.7}$$

thus yielding the following unbiased estimator

$$\hat{\boldsymbol{\beta}}_{OLS} = (\boldsymbol{\Phi}^T \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^T \mathbf{y} \tag{4.8}$$

Despite its simplicity and unbiasedness, the OLS estimator is, however, not always satisfactory because it is not unique if the design matrix $\boldsymbol{\Phi}$ is less than full rank and the variance of the estimator $\mathbb{V}\mathrm{ar}(\hat{\boldsymbol{\beta}}_{OLS}) = \sigma^2(\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}$ is large if $\boldsymbol{\Phi}$ is close to collinear. Therefore, both prediction and interpretation based on $\hat{\boldsymbol{\beta}}_{OLS}$ often work poorly, especially when the sample size $n_{\mathbf{y}}$ is not large compared to the number of variables p.

To overcome these problems, penalized regression methods have been proposed.

4.2 Penalized Regression

In this section, the basic properties of some already established regularization schemes are reviewed. All these regularization approaches (when the observation noise is assumed to be Gaussian - this point will be discussed later) are based on penalized least squares

$$PLS(\gamma, \boldsymbol{\beta}) = (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta})^T (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta}) + P(\gamma, \boldsymbol{\beta}) \tag{4.9}$$

and estimates of the parameter vector $\boldsymbol{\beta}$ are obtained by minimizing this equation, i.e.,

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \{PLS(\gamma, \boldsymbol{\beta})\} \tag{4.10}$$

The penalty term $P(\gamma, \boldsymbol{\beta})$ depends on the positive tuning parameter γ (regularization coefficient) which controls the shrinkage intensity. For the tuning parameter $\gamma = 0$ we obtain the ordinary least squares solution. On the contrary, for large values of γ the influence of the penalty term on the coefficient estimate increases.

4.2.1 Ridge Regression

One of the most popular alternative solutions to OLS estimates is ridge regression introduced by [Hoerl and Kennard, 1970]. Ridge regression finds the coefficients

$\boldsymbol{\beta}$ minimizing the RSS subject to an ℓ_2 norm constraint on the coefficients. The solution $\hat{\boldsymbol{\beta}}_{ridge}$ can be written as follows

$$\hat{\boldsymbol{\beta}}_{ridge} = \arg\min_{\boldsymbol{\beta}} \left\{ (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta})^T (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta}) \right\}, \qquad s.t. \sum_{j=1}^{p} |\beta_j|^2 \leq t, \quad t \geq 0. \tag{4.11}$$

or equivalently

$$\hat{\boldsymbol{\beta}}_{ridge} = \arg\min_{\boldsymbol{\beta}} \left\{ (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta})^T (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta}) + \gamma \sum_{j=1}^{p} \beta_j^2 \right\}, \qquad \gamma \geq 0. \tag{4.12}$$

Thus, the parameter t is clearly related to the parameter γ. This means that for a specific value γ there exists a value t such that the estimation equations (4.11) and (4.12) exhibit the same solution, i.e.,

$$\hat{\boldsymbol{\beta}}_{ridge} = (\boldsymbol{\Phi}^T \boldsymbol{\Phi} + \gamma \boldsymbol{I}_{n_{\mathbf{y}}})^{-1} \boldsymbol{\Phi}^T \mathbf{y} \tag{4.13}$$

where $\boldsymbol{I}_{n_{\mathbf{y}}}$ is the $n_{\mathbf{y}} \times n_{\mathbf{y}}$ identity matrix. By adding $\gamma \boldsymbol{I}_{n_{\mathbf{y}}}$ to $\boldsymbol{\Phi}^T \boldsymbol{\Phi}$, this results in a regular and invertible matrix even in both cases of multi–collinearity. Thus, ridge regression provides unique estimates in such situations.

Contrary to the OLS estimates the ridge estimator is not unbiased. Hence this regularization method introduces a little bias to reduce the variance and the mean squared error, respectively of the estimates and possibly improves the prediction accuracy. Due to this, the resulting model is less sensitive to changes in the data. To summarize, ridge regression yields more stable estimates by shrinking coefficients, but does not select predictors and therefore does not give an easily interpretable model, especially when p is large.

4.2.2 LASSO

Another common penalized regression approach is the *least absolute shrinkage and selection operator* (LASSO) proposed by [Tibshirani, 1996]. As with the ridge regression, the lasso estimates are obtained by minimizing the RSS but here subject to a constraint based on ℓ_1-norm:

$$\hat{\boldsymbol{\beta}}_{LASSO} = \arg\min_{\boldsymbol{\beta}} \left\{ (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta})^T (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta}) \right\}, \qquad s.t. \sum_{j=1}^{p} |\beta_j| \leq t, \quad t \geq 0. \tag{4.14}$$

Or equivalently, the LASSO determines the coefficient vector $\hat{\boldsymbol{\beta}}_{LASSO}$ satisfying

$$\hat{\boldsymbol{\beta}}_{LASSO} = \arg\min_{\boldsymbol{\beta}} \left\{ (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta})^T (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta}) + \gamma \sum_{j=1}^{p} |\beta_j| \right\}, \qquad \gamma \geq 0. \tag{4.15}$$

Unlike the quadratic constraint, the ℓ_1 norm constraint yields a sparse solution.

With decreasing values of the parameter t the estimated lasso coefficients are shrunk towards zero and some coefficients are exactly set to zero; for $t = 0$ all of them are equal to zero. Otherwise, a value of $t \geq t_{OLS}$ results in the unpenalized least squares estimates if the OLS estimator exists. In comparison to the parameter t, the parameter γ has the opposite effect on the estimation.

4.2.3 Bridge Regression

In [Frank and Friedman, 1993], the authors introduced bridge regression which, subject to a constraint on the ℓ_q, with $q \geq 0$, minimizes RSS.

$$\hat{\boldsymbol{\beta}}_{bridge} = \arg\min_{\boldsymbol{\beta}} \left\{ (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta})^T (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\beta}) + \gamma \sum_{j=1}^{p} |\beta_j|^q \right\}, \qquad \gamma \geq 0. \tag{4.16}$$

The estimator from bridge regression is not explicit, but Frank and Friedman argued that the optimal choice of the parameter q yields reasonable predictors. The bridge regression include both the ridge and LASSO regression as special cases with $q = 2$ and $q = 1$, respectively.

4.2.4 Discussion

The ridge regression utilizes the ℓ_2 penalty and is best used where there are high correlations between predictor, or we can say, collinearity. The LASSO utilizes the ℓ_1 penalty and does both continuous shrinkage and automatic variable selection simultaneously. Both ℓ_1 and ℓ_2 penalized estimation methods shrink the estimates of some certain regression coefficients towards to zeros. The purpose of the shrinkage is to avoid the overfit of the data which could be due to some collinearity of the design matrix or to the high-dimensionality of the regression coefficients compared to the number of observations available. However, the effects of ℓ_1 and ℓ_2 penalization are quite different in practice. Applying an ℓ_2 penalty, some regression coefficients are shrunk to small but non zeros values. On the contrary, applying ℓ_1 penalty, usually results in many regression coefficients shrunk exactly to zero and a few others with comparatively little shrinkage. The LASSO continuously shrinks the coefficient toward 0 as γ increases, and some coefficients are exactly shrunk to 0 if γ is sufficiently large. Moreover, continuous shrinkage often improve the prediction accuracy due to the bias-variance trade–off. The LASSO is supported by much theoretical work. [Meinshausen and Bühlmann, 2006] showed that variable selection with the LASSO can be consistent if the underlying model satisfies some conditions.

However, a limitation in the LASSO is the use of identical penalization on each regression coefficient which can lead to unacceptable bias in the resulting estimates [Fan and Li, 2001]. Indeed, the classical ℓ_1-regularization can lead to an over-shrinkage of large regression coefficients even in the presence of many zeros. This has resulted in sparsity-inducing non-convex penalties as with the bridge regression

framework, i.e., $\gamma \sum_{i=1}^{p} |\beta_i|^q$ with $q \in (0,1)$, which leads to the ℓ_q-regularization problem [Polson et al., 2011]. Alternative non-convex approaches are using different penalty coefficients on each regression coefficient, i.e., $\sum_{i=1}^{p} \gamma_i |\beta_i|$ have been proposed, as have grouping regularization constraints, see adaptive and sequential estimation approaches in [Zou, 2006, Lee et al., 2012, Candès et al., 2008, Chartrand and Yin, 2008]. Compared to these methods, the bridge regression possesses the great advantage of not introducing additional variables that need to be either tuned nor estimated. In this work, we will focus our study on the use of non-convex penalty functions with the same penalty coefficient for each regression terms.

4.2.5 Bayesian Formulation

From the different expression of penalized regression defined previously, we can easily remark that such minimization problem can be placed into a Bayesian setting. Under a Bayesian modelling paradigm, in which the regression coefficients are treated as a random vector, one may recover the LASSO estimates from the maximum a posteriori (MAP) point estimator of the coefficients via a choice of prior on the coefficients given by the multivariate Laplace distribution:

$$p(\boldsymbol{\beta}) \propto \exp(-\gamma \sum_{i=1}^{p} |\beta_i|) \tag{4.17}$$

Indeed, maximizing the log of the product of a Gaussian likelihood and this prior is exactly equivalent of LASSO minimization equation defined in (4.15).

4.2.5.1 Exponential Power distribution as regularization prior

More generally, the bridge regression minimization can equivalently be written in a Bayesian setting by using the exponential power (EP) distribution:

$$f(\boldsymbol{\beta}; \gamma, q) = \prod_{i=1}^{p} \frac{q}{2\gamma\Gamma(1/q)} \exp\left(-\left|\frac{\beta_i}{\gamma}\right|^q\right) \tag{4.18}$$

As a consequence, some works have been proposed in the literature in order to propose some Bayesian solutions to this penalized regression problems. In particular, the authors in [Park and Casella, 2008, Tibshirani, 2011, Polson et al., 2011], propose some Monte-Carlo algorithm to obtain the posterior distribution of the unknown coefficients. From a Bayesian perspective the use of MAP estimates is not really exploiting the full posterior information, see [Tibshirani, 2011] and [Park and Casella, 2008] who explore full posterior distribution using a Laplace prior via Markov chain Monte Carlo (MCMC). Moreover, some confidence interval can be obtained regarding the estimation of the coefficient but also on the predicted response at some specific location of the predictors, $\mathbf{x}_*$ which is impossible to obtain by using only point estimate of the unknown coefficients. As a conse-

quence, we will focus on designing Bayesian solutions that will be able to give us an approximation of the posterior distribution regarding the parameters of interest. Moreover, we study more specifically the case in which a non-convex penalty function is obtained though the use of a regularization prior, as the exponential power distribution with $q \in (0, 1]$, in order to avoid the over-shrinkage problem of large regression coefficients with convex penalty function. In this work, we propose an alternative regularization prior based on heavy-tailed distribution that will induce non-convex penalization.

4.2.5.2 α-Stable distribution as regularization prior

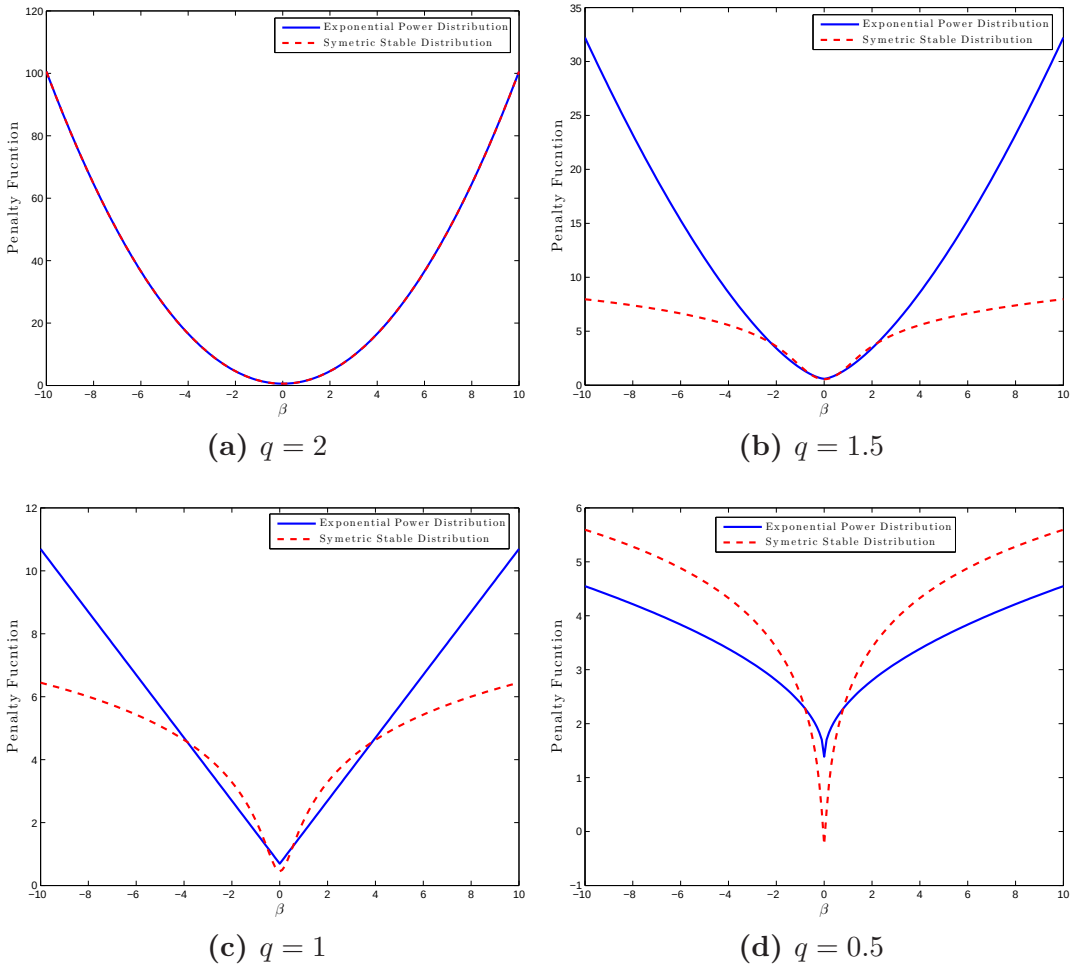
We propose to study the use of the symmetric α-Stable distribution as a new class of prior distributions for the regression coefficients. The α-stable distribution with characteristic exponent $0 < \alpha = q < 2$, dispersion parameter $\gamma > 0$, location parameter δ and skewness parameter $\beta \in [-1; 1]$, is only defined through its characteristic function :

$$\log \phi(t) = \begin{cases} i\delta t - \gamma^q |t|^q \left[1 - i\beta \text{sign}(t) \tan\left(\frac{q\pi}{2}\right)\right] & q \neq 1 \\ i\delta t - \gamma |t| \left[1 + i\beta \text{sign}(t) \frac{2}{\pi} \log |t|\right] & q = 1 \end{cases}$$

Since regularization prior are typically symmetric, we will considered symmetric α-stable ($\mathcal{S}\alpha\mathcal{S}$) distribution ($\delta = 0$, $\beta = 0$).

4.2.5.3 Comparison between α-Stable and Exponential distributions

To understand the behavior of these two different prior choices of Bayesian regularization we present below a few comparison of the influence of the prior with respect to the type of penalty, i.e., the shrinkage effect each choice may impose. To achieve this we present plots of the negative log densities (i.e., penalty functions) for the α-stable distribution and the exponential power distribution, see Figure 4.3 for different values of q. For $q = 2$, these two distributions are equivalent to the normal distribution, producing a convex penalty (Ridge regression). For $q < 1$ the penalty function from the exponential power distribution is non-convex whereas the one from the symmetric α-stable distribution is non-convex when the characteristic exponent of the distribution is $0 < q < 2$. In particular, for $q = 1$, we can see the greater Kurtosis and heavier tails provided by the stable distribution. As mentioned previously, the relatively light tails of the exponential power distribution prior is unattractive as it tends to shrink large values of the coefficients even when there is clear evidence from the likelihood that they corresponds to large values. This is an important motivation for the class of α-stable priors we introduce in this chapter.



(a) $q = 2$ **(b)** $q = 1.5$

(c) $q = 1$ **(d)** $q = 0.5$

Figure 4.3: Comparison of the penalty term induced by the log prior of the regression coefficient to be either the exponential power distribution or the α-sable distribution ($\gamma_{EP} = 2\gamma_{\mathcal{S}\alpha\mathcal{S}} = 1$)

4.3 Generalized Linear Models

4.3.1 Introduction and motivation

Until now, we have assumed that the errors ε_i are normally distributed but this is not generally true in practice. In this section, we describe a widely utilized class of regression models, the Generalized Linear Model (GLM) structure [Nelder and Baker, 1972], that removes this assumption.

Effectively, the generalized linear model (GLM) is a flexible generalization of ordinary linear regression that allows for response variables to be distributed from a more general distribution that the standard linear regression model which assumes normally distributed responses, see discussions in [McCullagh and Nelder, 1989, Denison et al., 2002]. In [Nelder and Wedderburn, 1972] original formulation, the distribution of each response variable is a member of an *exponential family*, such as Gaussian, binomial, Poisson,...etc. The GLM generalizes linear regression by allowing the linear model to be related to the response variable via a link function and by allowing the magnitude of the variance of each measurement to be a function of its predicted mean value. When specifying a GLM regression model one must consider three aspects the distribution for the response, the link function and the mean/variance relationships in terms of the covariates.

4.3.2 Definition of the Generalized Linear Model

A GLM model postulates that given $\mathbf{x}_i$, the response y_i has some probability distribution with mean μ_i. We consider a general basis function regression structure in which we need to perform model selection to assess the most suitable class of basis functions and we will jointly perform regularization of the regression coefficients associated with the basis functions to remove bases (transformed covariates) which are not explanatory of the variation in the response in a given model structure.

In the classical linear model, we assume that the expected value μ_i is a linear function of k predictors that take values $\mathbf{x}_i = \left[x_{i1} \dots x_{ik}\right]$ for the i–th case, so that

$$\mu_i = \sum_{j=1}^{p} \beta_j \Phi_{i,j}\left(\mathbf{x}_i\right) \tag{4.19}$$

On the contrary, a generalized linear model consists of three components:

1. A *random component*, specifying the conditional distribution of the response variable, y_i (for the i^{th} of $n_{\mathbf{y}}$ independently sampled observations), given the value of the explanatory variables, $\mathbf{x}_i$. In the initial formulation of GLMs, response variables $y_1, \dots, y_{n_{\mathbf{y}}}$ are assumed to share the same distribution from the exponential family (See Appendix E);

2. A *systematic component* - the covariates $\mathbf{x}_i$ transformed through the basis function are combined linearly with the coefficients $\boldsymbol{\beta}$ to form the linear *pre-*

dictor

$$\vartheta_i = \sum_{j=1}^{p} \beta_j \Phi_{i,j}\left(\mathbf{x}_i\right) \tag{4.20}$$

3. A *link between the random and systematic component.* A smooth and invertible linearizing *link function* $g(\cdot)$, which transforms the expectation of the response variable, $\mu_i \equiv \mathbb{E}(y_i)$

$$g(\mu_i) = \vartheta_i = \sum_{j=1}^{p} \beta_j \Phi_{i,j}\left(\mathbf{x}_i\right) \tag{4.21}$$

Because the link function is invertible, we can also write

$$\mu_i = g^{-1}(\vartheta_i) = g^{-1}\left(\sum_{j=1}^{p} \beta_j \Phi_{i,j}\left(\mathbf{x}_i\right)\right) \tag{4.22}$$

and, thus, the GLM may be thought of as a linear model for a transformation of the expected response or as a nonlinear regression model for the response. In theory, the link function can be any monotonic and invertible function. The inverse link g^{-1} is also called the *mean function.* Commonly employed link functions and their inverses are shown in Table 4.2. Note that the *identity link* simply returns its argument unaltered, $\vartheta_i = g(i) = \mu_i$, and thus $\mu_i = g^{-1}(\vartheta_i) = \vartheta_i$.

Table 4.2: Some common link functions and their inverses

Link	$\vartheta_i = g(\mu_i)$	$\boldsymbol{\mu} = g^{-1}(\vartheta_i)$
Identity	μ_i	ϑ_i
Log	$\ln\mu_i$	ϑ_i^{-1}
Inverse	μ_i^{-1}	ϑ_i^{-1}
Inverse–square	μ_i^{-2}	$\vartheta_i^{-\frac{1}{2}}$
Logit	$\ln\frac{\boldsymbol{\mu}_i}{1-\mu_i}$	$\frac{1}{1+\exp^{-\vartheta_i}}$
Probit	$\Upsilon^{-1}(\mu_i)$	$\Upsilon(\vartheta_i)$
Log–Log	$-\ln\left[-\ln(\mu_i)\right]$	$\exp\left[-\exp(-\vartheta_i)\right]$
Complementary log–log	$\ln\left[-\ln(1-\mu_i)\right]$	$1-\exp\left[-\exp(\vartheta_i)\right]$

Note: μ_i is the expected valued of response; ϑ_i is the linear predictor; and $\Upsilon(\cdot)$ is the cumulative distribution function of the standard–normal distribution.

Having presented basic details of the GLM model structure, one needs to consider how to perform basic parameter estimation in a Bayesian framework. Let us remark that the non-Bayesian methods described in Section 4.2 with ℓ_1 or ℓ_2-norm regularization have been extended in order to be able to deal with GLMs. However, these methods are still suffering from the limitations compared to Bayesian solutions as discussed in Section 4.2.5 - so we refer the readers to [Friedman et al., 2010b] for more details.

4.4 Proposed Bayesian Solution

In regression, the general aim is to find the best value of the unknown coefficient $\boldsymbol{\beta}$, given a specific basis function as well as a distribution for the data. But as discussed in the previous sections, there are a huge number of ways to approximate the truth: different basis functions (Section 4.1.2) as well as distributions to model the response variables via the use of the general GLM (Section 4.3). Thus the obvious questions of interest in such problems are:

- What kind of approximating functions should we use to accurately model the relationship between the input and output variables ?
- How do we know when we have found the "best" approximation to the truth?

Unlike most of the existing approaches that are only interested in finding the best value of the coefficients $\boldsymbol{\beta}$, we design a Bayesian solution in order to have some answers to these questions by formulating the challenging choice of the basis functions as well as the distribution of the response variable within a model selection problem.

4.4.1 Bayesian Model Selection

We consider several families of non-nested regression models, each specified by the choice of basis function transforming the covariates as well as the distribution of the response variables. We utilize regularization to remove non-explanatory predictors and model selection for the most suitable choice of basis and data distribution. A solution has thus to decide between a set of K models, each of which representing a different basis function and distribution.

Model selection is performed in a Bayesian framework in which we aim to approximate $p(\mathcal{M}_k|\mathbf{y})$, for each of the models $k \in \{1, 2, \ldots, K\}$, which corresponds to the posterior model probability as discussed in Section 1.1.3. Using Bayes' theorem,

$$p(\mathcal{M}_k|\mathbf{y}) \propto p(\mathbf{y}|\mathcal{M}_k)p(\mathcal{M}_k) \tag{4.23}$$

where $p(\mathbf{y}|\mathcal{M}_k)$ denotes the marginal likelihood under model $\mathcal{M}_k$, also known as Bayes evidence, and $p(\mathcal{M}_k)$ corresponds to the model prior. Moreover, we are also interested in estimating the parameters that define each model through the parameter posterior $p(\boldsymbol{\theta}|\mathbf{y}, \mathcal{M}_k)$. For example, the parameter is defined as follows for two models

$$\begin{array}{ll} \text{Multiquadric basis function and Normal model} & \boldsymbol{\theta} = \left\{\boldsymbol{\beta}, \sigma_y^2, \gamma\right\} \\ \text{Inverse quadratic and Poisson model} & \boldsymbol{\theta} = \{\boldsymbol{\beta}, \gamma\} \end{array} \tag{4.24}$$

In order to achieve this inference task, the two distributions of interest, i.e., the conditional parameter posterior, $p(\boldsymbol{\theta}|\mathbf{y}, \mathcal{M}_k)$, and the associated marginal likelihood $p(\mathbf{y}|\mathcal{M}_k)$ are required but unfortunately they are intractable. Therefore, we resort

to an Importance-Sampling (IS) based Monte Carlo solution to jointly approximate these two quantities. As in the previous chapter dealing with target localization in WSNs, we propose to use an SMC sampler in order to have an accurate approximation of both quantities.

4.4.2 Proposed Bayesian algorithm

A strategy similar to the one derived for multiple source localization is employed to perform this joint model selection and parameter estimation in penalized regression model.

As a consequence, we propose to use the following procedure:

1. For each model $\mathcal{M}_k$, $k \in 1, \ldots, K$: approximate the conditional parameter posterior distribution $p(\boldsymbol{\theta}_k|\boldsymbol{y}, \mathcal{M}_k)$ as well as the marginal likelihood $p(\boldsymbol{y}|\mathcal{M}_k)$ using an SMC sampler algorithm.

2. Approximate the model posterior $p(\mathcal{M}_k|\boldsymbol{y})$, via the approximation of $p(\boldsymbol{y}|\mathcal{M}_k)$ and model prior $p(\mathcal{M}_k)$ - Eq. (3.13).

As summarized in Algo 4.1, we propose to use the proposed strategies described in Chapter 2 in order to improve:

- the variance of the estimator of the normalizing constant (i.e. the evidence of the model, $p(\boldsymbol{y}|\mathcal{M}_k)$) by using the adaptive cooling schedule of the SMC sampler,

- the variance of the final approximation of the posterior of the parameters by using the proposed recycling schemes.

The great advantage of using this proposed SMC sampler algorithm (compared to classical frequentist approaches described in Section 4.2) is to be able to obtain the posterior distribution of the predictive curve at some new predictor location, $\mathbf{x}_*$:

$$p(f(\mathbf{x}_*)|\mathbf{x}_*, \mathbf{y}, \mathcal{M}_k) \approx \sum_{i=1}^{N} \widetilde{W}_T^{(i)} \delta_{f_k^{(i)}} \left(df(\mathbf{x}_*)\right) \tag{4.25}$$

where

$$f_k^{(i)} = \sum_{j=1}^{p} \beta_j^{(i)} \Phi_{i,j}^k \left(\mathbf{x}_i\right) \tag{4.26}$$

and where elements $\Phi_{i,j}^k$ correspond to the basis function associated with the model $\mathcal{M}_k$. Let us note that a similar approximation of Eq. (4.25) is straightforwardly obtained with the proposed recycling schemes by using the adapted weights defined in Section 2.3. This posterior distribution of the predicted curve can thus be used to obtained the mean predicted curve but also some confidence interval which is of great interest in regression.

Algorithm 4.1 SMC Sampler Algorithm for Model $\mathcal{M}_k$ in Penalized regression models

1: Find the optimal parameter value γ^* of the parametric cooling schedule using the strategy described in Section 2.2.2
2: Initialize particle system from the prior
3: $\left\{\boldsymbol{\theta}_1^{(i)}\right\}_{i=1}^N \sim p(\boldsymbol{\theta}|\mathcal{M}_k)$ and set $\left\{\widetilde{W}_1^{(i)}\right\}_{i=1}^N = 1/N$
4: **for** $t = 2, \ldots, T$ **do**
5: Computation of the weights: for each $i = 1, \ldots, N$

$$W_t^{(i)} = \widetilde{W}_{t-1}^{(i)} \frac{\pi_t(\boldsymbol{\theta}_{t-1}^{(i)})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1}^{(i)})} = \widetilde{W}_{t-1}^{(i)} \frac{p(\boldsymbol{y}|\boldsymbol{\theta}_{t-1}, \mathcal{M}_k)^{\phi_t}}{p(\boldsymbol{y}|\boldsymbol{\theta}_{t-1}, \mathcal{M}_k)^{\phi_{t-1}}}$$

Normalization of the weights : $\widetilde{W}_t^{(i)} = W_t^{(i)} \left[\sum_{j=1}^N W_t^{(j)}\right]^{-1}$
6: Selection: if $ESS < N/2$ then Resample
7: Mutation: for each $i = 1, \ldots, N_p$: Sample $\theta_t^{(i)} \sim \mathcal{K}_t(\boldsymbol{\theta}_{t-1}^{(i)}; \cdot)$ where $\mathcal{K}_t(\cdot;\cdot)$ is a $\pi_t(\cdot)$ invariant Markov kernel using a series of Adaptive Metropolis within Gibbs algorithms for each of B sub-blocks of the state $\boldsymbol{\theta}$ - see details in Algo. 1.9
8: **end for**
9: Approximate the model evidence, $p(\boldsymbol{z}|\mathcal{M}_k)$, using Eq. (1.62)
10: Use the proposed recycling schemes described in Section 2.3 in order to combine all simulated particles from iteration 1 to T in order to obtain an approximation of $p(\boldsymbol{\theta}_k|\boldsymbol{y}, \mathcal{M}_k)$

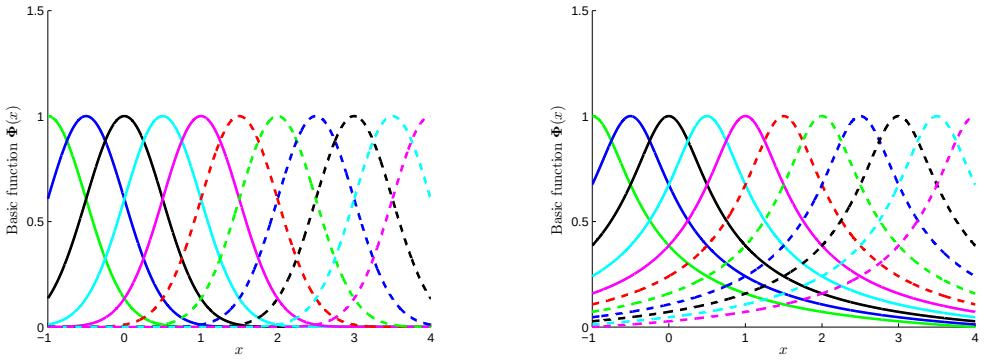
4.5 Numerical Simulation

In this section we present two different settings from GLM to investigate the performance of the proposed SMC sampler for the joint model selection and parameter estimation in a non-convex penalized regression model with the use of EP and $\mathcal{S}\alpha\mathcal{S}$ prior distributions: continuous data and count data. The performances of both the proposed adaptive cooling strategy and the recycling scheme will be also assessed through the two different cases. We have chosen $N_{\text{MCMC}} = 5$ and $B = 6$ for the adaptive MWG (summarized in Algo. 1.9) used in the SMC sampler as forward kernel. In all experiments, the input variable is considered as univariate such that $x \in [-1; 4]$ and 100 replications are performed. In addition, regarding the parameters of the priors, $p(\mathcal{M}_k) = 1/K$ has been chosen and the hyperparameter related to the dispersion of both EP and $\mathcal{S}\alpha\mathcal{S}$ priors defined respectively in Eqs (4.18) and (4.19) is such that $\gamma^2 \sim \mathcal{IG}(2, 1.3)$.

4.5.1 Case 1: Continuous data

In this first scenario, the performance of the proposed SMC sampler is analyzed for deciding between the 6 competing models described in Table 4.3. The true $n_{\mathbf{y}} = 40$ observations have been generated on some random locations under model $\mathcal{M}_1$ ($\sigma_y^2 = 0.5$) with regression coefficients set to zeros except $\beta_0 = 1$, $\beta_2 = \beta_9 = 5$, $\beta_4 = -5$, $\beta_6 = 3$ and $\beta_7 = -2$. For the basis functions ($\mathcal{M}_1$ to $\mathcal{M}_6$), 11 equally spaced centers c_j with the same scale parameter have been used $r_j = r = 0.5$. An

illustration of the basis functions is given in Fig. 4.4 for these specific settings.



(a) Gaussian **(b) Sigmoidal**

Figure 4.4: Illustration of the Gaussian basic function and Sigmoidal basic function with 11 centers.

Model	Likelihood function	Basis function
$\mathcal{M}_1$	Gaussian	Gaussian
$\mathcal{M}_2$	Gaussian	Inverse quadratic
$\mathcal{M}_3$	Gaussian	Sigmoidal
$\mathcal{M}_4$	Laplace	Gaussian
$\mathcal{M}_5$	Laplace	Inverse quadratic
$\mathcal{M}_6$	Laplace	Sigmoidal

Table 4.3: Description of the different models (basis functions and distributions) used in the continuous data regression scenario.

These 6 models are composed of 3 different basis functions as well as two different distribution for the error term of the response variable. The first is the Gaussian distribution defined as

$$y_i \sim \mathcal{N}(y_i|\sigma^2_{\mathbf{y}}, \mu_i) \qquad \text{with} \qquad \mu_i = \sum_{j=1}^{p} \beta_j \mathbf{\Phi}_k^j(\mathbf{x}_{i,j}) \quad . \tag{4.27}$$

and the second one is the Laplace distribution,

$$y_i \sim Laplace(y_i|\mu_i, b) \qquad \text{with} \qquad \mu_i = \sum_{j=1}^{p} \beta_j \mathbf{\Phi}_k^j(\mathbf{x}_{i,j}) \qquad \text{and } b = \sqrt{\frac{\sigma^2_{\mathbf{y}}}{2}}. \tag{4.28}$$

defined as

$$Laplace(y_i|\mu_i, b) = \frac{1}{2b} \exp\left(-\frac{|y_i - \mu_i|}{b}\right) \tag{4.29}$$

Both distributions use the identity link function shown in Table 4.2. As a consequence, the dimension of the parameter vector $\boldsymbol{\theta}$ to estimate is 14 in each model:

12 coefficients $\boldsymbol{\beta}$, $\sigma_{\mathbf{y}}$ and γ for each model. The prior for the likelihood dispersion is $\sigma_{\mathbf{y}}^2 \sim \mathcal{IG}(3, 0.5)$.

4.5.1.1 Performance for model selection

In this section, we study the performance of the proposed approach for the model selection, and more especially on its accuracy to correctly estimate the model evidence, $p(\boldsymbol{y}|\mathcal{M}_k)$. In Tables 4.4 and 4.5, we firstly compare the performance of the SMC sampler obtained by using the proposed adaptive cooling schedule compared to the one using a linear cooling schedule. The results in both tables clearly show that the proposed approach outperforms the linear cooling schedule by having a significant lower variance of the estimator of $p(\boldsymbol{y}|\mathcal{M}_1)$. Moreover, as expected from the theoretical analysis in Chapter 2, the variance decreases as either the number of particles N or the number of iterations T increases.

		Linear cooling schedule			Proposed Adaptive cooling schedule		
	$N = 50$	-90,8131	$\pm$	220,9249	-76,7771	$\pm$	14,5416
$T = 50$	$N = 200$	-81,7028	$\pm$	16,0938	-75,6507	$\pm$	1,0210
	$N = 50$	-81,0393	$\pm$	37,7063	-76,1351	$\pm$	6,3601
$T = 100$	$N = 200$	-77,8343	$\pm$	6,6065	-75,6117	$\pm$	0,4601
	$N = 50$	-78,3324	$\pm$	8,1660	-75,7429	$\pm$	1,8114
$T = 200$	$N = 200$	-76,0246	$\pm$	1,8509	-75,5677	$\pm$	0,4239

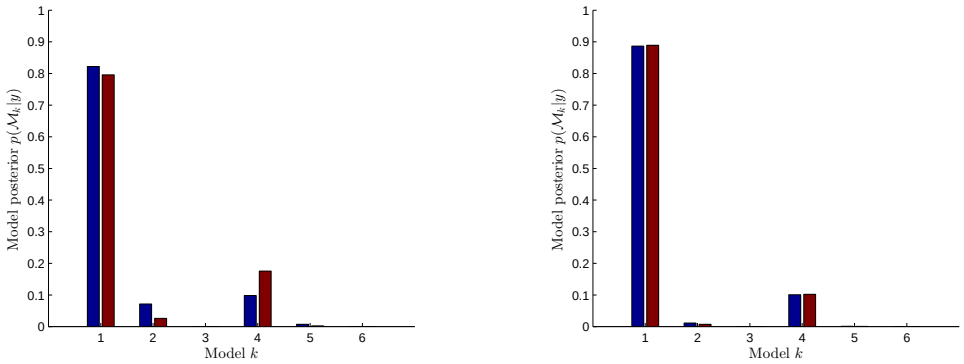
Table 4.4: The estimation of the marginal likelihood $\log p(\boldsymbol{y}|\mathcal{M}_1)$ (mean $\pm$ variance) in continuous data regression under model $\mathcal{M}_1$ [Prior: EP with $q = 0.5$].

		Linear cooling schedule			Proposed Adaptive cooling schedule		
	$N = 50$	-81,3636	$\pm$	16,3469	-78,2481	$\pm$	10,7360
$T = 50$	$N = 200$	-78,0741	$\pm$	5,4970	-76,4012	$\pm$	2,2658
	$N = 50$	-77,7423	$\pm$	5,6481	-76,6062	$\pm$	3,3293
$T = 100$	$N = 200$	-76,0461	$\pm$	1,6595	-75,9637	$\pm$	0,9066
	$N = 50$	-76,2887	$\pm$	1,7072	-76,0167	$\pm$	1,2064
$T = 200$	$N = 200$	-75,6649	$\pm$	0,3868	-75,6982	$\pm$	0,4953

Table 4.5: The estimation of the marginal likelihood $\log p(\boldsymbol{y}|\mathcal{M}_1)$ (mean $\pm$ variance) in continuous data regression under model $\mathcal{M}_1$ [Prior: $\mathcal{S}\alpha\mathcal{S}$ with $q = 1$].

Now, we study the model selection accuracy obtained by using the SMC sampler with the proposed adaptive cooling schedule. Fig. 4.5 shows a comparison of the average model posterior probability obtained with both EP and $\mathcal{S}\alpha\mathcal{S}$ priors and different values of the parameter q. The model used to generate the data is selected, thus validating the proposed procedure. From these results, we can also see that the algorithm is able to give the good model with high probability. There is some uncertainty with model $\mathcal{M}_4$ owing to the similarity of the two models: same basis functions but different distributions (Gaussian distribution versus Laplace

distribution). Both prior distributions give similar results.



(a) $q = 1$ **(b)** $q = 0.5$

Figure 4.5: Comparison between the approximated average model posteriors in continuous data regression (blue: EP , red:$\mathcal{S}\alpha\mathcal{S}$).

4.5.1.2 Performance for parameter estimation

We now investigate the performance of the SMC sampler to correctly estimate the unknown coefficients of regression and as a consequence to give some accurate prediction of the functional relationship between the input and output variables. Through this experiments, the proposed recycling schemes will be assessed.

Firstly, we study the stability of the estimator obtained using the recycling schemes proposed in this thesis. Table 4.6 and 4.7 show respectively the variance of the posterior mean estimator (i.e., trace $\{\mathbb{V}\text{ar}\left(\mathbb{E}_{\pi^N}\left[\boldsymbol{\beta}\right]\right)\}$) and the variance of the posterior mean of the predicted curve as:

$$\mathbb{V}\text{ar}_{\text{Approx. Curve}} = \frac{1}{L}\sum_{l=1}^{L}\mathbb{V}\text{ar}\left(\mathbb{E}_{\pi^N}\left[f(x_l)\right]\right) \tag{4.30}$$

where $\{x_l\}_{l=1}^{L}$ corresponds to the $L = 10000$ equally spaced grid points on the support of x (i.e., $[-1;4]$) and $\mathbb{E}_{\pi^N}\left[f(x_l)\right]$ is obtained from the posterior distribution approximation of predicted curve given in Eq. (4.25).

From these two tables, we can see that the number of iterations T of the SMC sampler does not really affect the stability of the estimator obtained without a recycling scheme since only the last iteration of the particles are used. On the contrary, the performance of the recycling schemes are improved both when the number of iterations and the number of particles increase. The proposed DeMix recycling scheme achieves the best performance by providing the lowest variance.

Table 4.8 and 4.9 compare the mean squared error obtained by using the different recycling schemes. As in the previous results, the DeMix recycling outperforms the other recycling approaches, especially the Naïve method and when no recycling is

		No Recycling	Naïve Recycling	ESS-based Recycling	Demix Recycling
$T = 50$	$N = 50$	7,188344	7,208748	7,120591	6,978712
	$N = 200$	3,385063	3,372960	3,212443	3,125740
$T = 100$	$N = 50$	5,383816	5,355592	4,579683	4,431033
	$N = 200$	2,295406	2,272483	1,902079	1,784975
$T = 200$	$N = 50$	6,593068	6,389972	4,188591	3,909149
	$N = 200$	2,146627	2,135784	1,572882	1,392894

Table 4.6: Variance of posterior mean estimator in continuous data regression under model $\mathcal{M}_1$ [Prior: EP with $q = 0.5$].

		No Recycling	Naïve Recycling	ESS-based Recycling	Demix Recycling
$T = 50$	$N = 50$	0,00649889	0,00649965	0,00561926	0,00535425
	$N = 200$	0,00237323	0,00235635	0,00206346	0,00197150
$T = 100$	$N = 50$	0,00538967	0,00518403	0,00358226	0,00332508
	$N = 200$	0,00170286	0,00165779	0,00114997	0,00104357
$T = 200$	$N = 50$	0,00635148	0,00568326	0,00265079	0,00232517
	$N = 200$	0,00156809	0,00144755	0,00079564	0,00066253

Table 4.7: Variance of approximated curve in continuous data regression under model $\mathcal{M}_1$ [Prior: EP with $q = 0.5$].

employed, in particular when the number of iterations T increases. Then in Table 4.10, we compare the impact on the mean squared error of the prior choice (EP vs $\mathcal{S}\alpha\mathcal{S}$). We can see from these results that a lower MSE is achieved when the $\mathcal{S}\alpha\mathcal{S}$ is used as prior for regularization.

		No Recycling	Naïve Recycling	ESS-based Recycling	Demix Recycling
$T = 50$	$N = 50$	25,93841016	25,93003297	25,95489229	25,86977971
	$N = 200$	18,84980134	18,84266031	19,06646358	19,01686333
$T = 200$	$N = 50$	23,15736387	23,04979798	21,79977296	21,66280214
	$N = 200$	17,86442937	17,85749678	17,77452217	17,68029717

Table 4.8: Average of the mean squared error between true regression coefficients and the estimated ones under the true model $\mathcal{M}_1$ in continuous data regression [Prior: $\mathcal{S}\alpha\mathcal{S}$ with $q = 0.5$].

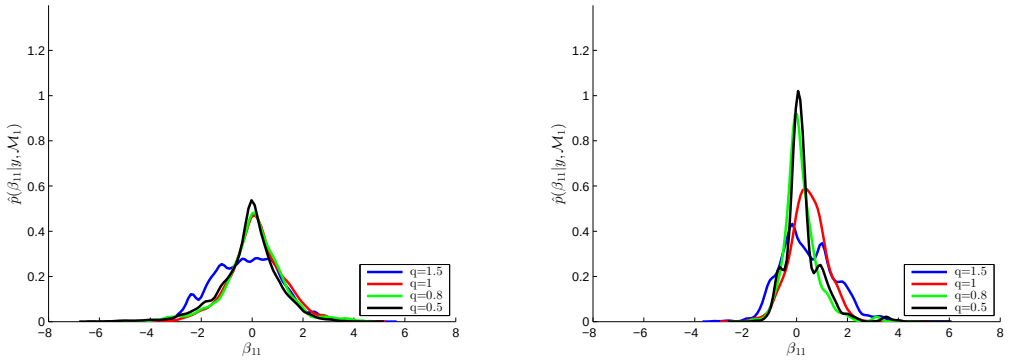
We can see in Fig. 4.6, the shrinkage effect on the marginal posterior distribution on one true zero coefficient. As expected from the discussion in Section 4.2.5, as q decreases, this marginal posterior distribution is shrinked around 0, a bit more rapidly for the same value of q with the use of $\mathcal{S}\alpha\mathcal{S}$.

		No Recycling	Naïve Recycling	ESS-based Recycling	Demix Recycling
$T = 50$	$N = 50$	20,45774418	20,44398663	20,07934861	20,14180226
	$N = 200$	14,99269283	14,97864074	14,70332751	14,78101050
$T = 200$	$N = 50$	22,33612119	22,17224515	18,16607751	17,80322402
	$N = 200$	16,05319454	16,02756772	14,85848058	14,58962115

Table 4.9: Average of the mean squared error between true regression coefficients and the estimated ones under the true model $\mathcal{M}_1$ in continuous data regression [Prior: EP with $q = 0.5$].

		EP	Stable
$T = 50$	$N = 50$	13,11983129	16,06304859
	$N = 200$	10,54469181	13,01181589
$T = 100$	$N = 50$	13,72176273	13,79308301
	$N = 200$	10,76843527	10,10960093
$T = 200$	$N = 50$	12,22612934	11,61485744
	$N = 200$	12,2562075	9,245580013

Table 4.10: Median of the mean squared error between true regression coefficients and the estimated ones under the true model $\mathcal{M}_1$ in continuous data regression.



(a) EP Prior **(b) $\mathcal{S}\alpha\mathcal{S}$ Prior**

Figure 4.6: Comparision of the shrinkage results obtained with the two different priors as q decrease by using Demix recycling scheme in continuous data regression.

Finally we are interested in the performance of predicted curve by using SMC Sampler under the proposed recycling scheme. In order to analyze the predicted curve's performance, the commonly used criterion is Mean Squared Error Prediction (MSEP). The MSEP measures the expected squared distance between what our predictor provides for a specific value and the true value:

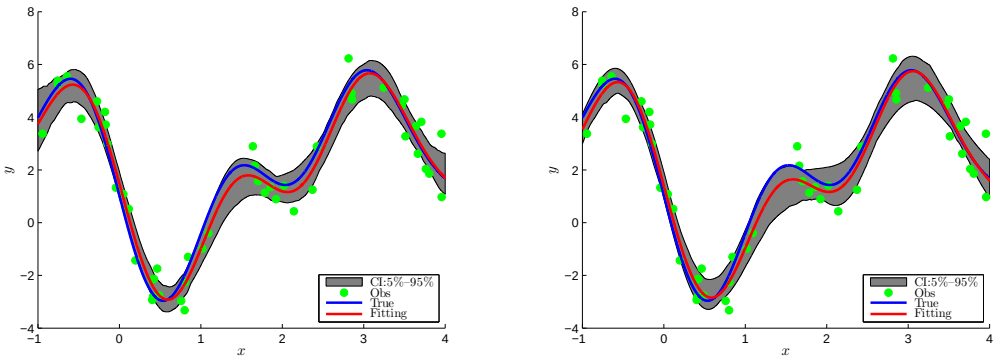
$$MSEP = \sum_{l=1}^{L} \mathbb{E}\left[\left(f(x_l) - \mathbb{E}_{\pi^N}\left[f(x_l)\right]\right)^2\right] \tag{4.31}$$

We conclude from Table 4.11 that the Demix recycling scheme give us a slightly lower MSEP than the other schemes.

		No Recycling	Naïve Recycling	ESS-based Recycling	Demix Recycling
$T = 50$	$N = 50$	0,10599208	0,10601201	0,10556323	0,10505292
	$N = 200$	0,09914624	0,09916341	0,09863060	0,09864062
$T = 100$	$N = 50$	0,10743767	0,10733964	0,10369331	0,10324023
	$N = 200$	0,10112909	0,10107449	0,09988531	0,09973361
$T = 200$	$N = 50$	0,10516423	0,10498927	0,10060531	0,10003071
	$N = 200$	0,09994505	0,09992371	0,09903866	0,09874603

Table 4.11: Mean squared error prediction in continuous data regression under model $\mathcal{M}_1$ [Prior: EP with $q = 0.5$].

As shown in Fig. 4.7, the SMC sampler is able to efficiently predict the unknown function even if only few observations are available. As opposed to frequentist LASSO, the proposed approach can give a confidence interval on the predicted curve which is of great interest in many applications. The true curve used to generate the observations is always within the confidence interval obtained from the proposed SMC sampler.



(a) EP Prior - $q = 0.8$ **(b)** $\mathcal{S}\alpha\mathcal{S}$ **Prior -** $q = 0.8$

Figure 4.7: Regression with continuous data [Prior: EP and $\mathcal{S}\alpha\mathcal{S}$ - $q = 0.8$] by using Demix recycling scheme: true function in blue - observed responses in green circles - posterior mean from SMC under model $\mathcal{M}_1$ in red and confidence region in gray 5% to 95% percentiles.

4.5.2 Case 2: Count Data

In this second scenario, we consider the regression problem which consists in finding the relationship between continuous input variables and count data as response variables. The 6 competing models used in this simulation scenario are described in Table 4.12 . The true $n_{\mathbf{y}} = 100$ observations have been generated with some

random locations under model $\mathcal{M}_1$ with regression coefficients set to zeros except $\beta_0 = 1$, $\beta_2 = 1.5$, $\beta_4 = -2$, $\beta_6 = 1$, $\beta_7 = -2$ and $\beta_9 = 1.2$. For the basis functions, 11 equally spaced centers c_j with the same scale parameters $r_j = r = 0.5$ have been used.

Model	Likelihood function	Basis function
$\mathcal{M}_1$	Poisson	Gaussian
$\mathcal{M}_2$	Poisson	Inverse quadratic
$\mathcal{M}_3$	Poisson	Sigmoidal
$\mathcal{M}_4$	Negative binomial	Gaussian
$\mathcal{M}_5$	Negative binomial	Inverse quadratic
$\mathcal{M}_6$	Negative binomial	Sigmoidal

Table 4.12: Description of the different models (basis functions and distributions) used in the count data regression scenario.

The 6 competing models are composed of 3 different basis functions as well as two different distribution for the error term of the response variable. The first is the Poisson distribution defined as

$$y_i \sim \mathcal{P}(y_i|\mu_i) \qquad \text{with} \qquad \mu_i = \exp\left(\sum_{j=1}^{p} \beta_j \mathbf{\Phi}_k^j(\mathbf{x}_{i,j})\right) \quad . \tag{4.32}$$

and the second one is the Negative Binomial distribution,

$$y_i \sim NB(y_i|\sigma_{\mathbf{y}}, \mu_i) \qquad \text{with} \qquad \mu_i = \exp\left(\sum_{j=1}^{p} \beta_j \mathbf{\Phi}_k^j(\mathbf{x}_{i,j})\right) \tag{4.33}$$

defined as

$$\mathbb{P}(\mathbf{Y}_i = y_i) = \left(\frac{\sigma_{\mathbf{y}}}{\sigma_{\mathbf{y}} + \mu_i}\right)^{\sigma_{\mathbf{y}}} \frac{\Gamma(\sigma_{\mathbf{y}} + y_i)}{y_i\,\Gamma(\sigma_{\mathbf{y}})} \left(\frac{\mu_i}{\sigma_{\mathbf{y}} + \mu_i}\right)^{y_i} \tag{4.34}$$

where $\sigma_{\mathbf{y}}$ corresponds to the dispersion parameter of this distribution. Let us note that the Poisson distribution is a limiting case of the negative binomial distribution, i.e.

$$\lim_{\sigma_{\mathbf{y}} \to +\infty} NB(y_i|\sigma_{\mathbf{y}}, \mu_i) = \mathcal{P}(y_i|\mu_i) \tag{4.35}$$

Both distributions use the log link function shown in Table 4.2. As a consequence, the dimension of the parameter vector $\boldsymbol{\theta}$ to estimate is 14 for :

- models $\mathcal{M}_1$ to $\mathcal{M}_3$: 13 coefficients $\boldsymbol{\beta}$ and γ
- models $\mathcal{M}_4$ to $\mathcal{M}_6$: 14 coefficients $\boldsymbol{\beta}$, γ $\sigma_{\mathbf{y}}$

The prior for the likelihood dispersion is $\sigma_{\mathbf{y}}^2 \sim \mathcal{IG}(3, 0.5)$.

4.5.2.1 Performance for model selection

Tables 4.13, 4.14 and 4.15 show the variance of the SMC sampler estimate of the model evidence, $p(\boldsymbol{y}|\mathcal{M}_1)$, obtained by using the proposed adaptive cooling schedule and the linear cooling schedule. As in case 1 with continuous data, the variance obtained by using the proposed adaptive cooling schedule is significantly lower than the one obtained with the linear cooling schedule.

		Linear cooling schedule			Proposed Adaptive cooling schedule		
	$N = 50$	-221,0070	±	124,3479	-203,3146	±	0,8215
$T = 50$	$N = 200$	-211,8227	±	34,1198	-203,2687	±	0,2325
	$N = 50$	-211,4072	±	28,2070	-203,1100	±	0,4598
$T = 100$	$N = 200$	-206,1121	±	10,0980	-203,1244	±	0,0698
	$N = 50$	-206,3567	±	13,6623	-202,9167	±	0,1627
$T = 200$	$N = 200$	-204,1015	±	3,8083	-203,0268	±	0,0530

Table 4.13: The estimation of the marginal likelihood log $p(\boldsymbol{y}|\mathcal{M}_1)$ (mean ± variance) in count data regression under model $\mathcal{M}_1$ [Prior: EP with $q = 0.5$].

		Linear cooling schedule			Proposed Adaptive cooling schedule		
	$N = 50$	-215,4884	±	176,6493	-202,2807	±	1,1212
$T = 50$	$N = 200$	-206,8129	±	17,9152	-202,0950	±	0,3469
	$N = 50$	-208,9877	±	55,5816	-202,1650	±	0,8234
$T = 100$	$N = 200$	-204,1774	±	6,5144	-201,9107	±	0,2360
	$N = 50$	-204,4954	±	8,9395	-202,1333	±	0,6031
$T = 200$	$N = 200$	-202,8712	±	1,4096	-201,8962	±	0,1376

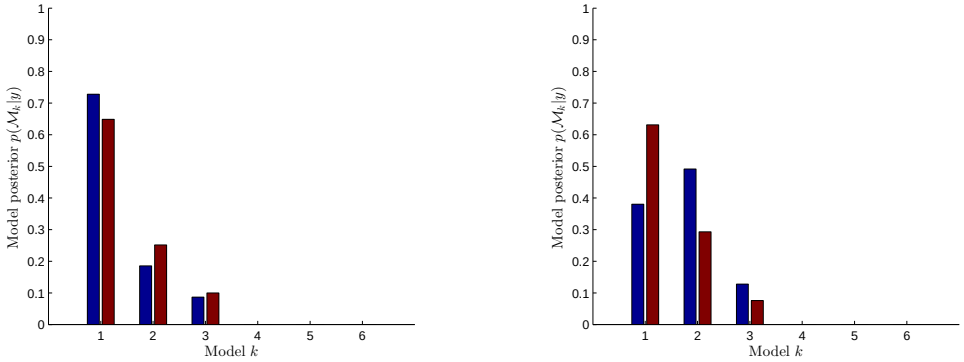
Table 4.14: The estimation of the marginal likelihood log $p(\boldsymbol{y}|\mathcal{M}_1)$ (mean ± variance) in count data regression under model $\mathcal{M}_1$ [Prior: $\mathcal{S}\alpha\mathcal{S}$ with $q = 0.5$].

		Linear cooling schedule			Proposed Adaptive cooling schedule		
	$N = 50$	-221,0070	±	124,3479	-203,3146	±	0,8215
$T = 50$	$N = 200$	-211,8227	±	34,1198	-203,2687	±	0,2325
	$N = 50$	-211,4072	±	28,2070	-203,1100	±	0,4598
$T = 100$	$N = 200$	-206,1121	±	10,0980	-203,1244	±	0,0698
	$N = 50$	-206,3567	±	13,6623	-202,9167	±	0,1627
$T = 200$	$N = 200$	-204,1015	±	3,8083	-203,0268	±	0,0530

Table 4.15: The estimation of the marginal likelihood log $p(\boldsymbol{y}|\mathcal{M}_1)$ (mean ± variance) in count data regression under model $\mathcal{M}_1$ [Prior: $\mathcal{S}\alpha\mathcal{S}$ with $q = 1$].

From Fig. 4.8, we analyze the average model posterior probability obtained by the algorithm using both regularization priors. With the $\mathcal{S}\alpha\mathcal{S}$ prior, the model used to generate has always the largest posterior probability (in average) in both case $q = 1$ and $q = 0.5$ whereas this is not the case with EP by choosing $q = 0.5$. In this

case, there are 31% of the total simulation runs in which we choose the true model and 69% of the total simulation runs the $\mathcal{M}_2$.



(a) $q = 1$ **(b)** $q = 0.5$

Figure 4.8: Comparision of the approximation of the model posterior (blue: EP , red:$\mathcal{S}\alpha\mathcal{S}$).

Even if the Poisson distribution is a limiting case (as $r \to \infty$) of the negative binomial distribution, the model posterior probability for $\mathcal{M}_4$ is closed to 0 which could be explained by the fact that the hyperparameters set for the prior of r in the inverse gamma lead to a very low a priori probability to have a large value.

4.5.2.2 Performance for parameter estimation

We now investigate the performance of the SMC sampler to correctly estimate the unknown coefficients of regression and as a consequence give some accurate prediction of the functional relationship between the input and output variables. Through this experiments, the proposed recycling schemes will be assessed.

As in the previous scenario, Tables 4.16 and 4.17 clearly demonstrate that our proposed scheme (ESS and DeMix) outperforms the two other schemes that were used in these simulations in terms of the stability of the posterior mean estimator. Again, the DeMix recycling scheme achieved a slightly better result than the ESS-based strategy in terms of variance.

Tables 4.18-4.20 and Table 4.21 present the ability of the proposed method to give an accurate estimate of the regression coefficients as well as an accurate prediction of the curve, respectively. We still can see from Table 4.18 an improvement gain with the proposed DeMix recycling scheme. In addition, a slightly lower MSE is obtained by using the symmetric stable distribution as regularization prior .

As in the first scenario, Fig. 4.9 shows that the marginal posterior distribution is shrinked around 0 when q decreases and a bit more rapidly with the use of $\mathcal{S}\alpha\mathcal{S}$ for the same value of q.

Fig 4.10 shows the resulting mean predicted curves (and associated confidence intervals) obtained by using the proposed SMC sampler under the true model. As

		No Recycling	Naïve Recycling	ESS-based Recycling	DeMix Recycling
	$N = 50$	1,938904	1,938904	1,651603	1,582488
$T = 50$	$N = 200$	0,410165	0,410165	0,370577	0,350994
	$N = 50$	1,569445	1,569433	1,214406	1,143985
$T = 100$	$N = 200$	0,552467	0,552448	0,447887	0,402980
	$N = 50$	2,247377	2,246568	1,638242	1,517514
$T = 200$	$N = 200$	0,722467	0,722348	0,568624	0,501554

Table 4.16: Variance of posterior mean estimator in count data regression under model $\mathcal{M}_1$ [Prior: EP with $q = 0.5$].

		No Recycling	Naïve Recycling	ESS-based Recycling	DeMix Recycling
	$N = 50$	0,00318883	0,00318883	0,00216641	0,00193355
$T = 50$	$N = 200$	0,00069063	0,00069063	0,00050476	0,00046064
	$N = 50$	0,00268185	0,00268197	0,00148822	0,00130908
$T = 100$	$N = 200$	0,00079279	0,00079267	0,00048881	0,00040874
	$N = 50$	0,00308443	0,00307905	0,00170143	0,00149617
$T = 200$	$N = 200$	0,00094644	0,00094534	0,00051125	0,00041064

Table 4.17: Variance of approximated curve in count data regression under model $\mathcal{M}_1$ [Prior: EP with $q = 0.5$].

		No Recycling	Naïve Recycling	ESS-based Recycling	Demix Recycling
	$N = 50$	5,29616600	5,29616598	5,23182229	5,22741009
$T = 50$	$N = 200$	4,286257907	4,286257906	4,263276843	4,263814081
	$N = 50$	4,937666355	4,937408961	4,780059818	4,72289285
$T = 100$	$N = 200$	3,968923666	3,968915643	4,006630363	3,984618102
	$N = 50$	5,328693098	5,325889885	5,001144984	4,929574759
$T = 200$	$N = 200$	3,834655466	3,832982929	3,691328281	3,658065713

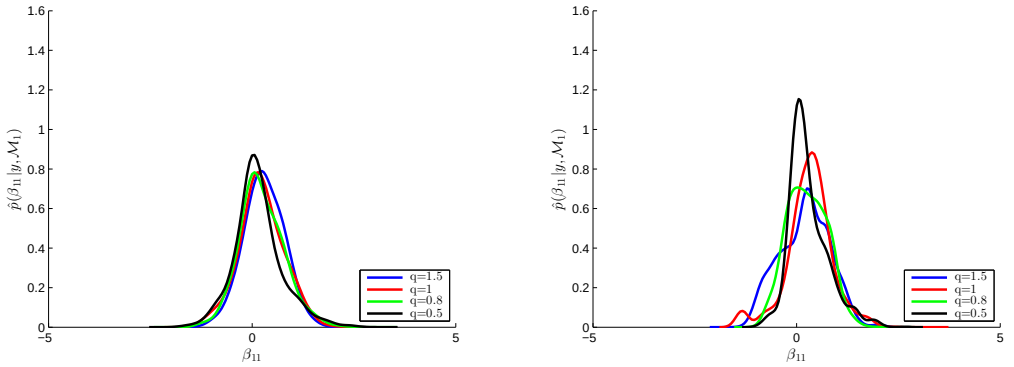
Table 4.18: Average of the mean squared error between true regression coefficients and the estimated ones under the true model in count data regression [Prior: $\mathcal{S}\alpha\mathcal{S}$ with $q = 1$].

		No Recycling	Naïve Recycling	ESS-based Recycling	Demix Recycling
	$N = 50$	7,488977527	7,488509712	7,071954105	6,98644176
$T = 50$	$N = 200$	5,131629407	5,131549613	4,988871483	4,979872341
	$N = 50$	6,518910284	6,518913368	5,922980086	5,875014888
$T = 100$	$N = 200$	5,322251966	5,322273791	5,167783205	5,152091357
	$N = 50$	6,250372415	6,250133006	5,484534075	5,453879921
$T = 200$	$N = 200$	5,184655426	5,183756676	4,90186597	4,850586809

Table 4.19: Average of the mean squared error between true regression coefficients and the estimated ones under the true model in count data regression [Prior: EP with $q = 0.5$].

		EP	Stable
	$N = 50$	4,283522621	4,769814942
$T = 50$	$N = 200$	3,499591349	3,399543736
	$N = 50$	4,774018074	3,941452355
$T = 100$	$N = 200$	3,561179957	3,189292366
	$N = 50$	4,466539824	3,859727923
$T = 200$	$N = 200$	3,916402394	3,255971671

Table 4.20: Median of mean squared error between true regression coefficients and the estimated ones under the true model $\mathcal{M}_1$ in count data regression.



(a) EP Prior **(b) $\mathcal{S}\alpha\mathcal{S}$ Prior**

Figure 4.9: Comparison of the shrinkage results obtained with the two different priors as q decrease by using Demix recycling scheme.

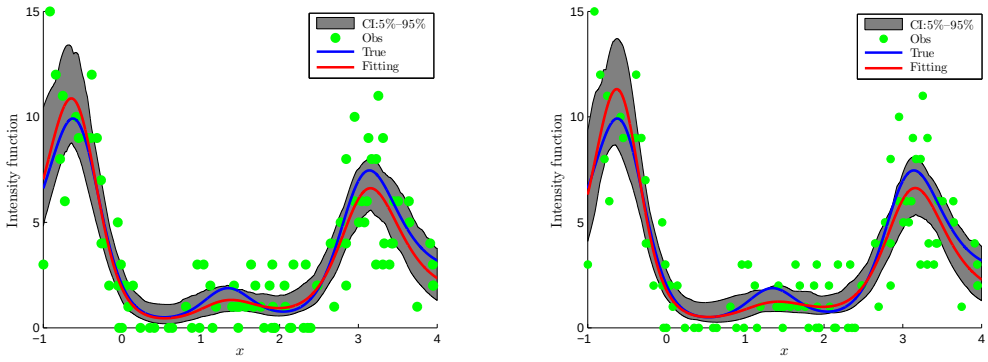
		No Recycling	Naïve Recycling	ESS-based Recycling	Demix Recycling
	$N = 50$	0,048757068	0,048758056	0,048501815	0,048239388
$T = 50$	$N = 200$	0,04666356	0,046665976	0,046438218	0,046361146
	$N = 50$	0,04863974	0,048632708	0,046806177	0,046813544
$T = 200$	$N = 200$	0,046908003	0,046904077	0,046374069	0,04628593

Table 4.21: Mean squared error prediction in count data regression under model $\mathcal{M}_1$ [Prior: EP with $q = 0.5$].

in the previous case, the true curve is always within the confidence region which clearly shows the ability of our algorithm to give a good prediction of the functional relationship between the input and output variables.

4.6 Conclusion

In this chapter, we have proposed an efficient algorithm for joint model selection and parameter estimation in penalized regression models based on SMC samplers. Firstly, we have proposed a new class of priors based on α-Stable distribution that



(a) EP Prior - $q = 0.8$ **(b)** $\mathcal{S}\alpha\mathcal{S}$ **Prior -** $q = 0.8$

Figure 4.10: Regression with count data [Prior: EP and $\mathcal{S}\alpha\mathcal{S}$ - $q = 0.8$] by using Demix recycling scheme: true function in blue - observed responses in green circles - posterior mean from SMC under model $\mathcal{M}_1$ in red and confidence region in gray 5% to 95% percentiles.

represents an alternative to exponential power distribution commonly used for ℓ_q- regularization in a Bayesian setting. Moreover, the proposed strategy based on parallel SMC samplers for dealing with the model selection allows us to obtain automatically some probabilistic criterion to decide the "best" basis function in order to approximate the relationship between the input and response variables and to choose the best probability distribution that fits the observation noise. Numerical simulations, with both continuous and count data, demonstrate the ability of the proposed algorithm to estimate the different quantities of interest as well to predict the response at some unobserved predictor location. These results also show that significant improvement can be obtained by using the proposed improvement strategies for the SMC sampler developed in Chapter 2.

Conclusion and future work

Conclusion

In many problems encountered in signal processing, it is possible to accurately describe the underlying statistical model using probability distributions. Statistical inference can then theoretically be performed based on the relevant posterior distribution in a Bayesian framework. However, most problems encountered in applied research require non-Gaussian and/or nonlinear models to correctly account for the observed data. In these cases, it is typically impossible to obtain the required statistical estimates of interest (conditional expectation, marginal likelihood, confidence interval, etc.) in closed form as it requires generally integration of complex multidimensional functions. A standard approach consists of making model simplifications or crude analytic approximations to obtain algorithms that can be easily implemented. With the recent availability of high-powered computers, numerical simulation-based approaches can now be considered and the full complexity of real problems can be addressed.

Monte Carlo algorithms are remarkably flexible and extremely powerful to solve such inference problems. The basic idea is to draw a large number of samples distributed according to some probability distribution(s) of interest so as to obtain consistent simulation-based estimates. Nevertheless in complex models with high-dimensional and/or multimodal posterior distribution, standard Monte-Carlo techniques like importance sampling or Markov Chain Monte-Carlo algorithms could lead to poor performance.

In this thesis, we have focused on more robust and efficient Monte Carlo algorithms that have been established in order to efficiently explore such high dimensional and multimodal spaces. In particular, we study a technique, named Sequential Monte-Carlo Sampler, that has been recently proposed in the statistic literature [Del Moral et al., 2006] as a promising alternative to standard MCMC methods. Although this approach presents many advantages over traditional MCMC methods as discussed through this thesis, the potential of this emergent technique is however largely underexploited in signal processing. In this thesis, we therefore focus our study on this technique by aiming at proposing some novel strategies that will improve the efficiency and facilitate practical implementation of the SMC sampler. Then, we apply the SMC sampler integrating our proposed improvement strategies to two challenging practical problems: Multiple source localization in wireless sensor networks and Bayesian penalized regression.

In Chapter 1, after reviewing the objectives of Bayesian inference, we describe standard Monte-Carlo algorithms. Then, the SMC sampler is described in more details with a presentation of the general principle as well as some discussion regarding its advantages over traditional Monte-Carlo algorithms as well as the different choices of parameters required for efficient practical implementation of the

algorithm.

In Chapter 2, we propose some novel strategies in order to improve the efficiency of the SMC sampler. Firstly, we present the general convergence results of the SMC sampler given in the original paper [Del Moral et al., 2006]. Then, we derive convergence results of the SMC sampler for some specific choice of the backward kernel used generally in practice as well as under a perfectly mixing forward kernel. These convergence results are derived for three variants of the SMC sampler: no resampling, resampling after the sampling and resampling before the sampling. The obtained results facilitate the analysis of the SMC sampler and in particular highlight the impact of the choice of the sequence of target distributions on the algorithm performance. From these results, we show that it is preferable to do the resampling before the sampling step. Then, by using these convergence results, we propose an automatic and adaptive strategy that selects the sequence of distributions (with tempered likelihood) that approximately minimizes the asymptotic variance of the estimator of the normalizing constant. Finally, we present another original contribution in order to improve the global efficiency of the SMC sampler. The idea developed in this thesis is to propose some correction mechanisms that allow the use of the particles generated through all the iterations of the algorithm (instead of only the particles from the last iteration) in order to improve the accuracy of the empirical approximation of the target distribution. The proposed strategies are assessed through numerical simulations. We demonstrate empirically that significant improvement can be obtained by using the different proposed approaches.

Then, in the two last chapters, the SMC sampler with our proposed strategies is applied to two challenging practical problems. In Chapter 3, we address the problem of localizing an unknown number of energy emitting source in wireless sensor networks with quantized data. We provide a generalization of recent existing works that deal only with a single source. We firstly derive the posterior Cramér-Rao bound for the estimation of the characteristics of these multiple energy emitting sources. Then, we propose a Bayesian solution based on SMC samplers for the joint estimation of the unknown number of sources as well as their associated parameters. Numerical simulations clearly illustrate the ability of the proposed SMC sampler to perform an accurate joint estimation in this challenging problem due to the high multimodality of the posterior distribution. Moreover, results show that the proposed adaptive cooling schedule as well as the proposed recycling schemes for SMC sampler allow to improve quite significantly the accuracy of the estimators that are required for the model selection (i.e., for estimating the number of sources) and the estimation of the source characteristics, respectively.

Finally, in Chapter 4, the problem of penalized regression in generalized linear models is tackled. After describing the regression modeling, we propose a new class of priors based on α-stable distributions in order to introduce a non-convex penalization for the regularization of the unknown regression coefficients. Thus, we propose a Bayesian method based on the use of parallel SMC samplers that is able to solve jointly model selection and parameter estimation of high interest in regression. The model selection procedure allows us to obtain, automatically, some

probabilistic criterion to decide the "best" basis function in order to approximate the relationship between the input and response variables and to choose the best probability distribution that fits the observation noise. Numerical simulations, with both continuous and count data, demonstrate the ability of the proposed algorithm to estimate the different quantities of interest as well to predict the response at some unobserved predictor location. These results also show that significant improvement can be obtained by using the proposed improvement strategies for the SMC sampler developed in Chapter 2.

Future Work

Some directions for future research are now discussed. In Chapter 2, we propose an interesting approach to combine all the particles that have been generated through the iterations of the SMC samplers in order to improve the accuracy of the empirical approximation of the target distribution of interest. The proposed strategy is performed once the algorithm has completely finished all its iterations. An interesting idea would be to utilize such strategies within the SMC sampler. For example, at iteration t, the idea would be firstly to recycle all the past simulated particles up to this iteration and to use them in order to approximate the target distribution π_t. Another interesting work that could be studied regarding SMC samplers is related to the resampling step and more specifically on when and how to resample. For example, the time parameter (i.e. iteration of the algorithm) is not always an appropriate way to decide when to resample since certain samples may have low importance weights, but as time t approaches these samples have higher weights.

In Chapter 3, the posterior Cramér-Rao bound for multiple emitting sources is derived. From a practical point of view, it could be interesting to use this bound in order to find the optimal value of the quantization thresholds of the sensors as well as their optimal location in order to obtain a better estimation performance. Additionally, this lower bound can also be used within the proposed SMC sampler in order to sequentially select the sensors that have to send their measurements to the fusion center for the localization of the multiple source. With such schemes, selecting sequentially only the most informative sensors could significantly reduce the communication requirements and energy consumption.

Finally, in Chapter 4, we design a Bayesian solution based on SMC samplers for penalized regression analysis. In this study we consider the regularization coefficient, γ, is the same for all the regression coefficients $\boldsymbol{\beta}$. It would be interesting to extend this work by designing a Bayesian solution that considers different regularization coefficients shared by group of regression coefficients, thus yielding to solutions that are sparse at both the group and individual feature levels, as in the group LASSO or sparse group LASSO [Friedman et al., 2010a].

Conclusion en Français

Dans de nombreux problèmes rencontrés en traitement du signal, il est possible de décrire avec précision le modèle statistique sous-jacent à l'aide de distributions de probabilité. L'inférence statistique peut alors être effectuée sur la base de la distribution pertinente, la loi a posteriori dans le cadre Bayésien. Cependant, la plupart des problèmes rencontrés en pratique nécessitent des modèles non-gaussiens et/ou non-linéaires afin de prendre en considération le système considéré. Dans de tels cas, il est généralement impossible d'obtenir les estimations statistiques d'intérêt (espérance conditionnelle, probabilité marginale, intervalle de confiance, etc) sous forme analytique puisque cela nécessite d'être capable de résoudre des intégrales de fonctions multidimensionnelles complexes. Une approche standard consiste à faire des simplifications du modèle afin d'obtenir des algorithmes pouvant être facilement mis en œuvre. Avec la montée en puissance des capacités de calcul, les approches fondées sur la simulation stochastique, peuvent plus que jamais être employées afin de résoudre ces problèmes d'inférence en considérant la réelle complexité du système sans avoir besoin de quelconques approximations.

Les algorithmes de type Monte-Carlo sont remarquablement flexibles et extrêmement puissants pour la résolution de ces problèmes d'inférence. L'idée de base est de générer un grand nombre d'échantillons distribués selon une distributionde probabilité d'intérêt de manière à obtenir des estimateurs empiriques basés sur ces échantillons. Néanmoins, dans des modèles complexes avec une distribution de grande dimension et/ou multimodale, les techniques classiques de Monte-Carlo comme l'échantillonnage d'importance ou encore les méthodes de Monte-Carlo par chaînes de Markov conduisent généralement vers des résultats non satisfaisants.

Dans cette thèse, nous nous sommes concentrés sur une technique robuste et efficace dans l'exploration d'espaces multimodaux et/ou de dimension élevée. En particulier, nous avons étudié une technique, appelée l'échantillonneur séquentiel Monte-Carlo (SMC), qui a été récemment proposée dans la littérature statistique [Del Moral et al., 2006] comme une alternative prometteuse aux méthodes traditionnelles MCMC. Bien que cette approche présente plusieurs avantages sur les méthodes MCMC comme discuté dans ce manuscrit de thèse, le potentiel de cette technique émergente est cependant largement sous-exploité dans le domaine du traitement du signal. Dans cette thèse, notre étude s'est donc concentrée sur cette technique en visant à proposer des stratégies novatrices permettant d'améliorer l'efficacité et de faciliter la mise en œuvre pratique de cet échantillonneur SMC. Ensuite, nous avons appliqué l'échantillonneur SMC intégrant les différentes stratégies d'amélioration proposées à deux problèmes pratiques difficiles: la localisation de multiple sources dans les réseaux de capteurs sans fil et la régression pénalisée.

Dans le chapitre 1, après avoir introduit les objectifs de l'inférence Bayésienne, nous avons décrit les principaux algorithmes de type Monte-Carlo. Nous nous sommes, ensuite, focalisés sur une description plus détaillée de l'échantillonneur

SMC en présentant tout d'abord son principe général, puis ses avantages par rapport aux algorithmes plus traditionnels de type Monte-Carlo. Nous avons enfin discuté des différents choix de paramètres nécessaires à sa mise en œuvre pratique.

Dans le chapitre 2, de nouvelles stratégies dans le but d'améliorer les performances de l'échantillonneur SMC et de faciliter son implémentation ont été proposées. Tout d'abord, nous avons présenté les résultats de convergence des estimateurs issus de cet algorithme développé dans [Del Moral et al., 2006]. Ensuite, nous avons obtenu les résultats de convergence pour certains choix spécifiques concernant les noyaux utilisés pour trois différentes variantes de l'algorithme: pas de rééchantillonnage, rééchantillonnage après l'échantillonnage et rééchantillonnage avant l'échantillonnage. Ces résultats nous ont permis de faciliter l'analyse de cet algorithme et surtout de mettre en évidence l'impact du choix de la séquence de distributions cibles intermédiaires sur les caractéristiques statistiques des estimateurs. A partir de ces résultats, nous avons montré notamment qu'il est préférable de faire le rééchantillonnage avant l'étape d'échantillonnage. Puis, en utilisant ces résultats de convergence, nous avons proposé une nouvelle stratégie permettant d'obtenir automatiquement la séquence de distributions intermédiaires qui minimise, sous certaines approximations, la variance asymptotique de l'estimateur de la constante de normalisation de la loi a posteriori. Finalement, nous avons présenté une autre contribution originale dans le but d'améliorer l'efficacité globale de l'échantillonneur SMC. L'idée développée dans cette thèse a été de proposer des mécanismes de correction qui permettent de construire des estimateurs plus efficaces de la distribution cible en prenant en compte les particules générées lors de l'ensemble des itérations de l'algorithme (au lieu de seulement les particules issues de la dernière itération). Les stratégies proposées ont été évaluées à travers de nombreuses simulations numériques qui ont permis de démontrer empiriquement une amélioration significative.

Dans les deux derniers chapitres, l'échantillonneur SMC incluant les stratégies proposées pour l'améliorer est appliqué à deux problèmes pratiques particulièrement difficiles. Dans le chapitre 3, nous avons abordé le problème de la localisation d'un nombre inconnu de sources émettrices d'énergie dans les réseaux de capteurs sans fil par l'observation de données quantifiées relatives au niveau de puissance reçu. Nous avons ainsi proposé une généralisation des travaux existants récents qui traitent seulement du problème mono-source. Pour résoudre ce problème, nous avons ainsi proposé une solution utilisant l'échantillonneur SMC qui permet d'estimer conjointement le nombre de sources présentes ainsi que leurs caractéristiques (position et puissance émise). Nous avons également dériver la borne de Cramér-Rao a posteriori représentant la limite inférieure des performances d'un estimateur Bayésien sur la position et la puissance émise de chaque source, au vu des observations obtenues au centre de fusion. Les simulations numériques ont montré clairement la capacité de l'échantillonneur SMC proposé à fournir une estimation précise dans ce problème difficile en raison de la forte multimodalité de la distribution a posteriori. En outre, les résultats ont démontré empiriquement que la stratégie adaptative du choix de la séquence de distributions intermédiaires ainsi que le système de recyclage proposés

permettent d'améliorer significativement la précision des estimateurs.

Enfin, dans le chapitre 4, le problème de la régression pénalisée dans des modèles linéaires généralisés est abordé. Après avoir décrit le problème de régression, nous avons notamment proposé une nouvelle classe de penalisation non-convexe pour la régularisation des coefficients de régression inconnus dans un cadre Bayésien par l'utilisation d'une loi a-priori α-stable. Ensuite, une méthode Bayésienne basée sur l'utilisation d'échantillonneurs SMC parallèles a été proposée afin de résoudre conjointement le problème de sélection de modèles et d'estimation des coefficients de regression. La procédure de sélection de modèles permet d'obtenir, automatiquement, un critère probabiliste pour décider de la meilleure "fonction de base" à utiliser afin d'approcher la relation entre les variables d'entrée et de sortie mais aussi de la meilleure distribution de probabilité correspondante au bruit d'observation et / ou au résidu de la régression. Des simulations numériques, dans le cadre de variables observées continues et discrètes, ont permis de démontrer la capacité de l'algorithme proposé à estimer les différentes quantités d'intérêt ainsi qu'à prédire la réponse du système. Ces résultats ont également montré qu'une amélioration significative pouvait être obtenue en utilisant les stratégies d'amélioration proposées pour l'échantillonneur SMC et développées dans le chapitre 2.

List of Publications

International conference papers:

- T. L. T. Nguyen, F. Septier, G. W. Peters and Y. Delignon. "Improving SMC Sampler Estimate by Recycling All Past Simulated Particles" in IEEE International Workshop on Statistical Signal Processing, Jul. 2014, Melbourne, Australia.
- T. L. T. Nguyen, F. Septier, G. W. Peters and Y. Delignon. "Bayesian Model Selection and Parameter Estimation in Penalized Regression Model Using SMC Samplers" in 21st European Signal Processing Conference (EUSIPCO), Sep 2013, Marrakech, Morocco.

Appendices

Appendix A

Proof of Proposition 2.1.1

In this appendix, we present the proof of Proposition 2.1.1 related to the asymptotic variances of the SMC sampler estimator when resampling is never performed. These asymptotic variances presented in this proposition are derived when the backward kernel used is given by:

$$\mathcal{L}_{t-1}(\boldsymbol{\theta}_t, \boldsymbol{\theta}_{t-1}) = \frac{\pi_t(\boldsymbol{\theta}_{t-1})\mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)}{\pi_t(\boldsymbol{\theta}_t)} \tag{A.1}$$

which is generally the one use when an MCMC kernel is used as forward kernel in order to be able to compute the incremental weight (as discussed in Chapter 1). Moreover, we assume that the mutation kernel used is mixing perfectly, i.e.:

$$\mathcal{K}_t(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t) = \pi_t(\boldsymbol{\theta}_t) \tag{A.2}$$

By plugging Eq. (A.2) into Eq. (A.1), the backward kernel can be written as:

$$\mathcal{L}_{t-1}(\boldsymbol{\theta}_t, \boldsymbol{\theta}_{t-1}) = \pi_t(\boldsymbol{\theta}_{t-1}) \tag{A.3}$$

On the estimation of an expectation

In order to obtain the variance of the estimate of an expectation when an SMC sampler without resampling is employed, we start from the general expression derived in the paper of Del Moral et al. [Del Moral et al., 2006] in Eq. (2.5):

$$N^{\frac{1}{2}}\left\{\mathbb{E}_{\pi_t^N}(\varphi) - \mathbb{E}_{\pi_t}(\varphi)\right\} \Rightarrow \mathcal{N}(0, \sigma^2_{IS,t}(\varphi)) \tag{A.4}$$

with

$$\sigma^2_{IS,t}(\varphi) = \int \frac{\tilde{\pi}_t^2(\boldsymbol{\theta}_{1:t})}{\eta_t(\boldsymbol{\theta}_{1:t})}\left\{\varphi(\boldsymbol{\theta}_t) - \mathbb{E}_{\pi_t}(\varphi)\right\}^2 d\boldsymbol{\theta}_{1:t} \tag{A.5}$$

By plugging Eq. (A.3) into Eqs (1.50) and (1.51), we obtain:

$$\begin{aligned}
\tilde{\pi}_t(\boldsymbol{\theta}_{1:t}) &= \pi_t(\boldsymbol{\theta}_t)\prod_{k=1}^{t-1}\mathcal{L}_k(\boldsymbol{\theta}_{k+1}, \boldsymbol{\theta}_k) \\
&= \pi_t(\boldsymbol{\theta}_t)\prod_{k=1}^{t-1}\pi_{k+1}(\boldsymbol{\theta}_k) \\
&= \pi_t(\boldsymbol{\theta}_t)\prod_{k=2}^{t}\pi_k(\boldsymbol{\theta}_{k-1})
\end{aligned} \tag{A.6}$$

Now by using the perfect mixing assumption in Eq. (A.2) into Eq. (1.53), the

joint proposal can be written as:

$$\begin{aligned}\eta_t(\boldsymbol{\theta}_{1:t}) &= \eta_1(\boldsymbol{\theta}_1)\prod_{k=2}^{t}\mathcal{K}_k(\boldsymbol{\theta}_{t-1},\boldsymbol{\theta}_t)\\ &= \eta_1(\boldsymbol{\theta}_1)\prod_{k=2}^{t}\pi_k(\boldsymbol{\theta}_k)\end{aligned} \tag{A.7}$$

Finally, by plugging the equations (A.6) and (A.7) into the general expression of the asymptotic variance given in (A.5), we obtain:

$$\begin{aligned}\sigma^2_{IS,t}(\varphi) &= \int \frac{\left(\pi_t(\boldsymbol{\theta}_t)\prod_{k=2}^{t}\pi_k(\boldsymbol{\theta}_{k-1})\right)^2}{\eta_1(\boldsymbol{\theta}_1)\prod_{k=2}^{t}\pi_k(\boldsymbol{\theta}_k)}\left\{\varphi(\boldsymbol{\theta}_t)-\mathbb{E}_{\pi_t}(\varphi)\right\}^2 d\boldsymbol{\theta}_{1:t}\\ &= \int\frac{\pi_2^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)}d\boldsymbol{\theta}_1\prod_{k=3}^{t}\int\frac{\pi_k^2(\boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1})}d\boldsymbol{\theta}_{k-1}\int\pi_t(\boldsymbol{\theta}_t)\left\{\varphi(\boldsymbol{\theta}_t)-\mathbb{E}_{\pi_t}(\varphi)\right\}^2 d\boldsymbol{\theta}_t\\ &= \int\frac{\pi_2^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)}d\boldsymbol{\theta}_1\prod_{k=3}^{t}\int\frac{\pi_k^2(\boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1})}d\boldsymbol{\theta}_{k-1}\underbrace{\left\{\mathbb{E}_{\pi_t}(\varphi^2)-\mathbb{E}^2_{\pi_t}(\varphi)\right\}}_{\mathbb{V}\mathrm{ar}_{\pi_t}(\varphi(\boldsymbol{\theta}))}\end{aligned} \tag{A.8}$$

On the estimation of the normalizing constant

The general asymptotic result for this estimator is given in Eq. (2.7) by:

$$N^{\frac{1}{2}}\left\{\log\left(\widehat{\frac{Z_t}{Z_1}}\right)-\log\left(\frac{Z_t}{Z_1}\right)\right\}\Rightarrow\mathcal{N}(0,\sigma^2_{IS,t}) \tag{A.9}$$

with

$$\sigma^2_{IS,t}=\int\frac{\tilde{\pi}_t(\boldsymbol{\theta}_{1:t})^2}{\eta_t(\boldsymbol{\theta}_{1:t})}d\boldsymbol{\theta}_{1:t}-1 \tag{A.10}$$

By using derivations of the previous Equation (A.8) since the same quantities are involved, it is straightforward to obtain the results for this estimator given in Proposition 2.1.1

$$\begin{aligned}\sigma^2_{IS,t} &= \int\frac{\pi_2^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)}d\boldsymbol{\theta}_1\prod_{k=3}^{t}\int\frac{\pi_k^2(\boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1})}d\boldsymbol{\theta}_{k-1}\underbrace{\int\pi_t(\boldsymbol{\theta}_t)d\boldsymbol{\theta}_t}_{=1}-1\\ &= \int\frac{\pi_2^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)}d\boldsymbol{\theta}_1\prod_{k=3}^{t}\int\frac{\pi_k^2(\boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1})}d\boldsymbol{\theta}_{k-1}-1\end{aligned} \tag{A.11}$$

Appendix B

Proof of Proposition 2.1.2

In this appendix, we present the proof of Proposition 2.1.2 related to the asymptotic variance of the SMC sampler estimator when resampling is performed after the sampling step.

On the estimation of an expectation

In order to obtain the variance of the expectation estimator with an SMC sampler when resampling is employed after the sampling step, we start from the general expression derived in the paper of Del Moral et al. [Del Moral et al., 2006] in Eq. (2.9):

$$N^{\frac{1}{2}}\left\{\mathbb{E}_{\pi_t^N}(\varphi)-\mathbb{E}_{\pi_t}(\varphi)\right\}\Rightarrow\mathcal{N}(0,\sigma^2_{SMC,t}(\varphi)) \tag{B.1}$$

with

$$\begin{aligned}\sigma^2_{SMC,t}(\varphi) = &\int\frac{\tilde{\pi}_t^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)}\left\{\int\varphi(\boldsymbol{\theta}_t)\tilde{\pi}_t(\boldsymbol{\theta}_t|\boldsymbol{\theta}_1)d\boldsymbol{\theta}_t-\mathbb{E}_{\pi_t}(\varphi)\right\}^2 d\boldsymbol{\theta}_1\\ &+\sum_{k=2}^{t-1}\int\frac{\tilde{\pi}_t^2(\boldsymbol{\theta}_k)\mathcal{L}^2_{k-1}(\boldsymbol{\theta}_k,\boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1})\mathcal{K}_k(\boldsymbol{\theta}_{k-1},\boldsymbol{\theta}_k)}\left\{\int\varphi(\boldsymbol{\theta}_t)\tilde{\pi}_t(\boldsymbol{\theta}_t|\boldsymbol{\theta}_k)d\boldsymbol{\theta}_t-\mathbb{E}_{\pi_t}(\varphi)\right\}^2 d\boldsymbol{\theta}_{k-1:k}\\ &+\int\frac{\pi_t^2(\boldsymbol{\theta}_t)\mathcal{L}^2_{t-1}(\boldsymbol{\theta}_t,\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})\mathcal{K}_t(\boldsymbol{\theta}_{t-1},\boldsymbol{\theta}_t)}\left\{\varphi(\boldsymbol{\theta}_t)-\mathbb{E}_{\pi_t}(\varphi)\right\}^2 d\boldsymbol{\theta}_{t-1:t}\end{aligned} \tag{B.2}$$

where

$$\tilde{\pi}_t(\boldsymbol{\theta}_k)=\int\tilde{\pi}_t(\boldsymbol{\theta}_{1:t})d\boldsymbol{\theta}_{1:k-1}d\boldsymbol{\theta}_{k+1:t} \tag{B.3}$$

$$\tilde{\pi}_t(\boldsymbol{\theta}_t|\boldsymbol{\theta}_k)=\frac{\int\tilde{\pi}_t(\boldsymbol{\theta}_{1:t})d\boldsymbol{\theta}_{1:k-1}d\boldsymbol{\theta}_{k+1:t-1}}{\tilde{\pi}_t(\boldsymbol{\theta}_k)} \tag{B.4}$$

Let us firstly derived B.3 under the perfect mixing assumption as well as the use of the specific backward kernel summarized by both Eqs. (A.2) and (A.3) - we have thus by using Eq. (A.6):

$$\begin{aligned}\tilde{\pi}_t(\boldsymbol{\theta}_k) &= \int\pi_t(\boldsymbol{\theta}_t)\prod_{k=2}^{t}\pi_k(\boldsymbol{\theta}_{k-1})d\boldsymbol{\theta}_{1:k-1}d\boldsymbol{\theta}_{k+1:t}\\ &= \pi_{k+1}(\boldsymbol{\theta}_k)\end{aligned} \tag{B.5}$$

Now, the conditional distribution defined in Eq. (B.4) can be rewritten as:

$$\begin{aligned}\tilde{\pi}_t(\boldsymbol{\theta}_t|\boldsymbol{\theta}_k) &= \frac{\pi_{k+1}(\boldsymbol{\theta}_k)\pi_t(\boldsymbol{\theta}_t)}{\pi_{k+1}(\boldsymbol{\theta}_k)}\\ &= \pi_t(\boldsymbol{\theta}_t)\end{aligned} \tag{B.6}$$

From this expression of the conditional, we have thus $\forall k \neq t$, :

$$\begin{aligned} \int \varphi(\boldsymbol{\theta}_t)\tilde{\pi}_t(x_t|\boldsymbol{\theta}_k)d\boldsymbol{\theta}_t &= \int \varphi(\boldsymbol{\theta}_t)\pi_t(\boldsymbol{\theta}_t)d\boldsymbol{\theta}_t \\ &= \mathbb{E}_{\pi_t}(\varphi) \end{aligned} \tag{B.7}$$

The expression of the variance in Eq. (B.2) can be simplified as:

$$\begin{aligned} \sigma^2_{SMC,t}(\varphi) &= \int \frac{\pi_t^2(\boldsymbol{\theta}_t)L_{t-1}^2(\boldsymbol{\theta}_t,\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})K_t(\boldsymbol{\theta}_{t-1},\boldsymbol{\theta}_t)}\left\{\varphi(\boldsymbol{\theta}_t)-\mathbb{E}_{\pi_t}(\varphi)\right\}^2 d\boldsymbol{\theta}_{t-1:t} \\ &= \int \frac{\pi_t^2(\boldsymbol{\theta}_t)\pi_t^2(\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})\pi_t(\boldsymbol{\theta}_t)}\left\{\varphi(x_t)-\mathbb{E}_{\pi_t}(\varphi)\right\}^2 d\boldsymbol{\theta}_{t-1:t} \\ &= \int \frac{\pi_t^2(\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})}d\boldsymbol{\theta}_{t-1}\underbrace{\left\{\mathbb{E}_{\pi_t}(\varphi^2)-\mathbb{E}^2_{\pi_t}(\varphi)\right\}}_{\mathbb{V}\mathrm{ar}_{\pi_t}(\varphi(\boldsymbol{\theta}))} \end{aligned} \tag{B.8}$$

On the estimation of the normalizing constant

The general asymptotic results for this estimator is given in Eq. (2.11) by:

$$N^{\frac{1}{2}}\left\{\log\left(\widehat{\frac{Z_t}{Z_1}}\right)-\log\left(\frac{Z_t}{Z_1}\right)\right\} \Rightarrow \mathcal{N}(0,\sigma^2_{SMC,t}) \tag{B.9}$$

with

$$\begin{aligned} \sigma^2_{SMC,t} = \int \frac{\tilde{\pi}_t^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)}d\boldsymbol{\theta}_1 - 1 + \sum_{k=2}^{t-1}\left[\int \frac{\tilde{\pi}_t^2(\boldsymbol{\theta}_k)\mathcal{L}_{k-1}^2(\boldsymbol{\theta}_k,\boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1})\mathcal{K}_k(\boldsymbol{\theta}_{k-1},\boldsymbol{\theta}_k)}d\boldsymbol{\theta}_{k-1:k} - 1\right] \\ + \int \frac{\pi_t^2(\boldsymbol{\theta}_t)\mathcal{L}_{t-1}^2(\boldsymbol{\theta}_t,\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})\mathcal{K}_t(\boldsymbol{\theta}_{t-1},\boldsymbol{\theta}_t)}d\boldsymbol{\theta}_{t-1:t} - 1 \end{aligned} \tag{B.10}$$

By using Eq. (B.5), the result in Proposition 2.1.2 is easily obtained as:

$$\begin{aligned} \sigma^2_{SMC,t} &= \int \frac{\pi_2^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)}d\boldsymbol{\theta}_1 - 1 + \sum_{k=2}^{t-1}\left[\int \frac{\pi_{k+1}^2(\boldsymbol{\theta}_k)\pi_k^2(\boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1})\pi_k(\boldsymbol{\theta}_k)}d\boldsymbol{\theta}_{k-1:k} - 1\right] \\ &\quad + \int \frac{\pi_t^2(\boldsymbol{\theta}_t)\pi_t^2(\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})\pi_t(\boldsymbol{\theta}_t)}d\boldsymbol{\theta}_{t-1:t} - 1 \\ &= \int \frac{\pi_2^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)}d\boldsymbol{\theta}_1 + \sum_{k=2}^{t-1}\int \frac{\pi_{k+1}^2(\boldsymbol{\theta}_k)}{\pi_k(\boldsymbol{\theta}_k)}d\boldsymbol{\theta}_k \int \frac{\pi_k^2(\boldsymbol{\theta}_{k-1})}{\pi_{k-1}(\boldsymbol{\theta}_{k-1})}d\boldsymbol{\theta}_{k-1} \\ &\quad + \int \frac{\pi_t^2(\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})}d\boldsymbol{\theta}_{t-1} - t \end{aligned} \tag{B.11}$$

Appendix C

Proof of Proposition 2.1.3

In this appendix, we present the proof of Proposition C related to the asymptotic variance of the SMC sampler estimator when resampling is performed before the sampling step. In [Del Moral et al., 2006], the authors does not study this case since the resampling cannot always be done before the sampling. In particular, as discussed in Section 1.3.2.3 we can do the resampling before the sampling when the weights does not depend on the current value of the particle as it is the case when the backward kernel is the one used in this proposition.

On the estimation of an expectation

This results is quite straightforward to obtain by using classical Monte-Carlo results since we use a perfectly mixing kernel, the particles are (asymptotically) drawn at the t-th iteration, for $i = 1, \ldots, N$:

$$\boldsymbol{\theta}_t^{(i)} \overset{\text{iid}}{\sim} \pi_t(\cdot) \tag{C.1}$$

which leads to the following particle estimate of the expectation:

$$\mathbb{E}_{\pi_t^N}(\varphi) = \frac{1}{N}\sum_{i=1}^{N} \varphi(\boldsymbol{\theta}_t^{(i)}) \tag{C.2}$$

All particles are equally weighted since we have performed the resampling before the sampling step. As a consequence, we obtain:

$$N^{\frac{1}{2}}\left\{\mathbb{E}_{\pi_t^N}(\varphi) - \mathbb{E}_{\pi_t}(\varphi)\right\} \Rightarrow \mathcal{N}(0, \sigma^2_{SMC_2,t}(\varphi)) \tag{C.3}$$

with

$$\sigma^2_{SMC_2,t}(\varphi) = \left\{\mathbb{E}_{\pi_t}(\varphi^2(\boldsymbol{\theta})) - \mathbb{E}^2_{\pi_t}(\varphi(\boldsymbol{\theta}))\right\} = \mathbb{V}\text{ar}_{\pi_t}(\varphi(\boldsymbol{\theta})) \tag{C.4}$$

On the estimation of the normalizing constant

In this section, we will derive the asymptotic variance related to the estimator of the normalizing constant. Let us firstly study the estimate of the ratio of normalizing constant,Z_t/Z_{t-1}, defined in Eq. (1.59) which is given in the context of the

proposition C by:

$$\begin{aligned}\widehat{\frac{Z_t}{Z_{t-1}}} &= \sum_{m=1}^{N} \widetilde{W}_{t-1}^{(m)} w_t(\boldsymbol{\theta}_{t-1}^{(m)}, \boldsymbol{\theta}_t^{(m)}) \\ &= \frac{1}{N}\sum_{m=1}^{N} \frac{\gamma_t(\boldsymbol{\theta}_{t-1}^{(m)})}{\gamma_{t-1}(\boldsymbol{\theta}_{t-1}^{(m)})} \end{aligned} \tag{C.5}$$

since the particles are equally weighted due to the resampling before the sampling and the unnormalized incremental weights are defined in Eq. (1.68) when the backward kernel in Eq. (1.67) is used. Moreover, owing to the perfect mixing assumption, we have: for $i = 1, \ldots, N$:

$$\boldsymbol{\theta}_{t-1}^{(i)} \overset{\text{iid}}{\sim} \pi_{t-1}(\cdot) \tag{C.6}$$

From (C.5) and (C.6), the unbiasedness of this estimator is obvious:

$$\begin{aligned}\mathbb{E}_{\pi_{t-1}}\left[\widehat{\frac{Z_t}{Z_{t-1}}}\right] &= \int \frac{\gamma_t(\boldsymbol{\theta}_{t-1})}{\gamma_{t-1}(\boldsymbol{\theta}_{t-1})}\pi_{t-1}(\boldsymbol{\theta}_{t-1})d\boldsymbol{\theta}_{t-1} \\ &= \int \frac{\gamma_t(\boldsymbol{\theta}_{t-1})}{\gamma_{t-1}(\boldsymbol{\theta}_{t-1})}\frac{\gamma_{t-1}(\boldsymbol{\theta}_{t-1})}{Z_{t-1}}d\boldsymbol{\theta}_{t-1} \\ &= \frac{Z_t}{Z_{t-1}} \end{aligned} \tag{C.7}$$

Let us now study the variance of this estimator:

$$\begin{aligned}\mathbb{V}\text{ar}\left(\widehat{\frac{Z_t}{Z_{t-1}}}\right) &= \frac{1}{N}\sum_{m=1}^{N} \mathbb{V}\text{ar}\left(\frac{\gamma_t(\boldsymbol{\theta}_{t-1})}{\gamma_{t-1}(\boldsymbol{\theta}_{t-1})}\right) \\ &= \frac{1}{N}\left\{\mathbb{E}_{\pi_{t-1}}\left[\frac{\gamma_t^2(\boldsymbol{\theta}_{t-1})}{\gamma_{t-1}^2(\boldsymbol{\theta}_{t-1})}\right] - \mathbb{E}_{\pi_{t-1}}^2\left[\frac{\gamma_t(\boldsymbol{\theta}_{t-1})}{\gamma_{t-1}(\boldsymbol{\theta}_{t-1})}\right]\right\} \end{aligned} \tag{C.8}$$

In this expression, the mean has already been derived in Eq. (C.7) and the second moment can be written as

$$\begin{aligned}\mathbb{E}_{\pi_{t-1}}\left[\frac{\gamma_t^2(\boldsymbol{\theta}_{t-1})}{\gamma_{t-1}^2(\boldsymbol{\theta}_{t-1})}\right] &= \int \pi_{t-1}(\boldsymbol{\theta}_{t-1})\frac{\gamma_t^2(\boldsymbol{\theta}_{t-1})}{\gamma_{t-1}^2(\boldsymbol{\theta}_{t-1})}d\boldsymbol{\theta}_{t-1} \\ &= \frac{Z_t^2}{Z_{t-1}^2}\int \frac{\pi_t^2(\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})}d\boldsymbol{\theta}_{t-1} \end{aligned} \tag{C.9}$$

which give the following expression for the variance:

$$\mathbb{V}\text{ar}\left(\widehat{\frac{Z_t}{Z_{t-1}}}\right) = \frac{1}{N}\left(\frac{Z_t}{Z_{t-1}}\right)^2\left[\int \frac{\pi_t^2(\boldsymbol{\theta}_{t-1})}{\pi_{t-1}(\boldsymbol{\theta}_{t-1})}d\boldsymbol{\theta}_{t-1} - 1\right] \tag{C.10}$$

In the results given in Proposition C, we want to have the variance of the log of

the normalizing constant at time t which can be rewritten using Eq. (1.60) as

$$\log\left(\frac{\widehat{Z_t}}{Z_1}\right) = \sum_{n=2}^{t} \log\left(\frac{\widehat{Z_n}}{Z_{n-1}}\right) \tag{C.11}$$

From this expression, we have to obtain the variance of the log ratio of the normalizing constant. This term can be obtained by using the *delta method* [Casella and Berger, 2002] that states that if

$$N^{\frac{1}{2}}\left(X_n - \mu\right) \Rightarrow \mathcal{N}(0, \sigma^2) \tag{C.12}$$

then for a given function g and a specific value of μ (by assuming that $g'(\mu)$ exists and is not 0)

$$N^{\frac{1}{2}}\left(g(X_n) - g(\mu)\right) \Rightarrow \mathcal{N}(0, \sigma^2\left[g'(\mu)\right]^2) \tag{C.13}$$

By using this delta method and Eqs. (C.10) and (C.11), we finally obtain the result presented in Proposition that C:

$$N^{\frac{1}{2}}\left\{\log\left(\frac{\widehat{Z_t}}{Z_1}\right) - \log\left(\frac{Z_t}{Z_1}\right)\right\} \Rightarrow \mathcal{N}(0, \sigma^2_{SMC_2,t}) \tag{C.14}$$

with

$$\sigma^2_{SMC_2,t} = \int \frac{\pi_2^2(\boldsymbol{\theta}_1)}{\eta_1(\boldsymbol{\theta}_1)} d\boldsymbol{\theta}_1 \quad + \sum_{k=2}^{t-1} \int \frac{\pi_{k+1}^2(\boldsymbol{\theta}_k)}{\pi_k(\boldsymbol{\theta}_k)} d\boldsymbol{\theta}_k - (t-1) \tag{C.15}$$

Appendix D

Proof of the Posterior Cramér-Rao bound

As presented in Section 3.3, the Fisher information matrix (FIM) for the posterior Cramér-Rao bound (PCRB) can be decomposed as follows:

$$\boldsymbol{J} = \underbrace{\int_{\Theta_k} \boldsymbol{J}_d(\boldsymbol{\theta}_K) p(\boldsymbol{\theta}_K|\mathcal{M}_k = K) d\boldsymbol{\theta}_K}_{\boldsymbol{J}_d} + \underbrace{\mathbb{E}\left[-\Delta^{\boldsymbol{\theta}}_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}_K|\mathcal{M}_k = K)\right]}_{\boldsymbol{J}_p} \tag{D.1}$$

In this appendix, we derive respectively $\boldsymbol{J}_d(\boldsymbol{\theta}_K)$ and $\boldsymbol{J}_p$.

D.1 Derivation of the likelihood information matrix

The information matrix $\boldsymbol{J}_d(\boldsymbol{\theta}_K)$ is defined as:

$$\boldsymbol{J}_d(\boldsymbol{\theta}_K) = -\mathbb{E}_{\boldsymbol{z}|\boldsymbol{\theta}_K}\left[\Delta^{\boldsymbol{\theta}}_{\boldsymbol{\theta}} \log p(\boldsymbol{z}|\boldsymbol{\theta}_K, \mathcal{M}_k = K)\right] \tag{D.2}$$

The first derivative of the log likelihood is given by:

$$\begin{aligned} \nabla^T_{\boldsymbol{\theta}} \log p(\boldsymbol{z}|\boldsymbol{\theta}_K, \mathcal{M}_k = K) &= \sum_{i=1}^{N} \nabla^T_{\boldsymbol{\theta}} \log p(z_i|\boldsymbol{\theta}_K, \mathcal{M}_k = K) \\ &= \sum_{i=1}^{N} \frac{\nabla^T_{\boldsymbol{\theta}} p(z_i|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{p(z_i|\boldsymbol{\theta}_K, \mathcal{M}_k = K)} \end{aligned} \tag{D.3}$$

Therefore, the second derivative can be written as:

$$\begin{aligned} \Delta^{\boldsymbol{\theta}}_{\boldsymbol{\theta}} \log p(\boldsymbol{z}|\boldsymbol{\theta}_K, \mathcal{M}_k = K) &= \nabla_{\boldsymbol{\theta}} \nabla^T_{\boldsymbol{\theta}} \log p(\boldsymbol{z}|\boldsymbol{\theta}_K, \mathcal{M}_k = K) \\ &= \sum_{i=1}^{N} \frac{\nabla_{\boldsymbol{\theta}} \nabla^T_{\boldsymbol{\theta}} p(z_i|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{p(z_i|\boldsymbol{\theta}_K, \mathcal{M}_k = K)} \\ &\quad - \frac{\nabla_{\boldsymbol{\theta}} p(z_i|\boldsymbol{\theta}_K, \mathcal{M}_k = K) \nabla^T_{\boldsymbol{\theta}} p(z_i|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{p(z_i|\boldsymbol{\theta}_K, \mathcal{M}_k = K)^2} \end{aligned} \tag{D.4}$$

To get Eq. (D.2), we now take the negative expectation of this second derivative with respect to $p(z_i|\boldsymbol{\theta}_K, \mathcal{M}_k = K)$:

$$\begin{aligned}\boldsymbol{J}_d(\boldsymbol{\theta}_K) &= \sum_{i=1}^{N}\sum_{j=0}^{L-1} p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)\left\{-\frac{\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\theta}}^T p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}\right.\\ &\quad \left.+\frac{\nabla_{\boldsymbol{\theta}} p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)\nabla_{\boldsymbol{\theta}}^T p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)^2}\right\}\\ &= \sum_{i=1}^{N}\sum_{j=0}^{L-1}\frac{\nabla_{\boldsymbol{\theta}} p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)\nabla_{\boldsymbol{\theta}}^T p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}\\ &\quad -\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\theta}}^T p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K) \qquad \text{(D.5)}\end{aligned}$$

The second term is equal to 0 since:

$$\begin{aligned}\sum_{i=1}^{N}\sum_{j=0}^{L-1}\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\theta}}^T p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K) &= \sum_{i=1}^{N}\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\theta}}^T \underbrace{\sum_{j=0}^{L-1} p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}_{=1}\\ &= 0 \qquad \text{(D.6)}\end{aligned}$$

As a consequence, we finally obtain:

$$\boldsymbol{J}_d(\boldsymbol{\theta}_K) = \sum_{i=1}^{N}\sum_{j=0}^{L-1}\frac{\nabla_{\boldsymbol{\theta}} p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)\nabla_{\boldsymbol{\theta}}^T p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K)} \qquad \text{(D.7)}$$

Using Eq. (3.7), the gradient term involved in this expression can be expressed as:

$$\nabla_{\boldsymbol{\theta}} p(z_i = j|\boldsymbol{\theta}_K, \mathcal{M}_k = K) = \sum_{l=0}^{L-1} p(z_i = j|b_i = l)\nabla_{\boldsymbol{\theta}} p(b_i = l|\boldsymbol{\theta}_K, \mathcal{M}_k = K) \qquad \text{(D.8)}$$

with

$$p(b_i = l|\boldsymbol{\theta}_K, \mathcal{M}_k = K) = Q\left(\frac{\lambda_{i,l} - a_i}{\sigma}\right) - Q\left(\frac{\lambda_{i,l+1} - a_i}{\sigma}\right) \qquad \text{(D.9)}$$

As a consequence, since the Q- function is the complementary Gaussian cumulative distribution, we can easily remark that:

$$\nabla_{\boldsymbol{\theta}} p(b_i = l|\boldsymbol{\theta}_K, \mathcal{M}_k = K) = \frac{1}{\sqrt{2\pi\sigma^2}}\underbrace{\left(e^{-\frac{(\lambda_{i,l} - a_i)^2}{2\sigma^2}} - e^{-\frac{(\lambda_{i,l+1} - a_i)^2}{2\sigma^2}}\right)}_{\rho_{i,l}}\nabla_{\boldsymbol{\theta}} a_i \qquad \text{(D.10)}$$

Finally from the definition of a_i in Eq. (3.2), we obtain, for $k = 1, \ldots, K$:

$$\begin{aligned}
\frac{\partial p(b_i = l|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{\partial P_k} &= \left(\frac{d_0}{d_{i,k}}\right)^{n/2} \frac{\rho_{i,l}}{2\sqrt{2\pi\sigma^2 P_k}} \\
\frac{\partial p(b_i = l|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{\partial x_k} &= \left(\frac{d_0}{d_{i,k}}\right)^{n/2} \frac{nP_k^{1/2} d_{i,k}^{-2} \rho_{i,l}(p_{x,i} - x_k)}{2\sqrt{2\pi\sigma^2}} \\
\frac{\partial p(b_i = l|\boldsymbol{\theta}_K, \mathcal{M}_k = K)}{\partial y_k} &= \left(\frac{d_0}{d_{i,k}}\right)^{n/2} \frac{nP_k^{1/2} d_{i,k}^{-2} \rho_{i,l}(p_{y,i} - y_k)}{2\sqrt{2\pi\sigma^2}}
\end{aligned} \tag{D.11}$$

which completes the analytical calculation of $\boldsymbol{J}_d(\boldsymbol{\theta}_K)$.

D.2 Derivation of the *a priori* information matrix

In this section, we derive the *a priori* information matrix given by:

$$\boldsymbol{J}_p = \mathbb{E}\left[-\Delta_{\boldsymbol{\theta}}^{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}_K|\mathcal{M}_k = K)\right] \tag{D.12}$$

From the prior distributions considered in this work - Eqs. 3.10 and 3.12, each target's location and power are independent and identically distributed. $\boldsymbol{J}_p$ will be therefore a $3K \times 3K$ block diagonal matrix with information associated to the location and the power defined respectively as:

$$\mathbb{E}\left[-\Delta_{[x_k,y_k]^T}^{[x_k,y_k]^T} \log \mathcal{N}([x_k, y_k]^T|\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)\right] = \boldsymbol{\Sigma}_p^{-1} \tag{D.13}$$

and

$$\begin{aligned}
\mathbb{E}\left[-\Delta_{P_k}^{P_k} \log \mathcal{IG}(P_k|a, b)\right] &= \mathbb{E}\left[-\frac{\partial^2}{\partial P_k^2} \log\left\{\frac{b^a}{\Gamma(a)} P_k^{-a-1} \exp\left(-\frac{b}{P_k}\right)\right\}\right] \\
&= \mathbb{E}\left[\frac{\partial^2}{\partial P_k^2}(a+1)\log(P_k) + \frac{b}{P_k}\right] \\
&= \mathbb{E}\left[2bP_k^{-3} - (a+1)P_k^{-2}\right] \\
&= 2b\mathbb{E}\left[P_k^{-3}\right] - (a+1)\mathbb{E}\left[P_k^{-2}\right]
\end{aligned} \tag{D.14}$$

Let us now derive the two moments involved in this expression. We have, for $n > 0$:

$$\begin{aligned}
\mathbb{E}\left[x^{-n}\right] &= \int_0^{+\infty} x^{-n} \frac{b^a}{\Gamma(a)} x^{-a-1} \exp\left(-\frac{b}{x}\right) dx \\
&= \frac{b^a}{\Gamma(a)} \int_0^{+\infty} x^{-(a+n)-1} \exp\left(-\frac{b}{x}\right) dx \\
&= \frac{b^a}{\Gamma(a)} \frac{\Gamma(a+n)}{b^{a+n}}
\end{aligned} \tag{D.15}$$

The last expression is obtained from the expression of the normalizing constant of

an inverse-gamma distribution, $\mathcal{IG}(a+n,b)$. By using the equality of the Gamma function, $\Gamma(a+1)=a\Gamma(a)$, we obtain:

$$\mathbb{E}\left[P_k^{-2}\right] = \frac{(a+1)a}{b^2} \tag{D.16}$$

$$\mathbb{E}\left[P_k^{-3}\right] = \frac{(a+2)(a+1)a}{b^3} \tag{D.17}$$

By plugging these expressions in Eq. (D.14), the prior information for the power is given by:

$$\mathbb{E}\left[-\Delta_{P_k}^{P_k}\log\mathcal{IG}(P_k|a,b)\right] = \frac{a(a+1)(a+3)}{b^2} = \xi \tag{D.18}$$

leading to

$$\boldsymbol{J}_p = \begin{bmatrix} \xi & & & \\ & \boldsymbol{\Sigma}_p^{-1} & 0 & \\ & \ddots & & \\ & 0 & \xi & \\ & & & \boldsymbol{\Sigma}_p^{-1} \end{bmatrix} \tag{D.19}$$

Appendix E

Exponential Family of distributions

In this appendix, we will describe the distributional models used in the Generalized Linear Model (GLM), described in Chapter 4 - Section 4.3, for the response μ_i's selected from the *exponential family* such that μ_i has density of the canonical form given by

$$f(y_i;\nu_i,\phi) = \exp\left[\frac{y_i\nu_i - b(\nu_i)}{a(\phi)} + c(y_i,\phi)\right] \ , \tag{E.1}$$

where

- $a(\cdot), b(\cdot)$ and $c(\cdot)$ are known functions that vary from one exponential family to another.

- $\nu_i = g_c(\mu_i)$, the *canonical parameter* for the exponential family in question, is a function of the expectation $\mu_i \equiv \mathbb{E}(y_i)$; moreover, the *canonical link* function $g_c(\cdot)$ does not depend on ϕ.

- $\phi > 0$ is the same for all i and is known as the *dispersion parameter*, which, in some families, takes on fixed, known value while in other families it is an unknown parameter to be estimated from the data along with ν.

A convenient property of distributions in the exponential families is that the conditional variance of y_i is a function of its mean μ_i and, possibly, a dispersion parameter ϕ, say, $\mathbb{V}\text{ar}(y_i) = \phi\mathbb{V}\text{ar}(\mu_i)$. The variance functions for the commonly used exponential families appear in Table E.1. The conditional variance of the response in the Gaussian family is constant, ϕ, which is simply alternative notation for what we previously termed the error variance, σ^2_ε. In the binomial and Poisson families, the dispersion parameter is set to the fixed value $\phi = 1$.

Table E.1 also shows the range of variation of the response variable in each family, and the so-called *canonical* (or "*natural*" *link function* associated with each family. The canonical link simplifies the GLM, but other link functions may be used as well. Indeed, one of the strengths of the GLM paradigm – in constract to transformations of the response variable in linear regression, $\mathbb{E}(g(y_i)) = \sum_{j=1}^{p} \beta_j \mathbf{\Phi}_k^j(\mathbf{x}_{i,j})$, is that the choice of linearizing transformation is partly separated from the distribution of y_j.

Table E.1: Canonical Link, Response Range and Conditional Variance Reduction Function for Exponential Families

Family	*Canonical Link*	*Range of* y_i	$\mathbb{V}\text{ar}(y_i\|\vartheta_i)$
Gaussian	Identity	$(-\infty, +\infty)$	ϕ
Binomial	Logit	$\frac{0, 1, \ldots, n_i}{n_i}$	$\frac{\mu_i(1-\mu_i)}{n_i}$
Poisson	Log	$0, 1, 2, \ldots,$	μ_i
Gamma	Inverse	$(0, \infty)$	$\phi\mu_i^2$
Inverse–Gamma	Inverse–square	$(0, \infty)$	$\phi\mu_i^3$

Note: ϕ is the dispersion parameter, ϑ_i is the linear predictor, and μ_i is the expectation of y_i (the response). In the binomial, n_i is the number of trials.

Under this formulation of the family the resulting mean is given by

$$\mu_i = \left.\frac{\partial b(\nu)}{\partial \nu}\right|_{\nu=\nu_i} \tag{E.2}$$

and the variance is given by

$$\mathbb{V}\text{ar}_i = a(\phi)\left.\frac{\partial^2 b(\nu)}{\partial \nu^2}\right|_{\nu=\nu_i} = a(\phi)\left.\frac{\partial \mu_i}{\partial \nu}\right|_{\nu=\nu_i} \tag{E.3}$$

This family of models contains many standard distributions allowing for continuous response distributions as well as discrete response distributions such as the normal, exponential, gamma, chi-squared, beta, Dirichlet, Bernoulli, categorical, Poisson, Wishart, Inverse Wishart and many others. Let us now illustrate two choices from this GLM: Normal regression model with identity link and Poisson regression model with log link.

1. **Normal Regression Model**: We consider the standard generalized linear basis regression model involving a link function $g(\cdot)$ given by the identity, as well as specifying a normal distribution for the responses y_i with mean μ_i and variance σ^2, the density function for which is given

$$f(y_i) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[\frac{(y_i - \mu_i)^2}{2\sigma^2}\right]. \tag{E.4}$$

To achieve this in the "exponential family" form of Equation (E.1) requires some heroic algebraic manipulation, eventually producing

$$f(y_i; \nu_i, \phi) = \exp\left\{\frac{y_i\nu_i - \frac{\nu_i^2}{2}}{\phi} - \frac{1}{2}\left[\frac{y_i^2}{\phi} + \ln(2\pi\phi)\right]\right\} \tag{E.5}$$

with $\nu_i = g_c(\mu_i) = \mu_i; \phi = \sigma^2; a(\phi) = \phi; b(\nu_i) = \frac{\nu_i^2}{2}$; and $c(y_i, \phi) = \frac{1}{2}\left[\frac{y_i^2}{\phi} + \ln(2\pi\phi)\right]$. Here, the dispersion parameter is just the variance and the assumption of a common ϕ is just the usual assumption of constant variance

which is known as Homoscedasticity.

2. **Poisson Regression Model**: We consider a discrete response model for observations which correspond to counts which in any given time or space increment are independent and distributed according to a Poisson distribution, where we denote the Poisson distribution intensity (mean) by λ, and the density function is given by

$$f(y_i) = \frac{\lambda^{y_i} \mathrm{e}^{-\lambda}}{y_i!}. \tag{E.6}$$

Under the GLM structure, our aim is to explain the observed counts with regard to the intensity function constructed via a link function in terms of a linear basis regression. To construct our Poisson regression model we consider the canonical link function in which $g(\cdot)$ is the logarithmic transformation of the linear basis function regression. In the exponential family formulation specified the Poisson distribution is obtained by considering $\nu_i = g_c(\mu_i) = \ln\mu_i = \ln\lambda$, $a(\phi) = 1$, $b(\nu_i) = \mathrm{e}^{\nu_i}$ and $c(y, \phi) = -\log y_i!$.

Many other commonly used distributions are in the exponential family and can all be put into the form of Equation (E.1). In addition to the examples listed in Table E.2, several distributions are in the exponential family, including the beta, multinomial, Dirichlet and Pareto. Distributions that are not in the exponential family but are used for statistical modeling include the student's t and uniform distributions.

Table E.2: Constructing for Exponential Families

Family	$a(\phi)$	$b(\nu)$	$c(y, \phi)$
Gaussian	ϕ	$\frac{\nu^2}{2}$	$-\frac{1}{2}\left[\frac{y^2}{\phi} + \ln(2\pi\phi)\right]$
Binomial	$\frac{1}{n}$	$\ln(1+\mathrm{e}^{\nu})$	$\ln\binom{n}{n_{\mathbf{y}}}$
Poisson	1	e^{ν}	$-\ln y!$
Gamma	ϕ	$-\ln(-\nu)$	$\phi^{-2}\ln\left(\frac{y}{\phi}\right) - \ln y - \ln\Gamma(\phi^{-1})$
Inverse–Gamma	ϕ	$-\sqrt{-2\nu}$	$-\frac{1}{2}\left[\ln\left(\pi\phi y^3\right) + \frac{1}{\phi y}\right]$

Note: n is the number of binomial observations, and $\Gamma(\cdot)$ is the gamma function.

Bibliography

[Andrieu and Moulines, 2006] Andrieu, C. and Moulines, É. (2006). On the ergodicity properties of some adaptive mcmc algorithms. *The Annals of Applied Probability*, 16(3):1462–1505. 37

[Bardenet, 2012] Bardenet, R. (2012). *Towards adaptive learning and inference: Applications to hyperparameter tuning and astroparticle physics*. PhD thesis, Universite Paris Sud XI. 72

[Candès et al., 2008] Candès, E., Wakin, M. B., and Boyd, S. P. (2008). Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier Analysis and Applications*, 14:877–905. 94

[Cappé et al., 2007] Cappé, O., Godsill, S. J., and Moulines, E. (2007). An overview of existing methods and recent advances in sequential monte carlo. *Proceedings of the IEEE*, 95(5):899–924. 2, 7, 31

[Cappé et al., 2004] Cappé, O., Guillin, A., Marin, J.-M., and Robert, C. P. (2004). Population monte carlo. *Journal of Computational and Graphical Statistics*, 13(4). 32

[Casella and Berger, 2002] Casella, G. and Berger, R. L. (2002). *Statistical Inference*. Cengage Learning, second edition. 133

[Celeux, 1998] Celeux, G. (1998). Bayesian inference for mixtures: The label-switching problem. In *In R. Payne and P. Green, editors, COMPSTAT 98. Physica-Verlag*. 73

[Chartrand and Yin, 2008] Chartrand, R. and Yin, W. (2008). Iteratively reweighted algorithms for compressive sensing. In *Proc. ICASSP*. 94

[Chopin, 2002] Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika*, 89(3):539–552. 32

[Cornuet et al., 2012] Cornuet, J., Marin, J.-M., Mira, A., and Robert, C. P. (2012). Adaptive multiple importance sampling. *Scandinavian Journal of Statistics*, 39(4):798–812. 52

[Del Moral et al., 2006] Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436. 2, 6, 7, 28, 31, 32, 33, 36, 37, 38, 39, 41, 42, 44, 115, 116, 119, 120, 127, 129, 131

[Del Moral and Miclo, 2000] Del Moral, P. and Miclo, L. (2000). Branching and interacting particle systems approximations of feynman-kac formulae with applications to non-linear filtering. *Seminaire de Probabilites XXXIV, Lecture notes in Mathematics*, pages 1–145. 36

[Denison et al., 2002] Denison, D. G. T., Holmes, C. C., Mallick, B. K., and Smith, A. F. M. (2002). *Bayesian Methods for Nonlinear Classification and Regression*. Wiley. 97

[Djuric and Chun, 2002] Djuric, P. and Chun, J.-H. (2002). An mcmc sampling approach to estimation of nonstationary hidden markov models. *IEEE Transactions on Signal Processing*, 50(5):1113–1123. 2, 6, 23

[Dobigeon et al., 2014] Dobigeon, N., Tourneret, J.-Y., Richard, C., Bermudez, J. C. M., McLaughlin, S., and Hero, A. O. (2014). Nonlinear unmixing of hyperspectral images: Models and algorithms. *IEEE Signal Processing Magazine*, 31(1):82–94. 2, 6, 23

[Doucet and De Freitas, 2001] Doucet, A. and De Freitas, N. (2001). *Sequential Monte Carlo methods in practice*, volume 1. Springer. 3, 15, 31, 34

[Doucet and Johansen, 2009] Doucet, A. and Johansen, A. M. (2009). A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 12:656–704. 2, 7, 31

[Doucet and Wang, 2005] Doucet, A. and Wang, X. (2005). Monte Carlo methods for signal processing: a review in the statistical signal processing context. *IEEE Signal Processing Magazine*, 22(6):152–170. 2, 6, 23

[Fan and Li, 2001] Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Amer. Statist. Assoc.*, 96:1348–1360. 93

[Frank and Friedman, 1993] Frank, I. E. and Friedman, J. H. (1993). A statistical view of some chemometrics regression tools. *Technometrics*, 35(2):109–135. 93

[Friedman et al., 2010a] Friedman, J., Hastie, T., and Tibshirani, R. (2010a). http://arxiv.org/abs/1001.0736. *arXiv:1001.0736 [math.ST]*. 117

[Friedman et al., 2010b] Friedman, J., Hastie, T., and Tibshirani, R. (2010b). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, 33(1). 98

[Friel and Wyse, 2012] Friel, N. and Wyse, J. (2012). Estimating the evidence–a review. *Statistica Neerlandica*, 66(3):288–308. 28

[Gamerman and Lopes, 2006] Gamerman, D. and Lopes, H. F. (2006). *Markov chain Monte Carlo: stochastic simulation for Bayesian inference*. CRC Press. 23

[Gelfand and Smith, 1990] Gelfand, A. E. and Smith, A. F. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American statistical association*, 85(410):398–409. 23, 26

[Gelman et al., 2003] Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2003). *Bayesian data analysis*. CRC press. 12

[Geman and Geman, 1984] Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):721–741. 26

[Geyer, 1991] Geyer, C. J. (1991). Markov chain Monte Carlo maximum likelihood. In *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, volume 1, pages 156–163. 2, 6, 28, 29

[Gil et al., 2013] Gil, M., Alajaji, F., and Linder, T. (2013). Rényi divergence measures for commonly used univariate continuous distributions. *Information Sciences*, 249(0):124 – 131. 46, 48

[Gordon et al., 1993] Gordon, N. J., Salmond, D. J., and Smith, A. F. (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proceedings F (Radar and Signal Processing)*, volume 140, pages 107–113. IET. 22

[Gramacy et al., 2010] Gramacy, R., Samworth, R., and King, R. (2010). Importance tempering. *Statistics and Computing*, 20(1):1–7. 50, 51

[Hastings, 1970] Hastings, W. K. (1970). Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109. 24

[Hoerl and Kennard, 1970] Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67. 91

[Jasra et al., 2007] Jasra, A., Stephens, D., and Holmes, C. (2007). On population-based simulation for static inference. *Statistics and Computing*, 17(3):263–279. 2, 3, 6, 28

[Jasra et al., 2011] Jasra, A., Stephens, D. A., Doucet, A., and Tsagaris, T. (2011). Inference for lévy-driven stochastic volatility models via adaptive sequential monte carlo. *Scandinavian Journal of Statistics*, 38(1):1–22. 37, 38, 47

[Kitagawa, 1996] Kitagawa, G. (1996). Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of computational and graphical statistics*, 5(1):1–25. 22

[Kunsch, 2005] Kunsch, H. R. (2005). Recursive Monte-Carlo filters: algorithms and theoretical analysis. *The Annals of Statistics*, 33(5):1983–2021. 50

[Lee et al., 2012] Lee, A., Caron, F., Doucet, A., and Holmes, C. (2012). Bayesian Sparsity-Path-Analysis of Genetic Association Signal using Generalized t Priors. *Statistical Applications in Genetics and Molecular Biology*, 11(2). 94

[Li and Hu, 2003] Li, D. and Hu, Y. H. (2003). Energy based collaborative source localization using acoustic microsensor array. *EURASIP Journal on Applied Signal Processing*, (4):321–337. 66, 68

[Li et al., 2002] Li, D., Wong, K. D., HU, Y. H., and Sayeed, A. M. (2002). Detection, classification and tracking of targets. *IEEE Signal Processing Magazine*, 19(3):17–29. 66

[Liang and Wong, 2001] Liang, F. and Wong, W. (2001). Real parameter evolutionary Monte Carlo with applications to Bayesian mixture models. *Journal of the American Statistical Association*. 2, 6, 28, 29

[Liang and Wong, 2000] Liang, F. and Wong, W. H. (2000). Evolutionary Monte Carlo: Applications to C_p Model Sampling and Change Point Problem. *Statistica Sinica*, 10:317–342. 2, 6, 28

[Liu, 2008] Liu, J. S. (2008). *Monte Carlo strategies in scientific computing*. springer. 3

[Liu and Chen, 1998] Liu, J. S. and Chen, R. (1998). Sequential monte carlo methods for dynamic systems. *Journal of the American statistical association*, 93(443):1032–1044. 21, 22

[Masazade et al., 2010] Masazade, E., Niu, R., Varshney, P. K., and Keskinoz, M. (2010). Energy Aware Iterative Source Localization for Wireless Sensor Networks. *IEEE Transactions on Signal Processing*, 58(9):4824–4835. 66, 76

[McCullagh and Nelder, 1989] McCullagh, P. and Nelder, J. A. (1989). *Generalized linear models (Second edition)*. London: Chapman & Hall. 97

[Meinshausen and Bühlmann, 2006] Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, pages 1436–1462. 93

[Metropolis et al., 1953] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092. 24

[Neal, 1993] Neal, R. M. (1993). Probabilistic inference using markov chain monte carlo methods. 12, 23

[Neal, 2001] Neal, R. M. (2001). Annealed importance sampling. *Statistics and Computing*, 11(2):125–139. 32, 36

[Nelder and Baker, 1972] Nelder, J. A. and Baker, R. (1972). *Generalized linear models*. Wiley Online Library. 97

[Nelder and Mead, 1965] Nelder, J. A. and Mead, R. (1965). A simplex method for function minimization. *Computer Journal*, (7):308–313. 48

[Nelder and Wedderburn, 1972] Nelder, J. A. and Wedderburn, R. W. M. (1972). Generalized linear models. *Journal of the Royal Statistical Society. Series A (General)*, 135(3):370–384. 97

[Nevat et al., 2014] Nevat, I., Eger, O., Peters, G. W., and Septier, F. (2014). NEPS: "Narrowband Efficient Positioning System" for Delivering Resource Efficient GNSS Receivers. In *IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, Singapore. 69

[Niu and Varshney, 2006] Niu, R. and Varshney, P. K. (2006). Target Location Estimation in Sensor Networks With Quantized Data. *IEEE Transactions on Signal Processing*. 66, 67, 76

[O'Ruanaidh and Gerald, 1996] O'Ruanaidh, J. J. and Gerald, W. J. F. (1996). *Numerical Bayesian methods applied to signal processing*, volume 5. Springer-Verlag New York. 13

[Owen and Zhou, 2000] Owen, A. and Zhou, Y. (2000). Safe and effective importance sampling. *Journal of the American Statistical Association*, 95(449):135–143. 52

[Ozdemir et al., 2009] Ozdemir, O., Niu, R., and Varshney, P. K. (2009). Channel Aware Target Localization With Quantized Data in Wireless Sensor Networks. *IEEE Transactions on Signal Processing*, 57(3):1190–1202. 66, 69, 73

[Park and Casella, 2008] Park, T. and Casella, G. (2008). The Bayesian Lasso. *J. Amer. Statist. Assoc.*, 103(482):681–686. 94

[Polson et al., 2011] Polson, N. G., Scott, J. G., and Windle, J. (2011). The Bayesian Bridge. *arXiv.org*, stat.ME. 94

[Robert, 2004] Robert, C. (2004). *The Bayesian choice*. Springer Texts in Statistics, second edition. 18, 20, 26

[Robert, 2007] Robert, C. (2007). *The Bayesian choice: from decision-theoretic foundations to computational implementation*. Springer. 28

[Robert and Casella, 2004] Robert, C. P. and Casella, G. (2004). *Monte Carlo statistical methods*. Springer, second edition. 2, 6, 24, 37

[Ruanaidh et al., 1996] Ruanaidh, Ó., Joseph, J., and Fitzgerald, W. J. (1996). *Numerical Bayesian methods applied to signal processing*. Springer. 15

[Rubin, 1987] Rubin, D. B. (1987). A noniterative sampling/importance resampling alternative to the data augmentation algorithm for creating a few imputations when fractions of missing information are modest: The SIR algorithm. *Journal of the American Statistical Association*, 82:543–546. 21

[Rubin, 1988] Rubin, D. B. (1988). Using the SIR algorithm to simulate posterior distributions. In Bernardo, M. H., Degroot, K. M., Lindley, D. V., and Smith, A. F. M., editors, *Bayesian Statistics 3*. Oxford University Press. 21

[Septier and Delignon, 2011] Septier, F. and Delignon, Y. (2011). MCMC sampling for joint estimation of phase distortions and transmitted symbols in OFDM systems. *Digital Signal Processing*, 21(2):341–353. 2, 6, 23

[Sheng and Hu, 2005] Sheng, X. and Hu, Y. H. (2005). Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks. *IEEE Transactions on Signal Processing*, 53(1):44–53. 66, 67, 68

[Sisson et al., 2007] Sisson, S., Fan, Y., and Tanaka, M. M. (2007). Sequential monte carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 104(6):1760–1765. 31

[Stephens, 2000] Stephens, M. (2000). Dealing with label switching in mixture models. *Journal of the Royal Statistical Society, Series B*, 62:795–809. 72

[Tibshirani, 1996] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288. 92

[Tibshirani, 2011] Tibshirani, R. (2011). Regression shrinkage and selection via the lasso: a retrospective. *J. R. Stat. Soc. Series B*, 73(3):273–282. 94

[Tierney, 1994] Tierney, L. (1994). Markov chains for exploring posterior distributions. *the Annals of Statistics*, pages 1701–1728. 27

[Van Trees, 1968] Van Trees, H. L. (1968). *Detection, Estimation, and Modulation Theory, Part I.* New York: Wiley. 74

[Veach and Guibas, 1995] Veach, E. and Guibas, L. (1995). Optimally combining sampling techniques for monte-carlo rendering. In *Proc. SIGGRAPH'95*, pages 419–428. 52

[Zhou et al., 2013] Zhou, Y., Johansen, A. M., and Aston, J. A. (2013). Towards automatic model comparison: An adaptive sequential monte carlo approach. *arXiv preprint arXiv:1303.3123*. 47

[Zou, 2006] Zou, H. (2006). The adaptive lasso and its oracle properties. *J. Amer. Statist. Assoc.*, 101:1418–1429. 94