

Modèle d'îlots de particules et application en fiabilité

C. Vergé

▶ **To cite this version:**

C. Vergé. Modèle d'îlots de particules et application en fiabilité. Algorithme et structure de données [cs.DS]. ECOLE POLYTECHNIQUE, 2015. Français. NNT : . tel-01212765

HAL Id: tel-01212765
https://hal.science/tel-01212765

Submitted on 7 Oct 2015

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

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







THÈSE DE DOCTORAT DE L'ÉCOLE POLYTECHNIQUE

présentée par

Christelle VERGÉ

pour obtenir le grade de

DOCTEUR DE L'ÉCOLE POLYTECHNIQUE
Spécialité: **Mathématiques appliquées**

Modèles d'îlots de particules et leurs applications en fiabilité

Soutenue publiquement le : 15 juillet 2015
devant le jury composé de :

Pierre Del Moral	Professeur, Université New South Wales	Directeur de thèse
Emmanuel Gobet	Professeur, Ecole Polytechnique	Président du jury
Tony Lelièvre	Professeur, Ecole des Ponts ParisTech	Rapporteur
Jérôme Morio	Ingénieur de Recherche HDR, Onera	Encadrant
Eric Moulines	Professeur, Telecom ParisTech	Directeur de thèse
Wojciech Pieczynski	Professeur, Telecom SudParis	Examinateur
Bruno Tuffin	Directeur de Recherche, INRIA	Rapporteur

Directeurs de thèse:	Eric Moulines & Pierre Del Moral
Encadrant Onera:	Jérôme Morio
Encadrant CNES:	Juan Carlos Dolado Perez
Laboratoires:	Onera & Centre de Mathématiques appliquées

Remerciements

Je voudrais avant tout exprimer toute ma gratitude à mes directeurs de thèse, Éric Moulines et Pierre Del Moral, pour le temps qu'ils m'ont consacré, leur patience, les idées brillantes qu'ils ont partagées avec moi, les nombreux échanges concernant la recherche (mais pas seulement) que nous avons eus, pour tout le soutien et les conseils qu'ils m'ont apportés, et aussi pour les rencontres qu'il m'ont permis de faire.

Je remercie chaleureusement Jérôme Morio et Juan Carlos Dolado Perez qui sont à l'origine de cette thèse. Leur bienveillance, tout en me laissant une grande liberté m'a rendu cette thèse plus douce.

Je souhaite également remercier Tony Lelièvre et Bruno Tuffin de m'avoir fait l'honneur de rapporter ma thèse, ainsi qu'Emmanuel Gobet et Wojciech Pieczynski, d'avoir accepté d'être présents dans mon jury.

J'ai une pensée particulière pour mes collègues et amis doctorants de l'ONERA, avec qui j'ai partagé au quotidien mes peines et mes joies de doctorante, ainsi qu'un nombre incalculable de pauses thé, de gâteaux, etc. Parmi eux, je tiens à remercier tout particulièrement mes "co-bureaux" successifs, Damien et Loïc, ainsi qu'Ariane, Rata, Dalal, Sarah, Arthur, Yohan, Pawit et Evrard.

Je tiens à remercier mes collègues du CMAP pour leur bienveillance. Je pense en particulier à mes partenaires de salsa ou de Rythm & fight, Manon et Aymeric, qui m'ont permis d'oublier les tracas de la thèse quelques heures par semaine.

Je suis très reconnaissante aux copains doctorants de MétéoFrance, Cécile, Lucie, Antoine, de m'avoir chaleureusement accueillie un mois pendant ma thèse, ainsi qu'à tous mes passages à Toulouse.

J'ai une pensée particulière pour toutes les personnes qui ont influencé d'une manière ou d'une autre les choix que j'ai pu faire et à qui je dois de m'être lancé dans cette thèse.

Enfin, mais non des moindres, j'en profite pour remercier ma famille et mes amis pour m'avoir encouragée jusqu'au jour de ma soutenance.

Table des matières

Table des acronymes **9**

1 Introduction **15**
1.1 Présentation générale . . . 15
1.2 Structure du document . . . 19
1.3 Cadre et notations . . . 19
1.3.1 Notations générales . . . 19
1.3.2 Méthodes de Monte-Carlo . . . 20
1.3.3 Échantillonnage préférentiel . . . 21
1.3.4 Méthodes de Monte-Carlo par chaînes de Markov (MCMC) . . . 22
1.3.5 Modèles de Markov cachés . . . 24
1.3.6 Modèles de Feynman-Kac . . . 25
1.3.7 Méthodes de Monte-Carlo séquentielles (SMC) . . . 27
1.3.8 Inégalité de Hoeffding . . . 29

2 Parallélisation des méthodes de Monte-Carlo séquentielles : les îlots de particules **33**
2.1 Introduction . . . 37
2.2 Algorithm derivation . . . 39
2.3 Asymptotic analysis of the double bootstrap (B^2) algorithm . . . 42
2.4 Extensions . . . 47
2.4.1 Epsilon-bootstrap interaction . . . 47
2.4.2 Effective Sample Size interaction . . . 49
2.5 Numerical simulations . . . 52
2.6 Proofs . . . 57
2.6.1 Proof of Theorem 2.2.1 . . . 57
2.6.2 Proof of Theorem 2.3.3 . . . 57
2.6.3 Proof of Theorem 2.3.4 . . . 58
2.6.4 Proof of Theorem 2.4.1 . . . 61
2.6.5 Proof of Proposition 2.4.3 . . . 61
2.6.6 Proof of Theorem 2.4.4 . . . 62

3 Étude des propriétés de convergence des îlots de particules **63**
3.1 Introduction . . . 66
3.2 Preliminaries . . . 68
3.2.1 Weighted particle islands and archipelagos . . . 68
3.2.2 Convergence properties of archipelagos . . . 69
3.3 Main results : operations on weighted archipelagos . . . 70
3.3.1 Selection on the island level . . . 70
3.3.2 Selection on the individual level . . . 71
3.3.3 Mutation . . . 72
3.4 Applications . . . 74
3.4.1 Feynman-Kac models . . . 74
3.4.2 The double bootstrap algorithm with adaptive selection (B^2ASIL) algorithm 74
3.4.3 Long-term stability of the double bootstrap (B^2) algorithm . . . 78
3.4.4 Uniform in time convergence under the strong mixing assumption . . . 79
3.4.5 Uniform in time convergence under a local Doeblin condition . . . 80
3.5 Technical results . . . 81
3.6 Proofs . . . 84
3.6.1 Proof of Theorem 3.3.1 . . . 84
3.6.2 Proof of Theorem 3.3.2 . . . 85
3.6.3 Proof of Theorem 3.3.3 . . . 85
3.6.4 Proof of Theorem 3.3.4 . . . 86
3.6.5 Proof of Theorem 3.3.5 . . . 87
3.6.6 Proof of Theorem 3.3.6 . . . 87
3.6.7 Proof of Theorem 3.3.7 . . . 89
3.6.8 Proof of Theorem 3.3.8 . . . 90
3.6.9 Proof of Theorem 3.3.9 . . . 90
3.6.10 Proof of Corollary 3.4.5 . . . 93
3.6.11 Proof of Corollary 3.4.8 . . . 93

4 Discussions autour des propriétés de convergence des îlots de particules **95**
4.1 Additional details concerning the preservation of asymptotic normality . . . 97
4.1.1 Checking the assumptions AN2-6 in the proof of Theorem 3.3.3 . . . 97
4.1.2 Checking the assumptions AN2-6 in the proof of Theorem 3.3.6 . . . 98
4.1.3 Checking the assumptions AN3-5 in the proof of Theorem 3.3.9 . . . 99
4.2 Some additional details for the proof of Corollary 3.4.5 . . . 101
4.3 Technical result in the proof of Lemma 3.5.3 . . . 102
4.4 Analysis of the double bootstrap (B^2) predictor . . . 102
4.5 Some additional convergence result for the *double bootstrap* (B²) filter algorithm . . . 105
4.5.1 Joint central limit theorem for the B² filter algorithm . . . 105
4.5.2 Technical lemmas . . . 108
4.6 Some additional convergence result for the B² filter algorithm . . . 109
4.6.1 Joint central limit theorem for the B² filter algorithm . . . 109

4.6.2 Technical lemmas 112

5 Un exemple d'algorithme d'îlots de particules pour l'analyse d'évènements rares 115
5.1 Introduction 119
5.2 General problem 121
5.3 The Particle Markov chain Monte Carlo (PMCMC) algorithm 122
5.3.1 The Metropolis-Hastings (MH) algorithm 122
5.3.2 The splitting algorithm 123
5.3.3 The PMCMC algorithm 126
5.4 The *sequential Monte Carlo square* (SMC^2) algorithm 127
5.4.1 Principle 127
5.4.2 The *iterated batch importance sampling* (IBIS) algorithm 128
5.4.3 The SMC^2 algorithm 130
5.5 Application on different test cases 132
5.5.1 Gaussian toy case 132
5.5.2 Estimation of launch vehicle booster fallout zone 133
5.5.3 Estimation of collision probability between orbiting objects 137
5.6 Conclusion 140

Conclusion et perspectives 141

Table des figures et des tableaux 145

Acronymes

a. k. a.	also known as ... 13
B^2	*double bootstrap* ... 141
CLT	central limit theorem ... 67
CV	coefficient of variation ... 64
CMC	Crude Monte Carlo ... 119
B^{2}ASIL	*double bootstrap with adaptive selection on the island level* ... 141
ESS	*effective sample size* ... 34
FORM	first order reliability methods ... 119
SORM	second order reliability methods ... 119
IBIS	*iterated batch importance sampling* ... 115
IPS	interacting particle systems ... 37
i. i. d.	indépendantes et identiquement distribuées ... 20
i. i. d.	independent and identically distributed ... 52
MCMC	*Markov chain Monte Carlo* ... 119
MCMC	Monte-Carlo par chaînes de Markov ... 116
MSE	erreur quadratique moyenne ... 18
MH	Metropolis-Hastings ... 115
PMH	Particle Metropolis-Hastings ... 126
MSE	*mean squared error* ... 47
PMCMC	Particle Markov chain Monte Carlo ... 115
PMCMC	Monte-Carlo à chaîne de Markov particulaires ... 117
SIMCMC	*sequential interacting Markov chain Monte Carlo* ... 35
SiL	multinomial selection on the individual level ... 65
SIL	multinomial selection on the island level ... 142
SISR	échantillonnage préférentiel séquentiel avec ré-échantillonnage ... 27

SMC sequential Monte Carlo 119
SMC Monte-Carlo séquentielles 116
SMC^2 *sequential Monte Carlo square* 115
TCL théorème central limite 141
w. r. t. with respect to 119

Résumé en français Les modèles de Feynman-Kac (généralisant les modèles de Markov cachés) sont aujourd'hui très largement utilisés afin de modéliser une grande diversité de séries temporelles dans différents domaines tels que l'aéronautique, l'analyse d'évènements rares, le traitement du signal, les mathématiques financières, la biologie, etc. La complexité croissante de ces modèles a conduit au développement d'approximations via différentes méthodes de Monte-Carlo, dont les méthodes de Monte-Carlo par chaînes de Markov (MCMC) et les méthodes de Monte-Carlo séquentielles (SMC). Les méthodes SMC appliquées au filtrage particulaire sont au centre de cette thèse. Elles consistent à approcher la loi d'intérêt à l'aide d'une population de particules en interaction définies séquentiellement. De nombreux algorithmes ont déjà été développés et étudiés dans la littérature. Nous proposons des techniques de parallélisation des méthodes SMC, en considérant des sous-populations de particules appelées *îlots* qui peuvent également interagir entre elles. Nous étudions les propriétés de convergence de ces algorithmes d'*îlots de particules*. En particulier, nous démontrons un théorème central limite (TCL) et la stabilité de la variance des estimateurs induits, grâce à des inégalités de déviation exponentielle et des tableaux triangulaires définis au niveau des îlots. Nous proposons également un nouvel algorithme de type îlots de particules en interaction pour estimer la loi de paramètres aléatoires conditionnellement à la réalisation d'un évènement rare. Nous illustrons sa convergence et nous l'appliquons à deux cas critiques en aérospatiale.

Mots-clés Approximation particulaire de mesures de Feynman-Kac, modèles d'îlots de particules, calcul en parallèle, méthodes de Monte-Carlo séquentielles, filtrage.

Titre en anglais Island particle models and their application in reliability

Résumé en anglais Feynman-Kac models (which generalize hidden Markov models) are nowadays widely used as they allow to model a large variety of time series in several fields such as aeronautics, rare event analysis, signal processing, finance, biology, and so on. Different approximations based on Monte Carlo principles have been developed as *Markov chain Monte Carlo* (MCMC) and sequential Monte Carlo (SMC). In this thesis, we focus on SMC methods. They consist in approximating a targeted law through an interacting particle system sequentially defined. Numerous algorithms have been developed and studied in the literature. We propose techniques of parallelization of such SMC methods, considering subpopulations of particles referred to by us as *islands* which can also interact. We study convergence properties of these island particle algorithms. Especially, we prove a central limit theorem (CLT) and the stability of the variance, thanks to exponential deviation inequality and triangular arrays defined on the island level. We also propose a novel algorithm of interacting island particles to estimate the law of random parameters conditionally to a rare event. We illustrate its convergence and we apply it to two critical cases in aerospace.

Key words Particle approximation of Feynman-Kac flow, island particle models, parallel implementation, sequential Monte Carlo methods, filtering.

RÉSUMÉ ÉTENDU en anglais Interacting particle systems, also known as (a. k. a.) particle filters or SMC methods are known to efficiently sample from sequences of complex distributions in a wide range of applications, including rare event analysis, non-linear filtering, hidden Markov chain parameter estimation, signal processing, financial mathematics (see [Doucet et al., 2001; Douc et al., 2014b] and the references therein). These algorithms evolve, recursively and randomly in time, a sample of random draws, called *particles*, with associated importance weights. The particle cloud is updated through *selection* and *mutation* operations, where the former duplicates or eliminates, through resampling, particles with large or small importance weights, respectively, while the latter disseminates randomly the particles over the state space and updates accordingly the importance weights for further selection. SMC methods are computationally intensive, as the estimation precision depends upon the particle swarm size, which may be critical in online applications where only a limited computational power is at hand. It is natural to study techniques allowing to reduce the particle swarm size, while ensuring good estimates.

Because of the particle interaction during the selection steps, running SMC methods in a parallel architecture is not straightforward. Considerable effort has been devoted in the past decade to the design of schemes for the parallel implementation of particle filters, from the totally heuristic to mathematically well-principled approaches. The natural idea we develop in [Vergé et al., 2015a], is to parallelize the algorithm by, instead of considering a single batch of N particles, simply dividing the particle population into N_1 batches of each N_2 particles, also called individuals (i.e., $N = N_1 N_2$), where each batch is referred to as an *island*. In this framework, each island evolves according to the standard SMC scheme subjecting alternately the subpopulation to selection and mutation. Unfortunately, the division of the particle population introduces additional bias which may be of significant for moderate island sizes N_2. Thus, we proposed to reduce this bias by performing additional selection also on the *island level* by resampling the islands according to probabilities proportional to the weight averages over the different subpopulations. Selection on the island level may be performed systematically, as in the *double bootstrap* (B^2) *algorithm* or may be activated adaptively by some criterion measuring the skewness of the island weights (like for *effective sample size* (ESS) method). The latter approach will be referred to by us as the *double bootstrap with adaptive selection on the island level* (B^{2}ASIL) in [Vergé et al., 2014]. A sequence of Monte Carlo estimators is obtained by weighing up, using the island weights, the self-normalized empirical measures associated with the different particle islands. But, island interaction prevents island parallelization. We have proposed a criterion to determine when island interaction is needed. We chose to base our criterion on mean squared error. See [Vergé et al., 2015a] for further details.

Convergence, asymptotic and non asymptotic properties of SMC methods have been well studied over the past two decades, see [Del Moral, 2004] and the references therein. In [Vergé et al., 2014], we present some novel convergence results for island particle models introduced in [Vergé et al., 2015a]. In particular we establish a CLT—as the number of islands and the common size of the islands tend jointly to infinity—of the B^{2}ASIL algorithm. For this purpose we introduce a notion of *archipelagos of weighted islands* that generalizes the particle models studied in [Vergé et al., 2015a] and consider three kinds of convergence properties of such archipelagos, namely *consistency* (convergence in probability), *asymptotic normality* (convergence in distribution in terms of a CLT with rate $\sqrt{N}$), and *large deviation* (an exponential inequality of Hoeffing-type that holds uniformly over all islands). The analysis of these properties is challenging due to the strong dependence among the particles through the different operations. So, we perform single-step analyses of three kinds of operations on archipelagos, namely *selection on the island level*, *selection on the individual level* and *mutation*, and we find conditions under which the previous set of convergence properties is preserved by these operations on archipelagos. This theory allows arbitrary compositions of these operations to be straightforwardly analyzed, providing a very flexible framework covering the B^2 algorithm as a special case. The proposed proofs, which rely on limit theorems for triangular arrays of dependent random variables, are inductive. We also establish the long-term numerical stability of the B^2 algorithm by bounding its asymptotic

variance under weak and easily checked assumptions that are typically satisfied for models with non-compact state space.

A potential application of island particle models concerns safety and reliability. Indeed, it is not just evaluating a risk or a probability but estimating the law of random phenomena that leads to critical events. Some parameters of the model or density parameters of input random variables in the system, may be fixed by an experimenter. From a risk analysis point of view, it is interesting to determine the impact of such tuning of parameters on the realisation of some critic event. We developed an algorithm which belongs to the island particle models, referred to as *sequential Monte Carlo square* (SMC^2) algorithm in [Vergé et al., 2015b]. This algorithm samples from the law of parameters of a system conditionally on a rare event. SMC^2 algorithm consists in running an SMC with N_1 realisations of parameters, and for each of them, running an other SMC with N_2 particles in order to approximate the importance weights (which are often not computable) by unbiased estimators. The interesting result is that the convergence of this algorithm has been established as soon as N_1 gets large, for any value of N_2. We check this convergence on a test case and we apply SMC^2 algorithm from the estimation of launch vehicle booster fallout zone.

Chapitre 1

Introduction

Sommaire

1.1 Présentation générale 15
1.2 Structure du document 19
1.3 Cadre et notations 19
1.3.1 Notations générales 19
1.3.2 Méthodes de Monte-Carlo 20
1.3.3 Échantillonnage préférentiel 21
1.3.4 Méthodes de Monte-Carlo par chaînes de Markov (MCMC) 22
1.3.5 Modèles de Markov cachés 24
1.3.6 Modèles de Feynman-Kac 25
1.3.7 Méthodes de Monte-Carlo séquentielles (SMC) 27
1.3.8 Inégalité de Hoeffding 29

1.1 Présentation générale

L'introduction des méthodes de SMC (aussi appelé filtres particulaires) a été motivée par la résolution des équations de filtrage dans un cadre général. Les deux modèles de filtrage les plus célèbres et pour lesquels une expression analytique exacte de la suite des distributions *a posteriori* est disponible sont, les chaînes de Markov à espace d'états fini qui se résolvent grâce au filtre de Markov caché (hidden Markov model filter, voir [Stratonovich, 1960]), et le modèle linéaire gaussien qui se résout grâce au filtre de Kalman (voir [Kalman and Bucy, 1961]). Cependant, les modèles réels sont souvent complexes (non linéaires et non gaussiens, multi-dimensionnel, *etc.*) et ne peuvent pas être résolus analytiquement. Depuis les années 1960, ce problème a été largement étudié, en proposant notamment le filtre de Kalman étendu [Jazwinski, 1970; Anderson and Moore, 1979] qui approche un modèle non linéaire et non gaussien par un modèle linéaire gaussien, à l'aide de développements de Taylor locaux. Il s'agit d'une approximation locale. Pour des modèles plus complexes, il est préférable d'utiliser des approximations par méthodes de Monte-Carlo ([Handschin and Mayne, 1969; Kitagawa, 1987]). Une présentation détaillée de ces différentes méthodes se trouve dans les ouvrages [Doucet et al., 2001] et [Künsch, 2000].

Les méthodes SMC [Gordon et al., 1993] sont reconnues pour être des méthodes efficaces pour simuler des lois complexes. L'intérêt pour le filtrage particulaire n'a cessé de grandir depuis les vingt dernières années, comme en témoignent les nombreuses applications pour l'estimation d'évènements rares [Del Moral

and Garnier, 2005], l'estimation de paramètres pour des modèles de Markov cachés, les mathématiques financières, les problèmes d'assimilation de données; voir par exemple [Doucet et al., 2001; Chopin, 2002; Del Moral, 2004; Cappé and Moulines, 2005; Del Moral et al., 2012]. Les méthodes SMC sont au cœur de cette thèse. Elles consistent à faire évoluer récursivement et aléatoirement, un échantillon de variables aléatoires, appelées *particules*, auxquelles sont associés des *poids d'importance*. Le nuage de particules est mis à jour à travers des étapes de *sélection* et *mutation* successives. L'étape de sélection consiste à ré-échantillonner les particules, *i.e.* à dupliquer ou éliminer celles qui ont respectivement des grands ou faibles poids d'importance. Après cette étape, les particules ont toutes le même poids, mais il y a plusieurs copies de certaines particules et d'autres ont disparu. Leur diversité a donc été réduite et la phase de mutation vise à réparer cet effet. L'étape de mutation consiste à faire évoluer les particules sur l'espace d'état de manière aléatoire et indépendante, puis à mettre à jour leur poids d'importance en vue de la prochaine étape de sélection. Différentes étapes de sélection ont été proposées, allant du ré-échantillonnage systématique comme dans l'algorithme *bootstrap particle filter* (abrégé *bootstrap* dans ce document) introduit par [Gordon et al., 1993], au ré-échantillonnage adaptatif comme dans [Liu and Chen, 1995; Liu, 2008; Del Moral et al., 2012; Douc and Moulines, 2008; Arnaud and Le Gland, 2009].

La convergence et l'étude des propriétés asymptotiques et non asymptotiques des méthodes SMC ont été largement étudiées durant les vingt dernières années, citons par exemple [Del Moral, 2004] et les références à l'intérieur de ce livre. Plusieurs démonstrations du théorème central limite (TCL) pour les méthodes SMC ont été proposées par [Del Moral and Guionnet, 1999], puis par [Chopin, 2004] pour l'inférence bayésienne, avant que ce résultat ne soit prouvé par [Douc and Moulines, 2008] sous des hypothèses affaiblies. Dans ce dernier article, les auteurs ont défini les propriétés de consistance et de normalité asymptotique pour les particules pondérées. Ils ont montré, en s'appuyant sur les théorèmes limites pour les tableaux triangulaires, que ces propriétés se propagent à travers les différentes opérations rencontrées dans les méthodes SMC, telles que le ré-échantillonnage et l'échantillonnage préférentiel. En particulier, ils ont pu établir la loi faible des grands nombres et le TCL pour le *bootstrap* avec éventuellement une sélection adaptative.

Un inconvénient des algorithmes particulaires est qu'ils nécessitent des ordinateurs avec une capacité de mémoire importante et une grande rapidité d'exécution, car la précision des estimations et les résultats de convergence dépendent en particulier de la taille de la population de particules considérée. Il est donc naturel de chercher des techniques permettant d'accélérer l'exécution des algorithmes particulaires, tout en conservant la précision des estimations. Cet enjeu est important notamment pour les applications en ligne. Depuis l'apparition des premiers simulateurs parallèles, à la fin des années 70, différentes approches ont été proposées, certaines heuristiques d'autres mathématiquement justifiées. Le développement d'ordinateurs multiprocesseurs a conduit naturellement à l'utilisation simultanée de plusieurs processeurs [Suchard and Rambaut, 2009]. Les calculs massivement parallèle ont pu être développé grâce au passage de processeurs classiques (CPU pour *central processing unit*) aux processeurs graphiques (GPU pour *graphics processing unit*). Une difficulté du calcul parallèle est néanmoins de coordonner la communication entre les différents processeurs. Les méthodes de Monte-Carlo faisant intervenir des moyennes sur des échantillons de variables aléatoires indépendantes, il est aisé de simuler autant de particules mais réparties sur plusieurs processeurs et ensuite de mettre en commun les simulations en pondérant correctement. Concernant les méthodes SMC, bien que l'étape de mutation soit complètement parallélisable, la sélection nécessite la communication des processeurs. La parallélisation des méthodes SMC est un sujet qui intéresse différents domaines tels que le décodage de l'information dans les réseaux neuronaux [Xu et al., 2014], la bio-mathématique [Suchard and Rambaut, 2009] ou l'économétrie [Casarin et al., 2013]. [Billio et al., 2013] ont proposé d'utiliser une combinaison pondérée d'estimateurs de densité sur des données en macroéconomie. Une toolbox matlab appelée DeCo (pour *Density Combination*), implémentée sur CPU et GPU, a d'ailleurs été créée. Elle est présentée dans l'article [Casarin et al., 2013]. Durham et Geweke ont proposé une nouvelle approche pour la simulation des méthodes SMC en séparant les particules en différents groupes pour faciliter la simulation en

parallèle. Ils ont illustré leur méthode dans [Durham and Geweke, 2011] sur plusieurs exemples en économie et finance en utilisant l'extension CUDA du langage de programmation C. Ainsi, l'utilisation d'ordinateurs multi-cœur et le développement d'outils de programmation ont accéléré l'estimation par méthodes SMC dans une faible mesure au vu de l'interaction entre les particules (voir [Lee et al., 2010] pour plus de détails). Suite à ce constat, une diminution du nombre de particules en interaction dans les méthodes SMC a été envisagée, quitte à utiliser plusieurs filtres particulaires en parallèle ou en interaction. L'interaction des méthodes SMC est pour la première fois mise en œuvre dans l'article [Jasra et al., 2008], dans le but de favoriser la diversité des particules et d'améliorer l'exploration de l'espace d'état. L'idée de partitionner l'espace d'état pour les méthodes de Monte-Carlo n'est pas nouvelle, elle intervient pour l'échantillonnage stratifié (voir [Robert and Casella, 2004]) et sur des espaces stratifiés (voir [Wang and Landau, 2001]). Le principe est que plusieurs filtres particulaires sont d'abord simulés en parallèle où ils sont initialisés sur différentes régions de l'espace d'état. Quand les filtres atteignent une même région, les échantillons fusionnent et sont autorisés à interagir. Les deux points cruciaux de cet algorithme sont la manière dont l'espace est stratifié et le moment où les filtres particulaires commencent à interagir, et leurs réglages sont laissés à l'appréciation de l'expérimentateur. Dans le cas où la loi cible est un mélange de distributions, les méthodes SMC en interaction peuvent améliorer les performances d'un SMC classique. L'idée d'introduire de l'interaction à un niveau supérieur aux particules a ensuite été évoquée dans [Brockwell et al., 2010] avec l'algorithme *sequential interacting Markov chain Monte Carlo* (SIMCMC). Cet algorithme n'appartient pas à la classe des méthodes de Monte-Carlo par chaînes de Markov (abrégé MCMC, pour *Markov chain Monte Carlo*) car il n'est pas markovien. Il consiste à faire interagir des processus qui se comportent asymptotiquement comme des Metropolis-Hastings (MH) indépendants. A la différence des méthodes SMC où l'étape de correction fait intervenir un échantillonnage préférentiel, une étape de MH est utilisée pour le SIMCMC.

Le *sequential Monte Carlo square* (SMC^2) introduit par [Chopin et al., 2013] est une méthode SMC pour simuler selon la loi *a posteriori* des paramètres et des états latents dans les modèles de Markov cachés. Une approche idéale serait d'utiliser un filtre particulaire sur l'espace des paramètres (comme l'algorithme *iterated batch importance sampling* (IBIS) de [Chopin, 2002] pour l'inférence bayésienne). Mais, les poids d'importance étant la plupart du temps incalculables, ils sont remplacés par des estimateurs sans biais, obtenus pour tout paramètre fixé via un filtre particulaire auxiliaire. Le nom SMC^2 est dû à l'imbrication de deux algorithmes SMC. L'estimation n'est cependant pas en ligne car elle nécessite pour chaque proposition de paramètre de générer un filtre particulaire depuis l'instant initial jusqu'au temps de l'algorithme, afin d'estimer les poids d'importance.

Un algorithme SMC très utilisé dans le cadre des évènements rares est le *splitting*. Son principe repose sur le fait que la région menant à l'évènement rare peut être englobée dans des régions intermédiaires plus faciles à atteindre. Le *splitting* a été pour la première fois proposé en physique par [Kahn and Harris, 1951], puis des variantes sont apparues comme dans [Villén-Altamirano and Villén-Altamirano, 1994; Glasserman et al., 1999; Garvels, 2000; Cérou et al., 2006]. Citons par exemple les articles [Garvels and Kroese, 1998; L'Ecuyer et al., 2007] qui résument et comparent ces méthodes. Des preuves de convergence se trouvent dans les articles [Del Moral and Miclo, 2001; Cérou et al., 2006, 2005]. Une amélioration du splitting, le splitting adaptatif a été introduite par [Cérou and Guyader, 2007], puis analysée dans un cas idéalisé par [Cérou et al., 2012], et plus récemment par [Bréhier et al., 2014].

En conséquence, ces travaux font apparaître que les méthodes SMC ont été largement étudiées tant du point de vue théorique que pratique. L'idée de faire interagir des méthodes SMC ou de les paralléliser a été évoquée respectivement dans différents domaines. D'autre part, une approche pour estimer les lois *a posteriori* a été développée par Chopin et al. dans [Chopin et al., 2013] dans le cadre des modèles de Markov cachés, sans toutefois définir les propriétés et la performance des estimations de cette méthode. Néanmoins, aucun algorithme n'a encore été proposé pour estimer les lois *a posteriori* de paramètres sachant la réalisation d'un évènement rare.

C'est dans ce contexte que cette étude a été menée. Le présent travail consiste à définir, calibrer et analyser asymptotiquement les algorithmes de type îlots de particules qui permettent de paralléliser les méthodes SMC classiques. En particulier, cette étude a conduit au développement d'un algorithme permettant d'estimer les lois *a posteriori* de paramètres sachant la réalisation d'un évènement rare, et à établir des propriétés de cet algorithme.

La démarche a consisté dans un premier temps à développer un algorithme, appelé *double bootstrap* (B^2), basé sur des îlots de particules en interaction afin de cibler les mêmes lois de filtrage ou de prédiction qu'avec un algorithme SMC classique. Nous avons utilisé les expressions asymptotiques du biais et de la variance des estimateurs établies dans [Del Moral, 2004] dans le cadre du *bootstrap*, afin de déduire celles pour l'algorithme B^2 et celles des îlots qui évoluent en parallèle. Grâce à ces dernières, nous avons pu confirmer que l'interaction entre les îlots augmente la variance mais réduit le biais. Il faut donc faire un compromis entre les deux et nous avons pour cela choisi d'établir un critère basé sur l'erreur quadratique moyenne (MSE). Ce critère permet de déterminer, en fonction du nombre et de la taille des îlots, quand il est bénéfique de faire interagir ou de paralléliser les îlots. Ces résultats et des simulations sont rassemblés dans le chapitre 2 et ont été publié dans la revue *Statistics and Computing* (voir [Vergé et al., 2015a]).

Dans un deuxième temps, nous avons utilisé une autre méthode afin d'établir les propriétés asymptotiques des algorithmes généraux de type îlots. Pour ce faire, nous avons introduit la notion *d'archipel d'îlots pondérés* et nous avons défini des propriétés asymptotiques des îlots pondérés, à savoir la consistance, la normalité asymptotique et une inégalité de déviation exponentielle. L'analyse de ces propriétés est difficile du fait de la forte dépendance des particules à travers les différentes opérations sur les îlots. Pour pallier cette difficulté, l'idée a été de définir des tableaux triangulaires sur les îlots et d'utiliser les théorèmes limites pour les tableaux triangulaires démontrés par [Douc and Moulines, 2008]. Nous avons ainsi mis en évidence des conditions que l'on espère minimales, sous lesquelles, les différentes opérations élémentaires sur les îlots préservent ces propriétés asymptotiques. En conséquence, nous avons ainsi analysé les fluctuations des estimateurs dérivés de l'algorithme du double bootstrap avec une sélection adaptative (*double bootstrap with adaptive selection on the island level* (B^2ASIL)) au niveau des îlots. A partir de celles-ci, nous avons pu retrouver les expressions établies au chapitre 2 dans le cas particulier du B^2. Puis, nous avons établi la stabilité en temps long de l'algorithme B^2 en bornant de manière uniforme la variance des estimateurs sous de faibles conditions, typiquement vérifiées dans le cas des modèles de Markov cachés. Ces résultats sont rassemblés dans l'article [Vergé et al., 2014], et certaines preuves sont détaillées dans le papier supplémentaire [Vergé et al., 2014], soumis à la revue *Stochastic Systems*.

Enfin, nous avons introduit un analogue de l'algorithme SMC^2 dans le cadre de la fiabilité. Cet algorithme, également appelé SMC^2, permet de simuler les paramètres d'un modèle conditionnellement à la réalisation d'un évènement rare. Cette version du SMC^2 fait intervenir un algorithme SMC très utilisé dans le contexte des évènements rares, le *splitting*. Nous avons validé le SMC^2 sur un cas test et nous l'avons appliqué à deux cas critiques en aérospatial : la prédiction de position de retombée d'un étage de lanceur et le risque de collision spatiale entre un débris et un satellite. L'analyse de cet algorithme se fait en l'interprétant comme un algorithme SMC sur un espace étendu, caractéristique due aux nouvelles observations disponibles à chaque itération et qui doivent être conservées temporellement afin d'estimer les poids d'importance. Ce travail a donné lieu à un article [Vergé et al., 2015b] soumis dans la revue *Mathematics in Computer Science* et un autre article en préparation concernant l'analyse des propriétés du SMC^2.

1.2 Structure du document

Le travail de recherche présenté dans cette thèse se décompose en trois articles de revue soumis à publication, retranscrits dans leur intégralité dans les chapitres 2, 3 et 5. Chaque article est précédé d'un résumé en français mettant en relief les résultats importants et les contributions.

Le chapitre 2 introduit le formalisme des modèles d'îlots particules et différentes interactions possibles entre îlots. L'analyse asymptotique des variances et des biais permet d'établir un critère pour déterminer quand l'interaction entre les îlots est bénéfique.

L'analyse des modèles d'îlots de particules se poursuit au chapitre 3 avec une technique différente. Celle-ci permet d'établir une loi faible des grands nombres, un TCL et une inégalité de déviation exponentielle. Les démonstrations étant relativement techniques, par souci de concision et de lisibilité, seules les preuves remarquables sont conservées dans le chapitre 3. Les autres preuves sont regroupées dans le chapitre 4. Des résultats complémentaires à l'article [Vergé et al., 2014], qui donneront prochainement lieu à un article, ont également été ajoutés au chapitre 4.

Par ailleurs, le chapitre 5 présente un nouvel algorithme d'analyse d'évènement critique. Ce chapitre est sans doute le plus pratique. Des résultats plus théoriques d'analyse de cet algorithme sont en cours.

Dans la suite du présent chapitre, les notations et les théorèmes qui serviront de base aux résultats démontrés dans cette thèse seront exposés.

Chapitre 2. (Article) *On parallel implementation of sequential Monte Carlo methods: the island particle model* (C. Vergé, C. Dubarry, P. Del Moral, E. Moulines).
Article publié dans la revue Statistics and Computing. Ce travail a été présenté au workshop Weather forecast and Geophysical fluids Workshop (CIMI, Toulouse, janvier 2014). Certains résultats seront présentés à la conférence Signal Processing EUSIPCO (Nice, France, septembre 2015).

Chapitre 3. (Article) *Convergence properties of weighted particle islands with application to the double bootstrap algorithm* (C. Vergé, P. Del Moral, E. Moulines, J. Olsson).
Article soumis pour publication dans la revue Stochastic Systems. Ce travail a été présenté au workshop Advanced Monte Carlo Methods for Complex Inference Problems, (Cambridge, UK, avril 2014). Certains résultats seront présentés à la conférence Signal Processing EUSIPCO (Nice, France, septembre 2015).

Chapitre 4. (Appendice) *Supplement paper to "Convergence properties of weighted particle islands with application to the double bootstrap algorithm"* (C. Vergé, E. Moulines, J. Olsson).

Chapitre 5. (Article) *An island particle algorithm for safety analysis* (C. Vergé, J. Morio, P. Del Moral)
Article soumis pour publication dans la revue Mathematics in Computer Science. Ce travail a été présenté à la conférence Gdr Mascot-Num (École des Mines de Saint-Etienne, France, avril 2015) et à la conférence 18th INFORMS Applied Probability (Istanbul, Turquie, juillet 2015).

1.3 Cadre et notations

1.3.1 Notations générales

Nous rappelons quelques notations utilisées dans ce document. Pour $(m,n) \in \mathbb{Z}^2$ tels que $m \leq n$, on définit $[\![m,n]\!] \stackrel{\text{def}}{=} \{m, m+1, \ldots, n\} \subset \mathbb{Z}$. Les quantités $\{a_\ell\}_{\ell=m}^n$ seront remplacées par la notation sous forme vectorielle $a_{m:n} = (a_m, \ldots, a_n)$, en prenant comme convention $a_{m:n} = \varnothing$ si $m > n$.

Pour tout $N \in \mathbb{N}^*$, $\mathsf{Mult}(\{a(i)\}_{i=1}^N)$ désigne la loi de probabilité discrète définie sur un ensemble de N nombres positifs $\{a(i)\}_{i=1}^N$ éventuellement non normalisés. Ainsi, la notation $V \sim \mathsf{Mult}(\{a(i)\}_{i=1}^N)$ signifie que la variable aléatoire V prend la valeur $i \in [\![1,N]\!]$ avec la probabilité $a(i)/\sum_{i'=1}^N a(i')$.

Dans tout ce document les variables aléatoires sont définies sur un même espace de probabilité $(\Omega, \mathcal{F}, \mathbb{P})$.

Pour tout espace mesurable $(\mathsf{X}, \mathcal{X})$ on note respectivement $\mathsf{M}(\mathsf{X})$ et $\mathcal{P}(\mathsf{X}) \subset \mathsf{M}(\mathsf{X})$ les ensembles de mesures et de mesures de probabilités sur $(\mathsf{X}, \mathcal{X})$. De plus, $\mathsf{F}(\mathsf{X})$ désigne l'ensemble des fonctions mesurables à valeurs réelles définies sur $(\mathsf{X}, \mathcal{X})$ et $\mathsf{F}_\mathrm{b}(\mathsf{X}) \subset \mathsf{F}(\mathsf{X})$ désigne le sous-ensemble des fonctions bornées. Pour toute fonction $h \in \mathsf{F}_\mathrm{b}(\mathsf{X})$ la norme sup est notée $\|h\|_\infty \stackrel{\text{def}}{=} \sup_{x \in \mathsf{X}} |h(x)|$ et l'oscillation est définie par $\mathrm{osc}(h) \stackrel{\text{def}}{=} \sup_{(x,x') \in \mathsf{X}^2} |h(x) - h(x')|$. Pour $\nu \in \mathsf{M}(\mathsf{X})$ et $f \in \mathsf{F}(\mathsf{X})$, on note $\nu f \stackrel{\text{def}}{=} \int f(x)\, \nu(\mathrm{d}x)$ l'intégrale de Lebesgue de f sous ν quand elle est définie. Soit $(\mathsf{Y}, \mathcal{Y})$ un autre espace mesurable. Un noyau de transition non normalisé K de $(\mathsf{X}, \mathcal{X})$ sur $(\mathsf{Y}, \mathcal{Y})$ est une fonction $K : \mathsf{X} \times \mathcal{Y} \to \mathbb{R}_+$ telle que :

- pour tout $x \in \mathsf{X}$, $A \mapsto K(x, A)$ est une mesure de probabilité sur $(\mathsf{Y}, \mathcal{Y})$.
- pour tout $A \in \mathcal{Y}$, $x \mapsto K(x, A)$ est une fonction $\mathcal{X}$-mesurable.

Si $K(x, \mathsf{Y}) = 1$ pour tout $x \in \mathsf{X}$, K est appelé un noyau de transition (un noyau de Markov ou plus simplement un noyau). Le noyau K permet de définir deux opérateurs intégraux, le premier agissant sur les fonctions et le deuxième sur les mesures. Plus précisément, soit $f \in \mathsf{F}_\mathrm{b}(\mathsf{X})$ et $\nu \in \mathsf{M}(\mathsf{X})$ on définit la fonction mesurable

$$Kf : \mathsf{X} \ni x \mapsto \int f(y)\, K(x, \mathrm{d}y) \;,$$

et la mesure

$$\nu K : \mathcal{Y} \ni \mathsf{A} \mapsto \int K(x, \mathsf{A})\, \nu(\mathrm{d}x) \;,$$

quand les intégrales sont bien définies. Enfin, si K est le noyau défini ci dessus et L est un autre noyau de transition non normalisé défini sur $(\mathsf{Y}, \mathcal{Y})$ à valeurs dans un troisième espace mesurable $(\mathsf{Z}, \mathcal{Z})$, alors on définit le produit de K et L comme le noyau de transition non normalisé

$$KL : \mathsf{X} \times \mathcal{Z} \ni (x, \mathsf{A}) \mapsto \int K(x, \mathrm{d}y)\, L(y, \mathsf{A}) \;,$$

de $(\mathsf{X}, \mathcal{X})$ sur $(\mathsf{Z}, \mathcal{Z})$, quand toutes les quantités mises en jeu sont bien définies.

Pour tout $x \in \mathsf{X}$, δ_x désigne la masse de Dirac associée au point x.

Dans la suite de ce chapitre, nous allons introduire trois techniques de simulation qui servent de base aux algorithmes étudiés dans cette thèse et en particulier aux filtres particulaires. Il s'agit de deux méthodes particulaires (la méthode de Monte-Carlo et l'échantillonnage préférentiel) et d'une méthode de Monte-Carlo par chaînes de Markov (MCMC). Puis, nous détaillerons le principe des méthodes Monte-Carlo séquentielles (SMC).

1.3.2 Méthodes de Monte-Carlo

Les méthodes de Monte-Carlo sont des méthodes numériques utilisées pour approcher des intégrales complexes ou incalculables. Soit η une mesure de probabilité définie sur un espace mesurable $(\mathsf{X}, \mathcal{X})$. L'approximation par Monte-Carlo consiste à approcher l'intégrale

$$\eta h = \int_{\mathsf{X}} h(x) \eta(\mathrm{d}x) \;, \quad (h \in \mathsf{F}(\mathsf{X}))$$

en utilisant un échantillon de variables aléatoires $\{\xi^N(i)\}_{i=1}^N$ indépendantes et identiquement distribuées (i. i. d.) selon la loi η. La mesure empirique induite, notée η^N, est définie par $\eta^N \stackrel{\text{def}}{=} \sum_{i=1}^N \delta_{\xi^N(i)}/N$. Notons que la mesure η^N est par définition aléatoire, si bien que les résultats d'approximations seront probabilistes, comme la convergence presque sûre (notée $\xrightarrow{\text{a.s.}}$), la convergence en probabilité (notée $\xrightarrow{\mathbb{P}}$), la convergence en loi (notée $\xrightarrow{\mathcal{D}}$), *etc.*. D'après la loi des grands nombres, η^N est un estimateur sans biais de η, *i.e.* pour tout $h \in \mathsf{F}_\mathrm{b}(\mathsf{X})$,

$$\eta^N h \xrightarrow[N \to +\infty]{\text{a.s.}} \eta h \;.$$

D'après le théorème central limite (TCL),

$$\sqrt{N}(\eta^N h - \eta h) \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}(0, \sigma^2(h)) \ ,$$

où $\sigma^2(h) \stackrel{\text{def}}{=} \mathbb{V}\text{ar}\left(h(\xi^N(1)\right) = \eta(h^2) - \eta(h)^2 = \eta\{(h-\eta h)^2\}$. On peut résumer l'approximation de la loi cible η par la mesure empirique η^N, par

$$\int_{\mathsf{X}} h(x)\eta(\mathrm{d}x) \simeq \frac{1}{N}\sum_{i=1}^{N} h(\xi^N(i)) \ , \quad (h \in \mathsf{F}(\mathsf{X})) \ .$$

Il n'est pas toujours possible de simuler un échantillon de variables aléatoires directement selon la loi d'intérêt η. Nous présenterons dans la suite de ce chapitre des alternatives pour construire des approximations de la loi η, dites techniques de Monte-Carlo avancées.

1.3.3 Échantillonnage préférentiel

Considérons toujours la loi cible η définie sur l'espace mesurable $(\mathsf{X}, \mathcal{X})$. L'échantillonnage préférentiel est basé sur l'idée que dans certaines situations, il est plus approprié de simuler selon une loi auxiliaire μ. Cette méthode sert de fondement à de nombreuses techniques de simulation d'évènements rares, telles que dans les changements de probabilités par décentrage ou modification de la variance pour les vecteurs gaussiens (voir [Jourdain and Lelong, 2009]), les transformations inspirées des grandes déviations ou la transformation de Esscher pour les processus de poisson composés. Voir [Rubino et al., 2009; L'Ecuyer et al., 2007; Asmussen, 2003; Glasserman, 2004; Blanchet et al., 2009] pour une présentation de ces méthodes, et de nombreux exemples d'applications en finance, en assurance, en sûreté, *etc.*.

Supposons que la loi cible η est absolument continue par rapport à une loi instrumentale μ ($\eta \ll \mu$) et notons $w = \mathrm{d}\eta/\mathrm{d}\mu$ la dérivée de Radon- Nikodym de η par rapport à μ. La fonction w représente le *rapport de vraisemblances* et est souvent appelée *fonction d'importance*.

Ce changement de mesures de probabilités permet d'écrire, pour toute fonction positive $h \in \mathsf{F}(\mathsf{X})$,

$$\eta h = \int_{\mathsf{X}} h(x)\eta(\mathrm{d}x) = \int_{\mathsf{X}} h(x)w(x)\mu(\mathrm{d}x) = \mu(wh) \ .$$

Soit $\{\xi^N(i)\}_{i=1}^N$ un échantillon de variables aléatoires i. i. d. selon la loi μ. D'après la loi des grands nombres,

$$\frac{1}{N}\sum_{i=1}^{N} w(\xi^N(i))h(\xi^N(i)) \xrightarrow[N\to+\infty]{\text{a.s.}} \mu(wh) = \eta h \ .$$

Comme précédemment, des inégalités de déviations et un TCL peuvent être utilisés pour étudier les fluctuations de cet estimateur autour de sa moyenne. Nous référons le lecteur intéressé à la section 1.3.8 pour des rappels sur l'inégalité de Hoeffding ainsi qu'une extension souvent utilisée dans les preuves du chapitre 3.

Souvent, la mesure de probabilité η est connue à un facteur de normalisation près. La fonction d'importance w est donc également connue à une constante de normalisation près. Dans ce cas, l'estimateur auto-normalisé

$$\frac{\sum_{i=1}^{N} w(\xi^N(i))h(\xi^N(i))}{\sum_{i=1}^{N} w(\xi^N(i))}$$

permet d'approcher la quantité intégrée ηh. En effet, l'estimateur auto-normalisé est défini comme le rapport entre les deux estimateurs $N^{-1}\sum_{i=1}^{N} w(\xi^N(i))h(\xi^N(i))$ et $N^{-1}\sum_{i=1}^{N} w(\xi^N(i))$ qui convergent presque surement vers $\mu(wh) = \eta(h)\,\mu(w)$ et $\mu(w)$ respectivement.

1.3.4 Méthodes de Monte-Carlo par chaînes de Markov (MCMC)

Les méthodes de Monte-Carlo par chaînes de Markov (abrégé MCMC pour *Markov chain Monte Carlo*) consistent à approcher une loi cible η en construisant une chaîne de Markov ergodique admettant la loi cible η comme loi stationnaire. Ces méthodes ne reposent pas sur le fait de savoir simuler selon une loi d'intérêt η, mais sur le fait de construire un noyau de Markov K qui laisse invariant la loi cible η, *i.e.* tel que $\eta K = \eta$. Les deux méthodes les plus utilisées sont l'algorithme de Metropolis-Hastings (MH) ([Metropolis et al., 1953; Hastings, 1970]) et l'algorithme de Gibbs (voir [Geman and Geman, 1984; Andrieu et al., 2010; Robert and Casella, 2004] pour une présentation complète). L'algorithme de MH consiste à proposer un nouveau candidat et à l'accepter avec une probabilité assurant que le noyau soit réversible par rapport à η. Il sera détaillé dans la section 1.3.4. L'algorithme de Gibbs suggère d'actualiser les composantes de la chaîne de Markov une à une afin d'améliorer la probabilité d'acceptation.

Algorithme de MH

Le but de l'algorithme de MH est d'échantillonner selon une loi cible η dont on connait la densité également, notée η, à une constante de normalisation près, par rapport à une mesure dominante λ. Souvent, la forme trop complexe de la densité ne permet pas de calculer la constante de normalisation $\int \eta(x)\lambda(\mathrm{d}x)$. Dans ce cas, les techniques de simulation classiques, telles que la méthode de Monte-Carlo ou l'inversion de la fonction de répartition sont inefficaces. La méthode de rejet ou l'algorithme de MH ([Metropolis et al., 1953], [Hastings, 1970]) peuvent alors être utilisés. L'algorithme de MH consiste à créer une chaîne de Markov $(X_n)_{n\in\mathbb{N}}$ qui approche en un certain sens la loi η. A ce titre, cet algorithme est une méthode MCMC. La chaîne de Markov X_n est initialisée arbitrairement à l'aide d'une loi notée η_0. Puis, elle évolue selon des phases successives de proposition et d'acceptation-rejet. L'étape de proposition vise à explorer l'espace, en proposant des états selon une transition markovienne, notée Q. Comme son nom l'indique, l'étape d'acceptation-rejet consiste à accepter la proposition précédente avec une certaine probabilité, appelée *taux d'acceptation* α, si le mouvement est rejeté, la chaîne conserve l'état précédent la proposition. Pour tout $x \in \mathsf{X}$, on notera $q(x,\cdot)$ la densité de la loi $Q(x,\cdot)$ par rapport à la mesure dominante λ. L'algorithme MH procède comme suit.

Data: η_0, Q, q, n: le nombre d'itérations de l'algorithme, fixé par l'utilisateur
Result: $(X_p)_{0\leq p\leq n}$
/* Initialisation */
Simuler $X_0 \sim \eta_0$;
/* Transition */
for $p \leftarrow 0$ **to** n **do**
  Simuler $Y_{n+1} \sim Q(X_n,\cdot)$;
  Calculer la probabilité d'acceptation

$$\alpha(X_n, Y_{n+1}) \stackrel{\text{def}}{=} 1 \wedge \frac{\eta(Y_{n+1})q(Y_{n+1}, X_n)}{\eta(X_n)q(X_n, Y_{n+1})} ,$$

  avec la convention $\alpha(x,y) = 0$ si $\eta(x)q(x,y) = 0$;
  Simuler $U \sim \mathcal{U}([0,1])$;
  if $U \leq \alpha(X_n, Y_{n+1})$ **then**
    $X_{n+1} = Y_{n+1}$,
  else
    $X_{n+1} = X_n$.
  end
end

Algorithm 1: MH

La transition de MH est la composée de deux transitions markoviennes. Il s'agit d'un noyau markovien, noté K, défini pour tous $(x, z) \in \mathsf{X}^2$ par

$$K(x, \mathrm{d}z) = \alpha(x, z)\ Q(x, \mathrm{d}z) + \left(1 - \int_{\mathsf{X}} \alpha(x, y)\ Q(x, \mathrm{d}y)\right)\ \delta_x(\mathrm{d}z).$$

La transition de MH K est η-réversible donc en particulier la mesure de probabilité η est invariante par la transition de MH K, *i.e.*

$$\eta = \eta K\ . \tag{1.3.1}$$

La chaîne $(X_n)_{n\in\mathbb{N}}$ suit la loi $\eta_n \overset{\text{def}}{=} \eta_0 K^n$. Remarquons que d'après (1.3.1), à tout instant n,

$$\eta_n - \eta = \eta_0 K^n - \eta K^n = (\eta_0 - \eta) K^n\ .$$

Suite à ce constat, on peut espérer une convergence de la chaîne X_n vers la loi cible η, sous des hypothèses de contraction du noyau K. De plus, si la chaîne est irréductible, ou plus généralement si la chaîne admet une unique mesure de probabilité invariante, alors on peut démontrer que les trajectoires $(X_{n_0+1}, \cdots, X_{n_0+n})$, considérées à partir d'un temps de chauffe n_0, vérifient le théorème ergodique

$$\lim_{n\to\infty} \frac{1}{n} \sum_{p=n_0+1}^{n_0+n} h(X_p) = \eta h\ , \quad (h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})) \quad \mathbb{P} - p.s.\ .$$

L'introduction d'un "temps de chauffe" sert à ce que la chaîne X_n "oublie" sa loi initiale, et approche la loi invariante η. Pour simuler un échantillon i. i. d. de N variables aléatoires à peu près simulées selon la loi η, il faudrait considérer l'échantillon $(X_{n_0}, X_{2n_0}, \cdots X_{Nn_0})$. L'algorithme de MH est un algorithme en temps long, car la convergence a lieu quand le nombre d'itérations est grand. Il est simple à mettre en œuvre et très général puisqu'il n'y a pas de condition sur η_0 et Q (hormis qu'ils admettent des densités par rapport à une mesure de probabilité dominante). En revanche, la performance de cet algorithme dépend des choix et de l'expérience de l'utilisateur, comme pour les méthodes de Monte-Carlo. Plus la transition Q a de propriétés, plus la transition K en a également.

Les algorithmes MCMC deviennent inefficaces quand il s'agit de simuler une suite de lois d'intérêt $(\eta_i)_{0\leq i\leq n}$. En effet, il faudrait mettre en œuvre le même principe que pour le recuit simulé et changer les transitions de MH K_i telles que $\eta_i = \eta_i K_i \quad (i \in [\![1, n]\!])$ au cours de l'algorithme, tout en considérant des temps de chauffe n_i assez important pour approcher chaque loi cible $\eta_0, \eta_1, \cdots, \eta_n$. Ce procédé est schématisé sur la Figure 1.1

$$\eta_0 \xrightarrow[n_0 \text{ itérations}]{K_1} \eta_1 \xrightarrow[n_1 \text{ itérations}]{K_2} \eta_2 \xrightarrow[n_2 \text{ itérations}]{K_3} \eta_3 \cdots$$

Figure 1.1: Recuit simulé

Nous verrons dans la section 1.3.7 que les algorithmes particulaires sont plus adaptés pour approcher une suite de distributions.

Algorithme Monte-Carlo à chaîne de Markov particulaires (PMCMC)

Les méthodes MCMC ont plusieurs variantes. En particulier, lorsque la densité η n'est pas calculable mais qu'elle peut être estimée sans biais par un filtre particulaire auxilaire, l'algorithme devient l'algorithme PMCMC. Cet algorithme est une combinaison des méthodes MCMC et des méthodes SMC. Cette idée de fusionner ces deux types d'algorithmes avait déjà été utilisée par [Gilks and Berzuini, 2001]. L'algorithme PMCMC a été introduit dans l'article [Andrieu et al., 2010] pour des transitions de Gibbs. Dans la transition

de MH, le principe du Particle Markov chain Monte Carlo (PMCMC) est le même. Il suffit de remplacer les quantités η intervenant dans le rapport d'acception par des estimateurs sans biais. Cet algorithme servira de comparaison au *sequential Monte Carlo square* (SMC^2) dans le chapitre 5.

1.3.5 Modèles de Markov cachés

Les mesures de Feynman-Kac apparaissent naturellement dans les problèmes de filtrage pour les chaînes de Markov cachées. Un modèle de Markov caché (HMM pour *hidden Markov model*) est une paire de processus stochastiques à temps discret $\{(X_n, Y_n)\}_{n\geq 0}$ où la suite $(X_n)_{n\geq 0}$ est la *chaîne cachée* (souvent appelée *signal*) partiellement observée à travers les *observations* $(Y_n)_{n\geq 0}$. $(X_n)_{n\geq 0}$ est une chaîne de Markov à valeurs dans l'*espace d'état* X et les variables aléatoires $(Y_n)_{n\geq 0}$ sont indépendantes conditionnellement à $(Y_n)_{n\geq 0}$ et à valeurs dans l'*espace d'observation* Y. Pour tout $n \geq 0$, la loi de Y_n conditionnellement à la suite $(X_k)_{k\geq 0}$ ne dépend que de l'état courant X_n. Les modèles de Markov cachés sont utilisés dans une grande classe de problèmes en finance, en biologie ou en traitement du signal, voir [Cappé et al., 2005] et [MacDonald and Zucchini, 2009] et les références incluses dans ces livres, ainsi que les deux exemples décrits dans cette section.

On suppose que $(X_n)_{n\geq 0}$ est une chaîne de Markov non homogène de distribution initiale η_0 et de noyau de transition M_n. Pour tout $x \in \mathsf{X}$, soit $g(x, \cdot)$ une densité par rapport à une mesure de probabilité de référence λ. Le modèle de Markov caché est donc défini par les équations d'évolution et d'observation suivantes

$$X_{n+1}|X_n = x_n \sim M_n(x_n, \cdot) \,,$$
$$Y_n|X_n = x_n \sim g(x_n, \cdot) \,.$$

Pour toute observation y_n, on définit les *fonctions potentielles* g_n comme les vraisemblances des observations $g_n(x) \stackrel{\text{def}}{=} g(x, y_n)$.

On peut montrer en utilisant la formule de Bayes, que pour tout $h \in \mathsf{F}(\mathsf{X}^{n+1})$,

$$\mathbb{E}\left[h(X_0, \ldots, X_n) \mid Y_0, \ldots, Y_{n-1}\right] = \frac{\int \ldots \int h(x_0, \ldots, x_n) \prod_{p=0}^{n-1} g_p(x_p) \eta_0(\mathrm{d}x_0) M_0(x_0, \mathrm{d}x_1) \ldots M_{n-1}(x_{n-1}, \mathrm{d}x_n)}{\int \ldots \int \prod_{p=0}^{n-1} g_p(x_p) \eta_0(\mathrm{d}x_0) M_0(x_0, \mathrm{d}x_1) \ldots M_{n-1}(x_{n-1}, \mathrm{d}x_n)} \,. \tag{1.3.2}$$

Nous donnons ensuite deux exemples classiques de modèles linéaires gaussiens.

Modèle linéaire gaussien

Le modèle linéaire gaussien est défini par:

$$X_{p+1} = \phi X_p + \sigma_u U_p \,, \quad Y_p = X_p + \sigma_v V_p \,,$$

où $X_0 \sim \mathcal{N}\left(0, \sigma_u^2/(1-\phi^2)\right)$, $\{U_p\}_{p\geq 1}$ et $\{V_p\}_{p\geq 1}$ sont des suites indépendantes de variables aléatoires i. i. d. gaussiennes, centrées, réduites, indépendantes de X_0. Le problème de *filtrage* consiste à estimer la loi du signal caché X_n conditionnellement aux observations $Y_0, \cdots, Y_n$. Le problème de *prédiction* consiste à estimer la loi du signal caché X_n conditionnellement aux observations $Y_0, \cdots, Y_{n-1}$. Dans le cas particulier du modèle linéaire gaussien, le filtre de Kalman s'applique et permet de calculer de manière exacte les lois de filtrage et de prédiction.

Modèle de volatilité stochastique

Le modèle de volatilité stochastique (voir [Hull and White, 1987]) est défini par:

$$X_{p+1} = \alpha X_p + \sigma U_{p+1} \,, \quad Y_p = \beta \mathrm{e}^{\frac{X_p}{2}} V_p \,,$$

où $X_0 \sim \mathcal{N}\left(0, \sigma^2/(1-\alpha^2)\right)$, $\{U_p\}_{p\geq 0}$ et $\{V_p\}_{p\geq 0}$ sont des suites indépendantes de variables aléatoires i. i. d. gaussiennes, centrées, réduites, indépendantes de X_0. Pour ce modèle, il n'existe pas de technique de simulation exacte. Les algorithmes SMC ont été développés pour résoudre ce genre de problèmes.

1.3.6 Modèles de Feynman-Kac

Les modèles de Feynman-Kac sont des modèles plus généraux que les modèles de Markov cachés (voir [Doucet et al., 2001; Del Moral, 2004]), où la fonction de vraisemblance g définie sur $\mathsf{X}\times\mathsf{Y}$, est remplacée par une collection de *fonctions potentielles* g_n. Nous commençons par présenter le cadre général des modèles de Feynman-Kac, puis nous spécifierons deux cas particuliers, et enfin nous donnerons des exemples d'applications qui se modélisent de cette manière.

À une suite de noyaux de transition non normalisés $\{Q_n\}_{n\in\mathbb{N}}$ définis sur un espace mesurable $(\mathsf{X},\mathcal{X})$ et une mesure de probabilité $\eta_0\in\mathcal{P}(\mathsf{X})$, on associe la suite $\{\eta_n\}_{n\in\mathbb{N}}$ des *mesures de Feynman-Kac* définies pour tout $h\in\mathsf{F}_{\mathrm{b}}(\mathsf{X})$ et $n\in\mathbb{N}$ par

$$\eta_n h \stackrel{\text{def}}{=} \frac{\gamma_n h}{\gamma_n \mathbb{1}_{\mathsf{X}}} , \tag{1.3.3}$$

où

$$\gamma_n h \stackrel{\text{def}}{=} \int\cdots\int h(x_n)\,\eta_0(\mathrm{d}x_0)\prod_{p=0}^{n-1} Q_p(x_p,\mathrm{d}x_{p+1}) , \tag{1.3.4}$$

(avec la convention standard $\prod_{p=m}^{n} a_p = 1$ quand $m>n$). Les suites de Feynman-Kac γ_n et η_n, respectivement non normalisées et normalisées, satisfont les relations de récurrence suivantes. Pour $h\in\mathsf{F}_{\mathrm{b}}(\mathsf{X})$ et $(m,n)\in\mathbb{N}$ avec $m\le n$,

$$\gamma_n h = \gamma_m Q_m\cdots Q_{n-1}h \quad\text{et}\quad \eta_n h = \frac{\gamma_m Q_m\cdots Q_{n-1}h}{\gamma_m Q_m\cdots Q_{n-1}\mathbb{1}_{\mathsf{X}}} = \frac{\eta_m Q_m\cdots Q_{n-1}h}{\eta_m Q_m\cdots Q_{n-1}\mathbb{1}_{\mathsf{X}}} ,$$

avec la convention $Q_m\cdots Q_\ell = \mathrm{id}$ si $m>\ell$. En particulier,

$$\eta_{n+1}h = \frac{\eta_n Q_n h}{\eta_n Q_n \mathbb{1}_{\mathsf{X}}} \quad (h\in\mathsf{F}_{\mathrm{b}}(\mathsf{X}), n\in\mathbb{N}) . \tag{1.3.5}$$

Beaucoup de problèmes d'estimation peuvent être formalisés à l'aide de mesures de Feynman-Kac. Nous détaillons dans la suite les modèles de Markov cachés et des problèmes d'estimation de lois conditionnelles qui se rencontrent typiquement en analyse d'évènements rares.

Exemple 1: les modèles de Markov cachés
Reprenons les notations de la section 1.3.5. Les lois de prédiction s'expriment en fonction des mesures de Feynman-Kac. En effet, d'après les formules (1.3.2) et (1.3.4) la loi de $X_n|(Y_0=y_0,\cdots Y_{n-1}=y_{n-1})$, s'écrit $\gamma_n/\gamma_n\mathbb{1}_{\mathsf{X}}=\eta_n$. η_n représente donc la loi de prédiction de X_n conditionnellement à $Y_0,...,Y_{n-1}$, γ_n représente la loi jointe de X_n et $Y_0,...,Y_{n-1}$ et la constante de normalisation $\gamma_n(\mathbb{1}_{\mathsf{X}})$ est la vraisemblance de la séquence d'observations $Y_0,...,Y_{n-1}$. Dans ce cas, pour tout $p\in[\![0,n-1]\!]$, $Q_p(x_p,\mathrm{d}x_{p+1}) = g_p(x_p)M_p(x_p,\mathrm{d}x_{p+1})$.

Le modèle linéaire gaussien est un modèle de Feynman-Kac particulier où pour tout $p\ge 0$

$$M_p(x_p,\mathrm{d}x_{p+1}) = \frac{1}{\sqrt{2\pi}\sigma_u}\exp\left[-\frac{(x_{p+1}-\phi x_p)^2}{2\sigma_u^2}\right]\mathrm{d}x_{p+1} ,$$
$$g_p(x_p) = \frac{1}{\sqrt{2\pi}\sigma_v}\exp\left[-\frac{(y_p-x_p)^2}{2\sigma_v^2}\right] .$$

Le modèle de volatilité stochastique se modélise comme un problème de Feynman-Kac en posant pour tout $p\ge 0$

$$M_p(x_p,\mathrm{d}x_{p+1}) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left[-\frac{(x_{p+1}-\alpha x_p)^2}{2\sigma^2}\right]\mathrm{d}x_{p+1} ,$$
$$g_p(x_p) = \frac{1}{\sqrt{2\pi}\beta}\exp\left[-\frac{\beta^2 x_p + y_p^2\mathrm{e}^{-x_p}}{2\beta^2}\right] .$$

Exemple 2: L'algorithme du splitting
Soit $\{A_n\}_{n\in\mathbb{N}}$ une suite décroissante d'ensembles ($A_n \subset A_{n-1}$) et X une variable aléatoire définie sur l'espace de probabilité $(\Omega, \mathcal{F}, \mathbb{P})$. On cherche à estimer les lois η_p de la variable aléatoire X conditionnellement aux évènements A_p, Remarquons que pour tout $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$,

$$\eta_p(h) = \frac{1}{\mathbb{P}(A_p)} \int \mathbb{1}_{A_p}(x) h(x) \mathbb{P}_X(\mathrm{d}x) = \frac{\mathbb{E}\left[h(X)\mathbb{1}_{A_p}(X)\right]}{\mathbb{E}\left[\mathbb{1}_{A_p}(X)\right]} .$$

Les lois η_p vérifient une relation de récurrence. En effet, soit $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$, puisque $\mathbb{1}_{A_{p+1}}\mathbb{1}_{A_p} = \mathbb{1}_{A_{p+1}}$,

$$\begin{aligned}\eta_{p+1}h &= \frac{\mathbb{E}\left[h(X)\mathbb{1}_{A_{p+1}}(X)\right]}{\mathbb{E}\left[\mathbb{1}_{A_{p+1}}(X)\right]} = \frac{\mathbb{E}\left[h(X)\mathbb{1}_{A_{p+1}}(X)\mathbb{1}_{A_p}(X)\right]}{\mathbb{E}\left[\mathbb{1}_{A_p}(X)\right]} \times \frac{\mathbb{E}\left[\mathbb{1}_{A_p}(X)\right]}{\mathbb{E}\left[\mathbb{1}_{A_{p+1}}(X)\mathbb{1}_{A_p}(X)\right]} \\ &= \frac{\eta_p(h\mathbb{1}_{A_{p+1}})}{\eta_p(\mathbb{1}_{A_{p+1}})} .\end{aligned}$$

Soit M_p, un noyau de Markov laissant invariant la loi η_p (par exemple une transition de MH qui cible la loi η_p, cf section 1.3.4) alors $\eta_p M_p = \eta_p$, et

$$\eta_{p+1}h = \frac{\eta_p M_p(\mathbb{1}_{A_{p+1}}h)}{\eta_p M_p(\mathbb{1}_{A_{p+1}})} = \frac{\eta_p Q_p h}{\eta_p Q_p \mathbb{1}_{\mathsf{X}}} ,$$

où

$$Q_p h = M_p(\mathbb{1}_{A_{p+1}}h) \quad i.e. \quad Q_p(x,B) = \int_B M_p(x,\mathrm{d}y)\mathbb{1}_{A_{p+1}}(y) \quad (\forall B \in \mathcal{X}) .$$

Les lois conditionnelles de X sachant A_p sont donc des mesures de Feynman-Kac. La constante de normalisation $\gamma_p \mathbb{1}_{\mathsf{X}}$ est la probabilité $\mathbb{P}(A_p)$ d'atteindre l'ensemble A_p.

Les modèles de Feynman-Kac se définissent soit à l'aide d'une suite de noyaux $\{Q_n\}_{n\in\mathbb{N}}$, soit à l'aide de suites de fonctions potentielles $\{g_n\}_{n\in\mathbb{N}^*}$ et de noyaux de Markov $\{M_n\}_{n\in\mathbb{N}}$. Un cas particulier est le *modèle de Feynman-Kac mis à jour* défini lorsque pour tout $p \in \mathbb{N}^*$, et $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$,

$$Q_p h \stackrel{\text{def}}{=} M_p(g_{p+1}h) \quad i.e. \quad Q_p(x,B) = \int_B M_p(x,\mathrm{d}y) g_{p+1}(y) \quad (\forall B \in \mathcal{X}) ,$$

où $\{M_n\}_{n\in\mathbb{N}}$ est une suite de noyaux (normalisés) de Markov définis sur $(\mathsf{X}, \mathcal{X})$, η_0 est une mesure de probabilité et $\{g_n\}_{n\in\mathbb{N}^*}$ est une suite de fonctions appelées *fonctions potentielles*, définies par $g_n : \mathsf{X} \to \mathbb{R}_+^*$ pour tout $n \in \mathbb{N}^*$. Ce cas particulier couvre une large variété de modèles en probabilité et statistiques, tels que *le filtrage optimal* pour les modèles de Markov cachés [Cappé et al., 2005] et l'*analyse d'évènements rares* (voir [Del Moral and Garnier, 2005; Cérou et al., 2012] et l'exemple du splitting). Cette configuration particulière sera utilisée à différents endroits de ce manuscrit, notamment dans le chapitre 5 ou pour étudier la stabilité en temps long de la variance du *double bootstrap* (B^2) dans subsection 3.4.3.

Une alternative (dépendant du problème d'intérêt) est de définir

$$Q_p h(x) = g_p(x) M_p h(x) \quad (h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X}), x \in \mathsf{X}, p \in \mathbb{N}^*) , \tag{1.3.6}$$

avec $Q_0 = M_0$. Ce modèle est le *modèle de Feynman-Kac prédictif*. Il est étudié au chapitre 2. Les deux modèles de Feynman-Kac prédictif et mis à jour sont équivalents, et on peut facilement passer de l'un à l'autre, comme cela a été fait dans la section 4.4.

A l'exception du cas du filtrage pour des modèles linéaires, pour lesquels le *filtre de Kalman*, fournit une solution ou pour les modèles de Markov cachés à espace d'état fini, utiliser un modèle de Feynman-Kac est en pratique non trivial si les mesures $\{\gamma_n\}_{n\in\mathbb{N}}$ ou $\{\eta_n\}_{n\in\mathbb{N}}$ ne peuvent pas être calculées de manière analytique. Dans ce cas, les techniques SMC permettent d'estimer récursivement les suites de Feynman-Kac.

1.3.7 Méthodes de SMC

Dans cette partie, nous décrivons les méthodes SMC qui permettent d'estimer les lois définies en (1.3.5) par

$$\eta_{n+1}h = \frac{\eta_n Q_n h}{\eta_n Q_n \mathbb{1}_{\mathsf{X}}} \quad (h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X}), n \in \mathbb{N}) ,$$

où $\{Q_n\}_{n\in\mathbb{N}}$ est une suite de noyaux de transition non normalisés définis sur un espace mesurable $(\mathsf{X}, \mathcal{X})$ et $\eta_0 \in \mathcal{P}(\mathsf{X})$. Supposons qu'il existe une suite de noyaux de transition (normalisés) $R_n : \mathsf{X}_n \times \mathcal{X}_{n+1} \to \mathbb{R}_+$ tels que $Q_n(x_n, \cdot) \ll R_n(x_n, \cdot)$ pour tout $x_n \in \mathsf{X}_n$, notons alors w_n les dérivées de Radon-Nikodym définies pour tout $n \in \mathbb{N}$ par

$$w_n(x_n, x_{n+1}) \stackrel{\text{def}}{=} \frac{\mathrm{d}Q_n(x_n, \cdot)}{\mathrm{d}R_n(x_n, \cdot)}(x_{n+1}) \quad ((x_n, x_{n+1}) \in \mathsf{X}_n \times \mathsf{X}_{n+1}) .$$

Les applications w_n sont appelées *fonctions d'importance*.

Les méthodes SMC produisent une suite d'échantillons de variables aléatoires pondérées, appelées *particules*. L'échantillon à l'instant n est noté $\{(\xi_n^N(i), \omega_n^N(i))\}_{i=1}^N$ où N est le nombre de particules. $\xi_n^N(i)$ est la $i^{\text{ème}}$ particule appartenant à l'espace d'état X sur lequel la loi cible est définie et $\omega_n^N(i)$ est le poids, positif, associé à cette particule. Soit $n \in \mathbb{N}$. Si l'échantillon de particules pondérées $\{(\xi_n^N(i), \omega_n^N(i))\}_{i=1}^N$ approche la loi η_n, il est possible de construire un échantillon de particules $\{(\xi_{n+1}^N(i), \omega_{n+1}^N(i))\}_{i=1}^N$, qui approche la loi η_{n+1} en utilisant une étape d'échantillonnage préférentiel décrit en section 1.3.3. En itérant cette étape, on obtient l'algorithme d'échantillonnage préférentiel séquentiel (SIS pour *sequential importance sampling*) qui construit une séquence d'échantillons de particules définis récursivement et qui approchent les lois η_n.

```
/* Initialisation                                                    */
for i ← 1 to N do
    ξ_0^N(i) ~ η_0;
    ξ_1^N(i) ~ R_0(ξ_0^N(i), ·) ;
    ω_1^N(i) ← w_0(ξ_0^N(i), ξ_1^N(i));
end
for p ← 1 to n − 1 do
    /* Échantillonnage préférentiel                                  */
    for i ← 1 to N do
        Simuler indépendamment ξ_{p+1}^N(i) ~ R_p(ξ_p^N(i), ·);
        Mettre à jour des poids d'importance : ω_{p+1}^N(i) ← ω_p^N(i) × w_p(ξ_p^N(i), ξ_{p+1}^N(i));
    end
end
```

Algorithm 2: Échantillonnage préférentiel séquentiel (SIS)

Les premières méthodes d'échantillonnage préférentiel séquentiel ont été développées dès le milieu des années 1950 ([Rosenbluth and Rosenbluth, 1955]) pour simuler numériquement des polymères (*self-avoiding random walk*) à longues chaines. Cet algorithme utilise les observations uniquement lors du calcul des poids d'importance, mais ne les prend pas en compte lors de l'évolution des particules sur l'espace d'état. Il est bien connu que les poids d'importance dégénèrent dans certaines situations, c'est pourquoi [Gordon et al., 1993] ont proposé d'ajouter une étape de ré-échantillonnage des particules à chaque itération : cela a donné l'algorithme d'échantillonnage préférentiel séquentiel avec ré-échantillonnage (SISR) aussi appelé *bootstrap*.

/* Initialisation */
for $i \leftarrow 1$ **to** N **do**
  $\xi_0^N(i) \sim \eta_0$;
  $\xi_1^N(i) \sim R_0(\xi_0^N(i), \cdot)$;
  $\omega_1^N(i) \leftarrow w_0(\xi_0^N(i), \xi_1^N(i))$;
end
for $p \leftarrow 1$ **to** $n-1$ **do**
  /* Sélection (ré-échantillonnage) */
  Simuler $\{I_p^N(i)\}_{i=1}^N \sim_{i.i.d.} \mathsf{Mult}(\{\omega_p^N(k)\}_{k=1}^N)$;
  /* Mutation (échantillonnage préférentiel) */
  for $i \leftarrow 1$ **to** N **do**
    Simuler indépendamment $\xi_{p+1}^N(i) \sim R_p(\xi_p^N(I_p^N(i)), \cdot)$;
    Mettre à jour des poids d'importance : $\omega_{p+1}^N(i) \leftarrow w_p(\xi_p^N(I_p^N(i)), \xi_{p+1}^N(i))$;
  end
end

Algorithm 3: Bootstrap

Des variantes de ré-échantillonnage évitant de ré-échantillonner de façon systématique les systèmes de particules ont été introduites comme l'*effective sample size* (ESS) (voir [Liu and Chen, 1995; Liu, 2008; Douc and Moulines, 2008; Arnaud and Le Gland, 2009; Del Moral et al., 2012]) ou l'ε-bootstrap (voir [Del Moral, 2004]). Elles seront détaillées et étudiées au chapitre 2.

Ces algorithmes étant inspirés des algorithmes d'optimisation évolutionnaires, la terminologie des algorithmes génétiques est généralement employée. L'étape de ré-échantillonnage est également appelée *sélection* et l'étape d'échantillonnage préférentiel est également appelée *mutation*.

Les méthodes SMC sont des *méthodes d'estimation en ligne*, ce qui signifie que les estimations sont produites sans avoir besoin de conserver toutes les observations passées et sans devoir toutes les utiliser pour produire une seule estimation. Ce genre de procédure est très utile dans le cas de grands ensembles de données (pour lesquelles le parcours de toutes les observations serait trop coûteux) ou dans des situations d'estimation en ligne sur des flux de données (pour lesquelles la mise à jour d'un paramètre et le calcul de toutes les observations associées est très coûteux en temps de calcul).

Les propriétés de ces algorithmes ont été largement étudiées. Nous rappelons dans la suite de cette section certaines propriétés du *bootstrap*. Soit $n \in \mathbb{N}$ et $h \in \mathsf{F}_\mathrm{b}(\mathsf{X})$. Notons η_n^N l'estimateur de η_n défini par

$$\eta_n^N h \stackrel{\mathrm{def}}{=} \frac{\sum_{i=1}^N \omega_n^N(i) h(\xi_n^N(i))}{\sum_{i=1}^N \omega_n^N(i)} .$$

On parle de consistance lorsque $\eta_n^N h \xrightarrow[N\to+\infty]{\mathbb{P}} \eta_n h$ et de convergence faible lorsque $\eta_n^N h \xrightarrow[N\to+\infty]{\text{a.s.}} \eta_n h$. Une preuve de la convergence faible se trouve dans le livre [Del Moral, 2004] et une preuve simplifiée de la consistance se trouve dans l'article [Douc and Moulines, 2008]. $\eta_n^N h$ est un estimateur biaisé de $\eta_n h$. Des contrôles du biais ont été obtenus, voir par exemple [Olsson and Rydén, 2004; Del Moral et al., 2012].

Plusieurs démonstrations du TCL pour les méthodes SMC ont été proposées. La première preuve a été proposée par [Del Moral and Guionnet, 1999]. L'argument a été considérablement simplifié par [Chopin, 2004] puis par [Künsch, 2005]. Cette méthode de preuve a été raffinée et étendue dans [Douc and Moulines, 2008] où la preuve de normalité asymptotique est ramenée à des résultats classiques de théorème limite pour des tableaux de martingales.

Dans ce contexte, le TCL prend la forme:

$$\sqrt{N}\left(\frac{\sum_{i=1}^N \omega_n^N(i) h(\xi_n^N(i))}{\sum_{k=1}^N \omega_n^N(k)} - \eta_n(h)\right) \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}(0, \sigma_n^2(h)) ,$$

où $\sigma_n^2(h)$ est une quantité positive désignant la variance asymptotique. Dans le cas du *bootstrap filter*, elle est définie par

$$\sigma_n^2(h) \stackrel{\text{def}}{=} \sum_{\ell=0}^{n-1} \frac{\eta_\ell R_\ell\{w_\ell^2 Q_{\ell+1}\cdots Q_{n-1}(h-\eta_n h)^2\}}{(\eta_\ell Q_\ell \cdots Q_{n-1}\mathbb{1}_{\mathsf{X}})^2} \ .$$

On retrouve ce résultat au [chapitre 3, Equation 3.4.5], comme conséquence du TCL établi pour le B^2.

1.3.8 Inégalité de Hoeffding

Soit X une variable aléatoire. On veut estimer la concentration de la loi de X autour de son espérance. Pour une suite de variables aléatoires (X_n) i. i. d. admettant une espérance m et une variance σ^2, on sait d'après le TCL que

$$\sqrt{N}\left(\frac{1}{N}\sum_{i=1}^{N} X_i - m\right) \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}(0,\sigma^2) \ .$$

Le TCL donne un contrôle asymptotique des déviations autour de la moyenne de l'ordre de $\sigma/\sqrt{N}$. L'inégalité de Hoeffding démontrée au théorème 1.3.2 précise quand à elle des contrôles non asymptotiques (*i.e.* valables quel que soit N). C'est une inégalité de déviation exponentielle au sens où elle donne un contrôle de la déviation d'une somme de variables aléatoires par rapport à leur moyenne sans aucune hypothèse sur la loi des variables aléatoires. La borne obtenue est de forme exponentielle. Une condition nécessaire est que les variables aléatoires soient bornées. On remarque que la borne obtenue ne dépend pas de la variance des variables aléatoires considérées. Cela signifie qu'on obtient le même majorant pour deux variables ayant le même support borné mais avec des variances différentes. Les inégalités de Bennett et de Bernstein seront préférées pour des bornes de concentrations plus fines. Néanmoins, pour notre étude au chapitre 3, l'inégalité de Hoeffding sera suffisante. Nous rappelons son énoncé et une preuve dans le reste de cette section, ainsi qu'une extension à un quotient de variables aléatoires.

Lemma 1.3.1. *Soit Z une variable aléatoire centrée et bornée. Soient $(a,b) \in \mathbb{R}^2$, tels que $a < b$ et $\mathbb{P}(a \le Z \le b) = 1$. Alors pour tout réel $s > 0$,*

$$\mathbb{E}\left[e^{sZ}\right] \le e^{\frac{s^2(b-a)^2}{8}} \ .$$

Proof. Soit $s > 0$ et $z \in [a,b]$. On peut toujours écrire $z = (1-\lambda)\,a + \lambda\, b$, en posant $\lambda = \dfrac{z-a}{b-a}$. Notons que $\lambda \in]0,1[$. Par convexité de la fonction $(z \mapsto e^{sz})$, on a

$$e^{sz} = e^{s((1-\lambda)a+\lambda b)} \le (1-\lambda)e^{sa} + \lambda e^{sb} \ .$$

Puis,

$$e^{sz} \le \frac{b-z}{b-a}e^{sa} + \frac{z-a}{b-a}e^{sb} \ .$$

Ainsi, en utilisant le fait que la variable aléatoire Z est centrée, on obtient la majoration suivante

$$\begin{aligned}
\mathbb{E}\left[e^{sZ}\right] &\le \mathbb{E}\left[\frac{b-Z}{b-a}\right]e^{sa} + \mathbb{E}\left[\frac{Z-a}{b-a}\right]e^{sb} \\
&\le \frac{b}{b-a}e^{sa} - \frac{a}{b-a}e^{sb} \\
&\le (1-\theta)e^{sa} + \theta e^{sb} = (1-\theta+\theta e^{s(b-a)})e^{sa}, \text{ où } \theta \stackrel{\text{def}}{=} -a/(b-a) \\
&\le (1-\theta+\theta e^{s(b-a)})e^{-s(b-a)\theta} \\
&= (1-\theta+\theta e^{u})e^{-u\theta} \text{ avec } u \stackrel{\text{def}}{=} s(b-a) \\
&= e^{\phi(u)}, \text{ où } \phi(u) \stackrel{\text{def}}{=} -\theta u + \ln(1-\theta+\theta e^{u})
\end{aligned}$$

Le développement de Taylor pour ϕ implique qu'il existe $v \in]0, u[$ tel que

$$\phi(u) = \phi(0) + u\phi'(0) + \frac{u^2}{2}\phi''(v).$$

Or, $\phi(0) = \phi'(0) = 0$ et en utilisant le fait que $\alpha(1-\alpha) \leq 1/4$ pour tout $\alpha \in \mathbb{R}$, on a pour tout $v > 0$, $\phi''(v) \leq 1/4$. D'où, $\phi(u) \leq u^2/8$. On conclut en remplaçant u par sa valeur $s(b-a)$. □

Theorem 1.3.2. *(Inégalité de Hoeffding) Soit $X_1, X_2, ..., X_n$ une suite de variables aléatoires indépendantes bornées telles que pour tout $i \in [\![1, n]\!]$, $X_i \in [a_i, b_i]$ avec probabilité 1. Posons $S_n \overset{\text{def}}{=} \sum_{i=1}^n X_i$. Alors pour tout $\varepsilon > 0$, on a*

$$\mathbb{P}(S_n - \mathbb{E}[S_n] \geq \varepsilon) \leq \exp\left(-\frac{2\varepsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right),$$

$$\mathbb{P}(S_n - \mathbb{E}[S_n] \leq -\varepsilon) \leq \exp\left(-\frac{2\varepsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right),$$

$$\mathbb{P}(|S_n - \mathbb{E}[S_n]| \geq \varepsilon) \leq 2\exp\left(-\frac{2\varepsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right).$$

Proof. Prouvons la première affirmation, les deux suivantes étant de simples conséquences de la première.

$$\begin{aligned}
\mathbb{P}(S_n - \mathbb{E}[S_n] \geq \varepsilon) &= \mathbb{P}(e^{s(S_n - \mathbb{E}[S_n])} \geq e^{s\varepsilon}) \quad \text{pour tout } s > 0 \\
&\leq e^{-s\varepsilon}\mathbb{E}\left[e^{s\sum_{i=1}^n (X_i - \mathbb{E}[X_i])}\right] \quad \text{d'après l'inégalité de Markov} \\
&\leq e^{-s\varepsilon}\prod_{i=1}^n \mathbb{E}\left[e^{s(X_i - \mathbb{E}[X_i])}\right] \quad \text{par indépendance} \\
&\leq e^{-s\varepsilon}\prod_{i=1}^n e^{\frac{s^2(b_i-a_i)^2}{8}}, \quad \text{grâce au Lemma 1.3.1} \\
&\leq e^{-s\varepsilon}\ e^{\frac{s^2}{8}\sum_{i=1}^n (b_i-a_i)^2} = e^{\psi(s)}, \quad \text{pour tout } s > 0,
\end{aligned}$$

où on a posé $\psi(s) \overset{\text{def}}{=} -s\varepsilon + s^2/8\sum_{i=1}^n (b_i - a_i)^2$. Sa dérivée s'annule en $s_0 = 4\varepsilon/\sum_{i=1}^n (b_i - a_i)^2$ et la dérivée seconde $\psi''(s) = \sum_{i=1}^n (b_i - a_i)^2/4$ est positive. Donc la fonction ψ est convexe, et atteint son minimum $-2\varepsilon^2/\sum_{i=1}^n (b_i - a_i)^2$ en s_0. □

Remark 3. Si on remplace l'hypothèse d'indépendance des $(X_i)_{i=1}^n$ par le fait qu'ils ont la même loi, le Theorem 1.3.2 est encore vrai.
En effet, notons m leur espérance commune et a et b les bornes. Ainsi, $na \leq S_n \leq nb$ puis $n(a-m) \leq S_n - \mathbb{E}[S_n] \leq n(b-m)$.

$$\begin{aligned}
\mathbb{P}(S_n - \mathbb{E}[S_n] \geq \varepsilon) &= \mathbb{P}(e^{s(S_n - \mathbb{E}[S_n])} \geq e^{s\varepsilon}) \quad \text{pour tout } s > 0 \\
&\leq e^{-s\varepsilon}\ \mathbb{E}\left[e^{s(S_n - \mathbb{E}[S_n])}\right] \\
&\leq e^{-s\varepsilon}\ e^{\frac{s^2 n^2 (b-a)^2}{8}}, \quad \text{en appliquant le Lemma 1.3.1 à } S_n \\
&\leq e^{-\frac{2\varepsilon}{n(b-a)^2}} \text{(en suivant les mêmes étapes que dans la preuve du Theorem 1.3.2).}
\end{aligned}$$

Corollary 1.3.4. *Soient $X_1, X_2, ..., X_n$ des variables aléatoires bornées, de même loi, telles que $X_1 \in [a, b]$ avec probabilité 1. Alors pour tout $\varepsilon > 0$,*

$$\mathbb{P}\left(\sum_{i=1}^n (X_i - \mathbb{E}[X_i]) \geq \varepsilon\right) \leq \exp\left(-\frac{2\varepsilon^2}{n(b-a)^2}\right).$$

Nous rappelons ici un lemma très utile pour obtenir une inégalité de type Hoeffding pour un quotient de variables aléatoires. Ce lemme sera notamment utilisé dans le chapitre 3.

Lemma 1.3.5. *[Douc et al., 2011, Extension of lemma 4] Soient A_n, B_n, a, b des variables aléatoires définies sur le même espace de probabilité. Supposons qu'il existe des constantes positives β, B, C et M telles que*

(i) $\left|\dfrac{A_n}{B_n}\right| \leq M$ *et* $b \geq \beta$ $\mathbb{P}-p.s.$

(ii) Pour tout $\varepsilon > 0$ *et tout* $n \geq 0$, $\mathbb{P}(|B_n - b| > \varepsilon) \leq 2e^{-C_1 n\varepsilon^2}$

(iii) Pour tout $\varepsilon > 0$ *et tout* $n \geq 0$, $\mathbb{P}(|A_n - a| > \varepsilon) \leq 2e^{-C_2 n(\varepsilon/M)^2}$

Alors

$$\mathbb{P}\left(\left|\frac{A_n}{B_n} - \frac{a}{b}\right| \geq \varepsilon\right) \leq 4\exp\left(-n(C_1 \wedge C_2)\frac{\varepsilon^2\beta}{2M}\right)$$

Proof. Écrivons la différence $\dfrac{A_n}{B_n} - \dfrac{a}{b} = \dfrac{1}{b}\dfrac{A_n}{B_n}(b - B_n) + \dfrac{1}{b}(A_n - a)$. Alors

$$\left|\frac{A_n}{B_n} - \frac{a}{b}\right| \leq \frac{1}{\beta}M|b - B_n| + \frac{1}{\beta}|A_n - a| \,,$$

puis

$$\begin{aligned}\left\{\left|\frac{A_n}{B_n} - \frac{a}{b}\right| \geq \varepsilon\right\} &\subset \left\{\frac{1}{\beta}M|b - B_n| + \frac{1}{\beta}|A_n - a| \geq \varepsilon\right\} \\ &\subset \left\{|b - B_n| \geq \frac{\varepsilon\beta}{2M}\right\} \cup \left\{|A_n - a| \geq \frac{\varepsilon\beta}{2}\right\} .\end{aligned}$$

On conclut que

$$\mathbb{P}\left(\left|\frac{A_n}{B_n} - \frac{a}{b}\right| \geq \varepsilon\right) \leq \mathbb{P}\left(|B_n - b| \geq \frac{\varepsilon\beta}{2M}\right) + \mathbb{P}\left(|A_n - a| \geq \frac{\varepsilon\beta}{2}\right) .$$

□

Chapitre 2

Parallélisation des méthodes de Monte-Carlo séquentielles : les îlots de particules

Sommaire

2.1 Introduction 37
2.2 Algorithm derivation 39
2.3 Asymptotic analysis of the double bootstrap (B^2) algorithm 42
2.4 Extensions 47
2.4.1 Epsilon-bootstrap interaction 47
2.4.2 Effective Sample Size interaction 49
2.5 Numerical simulations 52
2.6 Proofs 57
2.6.1 Proof of Theorem 2.2.1 57
2.6.2 Proof of Theorem 2.3.3 57
2.6.3 Proof of Theorem 2.3.4 58
2.6.4 Proof of Theorem 2.4.1 61
2.6.5 Proof of Proposition 2.4.3 61
2.6.6 Proof of Theorem 2.4.4 62

Les méthodes de Monte-Carlo séquentielles (abrégé SMC, pour *sequential Monte Carlo*) sont utilisées depuis plusieurs années dans différents domaines tels que la fiabilité, le filtrage non linéaire, la théorie du signal, les mathématiques financières, etc. Elles utilisent un échantillon de variables aléatoires, appelées *particules* en interaction, qui approchent séquentiellement une suite de lois d'intérêts. Un poids d'importance est associé à chaque particule. Ce nuage de particules est mis à jour selon des phases successives de *sélection* et de *mutation*. La sélection consiste à dupliquer ou éliminer, via un ré-échantillonnage, les particules ayant un grand ou faible poids d'importance. La mutation fait évoluer les particules sur l'espace d'état et met à jour les poids d'importance pour la prochaine étape de sélection.

Une préoccupation constante est d'accélérer l'exécution des algorithmes afin de pouvoir simuler des lois plus complexes. C'est aussi un enjeu important pour les applications en ligne. En effet, la performance des méthodes Monte-Carlo séquentielles (SMC) croit avec le nombre de particules considérées, ce qui peut être critique pour des applications en ligne où la puissance de calcul est limitée. Dans ce cas, il est naturel d'étudier des techniques de réduction du nombre de particules, tout en assurant une bonne qualité d'estimation.

Depuis l'apparition à la fin des années 70 des premiers simulateurs parallèles, différentes approches ont été proposées, certaines heuristiques d'autres mathématiquement justifiées.

Le développement d'ordinateurs multiprocesseurs a conduit naturellement à l'utilisation simultanée de plusieurs processeurs [Suchard and Rambaut, 2009]. Par exemple, la commande *parfor* en Matlab permet de lancer des simulations en parallèle, ce qui est utile sur un ordinateur multi-cœur ou lorsqu'on a accès à un *cluster* d'ordinateurs. Le passage de processeurs classiques (CPU pour central processing unit) aux processeurs graphiques (GPU pour graphics processing unit) a permis de développer les calculs massivement parallèles. Une difficulté du calcul parallèle est de coordonner la communication entre les différents processeurs. Les méthodes de Monte-Carlo faisant intervenir des moyennes sur des échantillons de variables aléatoires indépendantes, il est aisé de simuler autant de particules mais réparties sur plusieurs processeurs et ensuite de mettre en commun les simulations en pondérant correctement. Concernant les méthodes SMC, bien que l'étape de mutation soit complètement parallélisable, la sélection nécessite la communication des processeurs.

La parallélisation des méthodes SMC est un sujet qui intéresse différents domaines tels que le décodage de l'information dans les réseaux neuronaux [Xu et al., 2014], la bio-mathématique [Suchard and Rambaut, 2009] ou l'économétrie [Casarin et al., 2013]. [Billio et al., 2013] ont proposé d'utiliser une combinaison pondérée d'estimateurs de densité sur des données en macroéconomie. Une toolbox matlab appelée DeCo (pour Density Combination) implémentée sur CPU et GPU, a d'ailleurs été créée. Elle est présentée dans l'article [Casarin et al., 2013]. Durham et Geweke ont proposé une nouvelle approche pour la simulation des méthodes SMC en séparant les particules en différents groupes pour faciliter la simulation en parallèle. Ils ont illustré leur méthode dans [Durham and Geweke, 2011] sur plusieurs exemples en économie et finance en utilisant l'extension CUDA du langage de programmation C.

Ainsi, l'utilisation d'ordinateurs multi-cœur et le développement d'outils de programmation ont accéléré l'estimation par méthodes SMC dans une faible mesure au vu de l'interaction entre particules (voir [Lee et al., 2010] pour plus de détails).

Suite à ce constat, deux aspects ont été étudiés. Le premier est l'utilisation d'un ré-échantillonnage adaptatif. Des variantes de sélection dans les méthodes SMC ont été proposées, citons par exemple l'*effective sample size* (ESS) introduit par Liu et Chen dans [Liu and Chen, 1995] (voir aussi [Liu, 2008; Del Moral et al., 2012; Douc and Moulines, 2008; Arnaud and Le Gland, 2009]) ou encore l'ε-bootstrap présenté dans [Del Moral, 2004]. Il y a essentiellement deux motivations au fait de ne pas ré-échantillonner systématiquement les particules: l'interaction entre les particules augmente la variance du système de particules, et empêche la parallélisation. En revanche, l'absence d'interaction rend les algorithmes instables et non-convergents. La sélection ESS qui ré-échantillonne les particules uniquement quand il y a trop de diversité au sein de leurs poids semble être un bon compromis et est par exemple utilisée dans l'algorithme αSMC, introduit

dans l'article [Whiteley et al., 2013] puis étudié dans [Lee and Whiteley, 2014]. Cet algorithme généralise différents algorithmes tels que le *sequential importance sampling*, le *bootstrap particle filter* et l'*adaptive resampling particle filter*.

Le second aspect est la diminution du nombre de particules en interaction dans les méthodes SMC, quitte à utiliser plusieurs filtres particulaires parallèles ou en interaction. L'interaction des méthodes SMC est pour la première fois mise en œuvre dans l'article [Jasra et al., 2008], dans le but de favoriser la diversité des particules et d'améliorer l'exploration de l'espace d'état lorsque les noyaux de transition markoviens sont peu mélangeants. L'idée de partitionner l'espace d'état pour les méthodes de Monte-Carlo n'est pas nouvelle, elle intervient pour l'échantillonnage stratifié (voir [Robert and Casella, 2004]) et pour des espaces stratifiés (voir [Wang and Landau, 2001]). Le principe est que plusieurs filtres particulaires sont d'abord simulés en parallèle, initialisés sur différentes régions de l'espace d'état. Quand les filtres atteignent une même région, les échantillons fusionnent et sont autorisés à interagir. Les deux points cruciaux de cet algorithme sont la manière dont l'espace est stratifié et le moment où les filtres particulaires commencent à interagir, et leurs réglages sont laissés à l'appréciation de l'expérimentateur. Dans le cas où la loi cible est un mélange de distributions, les méthodes SMC en interaction peuvent améliorer les performances d'un SMC classique. L'idée d'introduire de l'interaction à un niveau supérieur aux particules a ensuite été évoquée dans [Brockwell et al., 2010] avec l'algorithme *sequential interacting Markov chain Monte Carlo* (SIMCMC). Cet algorithme n'appartient pas à la classe des méthodes de Monte-Carlo par chaînes de Markov (abrégé MCMC, pour Markov chain Monte Carlo) car il n'est pas markovien. Il consiste à faire interagir des processus qui se comportent asymptotiquement comme des Metropolis-Hastings (MH) indépendants. A la différence des méthodes SMC où l'étape de correction fait intervenir un échantillonnage préférentiel, une étape de MH est utilisée pour le SIMCMC.

Dans le chapitre 2, nous étudions la parallélisation des méthodes SMC au sens où nous introduisons N_1 filtres particulaires avec chacun N_2 particules. On appelle *îlot* l'ensemble des particules d'un filtre particulaire et nous introduisons les *modèles d'îlots de particules*. Les différents filtres particulaires peuvent être considérés en parallèle ou bien en interaction. Nous étudions dans le chapitre 2 la possibilité d'introduire une interaction au niveau des îlots. Nous montrons que les modèles d'îlots de particules appartiennent au modèle de Feynman-Kac lorsque le potentiel de chaque îlot est le potentiel moyen des particules à l'intérieur de cet îlot. L'observation fondamentale est que les distributions marginales de Feynman-Kac au niveau des îlots et des particules coïncident. Cela signifie que considérer des îlots de particules en interaction permet de cibler la même suite de distributions qu'avec un système de particules en interaction classique ou des algorithmes SMC en parallèle. En ce sens, ce travail diffère de l'article [Jasra et al., 2008].

Dans la discussion précédente, nous avons vu que l'on peut utiliser différentes sélections entre les particules. Nous avons choisi de mener notre étude en considérant du bootstrap (ré-échantillonnage systématique) entre les particules mais en laissant une certaine latitude sur le type d'interaction utilisée entre les îlots: le bootstrap, l'ESS, l'ε-bootstrap ou bien aucune.

Dans un premier temps, nous avons étudié dans quelles circonstances il est avantageux de considérer des îlots en parallèle ou de les ré-échantillonner systématiquement. Les algorithmes induits sont respectivement appelés bootstraps indépendants et *double bootstrap* (B^2). Une analyse asymptotique de ces algorithmes se trouve dans [Vergé et al., 2015a] et [Vergé et al., 2014]. L'interaction entre les îlots rend plus difficile la simulation en parallèle car elle nécessite des points de communication, augmente la variance mais permet de réduire le biais par rapport à des îlots en parallèle. Il faut donc faire un compromis entre le biais et la variance. Pour cela nous avons choisi d'établir un critère basé sur l'erreur quadratique moyenne (MSE pour mean square error). Nous avons établi les expressions asymptotiques du biais et de la variance pour le bootstrap en faisant apparaître les erreurs d'échantillonnage locales et en utilisant le résultat établissant la convergence de la suite des erreurs locales vers une suite de champs gaussiens centrés, *cf* [Del Moral, 2004]. À partir de ces expressions, il est facile d'en déduire les expressions du biais et de la variance asymptotiques

	Bootstrap	N_1 boostraps indépendants	B^2
Biais	$\frac{B}{N_2}$	$\frac{B}{N_2}$	$\frac{B+\widetilde{B}}{N_1 N_2}$
Variance	$\frac{V}{N_2}$	$\frac{V}{N_1 N_2}$	$\frac{V+\widetilde{V}}{N_1 N_2}$

Table 2.1: Comparaison du biais et de la variance

pour des bootstraps indépendants. Celles du B^2 s'obtiennent à partir de celles du bootstrap, en remplaçant les noyaux de Feynman-Kac par leurs versions au niveau des îlots. La multiplication par le bon facteur et le passage à la limite en N_1 puis N_2 permettent de faire apparaître le premier terme des développements asymptotiques. Le tableau 2.1 présente la forme des expressions obtenues. Le critère prend alors la forme suivante : lorsque le nombre de particules à l'intérieur des îlots est assez grand, on peut garder les îlots indépendants. Dans le cas contraire, l'interaction entre îlots devient souhaitable car elle permet de réduire fortement le biais.

Dans un second temps, nous avons étudié d'autres interactions au niveau des îlots : l'ESS et l'ε-bootstrap. Ces interactions entre îlots proposent un ré-échantillonnage adaptatif. L'algorithme faisant interagir des bootstraps avec une sélection au niveau des îlots basée sur l'ESS sera appelé dans le reste du document *double bootstrap with adaptive selection on the island level* (B^{2}ASIL). L'ESS ne ré-échantillonne les îlots que lorsque leurs potentiels sont hétérogènes. L'ε-bootstrap consiste à introduire une variable aléatoire de Bernoulli. Avec une probabilité égale à son potentiel normalisé, chaque particule est dupliquée, sinon elle est ré-échantillonnée. Cette interaction permet aux "bonnes" particules de ne pas être automatiquement (et inutilement) ré-échantillonnées. En particulier, on a montré que la variance obtenue en utilisant l'ε-bootstrap est toujours inférieure à celle obtenue en utilisant le bootstrap.

Nous comparons également le nombre de ré-échantillonnages entre les îlots dans le cas du B^2 et du B^{2}ASIL sur deux exemples : un modèle linéaire gaussien et un modèle de volatilité stochastique. Le point remarquable qui est observé par simulation pour le B^{2}ASIL, est que lorsque le nombre de particules N_2 à l'intérieur des îlots est assez grand, les îlots ne sont jamais ré-échantillonnés et évoluent donc comme s'ils étaient en parallèle.

Auteurs:

- Christelle Vergé (Onera, CNES, CMAP, Palaiseau)
- Cyril Dubarry (Criteo, Paris)
- Pierre Del Moral (Université UNSW, Sydney)
- Eric Moulines (Telecom ParisTech, Paris)

Statut: Le présent chapitre est basé sur l'article "On parallel implementation of sequential Monte Carlo methods: the island particle model", publié à *Statistics and Computing*, Issue 2, volume 25, pages 243-260, mars 2015, [Vergé et al., 2015a]

Abstract The approximation of the Feynman-Kac semigroups by systems of interacting particles is a very active research field, with applications in many different areas. In this chapter, we study the parallelization of such approximations. The total population of particles is divided into sub-populations, referred to as *islands*. The particles within each island follow the usual selection / mutation dynamics. We show that the evolution of each island is also driven by a Feynman-Kac semigroup, whose transition and potential can be explicitly related to ones of the original problem. Therefore, the same genetic type approximation of the Feynman-Kac semi-group may be used at the island level; each island might undergo selection / mutation algorithm. We investigate the impact of the population size within each island and the number of islands, and study different type of interactions. We find conditions under which introducing interactions between islands is beneficial. The theoretical results are supported by some Monte Carlo experiments.

Keywords: Particle approximation of Feynman-Kac flow, island particle models, parallel implementation.

2.1 Introduction

Numerical approximation of Feynman-Kac semigroups by systems of interacting particles is a very active field of researches. Interacting particle systems are increasingly used to sample complex high dimensional distributions in a wide range of applications including nonlinear filtering, data assimilation problems, rare event sampling, hidden Markov chain parameter estimation, stochastic control problems, financial mathematics; see for example Doucet et al. [2001]; Chopin [2002]; Del Moral [2004]; Cappé and Moulines [2005]; Del Moral et al. [2012] and the references therein.

Let $(\mathsf{X}_n, \mathcal{X}_n)_{n\geq 0}$ be a sequence of measurable spaces. Denote by $\mathcal{B}_b(\mathsf{X}_n)$ the Banach space of all bounded and measurable real valued functions f on X_n, equipped with the uniform norm. Let $(g_n)_{n\in\mathbb{N}}$ be a sequence of measurable *potential functions*, $g_n : \mathsf{X}_n \to \mathbb{R}^+$. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space. In the sequel, all the processes are defined on this probability space. Let $(X_n)_{n\in\mathbb{N}}$ be a non-homogeneous Markov chain on the sequence of state-spaces $(\mathsf{X}_n)_{n\in\mathbb{N}}$ with initial distribution η_0 on $(\mathsf{X}_0, \mathcal{X}_0)$ and Markov kernels $(M_n)_{n\in\mathbb{N}^*}$ [1]. We associate to the sequences of potential functions $(g_n)_{n\in\mathbb{N}}$ and Markov kernels $(M_n)_{n\in\mathbb{N}^*}$ the sequence of *Feynman-Kac measures*, defined for all $n \geq 1$ and for any $f_n \in \mathcal{B}_b(\mathsf{X}_n)$ by

$$\eta_n(f_n) \stackrel{\text{def}}{=} \gamma_n(f_n)/\gamma_n(\mathbb{1}_{\mathsf{X}_n}) \,, \tag{2.1.1}$$

$$\gamma_n(f_n) \stackrel{\text{def}}{=} \mathbb{E}\left[f_n(X_n) \prod_{0\leq p<n} g_p(X_p)\right] \tag{2.1.2}$$

$$= \int \gamma_0(\mathrm{d}x_0) \left[\prod_{0\leq p<n} g_p(x_p) M_{p+1}(x_p, \mathrm{d}x_{p+1})\right] f_n(x_n) \,, \tag{2.1.3}$$

where we have set by convention $\eta_0(f_0) = \gamma_0(f_0) \stackrel{\text{def}}{=} \mathbb{E}\left[f_0(X_0)\right]$.

The sequences of distributions $(\eta_n)_{n\geq 0}$ and their unnormalized versions $(\gamma_n)_{n\geq 0}$ are approximated sequentially using interacting particle systems (IPS). Such particle approximations are often referred to as sequential Monte Carlo (SMC) methods. The IPS consists in approximating for each $n \in \mathbb{N}$ the probability η_n by a set of N_2 *particles* $(X_n^i)_{i=1}^{N_2}$ which are generated recursively. Typically, the update of the particles may be decomposed into a mutation and a selection step. For example, the bootstrap algorithm proceeds as follows. In the *selection step* the particles are first sampled with weights proportional to the potential functions. In the *mutation step*, a new generation of particles $(X_{n+1}^i)_{i=1}^{N_2}$ is generated from the selected particles using

[1] a Markov kernel on $\mathsf{X}_n \times \mathcal{X}_{n+1}$ is a function $M_{n+1} : \mathsf{X}_n \times \mathcal{X}_{n+1} \to [0;1]$, such that, for all $x_n \in \mathsf{X}_n$, $A_{n+1} \mapsto M_{n+1}(x_n, A_{n+1})$ is a probability measure on $(\mathsf{X}_{n+1}, \mathcal{X}_{n+1})$ and for any $A_{n+1} \in \mathcal{X}_{n+1}$, $x_n \mapsto M_{n+1}(x_n, A_{n+1})$ is a measurable function.

the kernel M_{n+1}. The asymptotic behavior of such particle approximation is now well understood (see [Del Moral, 2004; Del Moral et al., 2012]).

Particle filtering is computationally an intensive method. Parallel computations provides an appealing solution to tackle this issue (see [Durham and Geweke, 2011] and the references therein for an in-depth description of parallelization of Bayesian computations). Moreover, parallel SMC has econometric applications. For instance, it is used in the Matlab package DeCo (Density Combination) to combine density forecasts with time-varying weights and different choices of scoring rule (see [Casarin et al., 2013] for further details). The basic idea to implement interacting particle system in parallel goes as follows: instead of considering a single large batch of $N = N_1 N_2$ particles, the population is divided into N_1 batches of N_2 particles. These batches are referred in the sequel to as *islands*. The terminology *island* is borrowed from dynamic populations theory (like the genetic type interacting particle model). The particles within each island are selected and mutate, as described above. We might also introduce interactions among *islands*.

In this chapter we introduce the *island particle models*. As we will see below, we may cast the island particle model in the Feynman-Kac framework, with appropriately defined potentials and transition kernels. The key observation is that the marginal distribution of the island Feynman-Kac model with respect to (w. r. t.) any individual coincide with (2.1.3). This interpretation allows to use the interacting particle model at the island level. The general idea of interacting SMC is already mentioned in [Jasra et al., 2008], but in this contribution each island targets a different distribution.

The study of the island particle model gives rise to several challenging theoretical questions. In this chapter, we investigate the impact of the number of particles in each island N_2 compared to the number of islands N_1 for a given total number of particles $N \stackrel{\text{def}}{=} N_1 N_2$, for an algorithm where the bootstrap mechanism is used both within and between the islands. We focus on the asymptotic bias and variance when both N_1 and N_2 goes to infinity. Fluctuation theorem will be present in chapter 3. We also investigate when and why introducing interactions at the island level improves the accuracy of the particle approximation. Intuitively, the trade-off might be understood as follows. When the N_1 islands are run independently, the bias induced in each island only depends on their population size N_2; when N_2 is small compared to the total number N, the bias will be large (and is of course not reduced by averaging across the islands). To reduce the bias, introducing an interaction between the islands is beneficial. However, this interaction increases the variance, due to the selection step. If we consider the mean squared error, the interaction between islands is beneficial when the improvement associated to the bias correction is not offset by the variance increase. When the number of particles N_2 within each island is *small* and the number of islands N_1 is *large*, then the interaction is typically beneficial. On the contrary, when $N_1 \ll N_2$, the interaction between islands may increase the mean squared error. We then propose a method whose mean-squared error can be shown to be lower than the independent island model for any choice of N_1 and N_2. In this algorithm, the islands are only sampled when required, based on a generalization of the effective sample size at the island level.

The chapter is organized as follows. In section 2.2 the interacting particle approximation of the Feynman-Kac model is first reviewed. The island Feynman-Kac model is then introduced. We first investigate the double bootstrap B^2 algorithm, in which selection and mutation are applied at each iteration within and across the islands. The asymptotic bias and variance of this algorithm is presented in section 2.3. The Feynman-Kac interpretation of the island model leads to several interacting island algorithms, based on different approximations of Feynman-Kac flows. Some of these are introduced and analyzed in section 2.4. Some numerical experiments are reported to support our findings and illustrate the impact of the numbers of islands and particles within each island in section 2.5.

2.2 Algorithm derivation

In this section, we introduce the island particle model. We first briefly recall the bootstrap approximation of Feynman-Kac measures.

According to the definitions (2.1.1) and (2.1.3) of the sequences of the Feynman-Kac measures $(\eta_n)_{n\in\mathbb{N}}$ and $(\gamma_n)_{n\in\mathbb{N}}$, for all $f_{n+1} \in \mathcal{B}_b(\mathsf{X}_{n+1})$ one gets

$$\gamma_{n+1}(f_{n+1}) = \eta_{n+1}(f_{n+1})\gamma_{n+1}(\mathbb{1}_{\mathsf{X}_{n+1}}),$$

and since,

$$\gamma_{n+1}(\mathbb{1}_{\mathsf{X}_{n+1}}) = \gamma_n(g_n) = \eta_n(g_n)\gamma_n(\mathbb{1}_{\mathsf{X}_n}),$$

an easy induction shows that

$$\gamma_{n+1}(\mathbb{1}_{\mathsf{X}_{n+1}}) = \prod_{0\leq p<n+1} \eta_p(g_p)$$

and then,

$$\gamma_{n+1}(f_{n+1}) = \eta_{n+1}(f_{n+1}) \prod_{0\leq p<n+1} \eta_p(g_p) \,. \tag{2.2.1}$$

Moreover, the sequence $(\eta_n)_{n\in\mathbb{N}}$ satisfy a nonlinear recursive relation. Indeed,

$$\eta_{n+1}(f_{n+1}) = \frac{\gamma_n(g_n M_{n+1} f_{n+1})}{\gamma_n(g_n M_{n+1} \mathbb{1}_{\mathsf{X}_{n+1}})} = \frac{\eta_n(g_n M_{n+1} f_{n+1})}{\eta_n(g_n)}. \tag{2.2.2}$$

Let $\mathcal{P}(\mathsf{X}_n)$ be the set of probability measures on X_n. Using the Boltzmann-Gibbs transformation $\Psi_n : \mathcal{P}(\mathsf{X}_n) \to \mathcal{P}(\mathsf{X}_n)$, defined for all $\mu_n \in \mathcal{P}(\mathsf{X}_n)$ by

$$\Psi_n(\mu_n)(\mathrm{d}x_n) \stackrel{\text{def}}{=} \frac{g_n(x_n)\ \mu_n(\mathrm{d}x_n)}{\mu_n(g_n)} \,, \tag{2.2.3}$$

the recursion (2.2.2) may be rewritten as

$$\eta_{n+1} = \Psi_n(\eta_n) M_{n+1} \,. \tag{2.2.4}$$

The sequence of probability $(\eta_n)_{n\in\mathbb{N}}$ can be approximated using the bootstrap algorithm. Other approximations can also be considered as well, but we only introduce the bootstrap for notational simplicity. Let N_2 be a positive integer. For any nonnegative integer n we denote by

$$(\mathbf{X}_n, \boldsymbol{\mathcal{X}}_n) \stackrel{\text{def}}{=} (\mathsf{X}_n^{N_2}, \mathcal{X}_n^{\otimes N_2}) \,, \tag{2.2.5}$$

the product space (the dependence of $\mathbf{X}_n$ and $\boldsymbol{\mathcal{X}}_n$ in N_2 is implicit). Thereafter, we omit to write the σ-field $\boldsymbol{\mathcal{X}}_n$ when there will be no confusion. We define the Markov kernel $\boldsymbol{M}_{n+1}(\mathbf{x}_n, \mathrm{d}\mathbf{x}_{n+1})$ from $\mathbf{X}_n$ into $\mathbf{X}_{n+1}$ as follows: for any $\mathbf{x}_n = (x_n^1, \ldots, x_n^{N_2}) \in \mathbf{X}_n$, we set

$$\boldsymbol{M}_{n+1}(\mathbf{x}_n, \mathrm{d}\mathbf{x}_{n+1}) \stackrel{\text{def}}{=} \prod_{1\leq i\leq N_2} \sum_{j=1}^{N_2} \frac{g_n(x_n^j)}{\sum_{k=1}^{N_2} g_n(x_n^k)} M_{n+1}(x_n^j, \mathrm{d}x_{n+1}^i) \,. \tag{2.2.6}$$

In other words, this transition can be interpreted as follows:

- In the *selection step*, the components of the vector $\mathbf{x}_n$ are selected with probabilities proportional to their potential $\{g_n(x_n^i)\}_{i=1}^{N_2}$;
- In the *mutation step*, the selected coordinates move conditionally independently to new positions using the Markov kernel M_{n+1}.

Let us introduce the particles and their evolution. Define by $(\boldsymbol{X}_n)_{n\geq 0}$ the Markov chain where for each $n \in \mathbb{N}$,

$$\boldsymbol{X}_n = (X_n^1, \ldots, X_n^{N_2}) \in \mathbf{X}_n, \tag{2.2.7}$$

with initial distribution $\boldsymbol{\eta}_0 \overset{\text{def}}{=} \eta_0^{\otimes N_2}$ and transition kernel $\boldsymbol{M}_{n+1}$. Denote by m^{N_2} the empirical measure on $\mathbf{X}_n$, defined as the kernel on $\mathbf{X}_n \times \mathsf{X}_n$ by

$$m^{N_2}(\mathbf{x}_n, \mathrm{d}z_n) \overset{\text{def}}{=} \frac{1}{N_2} \sum_{i=1}^{N_2} \delta_{x_n^i}(\mathrm{d}z_n) \ ,$$

where δ_{x_n} is the dirac mass at $x_n \in \mathsf{X}_n$. Equation (2.2.1) suggests the following N_2-particle approximations of the measures η_n and γ_n respectively defined for $f_n \in \mathcal{B}_b(\mathsf{X}_n)$ by

$$\eta_n^{N_2}(f_n) \overset{\text{def}}{=} m^{N_2} f_n(\boldsymbol{X}_n) = \frac{1}{N_2} \sum_{i=1}^{N_2} f_n(\boldsymbol{X}_n^i) \tag{2.2.8}$$

$$\gamma_n^{N_2}(f_n) \overset{\text{def}}{=} \eta_n^{N_2}(f_n) \prod_{0\leq p<n} \eta_p^{N_2}(g_p) = \eta_n^{N_2}(f_n)\ \gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n}) \ . \tag{2.2.9}$$

For $\mathbf{x}_n = (x_n^1, \cdots, x_n^{N_2}) \in \mathbf{X}_n$, define the potential function

$$\boldsymbol{g}_n(\mathbf{x}_n) \overset{\text{def}}{=} m^{N_2} g_n(\mathbf{x}_n) = \frac{1}{N_2} \sum_{i=1}^{N_2} g_n(x_n^i) \ . \tag{2.2.10}$$

The sequences of transition kernels $(\boldsymbol{M}_n)_{n\in\mathbb{N}}$ and potential functions $(\boldsymbol{g}_n)_{n\in\mathbb{N}}$ given by (2.2.6) and (2.2.10), respectively, define the Feynman-Kac process. The associated sequences of Feynman-Kac measures are defined, for each $\boldsymbol{f}_n \in \mathcal{B}_b(\mathbf{X}_n)$, by the following recursions

$$\boldsymbol{\eta}_0(\boldsymbol{f}_0) \overset{\text{def}}{=} \boldsymbol{\gamma}_0(\boldsymbol{f}_0) = \mathbb{E}\left[\boldsymbol{f}_0(\boldsymbol{X}_0)\right] \ , \tag{2.2.11}$$

$$\boldsymbol{\eta}_n(\boldsymbol{f}_n) \overset{\text{def}}{=} \boldsymbol{\gamma}_n(\boldsymbol{f}_n)/\boldsymbol{\gamma}_n(\mathbb{1}_{\mathsf{X}_n}), \quad \text{for all } n \geq 1, \tag{2.2.12}$$

$$\boldsymbol{\gamma}_n(\boldsymbol{f}_n) \overset{\text{def}}{=} \mathbb{E}\left[\boldsymbol{f}_n(\boldsymbol{X}_n) \prod_{0\leq p<n} \boldsymbol{g}_p(\boldsymbol{X}_p)\right] \ , \quad \text{for all } n \geq 1 \ , \tag{2.2.13}$$

where $(\boldsymbol{X}_n)_{n\geq 0}$ is a Markov chain with initial distribution $\boldsymbol{\gamma}_0$ and transition kernel $\boldsymbol{M}_n$. The key result, justifying the introduction of the island particle models, is the following theorem which links $(\eta_n, \gamma_n)_{n\geq 0}$ and $(\boldsymbol{\eta}_n, \boldsymbol{\gamma}_n)_{n\geq 0}$.

Theorem 2.2.1. *For any $\boldsymbol{f}_n \in \mathcal{B}_b(\mathbf{X}_n)$ of the form $\boldsymbol{f}_n(\mathbf{x}_n) = N_2^{-1} \sum_{i=1}^{N_2} f_n(x_n^i)$ where $f_n \in \mathcal{B}_b(\mathsf{X}_n)$,*

$$\forall n \in \mathbb{N}, \quad \boldsymbol{\gamma}_n(\boldsymbol{f}_n) = \gamma_n(f_n) \quad \textit{and} \quad \boldsymbol{\eta}_n(\boldsymbol{f}_n) = \eta_n(f_n) \ . \tag{2.2.14}$$

Proof. See subsection 2.6.1. □

The Feynman-Kac model $(\boldsymbol{\gamma}_n, \boldsymbol{\eta}_n)_{n\geq 0}$ can be approximated by an interacting particle system *at the island level*. We first describe the *double bootstrap* B^2 algorithm where the bootstrap is also applied across the islands (this algorithm shares some similarities with [Chopin et al., 2013]). This is only one of the many possible algorithms that can be derived from this interpretation of the Feynman-Kac model at the island level; see section 2.4 for other approximations.

Define by $\mathcal{P}(\mathbf{X}_n)$ the set of probabilities measures on $\mathbf{X}_n$. One can easily check that the sequence of measures $(\boldsymbol{\eta}_n)_{n\geq 0}$ satisfies the following recursion

$$\boldsymbol{\eta}_{n+1} = \boldsymbol{\Psi}_n(\boldsymbol{\eta}_n)\boldsymbol{M}_{n+1} \ , \tag{2.2.15}$$

where $\boldsymbol{\Psi}_n : \mathcal{P}(\mathbf{X}_n) \to \mathcal{P}(\mathbf{X}_n)$ is the Boltzmann-Gibbs transformation defined for any $\mu_n \in \mathcal{P}(\mathbf{X}_n)$ by

$$\boldsymbol{\Psi}_n(\mu_n)(\mathrm{d}\mathbf{x}_n) \stackrel{\text{def}}{=} \frac{\boldsymbol{g}_n(\mathbf{x}_n)\ \mu_n(\mathrm{d}\mathbf{x}_n)}{\mu_n(\boldsymbol{g}_n)} \ .$$

Let N_1 be a positive integer. We define the Markov kernel $\mathcal{M}_{n+1}$ from $(\mathbf{X}_n^{N_1}, \boldsymbol{\mathcal{X}}_n^{\otimes N_1})$ to $(\mathbf{X}_{n+1}^{N_1}, \boldsymbol{\mathcal{X}}_{n+1}^{\otimes N_1})$ as follows: for any $(\mathbf{x}_n^1, \ldots, \mathbf{x}_n^{N_1}) \in \mathbf{X}_n^{N_1}$ and $(\mathbf{x}_{n+1}^1, \ldots, \mathbf{x}_{n+1}^{N_1}) \in \mathbf{X}_{n+1}^{N_1}$, we put

$$\mathcal{M}_{n+1}((\mathbf{x}_n^1, \ldots, \mathbf{x}_n^{N_1}), \mathrm{d}(\mathbf{x}_{n+1}^1, \ldots, \mathbf{x}_{n+1}^{N_1})) \stackrel{\text{def}}{=} \prod_{1 \le i \le N_1} \sum_{j=1}^{N_1} \frac{\boldsymbol{g}_n(\mathbf{x}_n^j)}{\sum_{k=1}^{N_1} \boldsymbol{g}_n(\mathbf{x}_n^k)} \boldsymbol{M}_{n+1}(\mathbf{x}_n^j, \mathrm{d}\mathbf{x}_{n+1}^i) \ . \tag{2.2.16}$$

For each $n \in \mathbb{N}$, $(\boldsymbol{X}_n^1, \ldots, \boldsymbol{X}_n^{N_1}) \in \mathbf{X}_n^{N_1}$ is a population of N_1 interacting islands each with N_2 individuals. The process $\{(\boldsymbol{X}_n^1, \ldots, \boldsymbol{X}_n^{N_1})\}_{n \ge 0}$ is a Markov chain with the transition kernel $(\mathcal{M}_{n+1})_{n \ge 0}$.

In this interpretation, the N_1-particle model defined above can be seen as an interacting particle approximation of the island Feynman-Kac measures $\{(\boldsymbol{\eta}_n, \boldsymbol{\gamma}_n)\}_{n \ge 0}$. The transition $\mathcal{M}_{n+1}$ can be interpreted as follows:

- In the *selection step*, we sample randomly N_1 islands among the current islands $\left(\boldsymbol{X}_n^i\right)_{1 \le i \le N_1} \in \mathbf{X}_n^{N_1}$ with probability proportional to the empirical mean of the potentials in each island $\boldsymbol{g}_n(\boldsymbol{X}_n^i) = N_2^{-1} \sum_{j=1}^{N_2} g_n(X_n^{i,j})$, $1 \le i \le N_1$.

- In the *mutation transition*, the selected islands are independently updated using the Markov transition $\boldsymbol{M}_{n+1}$.

Also observe that for $N_2 = 1$, every island has a single particle. In this situation, the island Feynman-Kac model coincides with the N_1-particle model associated with the Feynman-Kac measures η_n.

Denote by $\boldsymbol{m}^{N_1}$ the empirical measure defined for any $\boldsymbol{f}_n \in \mathcal{B}_b(\mathbf{X}_n)$ and $(\mathbf{x}_n^1, \ldots, \mathbf{x}_n^{N_1}) \in \mathbf{X}_n^{N_1}$ by

$$\boldsymbol{m}^{N_1} \boldsymbol{f}_n(\mathbf{x}_n^1, \ldots, \mathbf{x}_n^{N_1}) \stackrel{\text{def}}{=} \frac{1}{N_1} \sum_{i=1}^{N_1} \boldsymbol{f}_n(\mathbf{x}_n^i) \ .$$

The N_1-particle approximations of the measures $\boldsymbol{\eta}_n$ and $\boldsymbol{\gamma}_n$ are defined for any $\boldsymbol{f}_n \in \mathcal{B}_b(\mathbf{X}_n)$ by

$$\boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n) \stackrel{\text{def}}{=} \boldsymbol{m}^{N_1} \boldsymbol{f}_n(\boldsymbol{X}_n^1, \ldots, \boldsymbol{X}_n^{N_1}) \ , \tag{2.2.17}$$

$$\boldsymbol{\gamma}_n^{N_1}(\boldsymbol{f}_n) \stackrel{\text{def}}{=} \boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n) \prod_{0 \le p < n} \boldsymbol{\eta}_p^{N_1}(\boldsymbol{g}_p) = \boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n)\ \boldsymbol{\gamma}_n^{N_1}(\mathbb{1}_{\mathbf{X}_n}) \ . \tag{2.2.18}$$

Data: η_0, $(g_p)_{p\geq 0}$, $(M_p)_{p\geq 1}$
Result: Empirical estimate of η_n

/* Initialization */
for $i \leftarrow 1$ **to** N_1 **do**
  Sample N_2 independent random variables $\boldsymbol{X}_0^i = \left(X_0^{i,j}\right)_{j=1}^{N_2}$ from η_0;
end
for $p \leftarrow 0$ **to** $n-1$ **do**
  /* Selection step between islands */
  Sample $\boldsymbol{I}_p = (I_p^i)_{i=1}^{N_1}$ multinomially with probability proportional to $\left(\frac{1}{N_2}\sum_{j=1}^{N_2} g_p(X_p^{i,j})\right)_{i=1}^{N_1}$;
  /* Island mutation step */
  for $i \leftarrow 1$ **to** N_1 **do**
    /* Particle selection within each island */
    Sample $\boldsymbol{J}_p^i = (J_p^{i,j})_{j=1}^{N_2}$ multinomially with probability proportional to $\left(g_p(X_p^{I_p^i,j})\right)_{j=1}^{N_2}$.
    /* Particle mutation */
    For $1 \leq j \leq N_2$, sample conditionally independently $X_{p+1}^{i,j}$ from the Markov kernel $M_{p+1}(X_p^{I_p^i, L_p^{i,j}}, \cdot)$, where $L_p^{i,j} = J_p^{I_p^i, j}$;
  end
end
Approximate $\eta_n(f_n)$ by $\sum_{i=1}^{N_1}\sum_{j=1}^{N_2} f_n\left(X_n^{i,j}\right)/N_1N_2$.

Algorithm 4: Double bootstrap (B^2) island filter

2.3 Asymptotic analysis of the double bootstrap (B^2) algorithm

The bootstrap particle approximation of the Feynman-Kac semigroup can be studied using the techniques introduced in [Del Moral, 2004] and further developed in [Del Moral et al., 2012]. For $\ell \in \mathbb{N}$, consider the finite kernel $Q_{\ell+1}$ from $(\mathsf{X}_\ell, \mathcal{X}_\ell)$ into $(\mathsf{X}_{\ell+1}, \mathcal{X}_{\ell+1})$ given for all $x_\ell \in \mathsf{X}_\ell$ by

$$Q_{\ell+1}(x_\ell, \mathrm{d}x_{\ell+1}) \stackrel{\text{def}}{=} g_\ell(x_\ell) M_{\ell+1}(x_\ell, \mathrm{d}x_{\ell+1}) \,. \tag{2.3.1}$$

For $p < n$, define by $Q_{p,n}$ the finite kernel from $(\mathsf{X}_p, \mathcal{X}_p)$ into $(\mathsf{X}_n, \mathcal{X}_n)$ as the following product

$$Q_{p,n} \stackrel{\text{def}}{=} Q_{p+1} Q_{p+2} \ldots Q_n \,, \tag{2.3.2}$$

and set by convention $Q_{n,n} \stackrel{\text{def}}{=} \mathrm{I}_n$ where I_n is the identity kernel on $(\mathsf{X}_n, \mathcal{X}_n)$. With this definition, the linear semigroup associated with the sequence of unnormalized Feynman-Kac measures $(\gamma_n)_{n\in\mathbb{N}}$ may be equivalently expressed as follows

$$\gamma_n = \gamma_p Q_{p,n} \,. \tag{2.3.3}$$

For any $x_p \in \mathsf{X}_p$, $A_n \in \mathcal{X}_n$, $Q_{p,n}$ may be written as the following conditional expectation,

$$Q_{p,n}(x_p, A_n) = \mathbb{E}\left[\mathbb{1}_{A_n}(X_n) \prod_{p\leq q<n} g_q(X_q) \mid X_p = x_p\right] \,,$$

where $(X_n)_{n\geq 0}$ is the non-homogenous Markov chain on the sequence of state-spaces $(\mathsf{X}_n, \mathcal{X}_n)_{n\geq 0}$ with initial distribution η_0 and Markov kernels $(M_n)_{n\geq 1}$.

According to (2.1.1), $\eta_n = \gamma_n/\gamma_n(\mathbb{1}_{\mathsf{X}_n})$ implies that $\eta_n = \gamma_p Q_{p,n}/\gamma_p Q_{p,n}(\mathbb{1}_{\mathsf{X}_n})$. Denote by Φ_{n+1} the mapping from $\mathcal{P}(\mathsf{X}_n)$ to $\mathcal{P}(\mathsf{X}_{n+1})$ given, for any $\mu_n \in \mathcal{P}(\mathsf{X}_n)$ by

$$\Phi_{n+1}(\mu_n) \stackrel{\text{def}}{=} \Psi_n(\mu_n) M_{n+1} = \frac{\mu_n Q_{n+1}}{\mu_n Q_{n+1}(\mathbb{1}_{\mathsf{X}_{n+1}})} \, . \tag{2.3.4}$$

Since $\eta_p = \gamma_p/\gamma_p(\mathbb{1}_{\mathsf{X}_p})$, these relations may be equivalently rewritten as

$$\eta_n = \frac{\eta_p Q_{p,n}}{\eta_p Q_{p,n}(\mathbb{1}_{\mathsf{X}_n})} = \Phi_{p,n}(\eta_p) \, , \tag{2.3.5}$$

where $\Phi_{p,n} = \Phi_n \circ \Phi_{n-1} \circ \cdots \circ \Phi_{p+1}$ is the nonlinear semigroup associated to the normalized Feynman-Kac measures $(\eta_n)_{n\geq 0}$. This nonlinear semigroup may be associated to the *potential kernels*

$$P_{p,n} \stackrel{\text{def}}{=} \frac{Q_{p,n}}{\eta_p Q_{p,n}(\mathbb{1}_{\mathsf{X}_n})} = \frac{\gamma_p(\mathbb{1}_{\mathsf{X}_p})}{\gamma_n(\mathbb{1}_{\mathsf{X}_n})} Q_{p,n} \, , \tag{2.3.6}$$

and therefore

$$\eta_n = \eta_p P_{p,n} \, . \tag{2.3.7}$$

For $\ell \in \mathbb{N}$, consider the finite kernel $\boldsymbol{Q}_{\ell+1}$ from $(\mathbf{X}_\ell, \boldsymbol{\mathcal{X}}_\ell)$ into $(\mathbf{X}_{\ell+1}, \boldsymbol{\mathcal{X}}_{\ell+1})$ for any $\mathbf{x}_\ell \in \mathbf{X}_\ell$ by

$$\boldsymbol{Q}_{\ell+1}(\mathbf{x}_\ell, \mathrm{d}\mathbf{x}_{\ell+1}) \stackrel{\text{def}}{=} \boldsymbol{g}_\ell(\mathbf{x}_\ell) \boldsymbol{M}_{\ell+1}(\mathbf{x}_\ell, \mathrm{d}\mathbf{x}_{\ell+1}) \, ,$$

where $\boldsymbol{M}_\ell$ is defined in (2.2.6) and $\boldsymbol{g}_\ell$ in (2.2.10). For $p \leq n$, define by $\boldsymbol{Q}_{p,n}$ the finite kernel from $(\mathbf{X}_p, \boldsymbol{\mathcal{X}}_p)$ into $(\mathbf{X}_n, \boldsymbol{\mathcal{X}}_n)$ by the equation $\boldsymbol{Q}_{p,n} \stackrel{\text{def}}{=} \boldsymbol{Q}_{p+1} \boldsymbol{Q}_{p+2} \ldots \boldsymbol{Q}_n$. Note that, for any $\mathbf{x}_p \in \mathbf{X}_p$, $\mathbf{A}_n \in \boldsymbol{\mathcal{X}}_n$,

$$\boldsymbol{Q}_{p,n}(\mathbf{x}_p, A_n) = \mathbb{E}\left[\mathbb{1}_{\mathbf{A}_n}(\boldsymbol{X}_n) \prod_{p \leq q < n} \boldsymbol{g}_q(\boldsymbol{X}_q) \mid \boldsymbol{X}_p = \mathbf{x}_p \right] \, ,$$

where $(\boldsymbol{X}_n)_{n\geq 0}$ is the island Markov chain defined in (2.2.7). With this notation, we may rewrite (2.2.11) as $\boldsymbol{\gamma}_n = \boldsymbol{\gamma}_p \boldsymbol{Q}_{p,n}$. According to (2.2.12), $\boldsymbol{\eta}_n = \boldsymbol{\gamma}_n/\boldsymbol{\gamma}_n(\mathbb{1}_{\mathbf{X}_n})$ implies that $\boldsymbol{\eta}_n = \boldsymbol{\gamma}_p \boldsymbol{Q}_{p,n}/\boldsymbol{\gamma}_p \boldsymbol{Q}_{p,n}(\mathbb{1}_{\mathbf{X}_n})$, and then

$$\boldsymbol{\eta}_n = \frac{\boldsymbol{\eta}_p \boldsymbol{Q}_{p,n}}{\boldsymbol{\eta}_p \boldsymbol{Q}_{p,n}(\mathbb{1}_{\mathbf{X}_n})} = \boldsymbol{\eta}_p \boldsymbol{P}_{p,n} \, ,$$

where $\boldsymbol{P}_{p,n}$ are given by

$$\boldsymbol{P}_{p,n} \stackrel{\text{def}}{=} \frac{\boldsymbol{Q}_{p,n}}{\boldsymbol{\eta}_p \boldsymbol{Q}_{p,n}(\mathbb{1}_{\mathbf{X}_n})} = \frac{\boldsymbol{\gamma}_p(\mathbb{1}_{\mathbf{X}_p})}{\boldsymbol{\gamma}_n(\mathbb{1}_{\mathbf{X}_n})} \boldsymbol{Q}_{p,n} \, .$$

According to Theorem 2.2.1, $\boldsymbol{\gamma}_p(\mathbb{1}_{\mathbf{X}_p}) = \gamma_p(\mathbb{1}_{\mathsf{X}_p})$ and $\boldsymbol{\gamma}_n(\mathbb{1}_{\mathbf{X}_n}) = \gamma_n(\mathbb{1}_{\mathsf{X}_n})$, which implies that

$$\boldsymbol{P}_{p,n} = \frac{\gamma_p(\mathbb{1}_{\mathsf{X}_p})}{\gamma_n(\mathbb{1}_{\mathsf{X}_n})} \boldsymbol{Q}_{p,n} \, .$$

To analyze the fluctuation of the interacting particle approximation $(\eta_n^{N_2})_{n\geq 0}$ around their limiting values

$(\eta_n)_{n\geq 0}$, we introduced the *local sampling errors*.

$$
\begin{array}{ccccccccc}
\gamma_0 & \xrightarrow{Q_1} & \gamma_1 = \gamma_0 Q_1 & \xrightarrow{Q_2} & \gamma_2 = \gamma_1 Q_2 & \to & \cdots & \xrightarrow{Q_n} & \gamma_n \\
\Downarrow & & & & & & & & \\
\gamma_0^{N_2} & \xrightarrow{Q_1} & \gamma_0^{N_2} Q_1 & \xrightarrow{Q_2} & \gamma_0^{N_2} Q_{0,2} & \to & \cdots & \xrightarrow{Q_n} & \gamma_0^{N_2} Q_{0,n} \\
 & & \Downarrow & & & & & & \\
 & & \gamma_1^{N_2} & \xrightarrow{Q_2} & \gamma_1^{N_2} Q_2 & \to & \cdots & \xrightarrow{Q_n} & \gamma_1^{N_2} Q_{1,n} \\
 & & & & \Downarrow & & & & \\
 & & & & \gamma_2^{N_2} & \to & \cdots & \xrightarrow{Q_n} & \gamma_2^{N_2} Q_{2,n} \\
 & & & & & & & \vdots & \\
 & & & & & & & \xrightarrow{Q_n} & \gamma_{n-1}^{N_2} Q_n \\
 & & & & & & & & \Downarrow \\
 & & & & & & & & \gamma_n^{N_2}
\end{array}
$$

Hence, we may decompose the difference $\gamma_n^{N_2} - \gamma_n$ as follows

$$
\gamma_n^{N_2} - \gamma_n = \sum_{p=1}^{n} \left[\gamma_p^{N_2} Q_{p,n} - \gamma_{p-1}^{N_2} Q_{p-1,n} \right] + \gamma_0^{N_2} Q_{0,n} - \gamma_n \; . \tag{2.3.8}
$$

For any $p \geq 1$, note that

$$
\begin{aligned}
\gamma_{p-1}^{N_2} Q_p &= \gamma_{p-1}^{N_2}(\mathbb{1}_{\mathsf{X}_{p-1}})\; \eta_{p-1}^{N_2} Q_p = \gamma_{p-1}^{N_2}(\mathbb{1}_{\mathsf{X}_{p-1}})\; \eta_{p-1}^{N_2}(g_{p-1})\; \Phi_p(\eta_{p-1}^{N_2}) \\
&= \gamma_{p-1}^{N_2}(\mathbb{1}_{\mathsf{X}_{p-1}})\; \frac{\gamma_{p-1}^{N_2}(g_{p-1})}{\gamma_{p-1}^{N_2}(\mathbb{1}_{\mathsf{X}_{p-1}})}\; \Phi_p(\eta_{p-1}^{N_2}) = \gamma_{p-1}^{N_2}(g_{p-1})\; \Phi_p(\eta_{p-1}^{N_2}) = \gamma_p^{N_2}(\mathbb{1}_{\mathsf{X}_p})\; \Phi_p(\eta_{p-1}^{N_2}) \; .
\end{aligned}
$$

Plugging in this relation in the local error yields to

$$
\begin{aligned}
\gamma_p^{N_2} Q_{p,n} - \gamma_{p-1}^{N_2} Q_{p-1,n} &= \gamma_p^{N_2} Q_{p,n} - \gamma_{p-1}^{N_2} Q_p Q_{p,n} \\
&= \left(\gamma_p^{N_2} - \gamma_p^{N_2}(\mathbb{1}_{\mathsf{X}_p}) \Phi_p(\eta_{p-1}^{N_2}) \right) Q_{p,n} = \gamma_p^{N_2}(\mathbb{1}_{\mathsf{X}_p}) \left(\eta_p^{N_2} - \Phi_p(\eta_{p-1}^{N_2}) \right) Q_{p,n} \; ,
\end{aligned}
$$

which, together with (2.3.8), imply that,

$$
W_n^{\gamma,N_2} \stackrel{\text{def}}{=} \sqrt{N_2} \left[\gamma_n^{N_2} - \gamma_n \right] = \sum_{p=0}^{n} \gamma_p^{N_2}(\mathbb{1}_{\mathsf{X}_p}) W_p^{N_2} Q_{p,n} \; , \tag{2.3.9}
$$

where the local errors $(W_p^{N_2})_{p\geq 0}$ are defined by

$$
W_0^{N_2} = \sqrt{N_2}(\eta_0^{N_2} - \eta_0) \quad \text{and} \quad W_p^{N_2} = \sqrt{N_2} \left[\eta_p^{N_2} - \Phi_p(\eta_{p-1}^{N_2}) \right], \quad \text{for all} \;\; p \geq 1 \; . \tag{2.3.10}
$$

The following results, adapted from [Del Moral, 2004, Corollary 9.3.1, pp. 295-298], establishes the convergence of $(W_p^{N_2})_{1\leq p\leq n}$ to centered Gaussian fields.

Theorem 2.3.1. *For the bootstrap filter, for any fixed time horizon $n \geq 1$, the sequence $(W_p^{N_2})_{1\leq p\leq n}$ converges in law, as N_2 goes to infinity, to a sequence of n independent centered Gaussian random fields $(W_p)_{0\leq p\leq n}$ with variance given, for any bounded function $f_p \in \mathcal{B}_b(\mathsf{X}_p)$, and $1 \leq p \leq n$, by*

$$
\mathbb{E}\left[W_p(f_p)^2 \right] = \eta_p \left[(f_p - \eta_p f_p)^2 \right] \; . \tag{2.3.11}
$$

Now, consider the sequence of random fields $(W_n^{\eta,N_2})_{n\geq 0}$ defined for any function $f_n \in \mathcal{B}_b(\mathsf{X}_n)$ by

$$W_n^{\eta,N_2}(f_n) \stackrel{\text{def}}{=} \sqrt{N_2}\left[\eta_n^{N_2} - \eta_n\right](f_n) = \sqrt{N_2}\eta_n^{N_2}[f_n - \eta_n(f_n)] \tag{2.3.12}$$

$$= \sqrt{N_2}\frac{\gamma_n^{N_2}(f_n - \eta_n(f_n))}{\gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n})} \; . \tag{2.3.13}$$

Using the fact that $\gamma_n(f_n - \eta_n(f_n)) = 0$ and (2.3.9), we may write

$$W_n^{\eta,N_2}(f_n) = \sqrt{N_2}\frac{(\gamma_n^{N_2} - \gamma_n)(f_n - \eta_n(f_n))}{\gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n})} = \frac{W_n^{\gamma,N_2}(f_n - \eta_n(f_n))}{\gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n})} \; . \tag{2.3.14}$$

The decomposition (2.3.9) and (2.3.12), combined with the Slutsky's lemma, imply the following asymptotic decomposition (which remains valid for more general algorithms than the bootstrap algorithm):

Theorem 2.3.2. *Assume that the sequence of local errors $(W_p^{N_2})_{1\leq p\leq n}$ converges in law, as N_2 goes to infinity, to a sequence of n independent centered Gaussian random fields $(W_p)_{1\leq p\leq n}$. Then, the sequence of random fields $(W_n^{\gamma,N_2})_{N_2\geq 0}$ converges in law, as N_2 goes to infinity, to the Gaussian random fields W_n^{γ} defined for any bounded function f_n in $\mathcal{B}_b(\mathsf{X}_n)$ by*

$$W_n^{\gamma}(f_n) \stackrel{\text{def}}{=} \sum_{p=0}^{n} \gamma_p(\mathbb{1}_{\mathsf{X}_p}) W_p(Q_{p,n} f_n) = \gamma_n(\mathbb{1}_{\mathsf{X}_n}) \sum_{p=0}^{n} W_p(P_{p,n} f_n) \; , \tag{2.3.15}$$

where $P_{p,n}$ is defined in (2.3.6). The sequence of random fields $(W_n^{\eta,N_2})_{N_2\geq 0}$ converges in law, as N_2 goes to infinity, to the Gaussian random fields W_n^{η} defined for any function $f_n \in \mathcal{B}_b(\mathsf{X}_n)$ by

$$W_n^{\eta}(f_n) \stackrel{\text{def}}{=} \sum_{p=0}^{n} W_p(P_{p,n}(f_n - \eta_n(f_n))) \; . \tag{2.3.16}$$

The asymptotic bias and variance for the single island interacting particle approximation of the sequence of Feynman-Kac measure formulated in the forthcoming theorem result almost immediately from Theorem 2.3.1.

Theorem 2.3.3. *Assume that the sequence of local errors $(W_p^{N_2})_{1\leq p\leq n}$ converges in law, as N_2 goes to infinity, to a sequence of n independent centered Gaussian random fields $(W_p)_{1\leq p\leq n}$. Then, for any time horizon $n \geq 0$ and any bounded function $f_n \in \mathcal{B}_b(\mathsf{X}_n)$, one has*

$$\lim_{N_2\to\infty} N_2 \mathbb{E}\left[\eta_n^{N_2}(f_n) - \eta_n(f_n)\right] = B_n(f_n) \; ,$$

$$\lim_{N_2\to\infty} N_2 \mathbb{V}\mathrm{ar}\left(\eta_n^{N_2}(f_n)\right) = V_n(f_n) \; ,$$

with

$$B_n(f_n) \stackrel{\text{def}}{=} -\sum_{p=0}^{n} \mathbb{E}\left[W_p(P_{p,n}(\mathbb{1}_{\mathsf{X}_n})) W_p(P_{p,n}(f_n - \eta_n(f_n)))\right] \; , \tag{2.3.17}$$

$$V_n(f_n) \stackrel{\text{def}}{=} \sum_{p=0}^{n} \mathbb{E}\left[\{W_p(P_{p,n}(f_n - \eta_n(f_n)))\}^2\right] \; . \tag{2.3.18}$$

When the bootstrap algorithm is applied, one gets the following expressions for $B_n(f_n)$ and $V_n(f_n)$ using Theorem 2.3.1:

$$B_n(f_n) = -\sum_{p=0}^{n} \eta_p\left(P_{p,n}(\mathbb{1}_{\mathsf{X}_n}) P_{p,n}(f_n - \eta_n(f_n))\right) \; , \tag{2.3.19}$$

$$V_n(f_n) = \sum_{p=0}^{n} \eta_p\left(P_{p,n}(f_n - \eta_n(f_n))^2\right) \; . \tag{2.3.20}$$

Proof. See subsection 2.6.2. □

We now compute the bias and the variance for the B^2 algorithm. The asymptotic behavior of the bias and the variance is derived in the following theorem using techniques adapted from [Del Moral, 2004].

Theorem 2.3.4. *For the B^2 algorithm, for any time horizon $n \geq 0$ and any $f_n \in \mathcal{B}_b(\mathsf{X}_n)$, one has*

$$\lim_{N_2\to\infty}\lim_{N_1\to\infty} N_1N_2\mathbb{E}\left[\boldsymbol{\eta}_n^{N_1}(m^{N_2}f_n)-\eta_n(f_n)\right] = B_n(f_n)+\widetilde{B}_n(f_n)\;,$$
$$\lim_{N_2\to\infty}\lim_{N_1\to\infty} N_1N_2\mathbb{V}\mathrm{ar}\left(\boldsymbol{\eta}_n^{N_1}(m^{N_2}f_n)\right) = V_n(f_n)+\widetilde{V}_n(f_n)\;,$$

where $B_n(f_n)$ and $V_n(f_n)$ are defined respectively in (2.3.17) and in (2.3.18), and where $\widetilde{B}_n(f_n)$ and $\widetilde{V}_n(f_n)$ are given by:

$$\widetilde{B}_n(f_n) \stackrel{\text{def}}{=} -\sum_{\ell=0}^{n}(n-\ell)\mathbb{E}\left[W_\ell(P_{\ell,n}(\mathbb{1}_{\mathsf{X}_n}))W_\ell(P_{\ell,n}(f_n-\eta_n(f_n)))\right] \tag{2.3.21}$$
$$+\sum_{\ell=0}^{n}\mathbb{E}\left[W_\ell\left(\sum_{p=\ell}^{n}P_{\ell,p}(\mathbb{1}_{\mathsf{X}_p})\right)W_\ell(P_{\ell,n}(f_n-\eta_n(f_n)))\right]\;,$$
$$\widetilde{V}_n(f_n) \stackrel{\text{def}}{=} \sum_{\ell=0}^{n}(n-\ell)\mathbb{E}\left[W_\ell(P_{\ell,n}(f_n-\eta_n(f_n)))^2\right]\;. \tag{2.3.22}$$

When the bootstrap algorithm is applied, one gets the following expressions for $\widetilde{B}_n(f_n)$ and $\widetilde{V}_n(f_n)$ using Theorem 2.3.1:

$$\widetilde{B}_n(f_n) = -\sum_{\ell=0}^{n}(n-\ell)\;\eta_\ell\left((P_{\ell,n}(\mathbb{1}_{\mathsf{X}_n})-\eta_n(\mathbb{1}_{\mathsf{X}_n}))P_{\ell,n}(f_n-\eta_n(f_n))\right) \tag{2.3.23}$$
$$+\sum_{\ell=0}^{n}\eta_\ell\left(\left(\sum_{p=\ell}^{n}(P_{\ell,p}(\mathbb{1}_{\mathsf{X}_p})-\eta_p(\mathbb{1}_{\mathsf{X}_p}))\right)P_{\ell,n}(f_n-\eta_n(f_n))\right)\;,$$
$$\widetilde{V}_n(f_n) = \sum_{\ell=0}^{n}(n-\ell)\;\eta_\ell\left(P_{\ell,n}(f_n-\eta_n(f_n))^2\right)\;. \tag{2.3.24}$$

Proof. See subsection 2.6.3. □

We can also consider the case where the N_1 islands are kept independent (a bootstrap filter is still applied within each island, but there is no interaction between islands). To that purpose, denote by $(\widetilde{\boldsymbol{X}}_n^i)_{i=1}^{N_1}$ N_1 independent islands of size N_2, each evolving using the bootstrap filter and, define the estimator of $\eta_n(f_n)$ for any $\boldsymbol{f}_n \in \mathcal{B}_b(\mathbf{X}_n)$, given by the empirical mean across islands

$$\widetilde{\boldsymbol{\eta}}_n^{N_1}(\boldsymbol{f}_n) \stackrel{\text{def}}{=} \frac{1}{N_1}\sum_{i=1}^{N_1}\boldsymbol{f}_n(\widetilde{\boldsymbol{X}}_n^i)\;.$$

For functions $\boldsymbol{f}_n$ on $\mathbf{X}_n$ of the form $\boldsymbol{f}_n(\mathbf{x}_n) = m^{N_2}f_n(\mathbf{x}_n)$, with $f_n \in \mathcal{B}_b(\mathsf{X}_n)$, one has

$$\widetilde{\boldsymbol{\eta}}_n^{N_1}(\boldsymbol{f}_n) = \frac{1}{N_1}\sum_{i=1}^{N_1}m^{N_2}f_n(\widetilde{\boldsymbol{X}}_n^i) = \frac{1}{N_1N_2}\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}f_n(\widetilde{\boldsymbol{X}}_n^{i,j})\;.$$

The asymptotic behavior of the bias and variance of $m^{N_2}f_n(\widetilde{\boldsymbol{X}}_n^i)$ may be easily deduced from the one of $\eta_n^{N_2}(f_n)$; Theorem 2.3.3 implies that

Theorem 2.3.5. *For any time horizon $n \geq 0$ and any $f_n \in \mathcal{B}_b(\mathsf{X}_n)$, one has*

$$\lim_{N_2 \to \infty} N_2 \left\{ \mathbb{E}\left[\widetilde{\boldsymbol{\eta}}_n^{N_1}(m^{N_2} f_n) \right] - \eta_n(f_n) \right\} = B_n(f_n) \; ,$$
$$\lim_{N_2 \to \infty} N_1 N_2 \mathbb{V}\mathrm{ar}\left(\widetilde{\boldsymbol{\eta}}_n^{N_1}(m^{N_2} f_n) \right) = V_n(f_n) \; ,$$

where $B_n(f_n)$ and $V_n(f_n)$ are defined respectively in (2.3.17) and (2.3.18).

The variance of the particle approximation is inversely proportional to $N_1 N_2$, but because the islands do not interact, the bias is independent of N_1 and is inversely proportional to N_2.

As shown by Theorem 2.3.4 and Theorem 2.3.5, a trade-off has to be made between the bias and the variance to decide which of the two estimators $\boldsymbol{\eta}_n^{N_1}$ and $\widetilde{\boldsymbol{\eta}}_n^{N_1}$ is the best. We can compare the mean squared error *mean squared error* (MSE) when the islands interact or when they are kept independent. The MSE for independent islands is given by $\frac{V_n(f_n)}{N_1 N_2} + \frac{B_n(f_n)^2}{N_2^2}$ whereas the MSE for the B^2 is given by $\frac{V_n(f_n) + \widetilde{V}_n(f_n)}{N_1 N_2}$. Therefore,

$$\frac{V_n(f_n) + \widetilde{V}_n(f_n)}{N_1 N_2} < \frac{V_n(f_n)}{N_1 N_2} + \frac{B_n(f_n)^2}{N_2^2} \quad \Leftrightarrow \quad N_2 < \frac{B_n(f_n)^2}{\widetilde{V}_n(f_n)} N_1 \; .$$

Consequently, the B^2 algorithm outperforms the independent islands when the number of particles N_2 within each island is small compared to the number of islands N_1; the interaction improves the bias (which is independent of N_1 when the islands are kept independent). On the contrary, when N_2 is larger than N_1, the variance increase introduced by the interaction (because of the selection step) may be larger than the bias reduction.

2.4 Extensions

In section 2.3 we have described and analyzed an interacting island model where the bootstrap algorithm is used both within and across the islands. Of course, other IPS approximations may be considered within and across islands. We will describe how the results of the previous sections may be adapted. The IPS approximation of each individual island may be cast in the Feynman-Kac framework. This section is devoted to check these conditions for various IPS approximations.

2.4.1 Epsilon-bootstrap interaction

ε-bootstrap interaction is a variant of the bootstrap, in which the selection step is slightly modified: only a fraction of the particles are resampled. Let ϵ_n be a nonnegative constant such that $\epsilon_n \; \|g_n\|_\infty \in [0,1]$, where $\|g_n\|_\infty = \sup_{x_n \in \mathsf{X}_n} |g_n(x_n)|$. For any measure $\mu_n \in \mathcal{P}(\mathsf{X}_n)$, define S_{n,μ_n} the Markov kernel on $(\mathsf{X}_n, \mathcal{X}_n)$ given for $x_n \in \mathsf{X}_n$ and $A_n \in \mathcal{X}_n$ by

$$S_{n,\mu_n}(x_n, A_n) \stackrel{\text{def}}{=} \epsilon_n \; g_n(x_n) \delta_{x_n}(A_n) + (1 - \epsilon_n \; g_n(x_n)) \, \Psi_n(\mu_n)(A_n) \; , \tag{2.4.1}$$

where Ψ_n is defined in (2.2.3). ε-bootstrap interaction algorithm proceeds as follows. At iteration n, a particle X_n^i is kept with a probability equal to $\epsilon_n \; g_n(X_n^i)$ or resampled with a probability $1 - \epsilon_n \; g_n(X_n^i)$. Resampling a particle consists in replacing it by a particle selected at random in the current population with weights proportional to their potential $(g_n(X_n^1), \ldots, g_n(X_n^{N_2}))$. Then, each selected particle is independently updated according to the Markov kernel M_{n+1}. When $\epsilon_n = 0$, all the particles are resampled, which corresponds to the bootstrap filter. Define the Markov kernel $\boldsymbol{M}_{n+1}(\mathbf{x}_n, \mathrm{d}\mathbf{x}_{n+1})$ from $\mathbf{X}_n$ into $\mathbf{X}_{n+1}$ by

$$\boldsymbol{M}_{n+1}(\mathbf{x}_n, \mathrm{d}\mathbf{x}_{n+1}) \stackrel{\text{def}}{=} \prod_{1 \leq i \leq N_2} S_{n,\eta_n^{N_2}} M_{n+1}(x_n^i, \mathrm{d}x_{n+1}^i) \; . \tag{2.4.2}$$

Consider a Markov chain $(\boldsymbol{X}_n)_{n\geq 0}$ where for each $n \in \mathbb{N}$, $\boldsymbol{X}_n = (X_n^1, \ldots, X_n^{N_2}) \in \mathbf{X}_n$, with initial distribution η_0 and transition kernel $\boldsymbol{M}_{n+1}$. Define the same approximations of the measures η_n and γ_n as in (2.2.8) and (2.2.9). Then, consider the island Feynman-Kac model associated to the Markov chain (2.2.11) and the potential function (2.2.10). The associated sequence $\{(\boldsymbol{\eta}_n, \boldsymbol{\gamma}_n)\}_{n\geq 0}$ of Feynman-Kac measures is given in (2.2.11), (2.2.12) and (2.2.13). We may establish the following extension of Theorem 2.2.1.

Theorem 2.4.1. *Let $\{(x_p^1, \ldots, x_p^{N_2})\}_{0\leq p\leq n}$ be a population of particles generated by the ε-bootstrap interaction algorithm specified by (2.4.2). Then, for any $\boldsymbol{f}_n \in \mathcal{B}_b(\mathbf{X}_n)$ of the form $\boldsymbol{f}_n(\mathbf{x}_n) = N_2^{-1} \sum_{i=1}^{N_2} f_n(x_n^i)$ with $f_n \in \mathcal{B}_b(\mathsf{X}_n)$, one gets $\boldsymbol{\gamma}_n(\boldsymbol{f}_n) = \gamma_n(f_n)$ and $\boldsymbol{\eta}_n(\boldsymbol{f}_n) = \eta_n(f_n)$.*

Proof. See subsection 2.6.4. □

For each $n \in \mathbb{N}$, let $(\boldsymbol{X}_n^1, \ldots, \boldsymbol{X}_n^{N_1}) \in \mathbf{X}_n^{N_1}$ be a population of N_1 islands each of N_2 individuals. The process $(\boldsymbol{X}_n^1, \ldots, \boldsymbol{X}_n^{N_1})$ is a Markov chain evolving according to selection and mutation steps, defined as follows

- *Selection step*: each island $\boldsymbol{X}_n^i$ is kept with a probability equal to $\epsilon_n\, \boldsymbol{g}_n(\boldsymbol{X}_n^i)$ or resampled with a probability $1 - \epsilon_n\, \boldsymbol{g}_n(\boldsymbol{X}_n^i)$. Resampling an island consists in replacing it by an island selected at random in the current population with weights proportional to their potential $(\boldsymbol{g}_n(\boldsymbol{X}_n^1)), \ldots, \boldsymbol{g}_n(\boldsymbol{X}_n^{N_2}))$.
- *Mutation step:* each selected island is updated independently according to the Markov transition $\boldsymbol{M}_{n+1}$.

These islands particles allow to build the N_1-particle approximation of the measures $\boldsymbol{\eta}_n$ and $\boldsymbol{\gamma}_n$, for any $\boldsymbol{f}_n \in \mathcal{B}_b(\mathbf{X}_n)$, as

$$\boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n) \stackrel{\text{def}}{=} \frac{1}{N_1} \sum_{i=1}^{N_1} \boldsymbol{f}_n(\boldsymbol{X}_n^i)\,,$$
$$\boldsymbol{\gamma}_n^{N_1}(\boldsymbol{f}_n) \stackrel{\text{def}}{=} \boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n) \prod_{0\leq p<n} \boldsymbol{\eta}_p^{N_1}(\boldsymbol{g}_p) = \boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n)\, \boldsymbol{\gamma}_n^{N_1}(\mathbb{1}_{\mathbf{X}_n})\,.$$

For this selection scheme, the following results, adapted from [Del Moral, 2004, Corollary 9.3.1, pp. 295-298], establishes the convergence of $(W_p^{N_2})_{1\leq p\leq n}$ to centered Gaussian fields:

Theorem 2.4.2. *For the ϵ_n-bootstrap filter, for any fixed time horizon $n \geq 1$, the sequence $(W_p^{N_2})_{1\leq p\leq n}$ defined in (2.3.10) converges in law, as N_2 goes to infinity, to a sequence of n independent centered Gaussian random fields $(W_p)_{0\leq p\leq n}$ with variance given, for any bounded function $f_p \in \mathcal{B}_b(\mathsf{X}_p)$, and $1 \leq p \leq n$, by*

$$\mathbb{E}\left[W_p(f_p)^2\right] = \eta_{p-1}\left[S_{p-1,\eta_{p-1}} M_p f_p^2 - \left(S_{p-1,\eta_{p-1}} M_p f_p\right)^2\right]\,. \tag{2.4.3}$$

This variance is smaller than the variance of the bootstrap algorithm.

Proposition 2.4.3. *The asymptotic variance of $\eta_n^{N_2}$ is smaller with respect to a non-zero sequence $(\epsilon_p)_{0\leq p\leq n-1}$ introduced in (2.4.1) than in the bootstrap algorithm.*

Proof. The proof is given in subsection 2.6.5. □

For example, for $\epsilon_p = \left(\text{essup}_{\eta_p}(g_p)\right)^{-1}$, $0 \leq p \leq n$ the asymptotic variance of $\eta_n^{N_2}(f_n)$, $0 \leq p \leq n$ is lower than for the bootstrap. We can also adapt it at the island level. For instance, Algorithm 5 describes

the $\epsilon_p = \left(\max_{1 \leq j \leq N_2} \boldsymbol{g}_p(\boldsymbol{X}_p^j) \right)^{-1}$-bootstrap islands interaction with ESS filter within the islands.

Data: η_0, $(g_p)_{p \geq 0}$, $(M_p)_{p \geq 1}$, $\alpha_{\text{Particles}}$
Result: Empirical estimate of η_n
/* Initialization */
for $i \leftarrow 1$ **to** N_1 **do**
  Set $\boldsymbol{\omega}_0^i = \left(\omega_0^{i,j}\right)_{j=1}^{N_2} = (1, \ldots, 1)$. Sample $\boldsymbol{X}_0^i = \left(X_0^{i,j}\right)_{j=1}^{N_2}$ independently distributed according to η_0.
end
for $p \leftarrow 0$ **to** $n-1$ **do**
  /* Island selection step */
  for $i \leftarrow 1$ **to** N_1 **do**
    - With probability $\boldsymbol{g}_p(\boldsymbol{X}_p^i) \,/\, \max_{1 \leq k \leq N_1} \boldsymbol{g}_p(\boldsymbol{X}_p^k)$, set $I_p^i = i$.
    - With probability $1 - \boldsymbol{g}_p(\boldsymbol{X}_p^i) \,/\, \max_{1 \leq k \leq N_1} \boldsymbol{g}_p(\boldsymbol{X}_p^k)$, sample I_p^i multinomially with probability proportional to $\{\boldsymbol{g}_p(\boldsymbol{X}_p^l) \,/\, \sum_{k=1}^{N_1} \boldsymbol{g}_p(\boldsymbol{X}_p^k)\}_{l=1}^{N_1}$.
  end
  /* Island mutation step */
  for $i \leftarrow 1$ **to** N_1 **do**
    /* Particle selection and weight updating within each island */
    Set $N_2^{\text{eff}} = \left(\sum_{j=1}^{N_2} \omega_p^{I_p^i,j} g_p(X_p^{I_p^i,j})\right)^2 \Big/ \sum_{j=1}^{N_2} \left(\omega_p^{I_p^i,j} g_p(X_p^{I_p^i,j})\right)^2$;
    if $N_2^{\text{eff}} \geq \alpha_{\text{Particles}} N_2$ **then**
      For $1 \leq j \leq N_2$, set $\omega_{p+1}^{i,j} = \omega_p^{I_p^i,j} g_p(X_p^{I_p^i,j})$;
      Set $\boldsymbol{J}_p^i = (J_p^{i,j})_{j=1}^{N_2} = (1, 2, \ldots, N_2)$;
    else
      Set $\boldsymbol{\omega}_{p+1}^i = \left(\omega_{p+1}^{i,j}\right)_{j=1}^{N_2} = (1, \ldots, 1)$;
      Sample $\boldsymbol{J}_p^i = (J_p^{i,j})_{j=1}^{N_2}$ multinomially with probability proportional to $\left(\omega_p^{I_p^i,j} g_p(X_p^{I_p^i,j})\right)_{j=1}^{N_2}$;
    end
    /* Particle mutation */
    For $1 \leq j \leq N_2$, sample independently $X_{p+1}^{i,j}$ according to $M_{p+1}(X_p^{I_p^i, L_p^{i,j}}, \cdot)$, where $L_p^{i,j} = J_p^{I_p^i,j}$.
  end
end
Approximate $\eta_n(f_n)$ by $\dfrac{1}{N_1} \displaystyle\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \frac{\omega_n^{i,j}}{\sum_{j=1}^{N_2} \omega_n^{i,j}} f_n\left(X_n^{i,j}\right)$.

Algorithm 5: ESS within ϵ_p-bootstrap interaction for $\epsilon_p = \left(\text{essup}_{\eta_p^{N_2}}(g_p)\right)^{-1}$

2.4.2 Effective Sample Size interaction

We describe the particle approximation of the probabilities $(\eta_n)_{n \geq 0}$ using the ESS method introduced in [Liu and Chen, 1995]; see also [Liu, 2008; Del Moral et al., 2012; Douc and Moulines, 2008; Arnaud and Le Gland, 2009]. The difference with the bootstrap filter stems from the selection step of the current particles which is

not performed at each step, but only when the importance weights do not satisfy some appropriately defined criterion. Contrary to the bootstrap filter, we now keep both the particles and the weights. Denote by x_n^i a particle and w_n^i its associated weight, assumed to be nonnegative. For a weighted sample $\{(w_n^i, x_n^i)\}_{i=1}^{N_2}$, the criterion

$$\left(\sum_{i=1}^{N_2} w_n^i g_n(x_n^i)\right)^2 / \sum_{i=1}^{N_2} \left(w_n^i g_n(x_n^i)\right)^2$$

is the ESS. The algorithm goes as follows. When the ESS is less than αN_2, for some $\alpha \in (0,1)$, the particles are multinomially resampled with probabilities proportional to their weights times their potential functions and the weights are all reset to 1. When the ESS is greater than αN_2, then the weights are simply multiplied by the potential function. The selected particles are then updated using the transition kernel M_{n+1}. For any nonnegative integer p we set $(\mathsf{X}_p, \boldsymbol{\mathcal{X}}_p) \stackrel{\text{def}}{=} ((\mathsf{X}_p \times \mathbb{R}^+)^{N_2}, (\mathcal{X}_p \otimes \mathcal{B}(\mathbb{R}^+))^{\otimes N_2})$. Introduce the following set

$$\Theta_{n,\alpha} = \left\{ \mathbf{x}_n = \left[(x_n^1, w_n^1), \ldots, (x_n^{N_2}, w_n^{N_2})\right] \in \mathsf{X}_n \middle| \frac{\left(\sum_{i=1}^{N_2} w_n^i g_n(x_n^i)\right)^2}{\sum_{i=1}^{N_2} \left(w_n^i g_n(x_n^i)\right)^2} \geq \alpha N_2 \right\} .$$

Define the Markov kernel $\boldsymbol{M}_{n+1}$ from X_n into X_{n+1} by

$$\begin{aligned} \boldsymbol{M}_{n+1}(\mathbf{x}_n, \mathrm{d}\mathbf{x}_{n+1}) \stackrel{\text{def}}{=} & \mathbb{1}_{\Theta_{n,\alpha}}(\mathbf{x}_n) \left[\prod_{1 \leq i \leq N_2} \delta_{w_n^i g_n(x_n^i)}(\mathrm{d}w_{n+1}^i) M_{n+1}(x_n^i, \mathrm{d}x_{n+1}^i) \right] \\ & + \mathbb{1}_{\Theta_{n,\alpha}^{\mathrm{c}}}(\mathbf{x}_n) \left[\prod_{1 \leq i \leq N_2} \delta_1(w_{n+1}^i) \sum_{j=1}^{N_2} \frac{w_n^j g_n(x_n^j)}{\sum_{k=1}^{N_2} w_n^k g_n(x_n^k)} M_{n+1}(x_n^j, \mathrm{d}x_{n+1}^i) \right] , \end{aligned} \tag{2.4.4}$$

where $\mathbf{x}_n = \left[(x_n^1, w_n^1), \ldots, (x_n^{N_2}, w_n^{N_2})\right] \in \mathsf{X}_n$ and $\Theta_{n,\alpha}^{\mathrm{c}}$ is the complement of $\Theta_{n,\alpha}$. We define a Markov chain $(\boldsymbol{X}_n)_{n \geq 0}$ where for each $n \in \mathbb{N}$,

$$\boldsymbol{X}_n = \left[(X_n^1, \omega_n^1), \ldots, (X_n^{N_2}, \omega_n^{N_2})\right] \in \mathsf{X}_n , \tag{2.4.5}$$

with initial distribution $\boldsymbol{\eta}_0 \stackrel{\text{def}}{=} (\eta_0 \otimes \delta_1)^{\otimes N_2}$ and transition kernel $\boldsymbol{M}_{n+1}$. Equation (2.2.1) suggests the following N_2-particle approximations of the measures η_n and γ_n defined for $f_n \in \mathcal{B}_b(\mathsf{X}_n)$ by

$$\eta_n^{N_2}(f_n) \stackrel{\text{def}}{=} \frac{1}{\sum_{i=1}^{N_2} \omega_n^i} \sum_{i=1}^{N_2} \omega_n^i f_n\left(X_n^i\right) = m^{N_2} f_n(\boldsymbol{X}_n) , \tag{2.4.6}$$

$$\gamma_n^{N_2}(f_n) \stackrel{\text{def}}{=} \eta_n^{N_2}(f_n) \prod_{0 \leq p < n} \eta_p^{N_2}(g_p) = \eta_n^{N_2}(f_n)\ \gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n}) , \tag{2.4.7}$$

where m^{N_2} stands for the operator given for any $f_n \in \mathcal{B}_b(\mathsf{X}_n)$ by

$$m^{N_2} f_n(\mathbf{x}_n) \stackrel{\text{def}}{=} \frac{1}{\sum_{i=1}^{N_2} w_n^i} \sum_{i=1}^{N_2} w_n^i f_n(x_n^i) .$$

For $\mathbf{x}_n = ((x_n^1, w_n^1), \ldots, (x_n^{N_2}, w_n^{N_2})) \in \mathsf{X}_n$, define the potential function

$$\boldsymbol{g}_n(\mathbf{x}_n) \stackrel{\text{def}}{=} m^{N_2} g_n(\mathbf{x}_n) = \frac{1}{\sum_{i=1}^{N_2} w_n^i} \sum_{i=1}^{N_2} w_n^i g_n\left(x_n^i\right) . \tag{2.4.8}$$

We can define the island Feynman-Kac model associated to the Markov chain (2.4.4) and the potential function (2.4.8), as in (2.2.11), (2.2.12) and(2.2.13).

Theorem 2.4.4. *For a particle system* $\mathbf{x}_n = ((x_n^1, w_n^1), \ldots, (x_n^{N_2}, w_n^{N_2})) \in \mathsf{X}_n$ *generated by the* ESS *algorithm and for any* $\boldsymbol{f}_n \in \mathcal{B}_b(\mathsf{X}_n)$ *of the form*

$$\boldsymbol{f}_n(\mathbf{x}_n) = \left(\sum_{i=1}^{N_2} w_n^i\right)^{-1} \sum_{i=1}^{N_2} w_n^i f_n\left(x_n^i\right)$$

where $f_n \in \mathcal{B}_b(\mathsf{X}_n)$,

$$\boldsymbol{\gamma}_n(\boldsymbol{f}_n) = \gamma_n(f_n) \quad \textit{and} \quad \boldsymbol{\eta}_n(\boldsymbol{f}_n) = \eta_n(f_n)\,. \tag{2.4.9}$$

Proof. See subsection 2.6.6. □

For each $n \in \mathbb{N}$, let $(\boldsymbol{X}_n^1, \ldots, \boldsymbol{X}_n^{N_1}) \in \mathsf{X}_n^{N_1}$ be a population of N_1 islands each of N_2 individuals. We associate to each island, a weight Ω_n^i, for $i \in \{1, \ldots, N_1\}$. We can also make the islands interact using an ESS criterion. The process $((\boldsymbol{X}_n^1, \Omega_n^1), \ldots, (\boldsymbol{X}_n^{N_1}, \Omega_n^{N_1}))$ is a Markov chain which evolves according to selection and mutation steps, defined as follows.

- *Selection step*: if the ESS criterion $\left(\sum_{i=1}^{N_1} \Omega_n^i \boldsymbol{g}_n(\boldsymbol{X}_n^i)\right)^2 / \sum_{i=1}^{N_1} \left(\Omega_n^i \boldsymbol{g}_n(\boldsymbol{X}_n^i)\right)^2$ is larger than βN_1 for one $\beta \in (0,1)$, we do not resample the islands and we update the weights thanks to the potential function $\Omega_{n+1}^i = \Omega_n^i \boldsymbol{g}_n(\boldsymbol{X}_n^i)$; otherwise, we resample the islands with probability proportional to $\{\Omega_n^i \boldsymbol{g}_n(\boldsymbol{X}_n^i)\}_{i=1}^{N_1}$ and the weights are all reset to 1.

- *Mutation step:* each selected island is updated independently according to the Markov transition $\boldsymbol{M}_{n+1}$.

These islands particles allow to define the N_1-particle approximation of the measures $\boldsymbol{\eta}_n$ and $\boldsymbol{\gamma}_n$, for any $\boldsymbol{f}_n \in \mathcal{B}_b(\mathsf{X}_n)$, as

$$\begin{aligned}
\boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n) &\stackrel{\text{def}}{=} \frac{1}{\sum_{i=1}^{N_1} \Omega_n^i} \sum_{i=1}^{N_1} \Omega_n^i \boldsymbol{f}_n(\boldsymbol{X}_n^i)\,, \\
\boldsymbol{\gamma}_n^{N_1}(\boldsymbol{f}_n) &\stackrel{\text{def}}{=} \boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n) \prod_{0 \le p < n} \boldsymbol{\eta}_p^{N_1}(\boldsymbol{g}_p) = \boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n)\, \boldsymbol{\gamma}_n^{N_1}(\mathbb{1}_{\mathbf{x}_n})\,.
\end{aligned}$$

Algorithm 6 describes the ESS within ESS island filter.

Data: η_0, $(g_p)_{p\geq 0}$, $(M_p)_{p\geq 1}$, $\alpha_{\text{Particles}}$
Result: Empirical estimate of η_n
/* Initialization */
for $i \leftarrow 1$ **to** N_1 **do**
  Set $\boldsymbol{\omega}_0^i = \left(\omega_0^{i,j}\right)_{j=1}^{N_2} = (1,\ldots,1)$;
  Sample $\boldsymbol{X}_0^i = \left(X_0^{i,j}\right)_{j=1}^{N_2}$ independently distributed according to η_0;
end
for $p \leftarrow 0$ **to** $n-1$ **do**
  /* Island selection step */
  Sample $\boldsymbol{I}_p = (I_p^i)_{i=1}^{N_1}$ multinomially with probability proportional to $\left(\boldsymbol{g}_p(\boldsymbol{X}_p^i, \boldsymbol{\omega}_p^i)\right)_{i=1}^{N_1}$;
  /* Island mutation step */
  for $i \leftarrow 1$ **to** N_1 **do**
    /* Particle selection and weight updating within each island */
    Set $N_2^{\text{eff}} = \left(\sum_{j=1}^{N_2} \omega_p^{I_p^i,j} g_p(X_p^{I_p^i,j})\right)^2 / \sum_{j=1}^{N_2} \left(\omega_p^{I_p^i,j} g_p(X_p^{I_p^i,j})\right)^2$;
    if $N_2^{\text{eff}} \geq \alpha_{\text{Particles}} N_2$ **then**
      For $1 \leq j \leq N_2$, set $\omega_{p+1}^{i,j} = \omega_p^{I_p^i,j} g_p(X_p^{I_p^i,j})$;
      Set $\boldsymbol{J}_p^i = (J_p^{i,j})_{j=1}^{N_2} = (1,2,\ldots,N_2)$.
    else
      Set $\boldsymbol{\omega}_{p+1}^i = \left(\omega_{p+1}^{i,j}\right)_{j=1}^{N_2} = (1,\ldots,1)$;
      Sample $\boldsymbol{J}_p^i = (J_p^{i,j})_{j=1}^{N_2}$ multinomially with probability proportional to $\left(\omega_p^{I_p^i,j} g_p(X_p^{I_p^i,j})\right)_{j=1}^{N_2}$
    end
    /* Particle mutation */
    For $1 \leq j \leq N_2$, sample independently $X_{p+1}^{i,j}$ according to $M_{p+1}(X_p^{I_p^i,L_p^{i,j}}, \cdot)$, where $L_p^{i,j} = J_p^{I_p^i,j}$.
  end
end
Approximate $\eta_n(f_n)$ by $\dfrac{1}{N_1}\displaystyle\sum_{i=1}^{N_1}\sum_{j=1}^{N_2} \frac{\omega_n^{i,j}}{\sum_{j'=1}^{N_2}\omega_n^{i,j'}} f_n\left(X_n^{i,j}\right)$.

Algorithm 6: ESS within bootstrap island filter (also called B^2ASIL)

2.5 Numerical simulations

Example 1 (Linear Gaussian Model). In order to assess numerically the previous results, we now consider the Linear Gaussian Model (LGM) defined by:

$$X_{p+1} = \phi X_p + \sigma_u U_p \ , \quad Y_p = X_p + \sigma_v V_p \ ,$$

where $X_0 \sim \mathcal{N}\left(0, \sigma_u^2/(1-\phi^2)\right)$, $\{U_p\}_{p\geq 1}$ and $\{V_p\}_{p\geq 1}$ are independent sequences of independent and identically distributed (i. i. d.) standard Gaussian random variables, independent of X_0. In the simulations, we have used $n = 150$ observations, generated using the model with $\phi = 0.9$, $\sigma_u = 0.6$ and $\sigma_v = 1$. We focus on the filtering problem, consisting in computing the distribution of the state X_n given $Y_0, \cdots, Y_n$. This

problem can be cast in the Feynman-Kac framework by setting for all $p \geq 0$

$$M_{p+1}(x_p, \mathrm{d}x_{p+1}) = \frac{1}{\sqrt{2\pi}\sigma_u} \exp\left[-\frac{(x_{p+1} - \phi x_p)^2}{2\sigma_u^2}\right] \mathrm{d}x_{p+1} \ ,$$
$$g_p(x_p) = \frac{1}{\sqrt{2\pi}\sigma_v} \exp\left[-\frac{(y_p - x_p)^2}{2\sigma_v^2}\right] \ .$$

We estimate the mean of the latent state $\mathbb{E}[X_n \mid Y_0, \ldots, Y_n]$. We compare the results obtained for different interactions across the islands and for different values of N_1 and N_2; in all the simulations, the bootstrap filter is used within the islands. We have run the simulations independently 250 times and we have compared these estimators with the value computed using the Kalman filter. Figure 2.2 displays the boxplots of the 250 values of these estimators. As expected, for small values of N_2 compared to N_1, the bias of independent islands

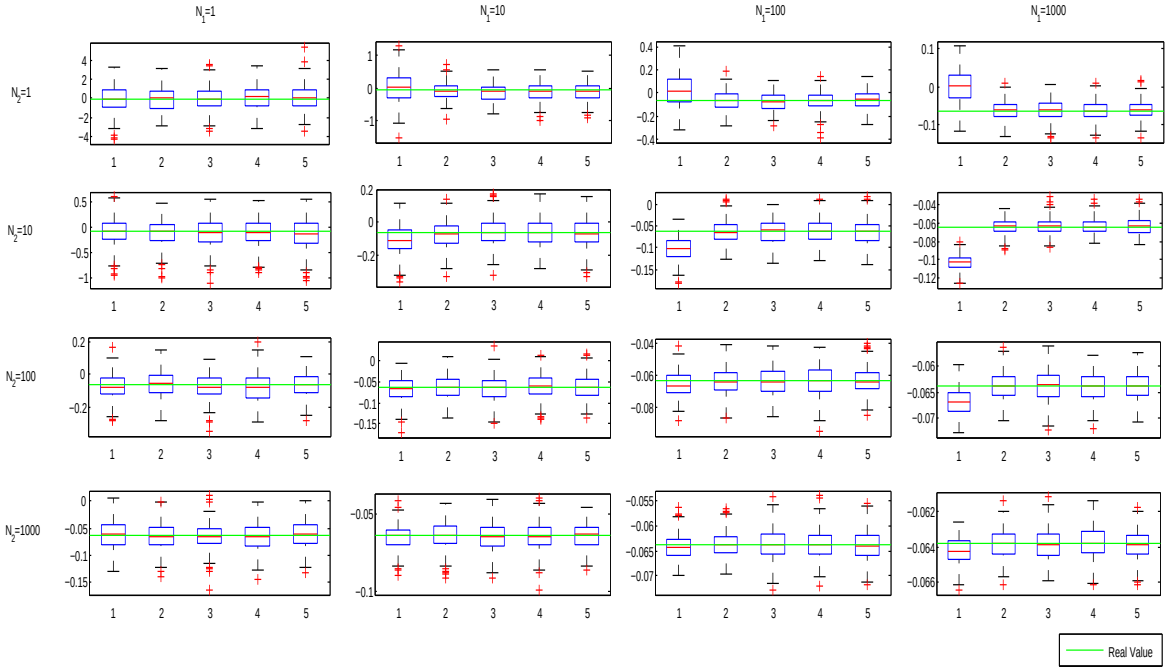


Figure 2.2: Comparison of different interactions across the islands with bootstrap within each island for the LGM (1) Bootstrap/independent; (2) B^{2}ASIL; (3) B^2; (4) Bootstrap/$(1/\|g_n\|_\infty)$)-bootstrap; (5) Bootstrap/essup$_{\eta_p^{N_2}}(g_n)$-bootstrap

is large compared to cases where islands interact; on the contrary, the variance is smaller for independent islands than for bootstrap island interaction. In this example, the type of interaction between islands does not have a significant impact on the dispersion of the estimator (the bias is negligible).

An important aspect for the efficiency of the algorithms is the number of interactions between islands. The smaller this number is, the quicker the algorithm will be. The number of interactions in the bootstrap case is nN_1. We have compared the island interactions number for the ϵ_p-bootstrap and the ESS interactions w. r. t. the bootstrap one, when we apply the bootstrap filter within the islands. We have computed the

N_2 \ N_1	1		10		100		1000	
1	0	150	631	1500	6853	15000	68851	150000
10	0	150	387	1500	5244	15000	58918	150000
100	0	150	162	1500	2473	15000	30262	150000
1000	0	150	57	1500	915	15000	11593	150000

Table 2.3: Island interactions number using bootstrap within ϵ_p-bootstrap and B^2 for the LGM.

N_2 \ N_1	1		10		100		1000	
1	0	150	718	1500	8052	15000	80656	150000
10	0	150	190	1500	2419	15000	25772	150000
100	0	150	21	1500	247	15000	2472	150000
1000	0	150	0	1500	0	15000	0	150000

Table 2.4: Island interactions number using B^{2}ASIL and B^2 for the LGM.

empirical number of interactions over the 250 simulations; the results are respectively given in tables 2.3 and 2.4. For a given number of islands, the island interactions number for the ESS and the ϵ_p-bootstrap decrease when the island size grows, whereas it is constant for the bootstrap. The island interactions number is always much smaller using the ESS or the ϵ_p-bootstrap than the bootstrap, across the islands. Moreover, as soon as the number of particles in each island is large enough, the ESS is no longer resampling the islands.

Theorem 2.4.2 assures that the variance is smaller using the ϵ_p-bootstrap than the bootstrap interaction. The variance gain using ϵ_p-bootstrap or ESS instead of bootstrap across the islands is given in table 2.5. The bootstrap interaction is applied within the islands. The variance is significantly reduced using the ϵ_p-bootstrap or the ESS interaction across the islands, instead of the bootstrap, up to 41 percent variance reduction.

Example 2 (Stochastic volatility model). We consider the stochastic volatility model:

$$X_{p+1} = \alpha X_p + \sigma U_{p+1} , \quad Y_p = \beta \mathrm{e}^{\frac{X_p}{2}} V_p ,$$

where $X_0 \sim \mathcal{N}\left(0, \sigma^2/(1-\alpha^2)\right)$, $\{U_p\}_{p\geq 0}$ and $\{V_p\}_{p\geq 0}$ are independent sequences of standard Gaussian random variables independent of X_0. In the simulations, we have used $n = 100$ observations, generated

N_2 \ N_1	10		100		1000	
10	24.7	27.4	27	14.7	23	3.2
100	24.7	29.7	16.1	12	23	27.6
1000	34.1	29.2	26.4	41.4	17	31.7

Table 2.5: Percentage of the variance gain using bootstrap within ϵ_p-bootstrap on the left side and B^{2}ASIL on the right side, compared to the B^2, in the LGM example.

using the model with $\alpha = 0.98$, $\sigma = 0.5$ and $\beta = 1$. We estimate the mean of the latent state X_n given the observations $Y_0, \cdots, Y_n$. This problem can be cast in the Feynman-Kac framework by setting for all $p \geq 0$

$$M_{p+1}(x_p, \mathrm{d}x_{p+1}) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(x_{p+1} - \alpha x_p)^2}{2\sigma^2}\right] \mathrm{d}x_{p+1} \ ,$$
$$g_p(x_p) = \frac{1}{\sqrt{2\pi}\beta} \exp\left[-\frac{\beta^2 x_p + y_p^2 \mathrm{e}^{-x_p}}{2\beta^2}\right] \ .$$

We have computed this quantity using a single run of bootstrap filter with 10^6 particles. In the following results, we always consider bootstrap interaction within each island, and we compare different interactions across the islands, for several values of N_1 and N_2. We have run the simulations independently 250 times. Figure 2.6 displays the boxplots of the 250 values of these estimators. The behavior of the different methods is similar to the one observed for the Linear Gaussian Model example.

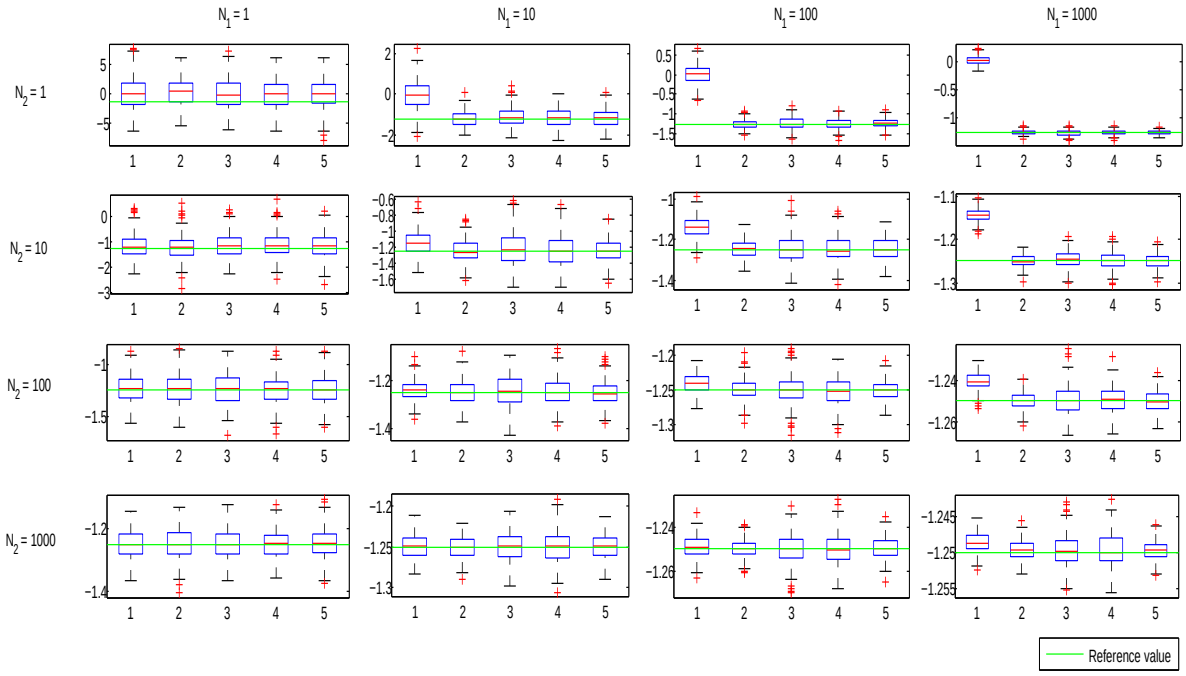


Figure 2.6: Comparison of different interactions across the islands with bootstrap within each island for the Stochastic volatility model (1) Bootstrap/independent; (2) B^{2}ASIL ; (3) B^2; (4) Bootstrap/$(1/\|g_n\|_\infty)$-bootstrap; (5) Bootstrap/essup$_{\eta_p^{N_2}}(g_n)$-bootstrap

We have compared the island interactions number for the ϵ_p-bootstrap and the ESS interactions w. r. t. the bootstrap one, when we apply the bootstrap filter within the islands. We have computed the empirical number of interactions over the 250 simulations; the results are respectively given in tables 2.7 and 2.8. The number of interactions in the bootstrap case is nN_1. The same phenomena are observed as for the Linear Gaussian Model example.

N_2 \ N_1	1		10		100		1000	
1	0	100	332	1000	4021	10000	42185	100000
10	0	100	221	1000	3069	10000	34789	100000
100	0	100	100	1000	1523	10000	18647	100000
1000	0	100	36	1000	577	10000	7332	100000

Table 2.7: Island interactions number using bootstrap within ϵ_p-bootstrap and B^2 for the Stochastic volatility model.

N_2 \ N_1	1		10		100		1000	
1	0	100	301	1000	3514	10000	36108	100000
10	0	100	109	1000	1229	10000	12096	100000
100	0	100	15	1000	186	10000	1956	100000
1000	0	100	0	1000	0	10000	0	100000

Table 2.8: Island interactions number using B^{2}ASIL and B^2 for the Stochastic volatility example.

The variance gain using the ϵ_p-bootstrap or the ESS instead of the bootstrap across the islands is given in table 2.9. The bootstrap interaction is applied within the islands. The variance is significantly reduced using the ϵ_p-bootstrap or the ESS interaction across the islands, instead of the bootstrap, up to 66 percent variance reduction.

N_2 \ N_1	10		100		1000	
10	44.2	57.8	35.3	57.2	30.4	50.7
100	46.4	49.3	52.2	44.6	46.8	65
1000	30.4	41.7	49.6	66.9	55.8	61.4

Table 2.9: Percentage of the variance gain using bootstrap within ϵ_p-bootstrap on the left side and B^{2}ASIL on the right side, compared to the B^2, in the Stochastic volatility example.

2.6 Proofs

2.6.1 Proof of Theorem 2.2.1

Using (2.2.8) and (2.2.10), $\boldsymbol{g}_n(\boldsymbol{X}_n)$ may be expressed as $\boldsymbol{g}_n(\boldsymbol{X}_n) = \eta_n^{N_2}(g_n)$ where $\boldsymbol{X}_n$ and $\eta_n^{N_2}$ are defined in (2.2.8) and (2.2.7), respectively. Similarly, for any $\boldsymbol{f}_n \in \mathcal{B}_b(\mathbf{X}_n)$ of the form $\boldsymbol{f}_n(\mathbf{x}_n) = N_2^{-1}\sum_{i=1}^{N_2} f_n(x_n^i)$ where $f_n \in \mathcal{B}_b(\mathsf{X}_n)$, $\boldsymbol{f}_n(\boldsymbol{X}_n)$ is given by $\boldsymbol{f}_n(\boldsymbol{X}_n) = \eta_n^{N_2}(f_n)$. Note that

$$\boldsymbol{\gamma}_n(\boldsymbol{f}_n) \stackrel{\text{def}}{=} \mathbb{E}\left[\boldsymbol{f}_n(\boldsymbol{X}_n)\prod_{0\le p<n}\boldsymbol{g}_p(\boldsymbol{X}_p)\right] = \mathbb{E}\left[\eta_n^{N_2}(f_n)\prod_{0\le p<n}\eta_p^{N_2}(g_p)\right], \tag{2.6.1}$$

and since by (2.2.9), it suffices to prove that $\gamma_n^{N_2}(f_n)$ is an unbiased estimator of $\gamma_n(f_n)$, i.e.

$$\mathbb{E}\left[\gamma_n^{N_2}(f_n)\right] = \gamma_n(f_n). \tag{2.6.2}$$

Define the filtration $\mathcal{F}_n^{N_2} \stackrel{\text{def}}{=} \sigma\left(\boldsymbol{X}_p, 0\le p\le n\right)$. Note that

$$\begin{aligned}\mathbb{E}\left[\eta_p^{N_2}(f_p) \mid \mathcal{F}_{p-1}^{N_2}\right] = \frac{1}{N_2}\sum_{i=1}^{N_2}\mathbb{E}\left[f_p(X_p^i) \mid \mathcal{F}_{p-1}^{N_2}\right] &= \mathbb{E}\left[f_p(X_p^1) \mid \mathcal{F}_{p-1}^{N_2}\right] \\ &= \frac{\sum_{i=1}^{N_2} g_{p-1}(X_{p-1}^i)M_p f_p(X_{p-1}^i)}{\sum_{i=1}^{N_2} g_{p-1}(X_{p-1}^i)} = \frac{\eta_{p-1}^{N_2}(Q_p f_p)}{\eta_{p-1}^{N_2}(g_{p-1})},\end{aligned} \tag{2.6.3}$$

where Q_p is defined in (2.3.1).

By the definition of $\gamma_n^{N_2}$ given in (2.2.9), one has

$$\begin{aligned}\mathbb{E}\left[\gamma_n^{N_2}(f_n)\right] = \mathbb{E}\left[\mathbb{E}\left[\eta_n^{N_2}(f_n) \mid \mathcal{F}_{n-1}^{N_2}\right]\prod_{0\le p<n}\eta_p^{N_2}(g_p)\right] &= \mathbb{E}\left[\frac{\eta_{n-1}^{N_2}(Q_n f_n)}{\eta_{n-1}^{N_2}(g_{n-1})}\prod_{0\le p<n}\eta_p^{N_2}(g_p)\right] \\ &= \mathbb{E}\left[\eta_{n-1}^{N_2}(Q_n f_n)\prod_{0\le p<n-1}\eta_p^{N_2}(g_p)\right].\end{aligned} \tag{2.6.4}$$

By iterating this step one gets

$$\begin{aligned}\mathbb{E}\left[\gamma_n^{N_2}(f_n)\right] &= \mathbb{E}\left[\eta_0^{N_2}(Q_1\cdots Q_n f_n)\right] = \mathbb{E}\left[Q_1\cdots Q_n f_n(X_0^1)\right] \\ &= \gamma_0 Q_1\cdots Q_n f_n = \gamma_n(f_n).\end{aligned}$$

2.6.2 Proof of Theorem 2.3.3

We preface the proof by the following Lemma.

Lemma 2.6.1. *For any $f_n^1, f_n^2 \in \mathcal{B}_b(\mathsf{X}_n)$, the pair $(W_n^{\gamma,N_2}(f_n^1), W_n^{\eta,N_2}(f_n^2))$ converges in law, as N_2 tends to infinity, to $(W_n^{\gamma}(f_n^1), W_n^{\eta}(f_n^2))$. In addition, for any polynomial function $\Phi : \mathbb{R}^2 \to \mathbb{R}$, one has:*

$$\lim_{N_2\to\infty}\mathbb{E}\left[\Phi\left(W_n^{\gamma,N_2}(f_n^1), W_n^{\eta,N_2}(f_n^2)\right)\right] = \mathbb{E}\left[\Phi\left(W_n^{\gamma}(f_n^1), W_n^{\eta}(f_n^2)\right)\right].$$

Proof. For any $(\alpha,\beta)\in\mathbb{R}^2$ by the definitions (2.3.9) of W_n^{γ,N_2} and (2.3.12) of W_n^{η,N_2} one has

$$\alpha W_n^{\gamma,N_2}(f_n^1) + \beta W_n^{\eta,N_2}(f_n^2) = \sum_{p=0}^{n}\left[\alpha\gamma_p^{N_2}(\mathbb{1}_{\mathsf{X}_p})W_p^{N_2}(Q_{p,n}f_n^1) + \beta\frac{\gamma_p^{N_2}(\mathbb{1}_{\mathsf{X}_p})}{\gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n})}W_p^{N_2}(Q_{p,n}(f_n^2 - \eta_n(f_n^2)))\right].$$

As in the proof of Theorem 2.3.1, a simple application of Slutsky's Lemma allows to show that $\alpha W_n^{\gamma,N_2}(f_n^1)+\beta W_n^{\eta,N_2}(f_n^2)$ converges in law to $\alpha W_n^{\gamma}(f_n^1)+\beta W_n^{\eta}(f_n^2)$. The proof follows from [Del Moral, 2004, Theorem 7.4.4], using that for any $p \geq 1$,

$$\sup_{N_2 \geq 1} \mathbb{E}\left[\left|W_n^{\gamma,N_2}(f_n^1)\right|^p\right]^{1/p} \leq c_p(n)||f_n^1|| \,, \tag{2.6.5}$$

$$\sup_{N_2 \geq 1} \mathbb{E}\left[\left|W_n^{\eta,N_2}(f_n^2)\right|^p\right]^{1/p} \leq c_p(n)||f_n^2|| \,, \tag{2.6.6}$$

for some finite constant $c_p(n)$ depending only on p and n. □

Proof. of Theorem 2.3.3 Consider first the bias term. We decompose the error as follows using (2.3.12):

$$\begin{aligned} N_2\left[\eta_n^{N_2}(f_n)-\eta_n(f_n)\right] &= \sqrt{N_2}W_n^{\eta,N_2}(f_n) = \sqrt{N_2}\frac{\gamma_n(\mathbb{1}_{\mathsf{X}_n})}{\gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n})}W_n^{\gamma,N_2}\left(\frac{f_n-\eta_n(f_n)}{\gamma_n(\mathbb{1}_{\mathsf{X}_n})}\right) \\ &= \sqrt{N_2}\left[\frac{\gamma_n(\mathbb{1}_{\mathsf{X}_n})}{\gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n})}-1\right]W_n^{\gamma,N_2}\left(\frac{f_n-\eta_n(f_n)}{\gamma_n(\mathbb{1}_{\mathsf{X}_n})}\right)+\sqrt{N_2}W_n^{\gamma,N_2}\left(\frac{f_n-\eta_n(f_n)}{\gamma_n(\mathbb{1}_{\mathsf{X}_n})}\right) . \end{aligned}$$

Since $W_n^{\gamma,N_2}=\sqrt{N_2}\left[\gamma_n^{N_2}-\gamma_n\right]$, Theorem 2.2.1 shows that, the expectation of the second term on the RHS of the previous equation, is zero. By noting that

$$\left[\frac{\gamma_n(\mathbb{1}_{\mathsf{X}_n})}{\gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n})}-1\right] = -\frac{1}{\gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n})}[\gamma_n^{N_2}-\gamma_n](\mathbb{1}_{\mathsf{X}_n}) = -\frac{1}{\sqrt{N_2}}\frac{1}{\gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n})}W_n^{\gamma,N_2}(\mathbb{1}_{\mathsf{X}_n}) \,,$$

where W_n^{γ,N_2} is defined in (2.3.9), one gets

$$\begin{aligned} N_2\mathbb{E}\left[\eta_n^{N_2}(f_n)-\eta_n(f_n)\right] &= -\mathbb{E}\left[\frac{1}{\gamma_n^{N_2}(\mathbb{1}_{\mathsf{X}_n})}W_n^{\gamma,N_2}(\mathbb{1}_{\mathsf{X}_n})W_n^{\gamma,N_2}\left(\frac{f_n-\eta_n(f_n)}{\gamma_n(\mathbb{1}_{\mathsf{X}_n})}\right)\right] \\ &= -\frac{1}{\gamma_n(\mathbb{1}_{\mathsf{X}_n})}\mathbb{E}\left[W_n^{\gamma,N_2}(\mathbb{1}_{\mathsf{X}_n})W_n^{\eta,N_2}(f_n)\right] \,, \end{aligned}$$

where W_n^{η,N_2} is given in (2.3.12). According to Lemma 2.6.1:

$$\lim_{N_2\to\infty} N_2\mathbb{E}\left[\eta_n^{N_2}(f_n)-\eta_n(f_n)\right] = -\frac{1}{\gamma_n(\mathbb{1}_{\mathsf{X}_n})}\mathbb{E}\left[W_n^{\gamma}(\mathbb{1}_{\mathsf{X}_n})W_n^{\eta}(f_n)\right] = B_n(f_n) \,, \tag{2.6.7}$$

by the definitions of W_n^{γ} and W_n^{η}. Consider now the variance. We use the decomposition

$$\mathbb{V}\mathrm{ar}\left(\eta_n^{N_2}(f_n)\right) = \mathbb{E}\left[\left(\eta_n^{N_2}(f_n)-\eta_n(f_n)\right)^2\right]-\left\{\mathbb{E}\left[\eta_n^{N_2}(f_n)-\eta_n(f_n)\right]\right\}^2 .$$

Using (2.6.7), one gets $\left\{\mathbb{E}\left[\eta_n^{N_2}(f_n)-\eta_n(f_n)\right]\right\}^2 = O(N_2^{-2})$. From the definition (2.3.12) of W_n^{η,N_2}, it follows $\mathbb{E}\left[\left(\eta_n^{N_2}(f_n)-\eta_n(f_n)\right)^2\right] = N_2^{-1}\mathbb{E}\left[W_n^{\eta,N_2}(f_n)^2\right]$, implying that $\lim_{N_2\to\infty} N_2\mathbb{V}\mathrm{ar}\left(\eta_n^{N_2}(f_n)\right) = \mathbb{E}\left[W_n^{\eta}(f_n)^2\right] = V_n(f_n)$, by the definition of W_n^{η} and using again Lemma 2.6.1. □

2.6.3 Proof of Theorem 2.3.4

We preface the proof of Theorem 2.3.4 by the following result on the usual Feynman-Kac model.

Lemma 2.6.2. *For any time horizon $n \geq 0$ and any functions $f_n^1, f_n^2 \in \mathcal{B}_b(\mathsf{X}_n)$ such that $\eta_n(f_n^1)=0$, one has*

$$\lim_{N_2\to\infty} N_2\mathbb{E}\left[\eta_n^{N_2}(f_n^1)\eta_n^{N_2}(f_n^2)\prod_{p=0}^{n-1}\eta_p^{N_2}(g_p)\right] = \gamma_n(\mathbb{1}_{\mathsf{X}_n})\sum_{p=0}^{n}\mathbb{E}\left[W_p(P_{p,n}(f_n^1))W_p(P_{p,n}(f_n^2-\eta_n(f_n^2)))\right] .$$

Proof. By the definition (2.2.9) of $\gamma_n^{N_2}$ one has $\eta_n^{N_2}(f_n^1)\prod_{p=0}^{n-1}\eta_p^{N_2}(g_p) = \gamma_n^{N_2}(f_n^1)$, and, according to (2.6.2), $\mathbb{E}\left[\gamma_n^{N_2}(f_n^1)\right] = \gamma_n(f_n^1) = \gamma_n(\mathbb{1}_{\mathsf{X}_n})\eta_n(f_n^1) = 0$, so that one gets

$$\begin{aligned}
&\mathbb{E}\left[\eta_n^{N_2}(f_n^1)\eta_n^{N_2}(f_n^2)\prod_{p=0}^{n-1}\eta_p^{N_2}(g_p)\right] = \mathbb{E}\left[\gamma_n^{N_2}(f_n^1)\eta_n^{N_2}(f_n^2)\right] \\
&= \mathbb{E}\left[\gamma_n^{N_2}(f_n^1)\left(\eta_n^{N_2}(f_n^2) - \eta_n(f_n^2)\right)\right] + \mathbb{E}\left[\gamma_n^{N_2}(f_n^1)\right]\eta_n(f_n^2) \\
&= \mathbb{E}\left[\left(\gamma_n^{N_2}(f_n^1) - \gamma_n(f_n^1)\right)\left(\eta_n^{N_2}(f_n^2) - \eta_n(f_n^2)\right)\right] = \frac{1}{N_2}\mathbb{E}\left[W_n^{\gamma,N_2}(f_n^1)W_n^{\eta,N_2}(f_n^2)\right] .
\end{aligned}$$

Then, Lemma 2.6.1 gives

$$\lim_{N_2\to\infty} N_2\mathbb{E}\left[\eta_n^{N_2}(f_n^1)\eta_n^{N_2}(f_n^2)\prod_{p=0}^{n-1}\eta_p^{N_2}(g_p)\right] = \mathbb{E}\left[W_n^{\eta}(f_n^2)W_n^{\gamma}(f_n^1)\right] ,$$

where W_n^{γ} and W_n^{η} are given by (2.3.15) and (2.3.16). □

Lemma 2.6.3. *For any time horizon $n \geq 1$, and any linear function $\boldsymbol{f}_n \in \mathcal{B}_b(\mathbf{X}_n)$ of the form*

$$\boldsymbol{f}_n(\mathbf{x}_n) = m^{N_2} f_n(\mathbf{x}_n) , \quad \text{where } f_n \in \mathcal{B}_b(\mathsf{X}_n) ,$$

one has

$$\boldsymbol{Q}_n\boldsymbol{f}_n(\mathbf{x}_{n-1}) = m^{N_2} Q_n f_n(\mathbf{x}_{n-1}) , \tag{2.6.8}$$

$$\boldsymbol{Q}_{p,n}\boldsymbol{f}_n(\mathbf{x}_p) = m^{N_2} Q_{p,n} f_n(\mathbf{x}_p) , \qquad \text{for any } p \leq n , \tag{2.6.9}$$

$$\boldsymbol{P}_{p,n}\boldsymbol{f}_n(\mathbf{x}_p) = m^{N_2} P_{p,n} f_n(\mathbf{x}_p) \qquad \text{for any } p \leq n . \tag{2.6.10}$$

Proof. One has from (2.6.3)

$$\boldsymbol{M}_n\boldsymbol{f}_n(\mathbf{x}_{n-1}) = \mathbb{E}\left[\boldsymbol{f}_n(\boldsymbol{X}_n) \mid \boldsymbol{X}_{n-1} = \mathbf{x}_{n-1}\right] = \mathbb{E}\left[m^{N_2} f_n(\boldsymbol{X}_n) \mid \boldsymbol{X}_{n-1} = \mathbf{x}_{n-1}\right] = \frac{m^{N_2} Q_n f_n(\mathbf{x}_{n-1})}{m^{N_2} g_{n-1}(\mathbf{x}_{n-1})} ,$$

which implies

$$\boldsymbol{Q}_n\boldsymbol{f}_n(\mathbf{x}_{n-1}) = \boldsymbol{g}_{n-1}(\mathbf{x}_{n-1})\ \boldsymbol{M}_n\boldsymbol{f}_n(\mathbf{x}_{n-1}) = m^{N_2} g_{n-1}(\mathbf{x}_{n-1}) \times \frac{m^{N_2} Q_n f_n(\mathbf{x}_{n-1})}{m^{N_2} g_{n-1}(\mathbf{x}_{n-1})} ,$$

showing (2.6.8). The proof of (2.6.9) follows by an induction since

$$\boldsymbol{Q}_{p,n}\boldsymbol{f}_n(\mathbf{x}_p) = \boldsymbol{Q}_{p,n-1}\boldsymbol{Q}_n\boldsymbol{f}_n(\mathbf{x}_p) .$$

Proof. of Theorem 2.3.4 □

Asymptotic bias behavior: For any fixed N_2, the asymptotic bias behavior of $\boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n)$ is given for any $\boldsymbol{f}_n \in \mathcal{B}_b(\mathbf{X}_n)$ by applying Theorem 2.3.3 to the island particle model in the bootstrap case:

$$\lim_{N_1\to\infty} N_1\mathbb{E}\left[\boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n) - \boldsymbol{\eta}_n(\boldsymbol{f}_n)\right] = -\sum_{p=0}^{n}\boldsymbol{\eta}_p\left[\boldsymbol{P}_{p,n}(\mathbb{1}_{\mathbf{x}_n})\boldsymbol{P}_{p,n}\left(\boldsymbol{f}_n - \boldsymbol{\eta}_n(\boldsymbol{f}_n)\right)\right] .$$

For linear functions $\boldsymbol{f}_n$ of the form $\boldsymbol{f}_n = m^{N_2} f_n$ where $f_n \in \mathcal{B}_b(\mathsf{X}_n)$, Lemma 2.6.3 states that

$$\boldsymbol{P}_{p,n}\left(\boldsymbol{f}_n - \boldsymbol{\eta}_n(\boldsymbol{f}_n)\right)(\boldsymbol{X}_p) = m^{N_2} P_{p,n}\left(f_n - \eta_n(f_n)\right)(\boldsymbol{X}_p) = \eta_p^{N_2}\left(P_{p,n}\left(f_n - \eta_n(f_n)\right)\right) , \tag{2.6.11}$$

and

$$\boldsymbol{P}_{p,n}(\mathbb{1}_{\mathbf{X}_n})(\boldsymbol{X}_p) = m^{N_2} P_{p,n}(\mathbb{1}_{\mathsf{X}_n})(\boldsymbol{X}_p) = \eta_p^{N_2}(P_{p,n}(\mathbb{1}_{\mathsf{X}_n})) . \tag{2.6.12}$$

Therefore, one gets

$$\begin{aligned}\boldsymbol{\eta}_p\left[\boldsymbol{P}_{p,n}(\mathbb{1}_{\mathbf{X}_n})\boldsymbol{P}_{p,n}\left(\boldsymbol{f}_n-\boldsymbol{\eta}_n(\boldsymbol{f}_n)\right)\right] &\overset{(1)}{=} \frac{\boldsymbol{\gamma}_p\left[\boldsymbol{P}_{p,n}(\mathbb{1}_{\mathbf{X}_n})\boldsymbol{P}_{p,n}\left(\boldsymbol{f}_n-\boldsymbol{\eta}_n(\boldsymbol{f}_n)\right)\right]}{\boldsymbol{\gamma}_p(\mathbb{1}_{\mathbf{X}_p})} \\ &\overset{(2)}{=} \frac{\mathbb{E}\left[\boldsymbol{P}_{p,n}(\mathbb{1}_{\mathbf{X}_n})(\boldsymbol{X}_p)\boldsymbol{P}_{p,n}\left(\boldsymbol{f}_n-\boldsymbol{\eta}_n(\boldsymbol{f}_n)\right)(\boldsymbol{X}_p)\prod_{\ell=0}^{p-1}\boldsymbol{g}_\ell(\boldsymbol{X}_\ell)\right]}{\boldsymbol{\gamma}_p(\mathbb{1}_{\mathbf{X}_p})} \\ &\overset{(3)}{=} \frac{\mathbb{E}\left[\eta_p^{N_2}(P_{p,n}\left(f_n-\eta_n(f_n)\right))\eta_p^{N_2}(P_{p,n}\left(\mathbb{1}_{\mathbf{X}_n}\right))\prod_{\ell=0}^{p-1}\eta_\ell^{N_2}(g_\ell)\right]}{\gamma_p(\mathbb{1}_{\mathbf{X}_p})}\,,\end{aligned} \tag{2.6.13}$$

where (1) is simply the definition (2.2.12) of $\boldsymbol{\eta}_p$, (2) stems from the definition (2.2.13) of $\boldsymbol{\gamma}_p$, and (3) follows from Theorem 2.2.1, the definition (2.2.10) of $(\boldsymbol{g}_\ell)_{\ell\geq 0}$ and equations (2.6.11) and (2.6.12). As $\eta_p(P_{p,n}\left(f_n-\eta_n(f_n)\right))=0$ one may apply Lemma 2.6.2 and

$$\begin{aligned}&\lim_{N_2\to\infty} N_2\boldsymbol{\eta}_p\left[\boldsymbol{P}_{p,n}(\mathbb{1}_{\mathbf{X}_n})\boldsymbol{P}_{p,n}\left(\boldsymbol{f}_n-\boldsymbol{\eta}_n(\boldsymbol{f}_n)\right)\right] \\ &\qquad=\sum_{\ell=0}^{p}\mathbb{E}\left[W_\ell(P_{\ell,p}(P_{p,n}(\mathbb{1}_{\mathbf{X}_n})-\eta_pP_{p,n}(\mathbb{1}_{\mathbf{X}_n})))W_\ell(P_{\ell,p}P_{p,n}(f_n-\eta_n(f_n)))\right] \\ &\qquad=\sum_{\ell=0}^{p}\mathbb{E}\left[W_\ell(P_{\ell,n}(\mathbb{1}_{\mathbf{X}_n})-P_{\ell,p}(\mathbb{1}_{\mathbf{X}_p}))W_\ell(P_{\ell,n}(f_n-\eta_n(f_n)))\right]\,,\end{aligned}$$

from which we conclude that

$$\begin{aligned}&\lim_{N_2\to\infty}\lim_{N_1\to\infty} N_2N_1\mathbb{E}\left[\boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n)-\boldsymbol{\eta}_n(\boldsymbol{f}_n)\right] \\ &\qquad=-\sum_{p=0}^{n}\sum_{\ell=0}^{p}\mathbb{E}\left[W_\ell(P_{\ell,n}(\mathbb{1}_{\mathbf{X}_n})-P_{\ell,p}(\mathbb{1}_{\mathbf{X}_p}))W_\ell(P_{\ell,n}(f_n-\eta_n(f_n)))\right] \\ &\qquad=-\sum_{\ell=0}^{n}\sum_{p=\ell}^{n}\mathbb{E}\left[W_\ell(P_{\ell,n}(\mathbb{1}_{\mathbf{X}_n})-P_{\ell,p}(\mathbb{1}_{\mathbf{X}_p}))W_\ell(P_{\ell,n}(f_n-\eta_n(f_n)))\right] \\ &\qquad=B_n(f_n)+\widetilde{B}_n(f_n)\,,\end{aligned}$$

where $B_n(f_n)$ is defined in (2.3.17) and $\widetilde{B}_n(f_n)$ is given in (2.3.21).

Asymptotic variance behavior: For any fixed N_2, the asymptotic variance behavior of $\boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n)$ is given for any $\boldsymbol{f}_n\in\mathcal{B}_b(\mathbf{X}_n)$ by applying Theorem 2.3.3 to the island particle model in the bootstrap case:

$$\lim_{N_1\to\infty} N_1\mathbb{V}\mathrm{ar}\left(\boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n)\right)=\sum_{p=0}^{n}\boldsymbol{\eta}_p\left[\boldsymbol{P}_{p,n}\left(\boldsymbol{f}_n-\boldsymbol{\eta}_n(\boldsymbol{f}_n)\right)^2\right]\,.$$

For linear functions $\boldsymbol{f}_n$ of the form $\boldsymbol{f}_n=m^{N_2}f_n$ where $f_n\in\mathcal{B}_b(\mathsf{X}_n)$, using the same steps as in (2.6.13), one gets

$$\boldsymbol{\eta}_p\left[\boldsymbol{P}_{p,n}\left(\boldsymbol{f}_n-\boldsymbol{\eta}_n(\boldsymbol{f}_n)\right)^2\right]=\frac{\mathbb{E}\left[\eta_p^{N_2}(P_{p,n}\left(f_n-\eta_n(f_n)\right))^2\prod_{\ell=0}^{p-1}\eta_\ell^{N_2}(g_\ell)\right]}{\gamma_p(\mathbb{1}_{\mathbf{X}_p})}\,.$$

As $\eta_p(P_{p,n}\left(f_n-\eta_n(f_n)\right))=0$ one may apply Lemma 2.6.2 and

$$\lim_{N_2\to\infty} N_2\boldsymbol{\eta}_p\left[\boldsymbol{P}_{p,n}\left(\boldsymbol{f}_n-\boldsymbol{\eta}_n(\boldsymbol{f}_n)\right)^2\right]=\sum_{\ell=0}^{p}\mathbb{E}\left[W_\ell(P_{\ell,p}P_{p,n}(f_n-\eta_n(f_n)))^2\right]=\sum_{\ell=0}^{p}\mathbb{E}\left[W_\ell(P_{\ell,n}(f_n-\eta_n(f_n)))^2\right]\,,$$

from which we conclude that

$$\lim_{N_2\to\infty}\lim_{N_1\to\infty} N_2N_1\mathbb{V}\mathrm{ar}\left(\boldsymbol{\eta}_n^{N_1}(\boldsymbol{f}_n)\right) = \sum_{p=0}^{n}\sum_{\ell=0}^{p}\mathbb{E}\left[W_\ell(P_{\ell,n}(f_n-\eta_n(f_n)))^2\right]$$
$$= \sum_{\ell=0}^{n}\sum_{p=\ell}^{n}\mathbb{E}\left[W_\ell(P_{\ell,n}(f_n-\eta_n(f_n)))^2\right] = V_n(f_n)+\widetilde{V}_n(f_n)\,,$$

where $V_n(f_n)$ is defined in (2.3.18) and $\widetilde{V}_n(f_n)$ is given in (2.3.22). □

2.6.4 Proof of Theorem 2.4.1

Lemma 2.6.4. *Let ϵ_n be a nonnegative constant such that $\epsilon_n\ g_n\in[0,1]$. Then*

$$\Psi_n(\mu_n) = \mu_n S_{n,\mu_n}\,,$$

where S_{n,μ_n} is defined in (2.4.1).

Proof. By (2.4.1) and (2.2.3) one has for any $A_n\in\mathcal{X}_n$

$$\begin{aligned}\mu_nS_{n,\mu_n}(A_n) &= \int \mu_n(\mathrm{d}x_n)S_{n,\mu_n}(x_n,A_n)\\ &= \int \mu_n(\mathrm{d}x_n)\left[\epsilon_n\ g_n(x_n)\delta_{x_n}(A_n)+(1-\epsilon_n\ g_n(x_n))\,\Psi_n(\mu_n)(A_n)\right]\\ &= \epsilon_n\int_{A_n}\mu_n(\mathrm{d}x_n)g_n(x_n)+(1-\epsilon_n\ \mu_n(g_n))\,\Psi_n(\mu_n)(A_n)\\ &= \epsilon_n\mu_n(g_n)\Psi_n(\mu_n)(A_n)+(1-\epsilon_n\ \mu_n(g_n))\,\Psi_n(\mu_n)(A_n) = \Psi_n(\mu_n)(A_n)\,.\quad\square\end{aligned}$$

□

Let $\mathcal{F}_n^{N_2}$ be the increasing filtration associated to the particle evolution $\mathcal{F}_n^{N_2}\overset{\text{def}}{=}\sigma\left(\boldsymbol{X}_p,0\le p\le n\right)$. As in the proofs of Theorem 2.2.1 and Theorem 2.4.4, the only point is to prove that

$$\mathbb{E}\left[\eta_p^{N_2}(f_p)\mid\mathcal{F}_{p-1}^{N_2}\right] = \frac{\eta_{p-1}^{N_2}(Q_pf_p)}{\eta_{p-1}^{N_2}(g_{p-1})}\,,$$

where Q_p is defined in (2.3.1). But,

$$\begin{aligned}\mathbb{E}\left[\eta_p^{N_2}(f_p)\mid\mathcal{F}_{p-1}^{N_2}\right] &= \frac{1}{N_2}\sum_{i=1}^{N_2}\mathbb{E}\left[f_p(X_p^i)\mid\mathcal{F}_{p-1}^{N_2}\right] = \frac{1}{N_2}\sum_{i=1}^{N_2}\boldsymbol{M}_p(f_p)(X_{p-1}^i)\\ &= \eta_{p-1}^{N_2}\boldsymbol{M}_p(f_p) = \eta_{p-1}^{N_2}S_{p-1,\eta_{p-1}^{N_2}}M_p(f_p) = \Psi_{p-1}(\eta_{p-1}^{N_2})M_p(f_p) = \frac{\eta_{p-1}^{N_2}(Q_pf_p)}{\eta_{p-1}^{N_2}(g_{p-1})}\,,\end{aligned}\tag{2.6.14}$$

using respectively (2.2.8), (2.4.2), Lemma 2.6.4 and (2.2.3).

2.6.5 Proof of Proposition 2.4.3

For the ϵ-interaction bootstrap, the sequence $(W_p^{N_2})_{1\le p\le n}$ converges in law, as N_2 tends to infinity, to a sequence of n independent centered Gaussian random fields $(W_p)_{0\le p\le n}$ with variance given by

$$\begin{aligned}\mathbb{E}\left[W_p(f_p)^2\right] &= \eta_{p-1}S_{p-1,\eta_{p-1}}M_pf_p^2-\eta_{p-1}\left[\left(S_{p-1,\eta_{p-1}}M_pf_p\right)^2\right]\\ &= \Psi_{p-1}(\eta_{p-1})(M_pf_p^2)-\eta_{p-1}\left[\left(S_{p-1,\eta_{p-1}}M_pf_p\right)^2\right]\,,\end{aligned}$$

thanks to Lemma 2.6.4.
In the special case $\epsilon_p = 0$ (the bootstrap case), the function $S_{p,\eta_p} g_p$ is constant and equal to $\Psi_p(\eta_p)(g_p)$ and the variance for the bootstrap is just

$$\Psi_{p-1}(\eta_{p-1})(M_p f_p^2) - (\Psi_{p-1}(\eta_{p-1})M_p f_p)^2 \ .$$

Therefore, the variance of the ϵ-interaction bootstrap may be decomposed as follows

$$\begin{aligned}\mathbb{E}\left[W_p(f_p)^2\right] &= \left(\Psi_{p-1}(\eta_{p-1})(M_p f_p^2) - (\Psi_{p-1}(\eta_{p-1})M_p f_p)^2\right) \\ &\quad - \left(\eta_{p-1}\left[\left(S_{p-1,\eta_{p-1}} M_p f_p\right)^2\right] - (\Psi_{p-1}(\eta_{p-1})M_p f_p)^2\right) \ .\end{aligned}$$

Observing that

$$\eta_{p-1}\left[\left(S_{p-1,\eta_{p-1}} M_p f_p\right)^2\right] - (\Psi_{p-1}(\eta_{p-1})M_p f_p)^2 = \eta_{p-1}\left(\left[S_{p-1,\eta_{p-1}} M_p f_p - \Psi_{p-1}(\eta_{p-1})(M_p f_p)\right]^2\right) \geq 0 \ ,$$

allows to conclude.

2.6.6 Proof of Theorem 2.4.4

Using (2.4.6), (2.4.8), (2.2.13), and for $\boldsymbol{f}_n$ such that $\boldsymbol{f}_n(\boldsymbol{X}_n) = \left(\sum_{i=1}^{N_2} w_n^i\right)^{-1} \sum_{i=1}^{N_2} w_n^i f_n\left(X_n^i\right) = m^{N_2} f_n(\boldsymbol{X}_n) = \eta_n^{N_2}(f_n)$, one gets

$$\boldsymbol{\gamma}_n(\boldsymbol{f}_n) \stackrel{\text{def}}{=} \mathbb{E}\left[\boldsymbol{f}_n(\boldsymbol{X}_n) \prod_{0\leq p<n} \boldsymbol{g}_p(\boldsymbol{X}_p)\right] = \mathbb{E}\left[\eta_n^{N_2}(f_n) \prod_{0\leq p<n} \eta_p^{N_2}(g_p)\right] .$$

By (2.4.7), it suffices to prove that $\mathbb{E}\left[\gamma_n^{N_2}(f_n)\right] = \gamma_n(f_n)$. We define by $\mathcal{F}_n^{N_2}$ the increasing filtration associated to the particle evolution $\mathcal{F}_n^{N_2} \stackrel{\text{def}}{=} \sigma\left(\boldsymbol{X}_p, 0 \leq p \leq n\right)$. One may show that for any $p > 0$ and $f_p \in \mathcal{B}_b(\mathsf{X}_p)$, one has $\mathbb{E}\left[\eta_p^{N_2}(f_p) \mid \mathcal{F}_{p-1}^{N_2}\right] = \eta_{p-1}^{N_2}(Q_p f_p)/\eta_{p-1}^{N_2}(g_{p-1})$, where Q_p is defined in (2.3.1). Indeed, by the definitions (2.4.4) of M_p and (2.4.6) of $\eta_p^{N_2}$,

$$\begin{aligned}&\mathbb{E}\left[\eta_p^{N_2}(f_p) \mid \mathcal{F}_{p-1}^{N_2}\right] = \sum_{i=1}^{N_2} \frac{\omega_p^i}{\sum_{j=1}^{N_2} \omega_p^j} \mathbb{E}\left[f_p(X_p^i) \mid \boldsymbol{X}_{p-1}\right] \\ &= \mathbb{1}_{\Theta_{p-1,\alpha}}(\boldsymbol{X}_{p-1}) \left[\frac{\sum_{i=1}^{N_2} \omega_{p-1}^i g_{p-1}(X_{p-1}^i) M_p f_p(X_{p-1}^i)}{\sum_{i=1}^{N_2} \omega_{p-1}^i g_{p-1}(X_{p-1}^i)}\right] \\ &\quad + \mathbb{1}_{\Theta_{p-1,\alpha}^{\mathrm{C}}}(\boldsymbol{X}_{p-1}) \left[\frac{1}{N_2} \sum_{i=1}^{N_2} \frac{\sum_{j=1}^{N_2} \omega_{p-1}^j g_{p-1}(X_{p-1}^j) M_p f_p(X_{p-1}^j)}{\sum_{j=1}^{N_2} \omega_{p-1}^j g_{p-1}(X_{p-1}^j)}\right] \\ &= \frac{\eta_{p-1}^{N_2}(Q_p f_p)}{\eta_{p-1}^{N_2}(g_{p-1})} \ .\end{aligned}$$

The proof follows exactly along the same lines as Theorem 2.2.1. By iterating this step one gets

$$\begin{aligned}\mathbb{E}\left[\gamma_n^{N_2}(f_n)\right] &= \mathbb{E}\left[\eta_0^{N_2}(Q_1 \cdots Q_n f_n)\right] = \mathbb{E}\left[Q_1 \cdots Q_n f_n(X_0^1)\right] \\ &= \gamma_0 Q_1 \cdots Q_n f_n = \gamma_n(f_n) \ .\end{aligned}$$

As the reader may have noticed, this unbias property doesn't depend on the definition of the sets $\Theta_{p,\alpha}$ defining the resampling times. From this observation, we underline that Theorem 2.4.4 is also true for more general classes of resampling time criterion.

Chapitre 3

Étude des propriétés de convergence des îlots de particules

Sommaire

3.1 Introduction . . . 66
3.2 Preliminaries . . . 68
3.2.1 Weighted particle islands and archipelagos . . . 68
3.2.2 Convergence properties of archipelagos . . . 69
3.3 Main results : operations on weighted archipelagos . . . 70
3.3.1 Selection on the island level . . . 70
3.3.2 Selection on the individual level . . . 71
3.3.3 Mutation . . . 72
3.4 Applications . . . 74
3.4.1 Feynman-Kac models . . . 74
3.4.2 The double bootstrap algorithm with adaptive selection (B^2ASIL) algorithm . . . 74
3.4.3 Long-term stability of the double bootstrap (B^2) algorithm . . . 78
3.4.4 Uniform in time convergence under the strong mixing assumption . . . 79
3.4.5 Uniform in time convergence under a local Doeblin condition . . . 80
3.5 Technical results . . . 81
3.6 Proofs . . . 84
3.6.1 Proof of Theorem 3.3.1 . . . 84
3.6.2 Proof of Theorem 3.3.2 . . . 85
3.6.3 Proof of Theorem 3.3.3 . . . 85
3.6.4 Proof of Theorem 3.3.4 . . . 86
3.6.5 Proof of Theorem 3.3.5 . . . 87
3.6.6 Proof of Theorem 3.3.6 . . . 87
3.6.7 Proof of Theorem 3.3.7 . . . 89
3.6.8 Proof of Theorem 3.3.8 . . . 90
3.6.9 Proof of Theorem 3.3.9 . . . 90
3.6.10 Proof of Corollary 3.4.5 . . . 93
3.6.11 Proof of Corollary 3.4.8 . . . 93

Ce chapitre analyse les propriétés des modèles d'îlots de particules introduits au chapitre 2, voir aussi [Vergé et al., 2015a]. Lorsque la puissance de calcul disponible est limitée, on peut être amené à répartir les $N \overset{\text{def}}{=} N_1 N_2$ particules d'un algorithme SMC classique en N_1 îlots de N_2 particules chacun. Chaque îlot évolue selon un SMC en alternant des phases de sélection et de mutation. D'après la discussion du chapitre 2, lorsque le nombre de particules N_2 dans chaque îlot est petit, il est intéressant de faire interagir les îlots afin de réduire le biais. Différentes interactions sont possibles, certaines systématiques comme pour le B^2 ou bien adaptatives comme pour le B^{2}ASIL. La dépendance introduite par les différentes opérations au niveau des îlots ou à l'intérieur des îlots rend difficile l'analyse des algorithmes B^2 ou B^{2}ASIL. La technique employée au chapitre 2, reformulant l'algorithme B^2 comme un modèle de Feynman-Kac sur un espace étendu et utilisant la double asymptotique (N_1 puis N_2 tend vers l'infini) a détourné la difficulté et a uniquement permis de déterminer les expressions asymptotiques du biais et de la variance du B^2 en fonction de N_1 et N_2, et d'en déduire sa consistance. La consistance de l'algorithme B^{2}ASIL ainsi qu'un théorème central limite (TCL) n'ont pas encore été établis : ce sont des objectifs de ce chapitre.

La propagation du chaos pour les méthodes SMC standards démontrée dans [Del Moral, 2004] implique que, malgré leur interaction, deux particules deviennent statistiquement indépendantes quand le nombre total de particules est grand. Cette propriété a permis de mettre en évidence des résultats de convergence pour les filtres particulaires, [Crisan and Doucet, 2002]. Dans le cas des modèles d'îlots de particules, on conjecture une loi des grands nombres et un TCL quand N_1 et N_2 tendent conjointement vers l'infini. De plus, par analogie avec les méthodes SMC avec un seul système de particules en interaction (voir [Del Moral and Guionnet, 1999; Chopin, 2004; LeGland and Oudjane, 2004; Künsch, 2005; Douc and Moulines, 2008]), on peut s'attendre à ce que le taux de convergence du TCL soit $\sqrt{N}$.

Dans ce chapitre nous présentons de nouveaux résultats de convergence pour les algorithmes de type îlots de particules. Pour ce faire, nous avons introduit la notion d'*archipel d'îlots pondérés* qui généralise les îlots de particules étudiés en [Vergé et al., 2015a] et trois propriétés de convergence sur les archipels : la consistance (convergence en probabilité), la normalité asymptotique (convergence en loi au sens d'un TCL en $\sqrt{N}$) et une inégalité de déviation exponentielle (une inégalité de type Hoeffding, uniforme sur les îlots). L'analyse de ces propriétés est difficile à cause de la dépendance entre les particules à chaque opération. C'est pourquoi, nous avons décomposé une itération du B^{2}ASIL en trois opérations élémentaires : la *sélection au niveau des îlots*, la *sélection au niveau des individus* (particules) et la *mutation*. Nous avons trouvé des conditions que l'on espère minimales, sous lesquelles les propriétés de convergence des archipels sont préservées par ces opérations élémentaires. Cette théorie permet d'analyser tout algorithme formé de ces opérations. Le B^{2}ASIL est un exemple.

Le B^{2}ASIL permet d'estimer en ligne les mesures de Feynman-Kac qui sont généralement non calculables, à l'exception du cas où le filtre de Kalman s'applique ou pour des modèles de chaînes de Markov cachées à espaces d'états finis. Le principe de l'algorithme B^{2}ASIL est qu'il ré-échantillonne les îlots uniquement lorsque leurs poids sont hétérogènes. Pour cela, on introduit un critère de dispersion des poids, appelé *coefficient de variation* (coefficient of variation (CV)) (voir [Kong et al., 1994] et [Liu, 2008]) défini par $\mathrm{CV}_N^2(\{\Omega_N(i)\}_{i=1}^{N_1})$, où $\Omega_N(i)$ est le poids de l'îlot i et

$$\mathrm{CV}_N^2 : (\mathbb{R}_+^*)^{N_1} \ni \{a(i)\}_{i=1}^{N_1} \mapsto N_1 \sum_{i=1}^{N_1} \left(\frac{a(i)}{\sum_{i'=1}^{N_1} a(i')} \right)^2 - 1 \,.$$

Le CV est proche de l'ESS proposé dans [Liu, 1996], et utilisé dans le chapitre 3. En effet, comme l'ESS peut s'exprimer comme $N_1/[1+\mathrm{CV}_N^2(\{a(i)\}_{i=1}^{N_1})]$, les deux critères sont équivalents. Le CV est minimal (nul) quand tous les poids des îlots sont égaux et maximal ($N_1 - 1$) quand les poids sont le plus épars, *i.e.*, quand la masse totale du système est portée par un seul îlot (une situation qui est cependant impossible dans le cadre de travail considéré puisque tous les poids sont supposés strictement positifs). Quand le CV est inférieur à un seuil fixé $\tau \geq 0$, les îlots n'interagissent pas. Dans le cas contraire, quand les poids des îlots sont trop

dispersés, les îlots sont mis à jour selon l'étape de sélection au niveau des îlots (multinomial selection on the island level (SIL)). Ensuite, les îlots évoluent de manière indépendante et totalement parallélisable selon la sélection au niveau des individus (multinomial selection on the individual level (SiL)) et la mutation. Le pseudo code de l'algorithme B^{2}ASIL est rappelé dans l'Algorithme 10.

La flexibilité du travail présenté au chapitre 3 généralise les résultats obtenus dans [Douc and Moulines, 2008] pour les méthodes SMC classiques avec un seul système de particules. Comme dans [Douc and Moulines, 2008], nos preuves sont inductives et se basent sur les théorèmes limites, versions conditionnelles du TCL de Lindeberg, pour les tableaux triangulaires de variables aléatoires dépendantes. La dépendance entre les particules nous a conduit à considérer des tableaux triangulaires au niveau des îlots.

En remarquant que le B^{2}ASIL se décompose en opérations élémentaires successives (SIL, SiL et mutation dont on a préalablement établi qu'elles propagent toutes les propriétés de convergence évoquées dans ce chapitre), il est facile d'en déduire la consistance, l'inégalité de déviation exponentielle et le TCL (quand les nombres de particules et d'îlots tendent conjointement vers l'infini) pour le B^{2}ASIL. Pour prouver la normalité asymptotique par exemple, on montre d'abord qu'elle est préservée par une itération de l'algorithme B^{2}ASIL et on détermine les formules récursives pour la variance asymptotique en appliquant les théorèmes 3.3.3, 3.3.6 et 3.3.9. On en déduit ensuite l'expression générale de la variance asymptotique du B^{2}ASIL. L'algorithme B^2 étant un cas particulier du B^{2}ASIL (pour $\tau = 0$), on en déduit facilement les mêmes résultats de convergence pour le B^2 et on simplifie l'expression de la variance en conséquence. Dans le cas où SIL n'est jamais utilisé, on retrouve la variance d'un SMC standard (aussi appelé, *bootstrap*, ou *sequential importance sampling with resampling*). En effet, dans ce cas, N_1 îlots indépendants (sans interaction) se propagent, chacun suivant un SMC standard utilisant N_2 particules.

Enfin, on établit la stabilité en temps long du B^2. D'une part, sous des hypothèses classiques [Del Moral and Guionnet, 2001; Del Moral, 2004] de mélange fort, valables typiquement pour des modèles de Feynman-Kac sur des espaces d'état compacts, on explicite un majorant uniforme de la variance asymptotique. D'autre part, en affaiblissant les hypothèses précédentes par des conditions de type *Doeblin local*, vérifiées pour une plus grande variété de modèles (y compris définis sur des espaces non compacts, voir [Douc et al., 2014a] pour des exemples), on montre que la variance asymptotique du B^2 est bornée. Pour cela, on commence par remarquer que la variance asymptotique du B^2 est la même que celle du bootstrap à un facteur $(n - \ell)$ près. La preuve pour le B^2 s'obtient en suivant les mêmes étapes que celle pour le bootstrap démontrée dans [[Douc et al., 2014a], Théorème 11].

Auteurs :

- Christelle Vergé (Onera, CNES, CMAP, Palaiseau)
- Pierre Del Moral (Université UNSW, Sydney)
- Eric Moulines (Telecom ParisTech, Paris)
- Jimmy Olsson (KTH Royal Institute of Technology, Stockholm, Suède)

Statut : Chapitre basé sur l'article "Convergence properties of weighted particle islands with application to the double bootstrap algorithm", soumis aux *Annales de l'Institut Henri Poincaré*, en octobre 2014, [Vergé et al., 2014]

Abstract Particle island models [Vergé et al., 2015a] provide a means of parallelization of sequential Monte Carlo methods, and in this chapter we present novel convergence results for algorithms of this sort. In particular we establish a central limit theorem—as the number of islands and the common size of the islands tend jointly to infinity—of the double bootstrap algorithm with possibly adaptive selection on the island level. For this purpose we introduce a notion of archipelagos of weighted islands and find conditions under which a set of convergence properties are preserved by different operations on such archipelagos. This theory allows arbitrary compositions of these operations to be straightforwardly analyzed, providing a very flexible framework covering the double bootstrap algorithm as a special case. Finally, we establish the long-term numerical stability of the double bootstrap algorithm by bounding its asymptotic variance under weak and easily checked assumptions satisfied typically for models with non-compact state space.

Keywords : Central limit theorem, exponential deviation, parallelization, particle island models, particle filter, sequential Monte Carlo methods

3.1 Introduction

This chapter discusses approaches to parallelization of SMC methods (or *particle filters*) approximating normalized *Feynman-Kac distribution flows*. At present, SMC methods are used successfully for online sampling from sequences of complex distributions in a wide range of applications, including nonlinear filtering, signal processing, data assimilation, e.g. see [Doucet et al., 2001; Chopin, 2002; Ristic et al., 2004; Cappé and Moulines, 2005; Crisan and Rozovskii, 2011] and the reference therein, and rare event analysis [Del Moral and Garnier, 2005; Cérou et al., 2012]. These algorithms evolve, recursively and randomly in time, a sample of random draws, *particles*, with associated *importance weights*, and the Feynman-Kac distribution flow is approximated by the weighted empirical measures associated with this sample. The particle cloud is updated through *selection* and *mutation* operations, where the former duplicates or eliminates, through resampling, particles with large or small importance weights, respectively, while the latter disseminates randomly the particles over the state space and updates accordingly the importance weights for further selection.

SMC methods are computationally intensive, which may be critical in online applications where only a limited computational power is at hand. In particular, since the particle interaction enforced by the selection operation is of "global" nature (as it draws, with replacement, each particle from the entire particle population rather than from a subset of the same), running SMC methods in parallel on multicore processors is not straightforward. The natural ideal of [Vergé et al., 2015a], which is the basis also for the present chapter, is to parallelize the algorithm by, instead of considering a single batch of N particles, simply dividing the particle population into N_1 batches of each N_2 particles (i.e., $N = N_1 N_2$), where each batch is referred to as a *particle island* (or simply an *island*). In this framework, each island evolves according to the standard SMC scheme subjecting alternatingly the subpopulation to selection and mutation. Unfortunately, the division of the particle population introduces additional bias which may be of note for moderate island sizes N_2. Thus, in [Vergé et al., 2015a] it is proposed to reduce this bias by performing additional selection also on the *island level* by resampling multinomially, when needed, the islands according to probabilities proportional to the weight averages over the different subpopulations. Selection on the island level may be performed systematically, as in the B^2 *algorithm* (in the present chapter we have chosen to denote the algorithm "B^2" rather than "2B", as we consider it more correctly described as a "square bootstrap" rather than a "double bootstrap"; nevertheless, the algorithm must not be confused with the *SMC square* (SMC^2) *algorithm* proposed in [Chopin et al., 2013], which is, if still of a related form, of a different nature) or may be activated adaptively by some criterion measuring the skewness of the island weights. The latter approach will be referred to by us as the *double bootstrap algorithm with adaptive selection on the island level* B^{2}ASIL. At the end of the day, a sequence of Monte Carlo estimators is obtained by weighing up, using the island weights, the self-normalized empirical

measures associated with the different particle islands.

Needless to say, the theoretical analysis of B^2-type algorithms is challenging due to the intricate dependence structure imposed by the island selection operation and the "double asymptotics" introduced by the two sample sizes N_1 and N_2. The authors of [Vergé et al., 2015a], who base their theoretical analysis on a reformulation of the particle island model as an extended Feynman-Kac model on an augmented space of dimension N_2, detour the latter difficulty by letting first the number N_1 of islands and then the number N_2 of individuals of each island tend to infinity. By *separating* the asymptotics in this manner, the analysis can, not surprisingly, at least in the case of the B^2 algorithm, be handled using classical techniques from SMC analysis, and in this way the authors establish convergence of bias and variance when these quantities are scaled with the size N of the system. However, working with this somewhat synthetic mode of convergence (with separated limits), the authors fail to supplement their consistency results with a central limit theorem (CLT). Moreover, they do not provide any convergence results for the B^{2}ASIL algorithm.

Nevertheless, even though the islands are allowed to interact through selection, any two individuals of the system should become more and more statistically independent as the number of islands as well as the size of the islands grow (cf. the *propagation of chaos* property of standard SMC methods [Del Moral, 2004]). Thus, we may expect a law of large number as well as a CLT to hold when N_1 and N_2 tend *jointly* to infinity. Moreover, in analogy with similar result for standard, single batch SMC methods, see [Del Moral and Guionnet, 1999; Chopin, 2004; Künsch, 2005; Douc and Moulines, 2008], we may expect the rate of such a CLT to be $\sqrt{N}$.

The aim of the present chapter is to improve the existing theoretical analysis of particle island models by establishing results of the previous type. For this purpose we will introduce a notion of *archipelagos of weighted islands* that generalizes the particle models studied in [Vergé et al., 2015a] and consider three kinds of convergence properties of such archipelagos, namely *consistency* (convergence in probability), *asymptotic normality* (convergence in distribution in terms of a CLT with rate $\sqrt{N}$), and *large deviation* (an exponential inequality of Hoeffing-type that holds uniformly over all islands). After this, we perform single-step analyses of three kinds of operations on archipelagos, namely *selection on the island level*, *selection on the individual level*, and *mutation*, and show how these operations preserve the convergence properties under consideration. As a consequence, we are able to establish that the convergence properties in question are preserved through an *arbitrary* composition of the mentioned operations, including the B^2 algorithm as a special case, and to provide explicit expressions of the associated asymptotic variance. Moreover, the flexibility of our results, which generalize those obtained in [Douc and Moulines, 2008] for standard, single batch SMC methods, makes these well-suited for analyzing particle island algorithms with adaptive resampling strategies such as the B^{2}ASIL scheme, for which we provide a detailed analysis (including a CLT). In our proofs, which rely on limit theorems for triangular arrays obtained in [Douc and Moulines, 2008], the working process is highly inductive. Since the intricate dependence structures of the particle model force us to define triangular arrays on the island level, we will often, when establishing the preservation of a certain convergence property of a certain operation, face a situation where the only way of obtaining some critical limit or bound is to add the same to the list of induction hypotheses. After this, one establishes that the operation in question preserves also this additional property (limit or bound), by possibly adding, if needed, further assumptions to the list, and so on. At the end of the day, we have obtained a more or less minimal set (a hexad in the case of asymptotic normality) of properties that need to be checked at each induction step. In this machinery, the large deviation property is a critical component, since it provides, as a consequence of the distribution-free character of Hoeffding-type inequalities, uniform control of the deviation of the empirical measures associated with the different islands from their common mean.

As a last contribution, we establish the numerical stability of the B^2 algorithm by bounding uniformly the asymptotic variance of its output. We carry through this analysis under a *strong mixing condition* as well a *local Doeblin condition* (see subsection 3.6.10 for details), where the latter is considerably weaker than the former and easily verified for a large variety of models with possibly non-compact state space.

When operating under the local Doeblin condition, we let the Feynman-Kac model be indexed by a strictly stationary sequence of random parameters (corresponding, e.g., to random observations in the case of optimal filtering in *hidden Markov models*) and show, using novel results in [Douc et al., 2014a], that the sequence of asymptotic variances is stochastically bounded (tight) in this setting. On the other hand, imposing the strong mixing assumption, which is classical in the literature of SMC analysis [Del Moral and Guionnet, 2001; Del Moral, 2004], allows an explicit, deterministic uniform variance bound to be obtained using standard methods.

To sum up, the contribution of the present chapter is threefold, since we

- introduce a general theory of archipelagos of weighted particle islands and analyze thoroughly the convergence properties, as the number N_1 of islands and the common size N_2 of the islands tend jointly to infinity, of such objects when subjected to certain operations. For this purpose, we develop a machinery that allows triangular arrays defined on the island level to be analyzed and which may be used for handling double asymptotics appearing in other kinds of island-type particle algorithms.

- apply the previous theoretical results to the B^2 and B^{2}ASIL algorithms, yielding laws of large numbers and CLTs for these schemes.

- establish the long-term stability of the B^2 algorithm under weak and easily checked assumptions.

The chapter is organized as follows. In section 3.2 we introduce, after some prefatory notation, the concept of archipelagos of weighted islands, and define the three different convergence properties of such archipelagos. Our main results are, along with the three different operations under consideration, presented in section 3.3, and section 3.4 discusses the application of these results to the B^{2}ASIL algorithm. In particular, in Corollary 3.4.2 we establish the asymptotic normality of this algorithm, which implies the asymptotic normality of the B^2 algorithm as a special case (see Corollary 3.4.3), and provide a formula for the asymptotic variance ; moreover, in subsection 3.4.3 establish the long-term stability of the algorithm by showing that the asymptotic variance of the B^2 algorithm may, under suitable assumptions, be bounded uniformly. The most significative proofs are gathered in section 3.6, and in order to avoid repetition we have put some additional proofs using similar techniques in chapter 3. Finally, section 3.5 provides some technical results that are used frequently in section 3.6.

3.2 Preliminaries

3.2.1 Weighted particle islands and archipelagos

Let $\{N_1(N)\}_{N\in\mathbb{N}^*}$ and $\{N_2(N)\}_{N\in\mathbb{N}^*}$ be sequences of positive integers such that $N_1(N)N_2(N) = N$ for all $N \in \mathbb{N}^*$ and $N_1(N) \to \infty$ and $N_2(N) \to \infty$ as $N \to \infty$. For the ease of notation we will often omit the index N from the notation and write simply N_1 and N_2. In the following, let $\{(\xi_N(i,j), \omega_N(i,j)); (i,j) \in [\![1, N_1]\!] \times [\![1, N_2]\!]\}$ be an array of X-valued random variables (the ξ_N) with associated nonnegative (possibly unnormalized) weights (the ω_N). For each $i \in [\![1, N_1]\!]$, the subset $\{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2}$ of the array will be referred to as an *island*. With this terminology, a random variable $\xi_N(i,j)$ in the array will be referred to as an *individual* or a *particle*. Finally, we associate each island $\{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2}$ with a nonnegative (possibly unnormalized) weight $\Omega_N(i)$. In the following, the set $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ of islands with associated weights will be referred to as an *archipelago* on $(\mathsf{X}, \mathcal{X})$. We will always require the island weights to be positive and the particle weights to be positive and uniformly bounded, i.e., there exists some constant $|\omega|_\infty$ such that $0 < |\omega_N(i,j)| \leq |\omega|_\infty$ for all $(i,j) \in [\![1, N_1(N)]\!] \times [\![1, N_2(N)]\!]$ and $N \in \mathbb{N}^*$.

3.2.2 Convergence properties of archipelagos

In the following, any limit ($\longrightarrow$), limit in probability ($\xrightarrow{\mathbb{P}}$), and limit in distribution ($\xrightarrow{\mathcal{D}}$) is supposed to hold as $N \to \infty$ if not specified differently.

Definition 3.2.1 (consistency). *An archipelago* $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *on* $(\mathsf{X}, \mathcal{X})$ *is said to be* consistent for $\eta \in \mathcal{P}(\mathsf{X})$ *if for all* $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$,

(C1) $$\sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\xi_N(i,j)) \xrightarrow[N\to+\infty]{\mathbb{P}} \eta h \ ,$$

(C2) $$\max_{i \in [\![1,N_1]\!]} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \xrightarrow[N\to+\infty]{\mathbb{P}} 0 \ .$$

Note that the estimator in (**C1**) assigns the weight $\Omega_N(i)/\sum_{i'=1}^{N_1} \Omega_N(i')$ to the self-normalized importance sampling estimator $\sum_{j=1}^{N_2} \omega_N(i,j) h(\xi_N(i,j))/\sum_{j'=1}^{N_2} \omega_N(i,j')$ associated with island $i \in [\![1, N_1]\!]$, and the smallness condition (**C2**) formalizes the fact that this weight, and thus the contribution of each island to the estimator associated with the archipelago as a whole, should vanish asymptotically as $N \to \infty$.

Definition 3.2.2 (exponential deviation). *In the following, let* $\eta \in \mathcal{P}(\mathsf{X})$ *and* ϱ *and* $\{c_\ell\}_{\ell=1}^2$ *be positive constants. An archipelago* $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *on* $(\mathsf{X}, \mathcal{X})$ *is said to satisfy* exponential deviation for $(\eta, \varrho, \{c_\ell\}_{\ell=1}^2)$ *if for all* $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$, $N_1 \in \mathbb{N}^*$, $N_2 \in \mathbb{N}^*$, *and* $\varepsilon > 0$,

(D) $$\mathbb{P}\left(\max_{i \in [\![1,N_1]\!]} \left| \frac{1}{N_2} \sum_{j=1}^{N_2} \omega_N(i,j) h(\xi_N(i,j)) - \varrho \times \eta h \right| \geq \varepsilon \right) \leq c_1 N_1 \exp\left(-c_2 N_2 \frac{\varepsilon^2}{\|h\|_\infty^2} \right) \ .$$

The exponential deviation inequality in (**D**) provides uniform control on the deviations of the unnormalized importance sampling estimators $\sum_{j=1}^{N_2} \omega_N(i,j) h(\xi_N(i,j))/N_2$, $i \in [\![1, N_1]\!]$, associated with the different islands from their common mean level $\varrho \times \eta h$. The factor N_1 on the right hand side of the equality is required to compensate for the maximum with respect to the island index. Notice that this inequality doesn't depend upon island weights. Assumption (**D**) implies, by a straightforward extension of the generalized Hoeffding inequality derived in [Douc et al., 2011, Lemma 4], that also the deviations of the properly normalized importance sampling estimators associated with the different islands from the expectations targeted by the archipelago can be uniformly controlled as follows.

Lemma 3.2.3. *Assume that* (**D**) *holds for* $(\eta, \varrho, \{c_\ell\}_{\ell=1}^2)$. *Then for all* $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$, $N_1 \in \mathbb{N}^*$, $N_2 \in \mathbb{N}^*$, *and* $\varepsilon > 0$,

$$\mathbb{P}\left(\max_{i \in [\![1,N_1]\!]} \left| \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{h(\xi_N(i,j)) - \eta h\} \right| \geq \varepsilon \right) \leq 2 c_1 N_1 \exp\left(-c_2 N_2 \frac{\varepsilon^2 \varrho^2}{4 \|h\|_\infty^2} \right) \ . \tag{3.2.1}$$

Finally, we introduce a third convergence property describing weak convergence in the sense of a CLT. Recall that $\mathcal{N}$ denotes the Gaussian distribution.

Definition 3.2.4 (asymptotic normality). *In the following, let*

- $\sigma^2 : \mathsf{F}_{\mathrm{b}}(\mathsf{X}) \to \mathbb{R}_+^*$ *and* $\nu^2 : \mathsf{F}_{\mathrm{b}}(\mathsf{X}) \to \mathbb{R}_+^*$ *be functionals.*
- $\eta \in \mathcal{P}(\mathsf{X})$ *and* $\{\mu_\ell\}_{\ell=1}^3 \subset \mathsf{M}(\mathsf{X})$ *be measures.*

An archipelago $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *on* $(\mathsf{X}, \mathcal{X})$ *is said to be* asymptotically normal for $(\eta, \sigma^2, \nu^2, \{\mu_\ell\}_{\ell=1}^3)$ *if for all* $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$,

(AN1) $$\sqrt{N}\sum_{i=1}^{N_1}\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\sum_{j=1}^{N_2}\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\{h(\xi_N(i,j))-\eta h\}\xrightarrow[N\to+\infty]{\mathcal{D}}\mathcal{N}(0,\sigma^2(h))$$

and, in addition,

(AN2) $$N_2\sum_{i=1}^{N_1}\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\left(\sum_{j=1}^{N_2}\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\{h(\xi_N(i,j))-\eta h\}\right)^2\xrightarrow[N\to+\infty]{\mathbb{P}}\nu^2(h)\ ,$$

(AN3) $$N_1\sum_{i=1}^{N_1}\left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\right)^2\sum_{j=1}^{N_2}\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}h(\xi_N(i,j))\xrightarrow[N\to+\infty]{\mathbb{P}}\mu_1 h\ ,$$

(AN4) $$N\sum_{i=1}^{N_1}\left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\right)^2\sum_{j=1}^{N_2}\left(\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\right)^2 h(\xi_N(i,j))\xrightarrow[N\to+\infty]{\mathbb{P}}\mu_2 h\ ,$$

(AN5) $$N_2\sum_{i=1}^{N_1}\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\sum_{j=1}^{N_2}\left(\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\right)^2 h(\xi_N(i,j))\xrightarrow[N\to+\infty]{\mathbb{P}}\mu_3 h\ ,$$

(AN6) $$\lim_{\lambda\to\infty}\sup_{N\in\mathbb{N}^*}\mathbb{P}\left(\max_{i\in[\![1,N_1]\!]}N_1\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\geq\lambda\right)=0\ .$$

Here (**AN1**) corresponds to a CLT and implies straightforwardly (**C1**). In addition, since (**AN6**) implies immediately (**C2**) we may conclude that asymptotic normality is stronger than consistency.

Remark 5. For a classic SMC with $N_1 = 1$ and $N_2 = N$ particles, this definition coincides with the asymptotic normality defined in [[Douc and Moulines, 2008], Définition 2] : (**AN1**) corresponds to the first assumption and (**AN4**) and (**AN5**) correspond to the second one.

3.3 Main results : operations on weighted archipelagos

We will next consider three different operations on archipelagos and show how these operations preserve the three convergence properties listed in the previous section.

3.3.1 Selection on the island level

The first operation, described in Algorithm 7, is referred to as SIL. This operation consists in converting an archipelago $\{(\Omega_N(i),\{(\xi_N(i,j),\omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ targeting some distribution η into an archipelago $\{(1,\{(\tilde{\xi}_N(i,j),\tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ with uniform island weights targeting the same distribution η. This step allows islands with small/large weights to be eliminated/duplicated, respectively. More precisely, a new family of islands is generated from the existing ones by selecting, conditionally independently given the input archipelago, new islands according to probabilities proportional to the island weights $\{\Omega_N(i)\}_{i=1}^{N_1}$. After this, the weights and the particles of the selected islands are copied deterministically (which of course implies that the particle weights of the new archipelago are bounded by the same constant $|\omega|_\infty$ as the ancestor

archipelago).

Data: $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$
Result: $\{(1, \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$
for $i \leftarrow 1$ **to** N_1 **do**
  draw $I_N(i) \sim \mathsf{Mult}(\{\Omega_N(i')\}_{i'=1}^{N_1})$;
  for $j \leftarrow 1$ **to** N_2 **do**
    set $\tilde{\xi}_N(i,j) \leftarrow \xi_N(I_N(i), j)$;
    set $\tilde{\omega}_N(i,j) \leftarrow \omega_N(I_N(i), j)$;
  end
end

Algorithm 7: Multinomial selection on the island level (SIL)

In the following we will abbreviate Algorithm 7 by writing

$$\text{“}\{(1, \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1} \leftarrow \mathrm{SIL}\left(\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}\right)\text{”}\,.$$

The following theorems state conditions under which SIL preserves consistency, exponential deviation, and asymptotic normality. The input and output in Algorithm 7 are respectively denoted by $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ and $\{(1, \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ and all proofs are found in section 3.6.

Theorem 3.3.1. *Assume that* $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *is consistent for* η. *Then also* $\{(1, \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *is consistent for* η.

Theorem 3.3.2. *Assume that* $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *satisfies exponential deviation for* $(\eta, \varrho, \{c_\ell\}_{\ell=1}^2)$. *Then also* $\{(1, \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *satisfies exponential deviation for* $(\eta, \varrho, \{c_\ell\}_{\ell=1}^2)$.

We impose the following assumption, guaranteeing that N_1 grows only subexponentially fast with respect to N_2.

(S) For all $\beta > 0$, $N_1 \exp(-\beta N_2) \to 0$ as $N \to \infty$.

Theorem 3.3.3. *Assume* ***(S)*** *and that* $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *satisfies exponential deviation for* $(\eta, \varrho, \{c_\ell\}_{\ell=1}^2)$ *and is asymptotically normal for* $(\eta, \sigma^2, \nu^2, \{\mu_\ell\}_{\ell=1}^3)$.
Then also $\{(1, \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *is asymptotically normal for* $(\eta, \tilde{\sigma}^2, \nu^2, \{\eta, \mu_3, \mu_3\})$, *where for all* $h \in \mathsf{F}_\mathrm{b}(\mathsf{X})$,

$$\tilde{\sigma}^2(h) = \sigma^2(h) + \nu^2(h)$$

(i.e. the SIL *operation modifies only* σ^2, μ_1, *and* μ_2*).*

3.3.2 Selection on the individual level

A second operation, described in Algorithm 8, is referred to as SiL, and consists in converting a weighted archipelago $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ targeting some distribution η into an archipelago $\{\Omega_N(i), \{(\tilde{\xi}_N(i,j), 1)\}_{j=1}^{N_2}\}_{i=1}^{N_1}$ with uniform particle weights targeting the same distribution η. This step allows particles with large/small weights to be duplicated/eliminated, respectively. Note that the island

weights remain unaffected.

Data: $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$
Result: $\{(\Omega_N(i), \{(\tilde{\xi}_N(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}$
for $i \leftarrow 1$ **to** N_1 **do**
  for $j \leftarrow 1$ **to** N_2 **do**
    draw $J_N(i,j) \sim \mathsf{Mult}(\{\omega_N(i,j')\}_{j'=1}^{N_2})$;
    set $\tilde{\xi}_N(i,j) \leftarrow \xi_N(i, J_N(i,j))$;
  end
end

Algorithm 8: Multinomial resampling on the individual level (SiL)

Trivially, the particle weights are bounded by $|\omega|_\infty = 1$ in this case. As for the SIL operation, we will express Algorithm 8 in a compact form by writing

$$\text{“}\{(\Omega_N(i), \{(\tilde{\xi}_N(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1} \leftarrow \mathrm{SiL}\left(\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}\right)\text{”} .$$

The following theorems state conditions under which SiL preserves consistency, exponential deviation inequality, and asymptotic normality.
Here, $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ and $\{(\Omega_N(i), \{(\tilde{\xi}_N(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ denote the input and output, respectively, of Algorithm 8.

Theorem 3.3.4. *Assume that* $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *is consistent for* η. *Then also* $\{(\Omega_N(i), \{(\tilde{\xi}_N(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *is consistent for* η.

Theorem 3.3.5. *Assume that* $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *satisfies exponential deviation for* $(\eta, \varrho, \{c_\ell\}_{\ell=1}^2)$. *Then also* $\{(\Omega_N(i), \{(\tilde{\xi}_N(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *satisfies exponential deviation for* $(\eta, 1, \{\tilde{c}_\ell\}_{\ell=1}^2)$, *where* $\tilde{c}_1 = 4(1 \vee c_1)$ *and* $\tilde{c}_2 = (1 \wedge (c_2\varrho^2/2))/8$.

Theorem 3.3.6. *Assume that* $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *satisfies exponential deviation for* $(\eta, \varrho, \{c_\ell\}_{\ell=1}^2)$ *and is asymptotically normal for* $(\eta, \sigma^2, \nu^2, \{\mu_\ell\}_{\ell=1}^3)$.
Then also $\{(\Omega_N(i), \{(\tilde{\xi}_N(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *is asymptotically normal for* $(\eta, \tilde{\sigma}^2, \tilde{\nu}^2, \{\mu_1, \mu_1, \eta\})$, *where for all* $h \in \mathsf{F}_\mathrm{b}(\mathsf{X})$,

$$\begin{cases} \tilde{\sigma}^2(h) = \sigma^2(h) + \mu_1\{(h - \eta h)^2\} , \\ \tilde{\nu}^2(h) = \nu^2(h) + \eta\{(h - \eta h)^2\} . \end{cases}$$

Again, proofs are found in section 3.6.

3.3.3 Mutation

The last operation we consider is *Mutation*, described in Algorithm 9. This operation converts, using importance sampling on the individual level, an archipelago $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ targeting $\eta \in \mathcal{P}(\mathsf{X})$ into another archipelago $\{\tilde{\Omega}_N(i), \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2}\}_{i=1}^{N_1}$ targeting some *other* probability distribution $\tilde{\eta}$, defined on another state space $(\tilde{\mathsf{X}}, \tilde{\mathcal{X}})$. The distribution $\tilde{\eta}$ is related to η through the identity

$$\tilde{\eta}h = \frac{\eta Q h}{\eta Q \mathbb{1}_\mathsf{X}} \quad (h \in \mathsf{F}_\mathrm{b}(\tilde{\mathsf{X}})) \tag{3.3.1}$$

where $Q : \mathsf{X} \times \tilde{\mathcal{X}} \to \mathbb{R}_+$ is a possibly unnormalized transition kernel. In the algorithm that follows, let $R : \mathsf{X} \times \tilde{\mathcal{X}} \to \mathbb{R}_+$ be a (normalized) transition kernel such that $Q(x, \cdot) \ll R(x, \cdot)$ for all $x \in \mathsf{X}$, and denote the

corresponding Radon-Nikodym derivatives by

$$w(x,\tilde{x}) \stackrel{\text{def}}{=} \frac{\mathrm{d}Q(x,\cdot)}{\mathrm{d}R(x,\cdot)}(\tilde{x}) \quad ((x,\tilde{x}) \in \mathsf{X}\times\tilde{\mathsf{X}}) \ .$$

In the sequel, we will refer to the mapping w as the *importance weight function* and assume that $w \in \mathsf{F}_{\mathrm{b}}(\mathsf{X}\otimes\tilde{\mathsf{X}})$ and $Q\mathbb{1}_{\tilde{\mathsf{X}}} \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$.

Data: $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$, Q, R
Result: $\{(\tilde{\Omega}_N(i), \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$
for $i \leftarrow 1$ **to** N_1 **do**
  for $j \leftarrow 1$ **to** N_2 **do**
    draw $\tilde{\xi}_N(i,j) \sim R(\xi_N(i,j),\cdot)$;
    set $\tilde{\omega}_N(i,j) \leftarrow w(\xi_N(i,j),\tilde{\xi}_N(i,j))\omega_N(i,j)$;
  end
  set $\tilde{\Omega}_N(i) \leftarrow \Omega_N(i)\dfrac{\sum_{j'=1}^{N_2}\tilde{\omega}_N(i,j')}{\sum_{j''=1}^{N_2}\omega_N(i,j'')}$;
end

Algorithm 9: Mutation

As before, we will abbreviate Algorithm 9 by writing

$$\text{“}\{(\tilde{\Omega}_N(i), \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1} \quad \leftarrow \quad \mathsf{Mut}\langle Q\rangle\left(\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}, R\right)\text{”} \quad ,$$

where the kernel Q is included in the notation for the sake of completeness. Note that the Mutation operation forms indeed a proper weighted archipelago with $|\tilde{\omega}|_\infty = |\omega|_\infty \, \|w\|_\infty$. In conformity with the SIL and SiL operations, the Mutation operation preserves consistency, exponential deviation, and asymptotic normality. This is established below, where $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ and $\{(\tilde{\Omega}_N(i), \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ denote consequently the input and output of Algorithm 9, respectively.

Theorem 3.3.7. *Assume that* $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *is consistent for* η. *Then* $\{(\tilde{\Omega}_N(i), \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *is consistent for* $\tilde{\eta}$ *defined in* (3.3.1).

Theorem 3.3.8. *Assume that* $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *satisfies exponential deviation for* $(\eta, \varrho, \{c_\ell\}_{\ell=1}^2)$. *Then* $\{(\tilde{\Omega}_N(i), \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *satisfies exponential deviation for* $(\tilde{\eta}, \tilde{\varrho}, \{\tilde{c}_\ell\}_{\ell=1}^2)$, *where* $\tilde{\varrho} = \varrho \times \eta Q\mathbb{1}_{\tilde{\mathsf{X}}}$, $\tilde{c}_1 = 2(2 \vee c_1)$, *and*

$$\tilde{c}_2 = \frac{1}{2}\left(\frac{1}{\delta^2} \wedge \frac{c_2}{2\|Q\mathbb{1}_{\tilde{\mathsf{X}}}\|_\infty^2}\right) \ ,$$

with $\delta \stackrel{\text{def}}{=} |\tilde{\omega}|_\infty + |\omega|_\infty \|Q\mathbb{1}_{\tilde{\mathsf{X}}}\|_\infty$.

Theorem 3.3.9. *Assume that* $\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *satisfies exponential deviation for* $(\eta, \varrho, \{c_\ell\}_{\ell=1}^2)$ *and is asymptotically normal for* $(\eta, \sigma^2, \nu^2, \{\mu_\ell\}_{\ell=1}^3)$. *Then the mutated archipelago* $\{(\tilde{\Omega}_N(i), \{(\tilde{\xi}_N(i,j), \tilde{\omega}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *is asymptotically normal for* $(\tilde{\eta}, \tilde{\sigma}^2, \tilde{\nu}^2, \{\tilde{\mu}_\ell\}_{\ell=1}^3)$, *where* $\tilde{\eta}$ *is defined in* (3.3.1) *and for all* $h \in \mathsf{F}_{\mathrm{b}}(\tilde{\mathsf{X}})$,

$$\begin{cases} \tilde{\sigma}^2(h) = \left(\sigma^2\{Q(h-\tilde{\eta}h)\} + \mu_2 R\{w^2(h-\tilde{\eta}h)^2\} - \mu_2\{Q^2(h-\tilde{\eta}h)\}\right)/(\eta Q\mathbb{1}_{\tilde{\mathsf{X}}})^2 \ , \\ \tilde{\nu}^2(h) = \left(\nu^2\{Q(h-\tilde{\eta}h)\} + \mu_3 R\{w^2(h-\tilde{\eta}h)^2\} - \mu_3\{Q^2(h-\tilde{\eta}h)\}\right)/(\eta Q\mathbb{1}_{\tilde{\mathsf{X}}})^2 \ , \\ \tilde{\mu}_1 h = \mu_1 Qh/\eta Q\mathbb{1}_{\tilde{\mathsf{X}}} \ , \\ \tilde{\mu}_2 h = \mu_2 R(w^2 h)/(\eta Q\mathbb{1}_{\tilde{\mathsf{X}}})^2 \ , \\ \tilde{\mu}_3 h = \mu_3 R(w^2 h)/(\eta Q\mathbb{1}_{\tilde{\mathsf{X}}})^2 \ , \end{cases}$$

(where $Q^2 h(x) \stackrel{\text{def}}{=} \{Qh(x)\}^2$ *and* $R(w^2 h)(x) \stackrel{\text{def}}{=} \int w^2(x,x')h(x')\,R(x,\mathrm{d}x')$ *for all* $x \in \mathsf{X}$ *and* $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$*).*

Remark 10. Note that Theorem 3.3.6 and Theorem 3.3.9 hold true regardless of the intermutual rates by which N_1 and N_2 tend to infinity with N. In particular, these results do not, on the contrary to Theorem 3.3.3, require the condition (**S**). This is in line with what we expect, as the SiL and Mutation operations do not involve any island interaction.

3.4 Applications

3.4.1 Feynman-Kac models

For a sequence of unnormalized transition kernels $\{Q_n\}_{n\in\mathbb{N}}$ defined on some common measurable space $(\mathsf{X},\mathcal{X})$ and some probability distribution $\eta_0 \in \mathcal{P}(\mathsf{X})$, a sequence $\{\eta_n\}_{n\in\mathbb{N}}$ of *Feynman-Kac measures* is defined by

$$\eta_n h \stackrel{\text{def}}{=} \frac{\gamma_n h}{\gamma_n \mathbb{1}_{\mathsf{X}}}\,, \quad n \in \mathbb{N},\ h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})\,, \tag{3.4.1}$$

where

$$\gamma_n h \stackrel{\text{def}}{=} \int \cdots \int h(x_n)\,\eta_0(\mathrm{d}x_0) \prod_{p=0}^{n-1} Q_p(x_p, \mathrm{d}x_{p+1}) \quad (h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X}))$$

(with usual convention $\prod_{p=m}^{n} a_p = 1$ when $m > n$). We may express recursively the sequences of unnormalized and normalized Feynman-Kac measures as, for $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$ and $(m,n) \in \mathbb{N}$ with $m \le n$,

$$\gamma_n h = \gamma_m Q_m \cdots Q_{n-1} h \quad \text{and} \quad \eta_n h = \frac{\gamma_m Q_m \cdots Q_{n-1} h}{\gamma_m Q_m \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}} = \frac{\eta_m Q_m \cdots Q_{n-1} h}{\eta_m Q_m \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}}\,,$$

respectively, with the convention $Q_m \cdots Q_\ell = \mathrm{id}$ if $m > \ell$. In particular,

$$\eta_{n+1} h = \frac{\eta_n Q_n h}{\eta_n Q_n \mathbb{1}_{\mathsf{X}}} \quad (h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X}), n \in \mathbb{N})\,, \tag{3.4.2}$$

which means that we may cast the model into the framework considered in subsection 3.3.3.

Example 3. A special instance of the previous framework is formed naturally by specifying, first, a sequence $\{M_n\}_{n\in\mathbb{N}}$ of normalized (Markov) transition kernels on $(\mathsf{X},\mathcal{X})$ with an associated initial distribution χ and, second, *potential functions* $\{g_n\}_{n\in\mathbb{N}^*}$, where $g_n : \mathsf{X} \to \mathbb{R}_+^*$ for all $n \in \mathbb{N}^*$, and letting $Q_n h(x) \stackrel{\text{def}}{=} M_n(g_{n+1} h)(x)$, $n \in \mathbb{N}^*$, $x \in \mathsf{X}$, and $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$. In addition, $\eta_0 \stackrel{\text{def}}{=} \chi$. This setup covers a large variety of important models in probability and statistics, such as *optimal filtering* in hidden Markov models (or *state-space models* ; see, e.g., [Cappé et al., 2005]) and models for the analysis of *rare events* [Del Moral and Garnier, 2005; Cérou et al., 2012]. We will return to this setting in subsection 3.4.3.

Using a Feynman-Kac model in practice is typically non-trivial as neither the distribution flow $\{\gamma_n\}_{n\in\mathbb{N}}$ nor $\{\eta_n\}_{n\in\mathbb{N}}$ can be computed in a closed-form in general (with the exception of the very specific cases of optimal filtering in linear state-space models, in which case the solution is provided by the *Kalman filter*, or hidden Markov models with finite state space).

3.4.2 The double bootstrap algorithm with adaptive selection (B^2ASIL) algorithm

In this section, our aim is to form online a sequence of archipelagos targeting the Feynman-Kac flow $\{\eta_n\}_{n\in\mathbb{N}}$ by using sequentially the operations described in section 3.3. A special feature of the approach that we

consider is that the SIL operation is not performed systematically at every iteration of the algorithm, but only when the island weights fail to satisfy some appropriately defined skewness criterion. In this way we avoid adding unnecessary variance to the estimator. More specifically, we will analyze an algorithm proposed in [Vergé et al., 2015a, Algorithm 3], where SIL is executed on the basis of the so-called CV (see [Kong et al., 1994] and [Liu, 2008]) given by $\mathrm{CV}_N^2(\{\Omega_N(i)\}_{i=1}^{N_1})$, where

$$\mathrm{CV}_N^2 : (\mathbb{R}_+^*)^{N_1} \ni \{a(i)\}_{i=1}^{N_1} \mapsto N_1 \sum_{i=1}^{N_1} \left(\frac{a(i)}{\sum_{i'=1}^{N_1} a(i')} \right)^2 - 1 \,. \tag{3.4.3}$$

The CV is closely related to the ESS proposed in [Liu, 1996], which is the criterion used in [Vergé et al., 2015a]; nevertheless, since the ESS can be expressed as $N_1/[1+\mathrm{CV}_N^2(\{a(i)\}_{i=1}^{N_1})]$, the two criteria are equivalent. Note that the CV is minimal (zero) when all island weights are perfectly equal and maximal (N_1-1) in the situation of maximal skewness, i.e., when the total mass of the system is carried by a single island (a situation which is however not possible in our framework, as we always assume the island weights to be strictly positive). More specifically, as long as the CV stays below a specified threshold $\tau > 0$, we let the N_1 islands evolve without interaction according to mutation and selection on the individual level. However, when the island weights get too dispersed as measured by the CV criterion, the islands are rejuvenated by SIL. The scheme, referred to by us as the B^{2}ASIL, is described in Algorithm 10, where we have added the iteration index p to the weighted archipelagos returned by the algorithm.

Data: $\{R_p\}_{p=0}^{n-1}$, τ
Result: $\{(\Omega_N^{(p)}(i), \{(\xi_N^{(p)}(i,j), \omega_N^{(p)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$, $p \in [\![0, n]\!]$
/* Initialization */
for $i \leftarrow 1$ **to** N_1 **do**
  for $j \leftarrow 1$ **to** N_2 **do**
    $\xi_N^{(0)}(i,j) \sim \eta_0$;
    $\omega_N^{(0)}(i,j) \leftarrow 1$;
  end
  $\Omega_N^{(0)}(i) \leftarrow 1$;
end
$\{(\Omega_N^{(1)}(i), \{(\xi_N^{(1)}(i,j), \omega_N^{(1)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1} \leftarrow \mathsf{Mut}\langle Q_0\rangle \left(\{(1, \{(\xi_N^{(0)}(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}, R_0 \right)$;
/* Main loop */
for $p \leftarrow 1$ **to** $n-1$ **do**
  /* Checking island weight skewness */
  if $\mathrm{CV}_N^2(\{\Omega_N^{(p)}(i)\}_{i=1}^{N_1}) > \tau$ **then**
    /* Selection on the island level */
    $\{(\tilde{\Omega}_N^{(p)}(i), \{(\tilde{\xi}_N^{(p)}(i,j), \tilde{\omega}_N^{(p)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1} \leftarrow \mathrm{SIL} \left(\{(\Omega_N^{(p)}(i), \{(\xi_N^{(p)}(i,j), \omega_N^{(p)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1} \right)$;
  else
    /* No action */
    $\{(\tilde{\Omega}_N^{(p)}(i), \{(\tilde{\xi}_N^{(p)}(i,j), \tilde{\omega}_N^{(p)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1} \leftarrow \{(\Omega_N^{(p)}(i), \{(\xi_N^{(p)}(i,j), \omega_N^{(p)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$;
  end
  /* Selection on the individual level */
  $\{(\tilde{\Omega}_N^{(p)}(i), \{(\check{\xi}_N^{(p)}(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1} \leftarrow \mathrm{SiL} \left(\{(\tilde{\Omega}_N^{(p)}(i), \{(\tilde{\xi}_N^{(p)}(i,j), \tilde{\omega}_N^{(p)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1} \right)$;
  /* Mutation */
  $\{(\Omega_N^{(p+1)}(i), \{(\xi_N^{(p+1)}(i,j), \omega_N^{(p+1)}(i,j))\}_{j=1}^{N_2}\}_{i=1}^{N_1} \leftarrow \mathsf{Mut}\langle Q_p\rangle \left(\{(\tilde{\Omega}_N^{(p)}(i), \{(\check{\xi}_N^{(p)}(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}, R_p \right)$;
end

Algorithm 10: The B^{2}ASIL algorithm

Using the theoretical results obtained in section 3.3 we may prove the following result, establishing that exponential deviation and asymptotic normality are preserved through one iteration of the B^{2}ASIL algorithm. As a by product we obtain the incremental asymptotic variance caused by an iteration. Since focus is set on asymptotic normality, we provide recursive formulas describing precisely the evolution of the functionals and measures involved in (**AN**1–5), while leaving the derivation of the analogous formulas for the constants of the exponential deviation bound (**D**) to the reader. The proof of this result provides a nice illustration of the efficiency by which the theoretical results obtained in section 3.3, despite appearing somewhat involved at a first sight, can be applied for analyzing sequences of archipelagos produced by executing alternatively the SIL, SiL, and Mutation operations in an arbitrary order.

Theorem 3.4.1. *Assume (**S**) and that* $\{(\Omega_N^{(n)}(i), \{(\xi_N^{(n)}(i,j), \omega_N^{(n)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *satisfies exponential deviation and is asymptotically normal for* $(\eta_n, \sigma_n^2, \nu_n^2, \{\mu_\ell^{(n)}\}_{\ell=1}^3)$, $n \in \mathbb{N}^*$.
Then the archipelago $\{(\Omega_N^{(n+1)}(i), \{(\xi_N^{(n+1)}(i,j), \omega_N^{(n+1)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ *generated through one iteration of Algorithm 10 is asymptotically normal for* $(\eta_{n+1}, \sigma_{n+1}^2, \nu_{n+1}^2, \{\mu_\ell^{(n+1)}\}_{\ell=1}^3)$, *where* η_{n+1} *is given by* (3.4.2) *and for all* $h \in \mathsf{F}_\mathrm{b}(\mathsf{X})$,

$$\begin{cases} \sigma_{n+1}^2(h) = \dfrac{\sigma_n^2\{Q_n(h-\eta_{n+1}h)\} + \varepsilon_n \nu_n^2\{Q_n(h-\eta_{n+1}h)\}}{(\eta_n Q_n \mathbb{1}_\mathsf{X})^2} \\ \qquad\qquad + \dfrac{\varepsilon_n \eta_n R_n\{w_n^2(h-\eta_{n+1}h)^2\} + (1-\varepsilon_n)\mu_1^{(n)} R_n\{w_n^2(h-\eta_{n+1}h)^2\}}{(\eta_n Q_n \mathbb{1}_\mathsf{X})^2} \,, \\ \nu_{n+1}^2(h) = \dfrac{\nu_n^2\{Q_n(h-\eta_{n+1}h)\} + \eta_n R_n\{w_n^2(h-\eta_{n+1}h)^2\}}{(\eta_n Q_n \mathbb{1}_\mathsf{X})^2} \,, \\ \mu_1^{(n+1)}h = (1-\varepsilon_n)\dfrac{\mu_1^{(n)} Q_n h}{\eta_n Q_n \mathbb{1}_\mathsf{X}} + \varepsilon_n \eta_{n+1} h \,, \\ \mu_2^{(n+1)}h = (1-\varepsilon_n)\dfrac{\mu_1^{(n)} R_n(w_n^2 h)}{(\eta_n Q_n \mathbb{1}_\mathsf{X})^2} + \varepsilon_n \mu_3^{(n+1)} h \,, \\ \mu_3^{(n+1)}h = \dfrac{\eta_n R_n(w_n^2 h)}{(\eta_n Q_n \mathbb{1}_\mathsf{X})^2} \,, \end{cases}$$

where $\varepsilon_n \overset{\text{def}}{=} \mathbb{1}\{\mu_1^{(n)}\mathbb{1}_\mathsf{X} > \tau + 1\}$.

Démonstration. First, note that since the input archipelago satisfies (**AN**3), it holds that

$$\mathrm{CV}_N^2(\{\Omega_N^{(n)}(i)\}_{i=1}^{N_1}) \xrightarrow{\mathbb{P}} \mu_1^{(n)}\mathbb{1}_\mathsf{X} - 1 \,,$$

which implies

$$\mathbb{1}\{\mathrm{CV}_N^2(\{\Omega_N^{(n)}(i)\}_{i=1}^{N_1}) > \tau\} \xrightarrow[N\to+\infty]{\mathbb{P}} \varepsilon_n \,,$$

where ε_n is defined in the statement of the theorem. Consequently, after the **if-else** statement in Algorithm 10, the resulting archipelago $\{(\tilde\Omega_N^{(p)}(i), \{(\tilde\xi_N^{(p)}(i,j), \tilde\omega_N^{(p)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ satisfies, by Theorem 3.3.2 and Theorem 3.3.3, exponential deviation and asymptotic normality, the latter for

$$\begin{cases} (\eta_n, \sigma_n^2, \nu_n^2, \{\mu_\ell^{(n)}\}_{\ell=1}^3) & \text{if } \varepsilon_n = 0 \,, \\ (\eta_n, \sigma_n^2 + \nu_n^2, \nu_n^2, \eta_n, \mu_3^{(n)}, \mu_3^{(n)}) & \text{if } \varepsilon_n = 1 \,. \end{cases}$$

Thus, the archipelago $\{(\tilde\Omega_N^{(p)}(i), \{(\check\xi_N^{(p)}(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ obtained after additional SiL satisfies, by Theorem 3.3.5 and Theorem 3.3.6, exponential deviation as well as asymptotic normality, the latter for

$$\begin{cases} (\eta_n, \sigma_n^2(\cdot) + \mu_1^{(n)}\{(\cdot - \eta_n\cdot)^2\}, \nu_n^2(\cdot) + \eta_n\{(\cdot - \eta_n\cdot)^2\}, \mu_1^{(n)}, \mu_1^{(n)}, \eta_n) & \text{if } \varepsilon_n = 0 \,, \\ (\eta_n, \sigma_n^2(\cdot) + \nu_n^2(\cdot) + \eta_n\{(\cdot - \eta_n\cdot)^2\}, \nu_n^2(\cdot) + \eta_n\{(\cdot - \eta_n\cdot)^2\}, \eta_n, \eta_n, \eta_n) & \text{if } \varepsilon_n = 1 \,. \end{cases}$$

Finally, considering also the final Mutation operation in Algorithm 10, and propagating, for the two different values of ε_n, the quantities of the previous display through the updating formulas of Theorem 3.3.9, establishes, together with Theorem 3.3.8, the statement of the theorem. □

Corollary 3.4.2. *Assume* **(S)**. *Then all archipelagos* $\{(\Omega_N^{(n)}(i), \{(\xi_N^{(n)}(i,j), \omega_N^{(n)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$, $n \in \mathbb{N}$, *produced by the* B^{2}ASIL *algorithm satisfies exponential deviation and asymptotic normality, where for* $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$ *and* $n \in \mathbb{N}^*$,

$$\begin{cases} \sigma_n^2(h) = \sum_{\ell=0}^{n-1} \left(1 + \sum_{p=\ell+1}^{n-1} \varepsilon_p\right) \frac{\eta_\ell R_\ell \{w_\ell^2 Q_{\ell+1} \cdots Q_{n-1}(h - \eta_n h)^2\}}{(\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}})^2} , \\ \nu_n^2(h) = \sum_{\ell=0}^{n-1} \frac{\eta_\ell R_\ell \{w_\ell^2 Q_{\ell+1} \cdots Q_{n-1}(h - \eta_n h)^2\}}{(\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}})^2} , \\ \mu_1^{(n)} h = \eta_n h \end{cases}$$

(under the standard conventions that $\prod_{\ell=m}^n a_\ell = 1$, $\sum_{\ell=m}^n a_\ell = 0$, *and* $Q_m \cdots Q_n = \mathrm{id}$ *if* $m > n$*), and* $\{\varepsilon_n\}_{n \in \mathbb{N}^*}$ *is given in Theorem 3.4.1. In addition,* $\mu_1^{(0)} = \eta_0$ *and*

$$\sigma_0^2(h) = \nu_0^2(h) = \eta_0\{(h - \eta_0 h)^2\} \quad (h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})) .$$

Démonstration. The non-recursive expression above are verified using induction. More specifically, one assumes that the given expressions of $(\sigma_n^2, \nu_n^2, \mu_1^{(n)})$ hold true for some $n \in \mathbb{N}$ (and for all $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$) and plug the same into the recursive expressions established in Theorem 3.4.1 under repeated use of the identities

$$Q_\ell \cdots Q_{n-1}\{Q_n(h - \eta_{n+1}h) - \eta_n Q_n(h - \eta_{n+1}h)\} = Q_\ell \cdots Q_n(h - \eta_{n+1}h) \qquad (h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X}), \ell \in \mathbb{N}) ,$$

and

$$\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}} \times \eta_n Q_n \mathbb{1}_{\mathsf{X}} = \eta_\ell Q_\ell \cdots Q_n \mathbb{1}_{\mathsf{X}} \quad (\ell \in \mathbb{N}) .$$

We leave this to the reader. To verify the base case $n = 1$, note that the initial archipelago $\{(1, \{(\xi_N^{(0)}(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}\}$ is, by the standard CLT and law of large numbers of for independent random variables, asymptotically normal for $(\eta_0, \sigma_0^2, \sigma_0^2, \eta_0, \eta_0, \eta_0)$, where $\sigma_0^2(h) = \eta_0\{(h - \eta_0 h)^2\}$, $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$, and satisfies, by Hoeffding's inequality, exponential deviation for $(\eta_0, 1, 2, 1/2)$. Now, by Theorem 3.3.9 and Theorem 3.3.9 also the weighted archipelago $\{(\Omega_N^{(1)}(i), \{(\xi_N^{(1)}(i,j), \omega_N^{(1)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$, obtained by mutating the initial archipelago, satisfies exponential deviation and asymptotic normality for $\mu_1^{(1)} = \eta_1$ and

$$\sigma_1^2(h) = \nu_1^2(h) = \frac{\eta_0 R_0 \{w_0^2 (h - \eta_1 h)^2\}}{(\eta_0 Q_0 \mathbb{1}_{\mathsf{X}})^2} \quad (h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})) .$$

Under the standard conventions, this is however in agreement with the formula in the statement of the theorem. This completes the proof. □

Of special interest is of course the special case where SIL is applied systematically at every iteration, corresponding to $\tau = 0$. This yields the standard B^2 algorithm, in which case the asymptotic variance is given by the following corollary.

Corollary 3.4.3. *Assume* **(S)**. *Then all archipelagos* $\{(\Omega_N^{(n)}(i), \{(\xi_N^{(n)}(i,j), \omega_N^{(n)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$, $n \in \mathbb{N}$, *produced by the* B^2 *algorithm satisfies exponential deviation and asymptotic normality, where for* $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$ *and* $n \in \mathbb{N}^*$,

$$\sigma_n^2(h) = \sum_{\ell=0}^{n-1} (n - \ell) \frac{\eta_\ell R_\ell \{w_\ell^2 Q_{\ell+1} \cdots Q_{n-1}(h - \eta_n h)^2\}}{(\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}})^2} , \tag{3.4.4}$$

and

$$\sigma_0^2(h) = \eta_0\{(h - \eta_0 h)^2\} \, .$$

Démonstration. The result is an immediate consequence of Corollary 3.4.2, as $\tau = 0$ implies that $\varepsilon_n = 1$ for all $n \in \mathbb{N}^*$. □

On the other hand, letting $\varepsilon_n = 0$ for all $n \in \mathbb{N}^*$ in Corollary 3.4.2, corresponding to the case where SIL is never applied, yields the variance

$$\sigma_n^2(h) = \sum_{\ell=0}^{n-1} \frac{\eta_\ell R_\ell\{w_\ell^2 Q_{\ell+1} \cdots Q_{n-1}(h - \eta_n h)^2\}}{(\eta_\ell Q_\ell \cdots Q_{n-1}\mathbb{1}_{\mathsf{X}})^2} \, , \tag{3.4.5}$$

which we recognize as the well-known formula for the asymptotic variance of the standard SMC algorithm (more specifically, the *sequential importance sampling with resampling*, SISR, algorithm). This is completely in line with our expectations, as such an algorithm would simply propagate N_1 independent (non-interacting) islands, each island evolving as a standard SMC algorithm based on N_2 particles.

3.4.3 Long-term stability of the double bootstrap (B^2) algorithm

As a last part of our study we establish the long-term numerical stability of the B^2 algorithm by providing a uniform in time bound on the asymptotic variance of its output. Throughout this section, in the spirit of Example 3, we will assume that each unnormalized transition kernel Q_p, $p \in \mathbb{N}$, can be decomposed into a normalized transition kernel $M_p : \mathsf{X} \times \mathcal{X} \to [0, 1]$ and a nonnegative potential potential function $g_{p+1} : \mathsf{X} \to \mathbb{R}_+$, i.e., for all $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$ and $x \in \mathsf{X}$,

$$Q_p h(x) = M_p(g_{p+1} h)(x) \, . \tag{3.4.6}$$

In this setting, given a sequence $\{R_p\}_{p\in\mathbb{N}}$ of proposal kernels such that $M_p(x, \cdot) \ll R_p(x, \cdot)$ for all $x \in \mathsf{X}$ and $p \in \mathbb{N}$, the importance weight function is given by

$$w_p(x, x') = g_{p+1}(x') \frac{\mathrm{d}M_p(x, \cdot)}{\mathrm{d}R_p(x, \cdot)} \quad (x, x') \in \mathsf{X}^2 \, .$$

Remark 4. Instead of letting the Feynman-Kac distribution flow be generated by the unnormalized kernel (3.4.6), one could, as in [Vergé et al., 2015a], consider an alternative model with a flow $\{\tilde{\eta}_p\}_{p\in\mathbb{N}}$ generated by

$$\tilde{Q}_p h(x) = g_{p-1}(x) M_p h(x) \quad (h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X}), x \in \mathsf{X}, p \in \mathbb{N}^*) \, , \tag{3.4.7}$$

with $\tilde{Q}_0 = M_0$ and $\tilde{\eta}_0 = \chi$. In [Del Moral, 2004] the two models (3.4.6) and (3.4.7) are referred to as *updated* and *prediction* Feynman-Kac models, respectively. For the prediction model, it is, in the case of the B^2 algorithm, possible to achieve *full adaptation* (borrowing the terminology of [Pitt and Shephard, 1999]) of the algorithm, i.e., to generate archipelagos $\{(1, \{(1, \xi_N^{(p)}(i, j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$, $p \in \mathbb{N}$, with uniformly weighted islands and individuals targeting the distribution sequence of interest, by letting $R_p = M_p$ for all $p \in \mathbb{N}$ and decomposing the dynamics (3.4.7) into the product

$$\tilde{Q}_p = G_p M_p \, , \tag{3.4.8}$$

where $G_p h(x) = g_p(x) h(x)$, $(x, h) \in \mathsf{X} \times \mathsf{F}_{\mathrm{b}}(\mathsf{X})$, is the *Boltzmann multiplicative operator* associated with the potential g_p. Now (3.4.8) allows also the Feynman-Kac transition according to $\tilde{Q}_p$ to be decomposed into *two* subsequent Feynman-Kac sub-transitions, the first according to G_p and the other according to M_p. The former corresponds to the Mutation operation

$$\{(\check{\Omega}_N^{(p)}(i), \{(\check{\omega}_N^{(p)}(i, j), \xi_N^{(p)}(i, j))\}_{i=1}^{N_2})\}_{i=1}^{N_1} \leftarrow \mathsf{Mut}\langle G_p\rangle \left(\{(1, \{(1, \xi_N^{(p)}(i, j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}, \mathrm{id}\right) \, , \tag{3.4.9}$$

which simply assigns each particle and island the weights $\check{\omega}_N(i,j) = g_p(\xi_N^{(p)}(i,j))$ and $\check{\Omega}_N^{(p)}(i) = \sum_{j=1}^{N_2} g_p(\xi_N^{(p)}(i,j))/N_2$, respectively (where we assumed that we start with uniformly weighted islands and individuals). After this *weighing operation*, the output (3.4.9) is, in accordance with Algorithm 10 (with $\tau = 0$), subjected to the SIL and SiL operations followed by the mutation operation

$$\{(1, \{(1, \xi_N^{(p+1)}(i,j))\}_{i=1}^{N_2})\}_{i=1}^{N_1} \leftarrow \mathsf{Mut}\langle M_p \rangle \left(\{(1, \{(1, \tilde{\xi}_N^{(p)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}, M_p \right) ,$$

yielding an archipelago with perfectly uniform island and individual weights approximating the Feynman-Kac distribution $\tilde{\eta}_{p+1}$ at the next time point. Also this algorithm may be analyzed straightforwardly using our results, and carrying through this analysis retrieves exactly the variance expression obtained in [Vergé et al., 2015a, Equation 43]. We leave this as an exercise to the interested reader.

The previous way of obtaining an archipelago with uniformly weighted islands and individuals approximating the prediction Feynman-Kac distribution flow can be viewed as a special instance of a general *auxiliary double bootstrap algorithm* (extending the so-called *auxiliary particle filter* proposed in [Pitt and Shephard, 1999]) based on the decomposition

$$Q_p = T_p \check{Q}_p ,$$

where $T_p h(x) = t_p(x) h(x)$, $(x,h) \in \mathsf{X} \times \mathsf{F}_\mathrm{b}(\mathsf{X})$, is a Boltzmann multiplicative operator associated with some positive auxiliary importance weight function $t_p \in \mathsf{F}_\mathrm{b}(\mathsf{X})$, and

$$\check{Q}_p(x,h) \stackrel{\mathrm{def}}{=} t_p^{-1}(x) Q_p h(x) \quad (x \in \mathsf{X}, h \in \mathsf{F}_\mathrm{b}(\mathsf{X})) .$$

In analogy with the previous, we may thus construct an alternative algorithm approximating $\{\eta_p\}_{p \in \mathbb{N}}$ by furnishing the main loop of the B^2 algorithm with a prefatory weighing operation

$$\{(\check{\Omega}_N^{(p)}(i), \{(\check{\omega}_N^{(p)}(i,j), \xi_N^{(p)}(i,j))\}_{i=1}^{N_2})\}_{i=1}^{N_1} \leftarrow \mathsf{Mut}\langle T_p \rangle \left(\{(\tilde{\Omega}_N^{(p)}(i), \{(\tilde{\omega}_N^{(p)}(i,j), \xi_N^{(p)}(i,j))\}_{i=1}^{N_2})\}_{i=1}^{N_1}, \mathrm{id} \right) , \tag{3.4.10}$$

and, after intermediate SIL and SiL operations, a terminating Mutation operation

$$\{(\Omega_N^{(p+1)}(i), \{(\omega_N^{(p+1)}(i,j), \xi_N^{(p+1)}(i,j))\}_{i=1}^{N_2})\}_{i=1}^{N_1} \leftarrow \mathsf{Mut}\langle \check{Q}_p \rangle \left(\{(1, \{(1, \tilde{\xi}_N^{(p)}(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}, R_p \right) ,$$

where, consequently, the all weights are given by the importance weight function

$$\check{w}_p(x,x') = t_p^{-1}(x) \frac{\mathrm{d}Q_p(x,\cdot)}{\mathrm{d}R_p(x,\cdot)}(x') \quad ((x,x') \in \mathsf{X}^2) .$$

Thus, choosing $t_p(x)$ as some prediction of the value of the derivative $\mathrm{d}Q_p(x,\cdot)/\mathrm{d}R_p(x,\cdot)$ in the support of $R_p(x,\cdot)$ yields close to uniformly weighted islands and individuals (i.e., a close to fully adapted algorithm); for instance, following [Pitt and Shephard, 1999], a possible design is $t_p(x) = \mathrm{d}Q_p(x,\cdot)/\mathrm{d}R_p(x,\cdot)(R_p\,\mathrm{id}(x))$. Of course, also this algorithm can be analyzed easily using our results (we refer to [Douc et al., 2009] for such an analysis of the standard auxiliary particle filter).

3.4.4 Uniform in time convergence under the strong mixing assumption

When studying the numerical stability of the B^2 algorithm we will first work under the following *strong mixing condition*.

(M) (i) There exist constants $0 < \sigma_- < \sigma_+ < \infty$ and $\varphi \in \mathcal{P}(\mathsf{X})$ such that for all $p \in \mathbb{N}$, $x \in \mathsf{X}$, and $\mathsf{A} \in \mathcal{X}$,

$$\sigma_- \varphi(\mathsf{A}) \leq M_p(x, \mathsf{A}) \leq \sigma_+ \varphi(\mathsf{A}) .$$

(ii) It holds that $w_+ \stackrel{\text{def}}{=} \sup_{p\in\mathbb{N}} \|w_p\|_\infty < \infty$.

(iii) It holds that $c_- \stackrel{\text{def}}{=} \inf_{(p,x)\in\mathbb{N}\times\mathsf{X}} Q_p \mathbb{1}_{\mathsf{X}}(x) > 0$.

The assumption (**M**)(i), implying that each M_p allows the whole state space X as a 1-*small set*, is rather restrictive and requires typically the state space X to be a compact set. Still, it plays a vital role in the literature of SMC analysis [Del Moral and Guionnet, 2001; Del Moral, 2004; Cappé et al., 2005; Douc et al., 2009; Olsson et al., 2008; Douc et al., 2014b]. On the other hand, the weaker assumption (**M**)(ii) is satisfied for most applications and (**M**)(iii) does not require the potential functions to be uniformly bounded from below; the latter is a condition that appears frequently in the literature. Under (**M**), denote

$$\rho \stackrel{\text{def}}{=} 1 - \frac{\sigma_-}{\sigma_+} \; ; \tag{3.4.11}$$

then the previous assumptions allow the following explicit uniform in time bound to be derived.

Corollary 3.4.5. *Suppose (**M**). Then the sequence of asymptotic variances of the* B^2 *algorithm (see Corollary 3.4.3) satisfies, for all $n \in \mathbb{N}$ and $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$,*

$$\sigma_n^2(h) \leq w_+ \frac{\operatorname{osc}^2(h)}{(1-\rho)^2(1-\rho^2)^2 c_-} \; ,$$

where ρ is defined in (3.4.11).

The proof is found in subsection 3.6.10.

3.4.5 Uniform in time convergence under a local Doeblin condition

The explicitness and simplicity of the variance bound in Corollary 3.4.5 are obtained at the cost of restrictive model assumptions that are rarely satisfied in real-world applications. Thus, in this section we will discuss how the assumptions of (**M**) can be lightened considerably and turned into easily verifiable conditions, satisfied for many models of interest, by considering assumptions under which the asymptotic variance is *stochastically bounded* (tight) rather than bounded by a deterministic constant. Since the asymptotic variance (3.4.4) of the B^2 algorithm differs only from that of the SISR algorithm (see (3.4.5)) by the factors $n-\ell$, the results obtained in this section will rely heavily on similar results obtained in [Douc et al., 2014a] for the standard bootstrap particle filter. For this purpose, assume that each potential function depends on time through some random parameter only, i.e., for all $p \in \mathbb{N}^*$, $g_p = g\langle Z_p \rangle$, where $\{Z_p\}_{p\in\mathbb{N}}$ is some stochastic process taking values in some state space $(\mathsf{Z}, \mathcal{Z})$ and $g\langle z\rangle \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$ for all $z \in \mathsf{Z}$. Moreover, we assume that the normalized transition kernels of the model are time homogeneous, i.e., $M_p = M$ for all $p \in \mathbb{N}$, and that Mutation is based on the underlying dynamics of the model, i.e., $R_p = R = M$, and, consequently, $w_p(x,x') = w_p\langle Z_{p+1}\rangle(x,x') = g\langle Z_{p+1}\rangle(x')$ for all $p \in \mathbb{N}$ and $(x,x') \in \mathsf{X}^2$. Thus, in this case the model generates a parameter dependent Feynman-Kac flow $\{\eta_p\langle Z_{0:p}\rangle\}_{p\in\mathbb{N}}$. (For instance, in the case of a hidden Markov model, the sequence $\{Z_p\}_{p\in\mathbb{N}}$ plays the role of noisy observations of some Markov chain (the *state process*) $\{X_p\}_{p\in\mathbb{N}}$ with transition kernel M on $(\mathsf{X}, \mathcal{X})$. Conditionally on the state process, the observations are assumed to be independent and such that the conditional density $\mathsf{Z} \ni z \mapsto g\langle z\rangle(x)$ of each Z_p depends on the corresponding state $X_p = x \in \mathsf{X}$ only. In this important framework, $\eta_p\langle Z_{0:p}\rangle$ is the so-called *filter distribution* at time p, i.e., the conditional distribution of the latent state X_p given the observations $Z_{0:p}$). In this case, the asymptotic variance $\sigma_n^2(h)$ of the B^2 algorithm is a function of the random vector $Z_{0:n}$, and we write $\sigma_n^2\langle Z_{0:n}\rangle(h)$ to emphasize this fact. We will replace the condition (**M**)(i) by a considerably weaker condition of the following type.

Definition 3.4.6. *A set $\mathsf{C} \in \mathcal{X}$ is* local Doeblin *with respect to M if there is $\varphi_{\mathsf{C}} \in \mathcal{P}(\mathsf{X})$ with $\varphi_{\mathsf{C}}(\mathsf{C}) = 1$ and constants $0 < \sigma_{\mathsf{C}}^- < \sigma_{\mathsf{C}}^+$ such that for all $x \in \mathsf{C}$ and $\mathsf{A} \in \mathcal{X}$,*

$$\sigma_{\mathsf{C}}^- \varphi_{\mathsf{C}}(\mathsf{A}) \leq M(x, \mathsf{A}\cap\mathsf{C}) \leq \sigma_{\mathsf{C}}^+ \varphi_{\mathsf{C}}(\mathsf{A}) \; .$$

Now, impose the following assumption.

(L) The process $\{Z_p\}_{p\in\mathbb{N}}$ is strictly stationary and ergodic. Moreover, there exists a set $\mathsf{K} \in \mathcal{Z}$ such that the following holds.

(i) $\mathbb{P}(Z_0 \in \mathsf{K}) > 2/3$.

(ii) For all $\varepsilon > 0$ there exists a local Doeblin set C such that for all $z \in \mathsf{Z}$,

$$\sup_{x\in\mathsf{C}^c} g\langle z\rangle(x) \leq \varepsilon \left\|g\langle z\rangle\right\|_\infty < \infty \,.$$

(iii) There exists a set $\mathsf{D} \in \mathcal{X}$ such that $\inf_{x\in\mathsf{D}} M(x,\mathsf{D}) > 0$ and

$$\mathbb{E}\left[\ln^- \inf_{x\in\mathsf{D}} g\langle Z_0\rangle(x)\right] < \infty \,.$$

The condition (**L**) can be checked easily for a large variety of models; see [Douc et al., 2014a, Section 4] for examples.

Remark 7. The condition (**L**) can be weakened further by requiring the local Doeblin condition to hold only for some iterate $Q\langle z_1\rangle \cdots Q\langle z_r\rangle$, $z_{1:r} \in \mathsf{Z}^r$, with a minorizing measure φ_{C} and constants σ_{C}^-, σ_{C}^+ possibly depending on the block $z_{1:r}$; we refer to [Douc et al., 2014a] for details. In this chapter we have however chosen to state the most basic version of the condition (corresponding to $r = 1$) for simplicity.

Under (**L**), define $\mathcal{P}(\mathsf{X},\mathsf{D}) \subset \mathcal{P}(\mathsf{X})$ as the set of all $\chi \in \mathcal{P}(\mathsf{X})$ for which there exists $\mathsf{D}' \in \mathcal{X}$ such that (i) $\inf_{x\in\mathsf{D}'} M(x,\mathsf{D}) > 0$, (ii) $\mathbb{E}[\ln^- \inf_{x\in\mathsf{D}'} g\langle Z_0\rangle(x)] < \infty$, and (iii) $\chi(\mathsf{D}') > 0$. Then the following holds true.

Corollary 3.4.8. *Assume (**L**) and suppose in addition that $\eta_0 \in \mathcal{P}(\mathsf{X},\mathsf{D})$. Then for all $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$, the sequence $\{\sigma_n^2\langle Z_{0:n}\rangle(h)\}_{n\in\mathbb{N}}$ of asymptotic variances of the output of the B^2 algorithm is tight, i.e., it satisfies*

$$\lim_{\lambda\to\infty} \sup_{n\in\mathbb{N}} \mathbb{P}\left(\sigma_n^2\langle Z_{0:n}\rangle(h) \geq \lambda\right) = 0 \,.$$

The previous result is obtained by inspecting the proof of [Douc et al., 2014a, Theorem 11]; see subsection 3.6.11 for some details.

3.5 Technical results

We first recall two results, obtained in [Douc and Moulines, 2008], which are essential for the developments of the present chapter.

Theorem 3.5.1 ([Douc and Moulines, 2008]). *Let $(\Omega,\mathcal{A},\{\mathcal{F}_N\}_{N\in\mathbb{N}},\mathbb{P})$ be a filtered probability space. In addition, let, for a given sequence $\{M_N\}_{N\in\mathbb{N}}$ of integers such that $M_N \to \infty$ as $N \to \infty$, $\{U_N(i)\}_{i=1}^{M_N}$, $N \in \mathbb{N}$, be a triangular array of random variables on $(\Omega,\mathcal{A},\mathbb{P})$ such that for all $N \in \mathbb{N}$, the variables $\{U_N(i)\}_{i=1}^{M_N}$ are conditionally independent given $\mathcal{F}_N$ with $\mathbb{E}[|U_N(i)| \mid \mathcal{F}_N] < \infty$, $\mathbb{P}$-a.s., for all $i \in [\![1, M_N]\!]$. Moreover, assume that*

(A1) $$\lim_{\lambda\to\infty} \sup_{N\in\mathbb{N}} \mathbb{P}\left(\sum_{i=1}^{M_N} \mathbb{E}[|U_N(i)| \mid \mathcal{F}_N] \geq \lambda\right) = 0 \,.$$

(A2) *For all $\varepsilon > 0$, as $N \to \infty$,*

$$\sum_{i=1}^{M_N} \mathbb{E}\left[|U_N(i)| \mathbb{1}\{|U_N(i)| \geq \varepsilon\} \mid \mathcal{F}_N\right] \xrightarrow{\mathbb{P}} 0 \,.$$

Then, as $N \to \infty$,

$$\max_{m \in [\![1, M_N]\!]} \left| \sum_{i=1}^{m} U_N(i) - \sum_{i=1}^{m} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right] \right| \xrightarrow{\mathbb{P}} 0 \, .$$

Theorem 3.5.2 ([Douc and Moulines, 2008]). *Let the assumptions of Theorem 3.5.1 hold with* $\mathbb{E}[U_N^2(i) \mid \mathcal{F}_N] < \infty$, $\mathbb{P}$*-a.s., for all* $i \in [\![1, M_N]\!]$, *and* (**A1**) *and* (**A2**) *replaced by :*

(B1) *For some constant* $\varsigma^2 > 0$, *as* $N \to \infty$,

$$\sum_{i=1}^{M_N} \left(\mathbb{E}[U_N^2(i) \mid \mathcal{F}_N] - \mathbb{E}^2\left[U_N(i) \mid \mathcal{F}_N\right] \right) \xrightarrow{\mathbb{P}} \varsigma^2 \, .$$

(B2) *For all* $\varepsilon > 0$, *as* $N \to \infty$,

$$\sum_{i=1}^{M_N} \mathbb{E}\left[U_N^2(i) \mathbb{1}\{|U_N(i)| \geq \varepsilon\} \mid \mathcal{F}_N\right] \xrightarrow{\mathbb{P}} 0 \, .$$

Then, for all $u \in \mathbb{R}$, *as* $N \to \infty$,

$$\mathbb{E}\left[\exp\left(\mathrm{i}\, u \sum_{i=1}^{M_N} \left\{ U_N(i) - \mathbb{E}[U_N(i) \mid \mathcal{F}_N] \right\} \right) \mid \mathcal{F}_N \right] \xrightarrow{\mathbb{P}} \exp\left(-u^2 \varsigma^2 / 2\right) \, .$$

The following lemma is useful when verifying the tightness conditions (**A2**) and (**B2**).

Lemma 3.5.3. *Let the* $(\{M_N\}_{N \in \mathbb{N}}, \{U_N(i)\}_{i=1}^{M_N}, \{\mathcal{F}_N\}_{N \in \mathbb{N}})$ *be the triangular array given in Theorem 3.5.1. Assume that there exist sequences* $\{V_N\}_{N \in \mathbb{N}}$, $\{X_N\}_{N \in \mathbb{N}}$, *and* $\{Y_N\}_{N \in \mathbb{N}}$ *of nonnegative random variables such that*

(i) for all $N \in \mathbb{N}$, $\mathbb{P}$*-a.s.,*

$$\max_{i \in [\![1, M_N]\!]} |U_N(i)| \leq V_N + X_N Y_N^2 \, ,$$

(ii) $\{V_N\}_{N \in \mathbb{N}}$ *and* $\{X_N\}_{N \in \mathbb{N}}$ *are* $\{\mathcal{F}_N\}_{N \in \mathbb{N}}$*-adapted and such that* $V_N \xrightarrow{\mathbb{P}} 0$ *as* $N \to \infty$ *and*

$$\lim_{\lambda \to \infty} \sup_{N \in \mathbb{N}} \mathbb{P}\left(X_N \geq \lambda\right) = 0 \, ,$$

(iii) for some $\alpha \in \{1, 2\}$, $\nu > 0$, $c > 0$, *and* $C > 0$, $\mathbb{P}$*-a.s.,*

$$\mathbb{P}\left(Y_N \geq y \mid \mathcal{F}_N\right) \leq C M_N \exp\left(-c M_N^{\nu} y^{2\alpha}\right) \, . \tag{3.5.1}$$

Then for $p \in \{1, 2\}$,

$$\lim_{\lambda \to \infty} \sup_{N \in \mathbb{N}} \mathbb{P}\left(\sum_{i=1}^{M_N} \mathbb{E}\left[|U_N^p(i)| \mid \mathcal{F}_N\right] \geq \lambda \right) = 0$$

$$\Rightarrow \sum_{i=1}^{M_N} \mathbb{E}\left[|U_N^p(i)| \mathbb{1}\{|U_N(i)| \geq \varepsilon\} \mid \mathcal{F}_N\right] \xrightarrow{\mathbb{P}} 0 \, , \quad \forall \varepsilon > 0, \text{ as } N \to \infty.$$

Démonstration. We start with the case $p = 1$. First, note that according to Proposition 4.3.2, for all $\upsilon > 0$,

$$\begin{aligned} \mathbb{E}\left[Y_N^2 \mathbb{1}\{Y_N^2 \geq \upsilon\} \mid \mathcal{F}_N\right] &= \int_{\upsilon}^{\infty} \mathbb{P}\left(Y_N \geq \sqrt{y} \mid \mathcal{F}_N\right) \mathrm{d}y + \upsilon \mathbb{P}\left(Y_N \geq \sqrt{\upsilon} \mid \mathcal{F}_N\right) \\ &\leq C M_N \int_{\upsilon}^{\infty} \exp\left(-c M_N^{\nu} y^{\alpha}\right) \mathrm{d}y + C M_N \upsilon \exp\left(-c M_N^{\nu} \upsilon^{\alpha}\right) \, , \end{aligned}$$

where we used the condition (iii) in the second step. Thus,

$$\mathbb{E}\left[Y_N^2 \mathbb{1}\{Y_N^2 \geq \upsilon\} \mid \mathcal{F}_N\right] \leq \begin{cases} \left(\upsilon + M_N^{-\nu}/c\right) C M_N \exp\left(-c M_N^{\nu} \upsilon\right) & \text{for } \alpha = 1 \,, \\ \left(\upsilon + M_N^{-\nu}/(2c\upsilon)\right) C M_N \exp\left(-c M_N^{\nu} \upsilon^2\right) & \text{for } \alpha = 2 \,, \end{cases} \tag{3.5.2}$$

using the standard upper tail bound for Gaussian distributions. In any case,

$$M_N \mathbb{E}\left[Y_N^2 \mathbb{1}\{Y_N^2 \geq \upsilon\} \mid \mathcal{F}_N\right] \xrightarrow[N \to +\infty]{\mathbb{P}} 0 \,. \tag{3.5.3}$$

In addition, note that (ii) implies, for all $\varepsilon' > 0$ and all $\delta > 0$, the existence of a constant $\lambda_\delta < \infty$ such that for all $\lambda \geq \lambda_\delta$,

$$\sup_{N \in \mathbb{N}} \mathbb{P}\left(\mathbb{1}\{X_N \geq \lambda\} \sum_{i=1}^{M_N} \mathbb{E}\left[|U_N(i)| \mid \mathcal{F}_N\right] \geq \varepsilon' \right) \leq \sup_{N \in \mathbb{N}} \mathbb{P}\left(X_N \geq \lambda\right) \leq \delta \,. \tag{3.5.4}$$

Now, for any $\varepsilon > 0$ and $\lambda > 0$ the quantity of interest may be bounded as

$$\begin{aligned} \sum_{i=1}^{M_N} \mathbb{E}\left[|U_N(i)| \mathbb{1}\{|U_N(i)| \geq \varepsilon\} \mid \mathcal{F}_N\right] &\leq M_N V_N \mathbb{P}\left(Y_N \geq \sqrt{\frac{\varepsilon}{2\lambda}} \mid \mathcal{F}_N\right) \\ &\quad + \left(\mathbb{1}\{X_N \geq \lambda\} + \mathbb{1}\{V_N \geq \varepsilon/2\}\right) \sum_{i=1}^{M_N} \mathbb{E}\left[|U_N(i)| \mid \mathcal{F}_N\right] + \lambda M_N \mathbb{E}\left[Y_N^2 \mathbb{1}\{Y_N^2 \geq \varepsilon/(2\lambda)\} \mid \mathcal{F}_N\right] \,, \end{aligned}$$

where the upper bound may, by (iii), (3.5.3), and (3.5.4), be made arbitrarily small in probability by increasing first λ and then N. This completes the proof in the case $p = 1$.

We turn to the case $p = 2$. However, by letting $\tilde{U}_N(i) \overset{\text{def}}{=} U_N^2(i)$, $i \in [\![1, M_N]\!]$, $N \in \mathbb{N}^*$, and noting that $\max_{i \in [\![1, M_N]\!]} \tilde{U}_N(i) \leq \tilde{V}_N + \tilde{X}_N \tilde{Y}_N^2$, where $\tilde{V}_N \overset{\text{def}}{=} 2V_N^2$, $\tilde{X}_N \overset{\text{def}}{=} 2X_N^2$, and $\tilde{Y}_N \overset{\text{def}}{=} Y_N^2$, we thus realise that the proof of the case $p = 1$ goes through if we can verify that (3.5.3) and (3.5.4) hold true when X_N, Y_N, and $\{U_N(i)\}_{i=1}^{M_N}$ are replaced by $\tilde{X}_N$, $\tilde{Y}_N$, and $\{\tilde{U}_N(i)\}_{i=1}^{M_N}$, respectively. Nevertheless, (3.5.4) holds straightforwardly as tightness of $\{X_N\}_{N \in \mathbb{N}}$ implies tightness of $\{\tilde{X}_N\}_{N \in \mathbb{N}}$. Moreover, using condition (iii) one shows, along previous lines, that

$$\begin{aligned} \mathbb{E}\left[\tilde{Y}_N^2 \mathbb{1}\{\tilde{Y}_N^2 \geq \upsilon\} \mid \mathcal{F}_N\right] &= \int_{\upsilon}^{\infty} \mathbb{P}\left(Y_N \geq \sqrt[4]{y} \mid \mathcal{F}_N\right) \mathrm{d}y + \upsilon \mathbb{P}\left(Y_N \geq \sqrt[4]{\upsilon} \mid \mathcal{F}_N\right) \\ &\leq C M_N \int_{\upsilon}^{\infty} \exp\left(-c M_N^{\nu} y^{\alpha/2}\right) \mathrm{d}y + \upsilon C M_N \exp\left(-c M_N^{\nu} \upsilon^{\alpha/2}\right) \,. \end{aligned}$$

For $\alpha = 1$,

$$\mathbb{E}\left[\tilde{Y}_N^2 \mathbb{1}\{\tilde{Y}_N^2 \geq \upsilon\} \mid \mathcal{F}_N\right] \leq \left(2\sqrt{\upsilon} M_N^{-\nu}/c + 2M_N^{-2\nu}/c^2 + \upsilon\right) C M_N \exp\left(-c M_N^{\nu} \sqrt{\upsilon}\right) \,.$$

while the case $\alpha = 2$ corresponds to the first case of (3.5.2). Consequently, as $N \to \infty$,

$$M_N \mathbb{E}\left[\tilde{Y}_N^2 \mathbb{1}\{\tilde{Y}_N^2 \geq \upsilon\} \mid \mathcal{F}_N\right] \xrightarrow{\mathbb{P}} 0 \,,$$

which completes the proof. □

Lemma 3.5.4. *Let $a \in \mathbb{R}$ be nonzero, $a \neq 0$, and let $\{X_N(i)\}_{i=1}^{N_1}$, $N_1(N) \in \mathbb{N}^*$, be random variables such that $X_N(i) \neq 0$ for all $i \in [\![1, N_1]\!]$. Assume that $\max_{i \in [\![1, N_1]\!]} |X_N(i) - a| \xrightarrow{\mathbb{P}} 0$ as $N \to \infty$. Then*

$$\max_{i \in [\![1, N_1]\!]} \left|X_N^{-1}(i) - a^{-1}\right| \xrightarrow{\mathbb{P}} 0.$$

Démonstration. Pick $\varepsilon > 0$; then we may write for all $\eta > 0$,

$$\mathbb{P}\left(\max_{i\in[\![1,N_1]\!]}\left|X_N^{-1}(i)-a^{-1}\right|\geq\varepsilon\right)\leq\mathbb{P}\left(\max_{i\in[\![1,N_1]\!]}|X_N(i)-a|\geq\eta\right)$$
$$+\mathbb{P}\left(\max_{i\in[\![1,N_1]\!]}\left|X_N^{-1}(i)-a^{-1}\right|\geq\varepsilon,\max_{i\in[\![1,N_1]\!]}|X_N(i)-a|<\eta\right),$$

where the first term tends to zero as N tends to infinity for any η by assumption. For all $i \in [\![1, N_1]\!]$, there exists, by Taylor's formula, $\zeta_N(i) \in (X_N(i) \wedge a, X_N(i) \vee a)$ such that $|X_N^{-1}(i) - a^{-1}| = \zeta_N^{-2}(i)|X_N(i) - a|$. Thus, if $a > 0$ and $0 < \eta < a$,

$$\mathbb{P}\left(\max_{i\in[\![1,N_1]\!]}\left|X_N^{-1}(i)-a^{-1}\right|\geq\varepsilon,\max_{i\in[\![1,N_1]\!]}|X_N(i)-a|<\eta\right)\leq\mathbb{P}\left(\max_{i\in[\![1,N_1]\!]}|X_N(i)-a|>\varepsilon\{a-\eta\}^2\right),$$

where the right hand side tends, by assumption, to zero as N tends to infinity. On the other hand, if $a < 0$ and $0 < \eta < -a$,

$$\mathbb{P}\left(\max_{i\in[\![1,N_1]\!]}\left|X_N^{-1}(i)-a^{-1}\right|\geq\varepsilon,\max_{i\in[\![1,N_1]\!]}|X_N(i)-a|<\eta\right)\leq\mathbb{P}\left(\max_{i\in[\![1,N_1]\!]}|X_N(i)-a|>\varepsilon\{a+\eta\}^2\right),$$

where again the right hand side tends to zero. This concludes the proof. □

Lemma 3.5.5. *Let $\{Z_N\}_{N\in\mathbb{N}}$ be a sequence of random variables such that for some constant $z_\infty \in \mathbb{R}$, as $N \to \infty$,*

$$\mathbb{E}[Z_N] \to z_\infty$$

and is uniformly bounded by some constant $z^+ \in \mathbb{R}$. Let $\{X_N\}$ be a sequence of random variables that converges in probability to some constant $x_\infty \in \mathbb{R}$ and is dominated by some integrate random variable. Then, as $N \to \infty$,

$$\mathbb{E}[X_N Z_N] \to x_\infty z_\infty .$$

Démonstration. The result is obtained straightforwardly by writing

$$|\mathbb{E}[X_N Z_N] - x_\infty z_\infty| \leq z^+ \mathbb{E}[|X_N - x_\infty|] + |x_\infty|\,|\mathbb{E}[Z_N] - z_\infty| ,$$

where the right hand side tends to zero as N tends to infinity by assumption and dominated convergence. □

3.6 Proofs

In this section we will use repeatedly the same notation $\{U_N(i)\}_{i=1}^{N_1}$ to denote triangular arrays (in the sense of Theorem 3.5.1 and Theorem 3.5.2), even though the roles of these arrays change throughout the proofs.

3.6.1 Proof of Theorem 3.3.1

We apply Theorem 3.5.1. For this purpose, define the triangular array and filtration

$$\begin{aligned} U_N(i) &\stackrel{\text{def}}{=} \sum_{j=1}^{N_2} \frac{\omega_N(I_N(i),j)}{N_1 \sum_{j'=1}^{N_2} \omega_N(I_N(i),j')} h(\xi_N(I_N(i),j)) \quad (i \in [\![1, N_1(N)]\!], N \in \mathbb{N}^*) , \\ \mathcal{F}_N &\stackrel{\text{def}}{=} \sigma\left(\{(\Omega_N(i), \{(\xi_N(i,j), \omega_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}\right) \quad (N \in \mathbb{N}^*) , \end{aligned} \tag{3.6.1}$$

respectively. Now, since the island indices $\{I_N(i)\}_{i=1}^{N_1}$ are, conditionally on $\mathcal{F}_N$, i.i.d. with common distribution $\mathsf{Mult}(\{\Omega_N(i')\}_{i'=1}^{N_1})$ it holds, as the ancestor sample is assumed to be consistent,

$$\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right] = N_1 \mathbb{E}\left[U_N(1) \mid \mathcal{F}_N\right] = \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\xi_N(i,j)) \xrightarrow{\mathbb{P}} \eta h \,. \quad (3.6.2)$$

Thus, since $|U_N(i)| \leq \|h\|_\infty / N_1 < \infty$ for all $i \in [\![1, N_1]\!]$, it is enough to check the conditions (**A1**) and (**A2**) in Theorem 3.5.1. The tightness condition (**A1**) is straightforwardly satisfied as sequences that converge in probability are tight. Moreover, to check (**A2**) we may apply Lemma 3.5.3 with $V_N = \|h\|_\infty / N_1$ and $Y_N = X_N = 0$. Thus, the limits, in probability as $N \to \infty$, of the series $\sum_{i=1}^{N_1} U_N(i)$ and $\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right]$ coincide, which completes the proof.

3.6.2 Proof of Theorem 3.3.2

Trivially, since $\{I_N(i)\}_{i=1}^{N_1} \subset [\![1, N_1]\!]$ it holds for all $\varepsilon > 0$,

$$\mathbb{P}\left(\max_{i \in [\![1,N_1]\!]} \left| \frac{1}{N_2} \sum_{j=1}^{N_2} \omega_N(I_N(i), j) h(\xi_N(I_N(i), j)) - \varrho \times \eta h \right| \geq \varepsilon \right)$$
$$\leq \mathbb{P}\left(\max_{i \in [\![1,N_1]\!]} \left| \frac{1}{N_2} \sum_{j=1}^{N_2} \omega_N(i, j) h(\xi_N(i, j)) - \varrho \times \eta h \right| \geq \varepsilon \right) \,,$$

where the right hand side has an exponential bound by assumption. This completes the proof.

3.6.3 Proof of Theorem 3.3.3

In order to check (**AN1**) using Theorem 3.5.2, define the array

$$U_N(i) \stackrel{\text{def}}{=} \sqrt{\frac{N_2}{N_1}} \sum_{j=1}^{N_2} \frac{\omega_N(I_N(i), j)}{\sum_{j'=1}^{N_2} \omega_N(I_N(i), j')} \{h(\xi_N(I_N(i), j)) - \eta h\} \quad (i \in [\![1, N_1]\!], N \in \mathbb{N}^*) \,, \quad (3.6.3)$$

and let $\{\mathcal{F}_N\}_{N \in \mathbb{N}^*}$ be the filtration (3.6.1). We first note that $\mathbb{E}[U_N^2(i) \mid \mathcal{F}_N] \leq 4 N_2 \|h\|_\infty^2 / N_1 < \infty$ for all $i \in [\![1, N_1]\!]$. Along the lines of (3.6.2),

$$\sum_{i=1}^{N_1} U_N(i) = \sum_{i=1}^{N_1} \{U_N(i) - \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right]\}$$
$$+ \sqrt{N} \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{h(\xi_N(i,j)) - \eta h\} \,. \quad (3.6.4)$$

By assumption, the second term on the right hand side of (3.6.4) converges in distribution to a Gaussian random variable with zero mean and variance $\sigma^2(h)$. To treat the first term using Theorem 3.5.2 we first consider

$$\sum_{i=1}^{N_1} \mathbb{E}\left[U_N^2(i) \mid \mathcal{F}_N\right] = N_2 \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \left(\sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{h(\xi_N(i,j)) - \eta h\} \right)^2 \xrightarrow{\mathbb{P}} \nu^2(h) \,, \quad (3.6.5)$$

where the limit holds as the ancestor archipelago is assumed to satisfy (**AN2**). In addition,

$$\sum_{i=1}^{N_1} \mathbb{E}^2 \left[U_N(i) \mid \mathcal{F}_N \right] = \frac{1}{N_1} \left(\sqrt{N} \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{ h(\xi_N(i,j)) - \eta h \} \right)^2 \xrightarrow{\mathbb{P}} 0 \,, \quad (3.6.6)$$

as the right hand side tends, as the ancestor archipelago satisfies (**AN1**), in distribution to a scaled χ^2-distributed random variable as $N \to \infty$. Combining the two previous displays shows that the condition (**B1**) in Theorem 3.5.2 holds with limit $\varsigma^2(h) = \nu^2(h)$. To check the condition (**B2**) in the same lemma, we note that $\max_{i \in [\![1,N_1]\!]} |U_N(i)| \leq V_N + X_N Y_N^2$, with $X_N = Y_N = 0$ and

$$V_N = \max_{i \in [\![1,N_1]\!]} \sqrt{\frac{N_2}{N_1}} \left| \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{ h(\xi_N(i,j)) - \eta h \} \right| \quad (N \in \mathbb{N}^*) \,. \quad (3.6.7)$$

Note that the sequence $\{V_N\}_{N \in \mathbb{N}^*}$ is $\mathcal{F}_N$-adapted and vanishes in probability as $N \to \infty$ since the ancestor archipelago is assumed to satisfy (**D**) and thus Equation 3.2.1. We may then apply Lemma 3.5.3 to check the condition (**B2**) in Theorem 3.5.2, implying that for all $u \in \mathbb{R}$,

$$\mathbb{E} \left[\exp \left(\mathrm{i}\, u \sum_{i=1}^{N_1} \{ U_N(i) - \mathbb{E} \left[U_N(i) \mid \mathcal{F}_N \right] \} \right) \mid \mathcal{F}_N \right] \xrightarrow{\mathbb{P}} \exp(-u^2 \nu^2(h)/2) \,.$$

Now, using this limit, the decomposition (3.6.4), and hypothesis that the ancestor archipelago satisfies (**AN1**), we conclude, via Lemma 3.5.5, that for all $u \in \mathbb{R}$,

$$\mathbb{E} \left[\exp \left(\mathrm{i}\, u \sum_{i=1}^{N_1} U_N(i) \right) \right] \longrightarrow \exp(-u^2 \{ \sigma^2(h) + \nu^2(h) \}/2) \,,$$

which concludes the proof of (**AN1**).

To check that the output of Algorithm 7 satisfies also (**AN2–5**) we proceed along similar lines. The assumption 1 is used there. We refer the reader to subsection 4.1.1 for all details.

3.6.4 Proof of Theorem 3.3.4

We first note that (**C2**) is trivially satisfied. In order to check (**C1**) we apply Theorem 3.5.1 to the array

$$U_N(i) \stackrel{\text{def}}{=} \frac{\Omega_N(i)}{N_2 \sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} h(\xi_N(i, J_N(i,j))) \quad (i \in [\![1,N_1]\!], N \in \mathbb{N}^*)$$

associated with $\{\mathcal{F}_N\}_{N \in \mathbb{N}}$ given by (3.6.1). Note that all indices $\{J_N(i,j) \in [\![1,N_1]\!] \times [\![1,N_2]\!]\}$ are conditionally independent given $\mathcal{F}_N$. Moreover, for all $i \in [\![1,N_1]\!]$ it holds that $\{J_N(i,j)\}_{j=1}^{N_2} \sim \mathsf{Mult}(\{\omega_N(i,j')\}_{j'=1}^{N_2})^{\otimes N_2}$. Hence,

$$\sum_{i=1}^{N_1} \mathbb{E} \left[U_N(i) \mid \mathcal{F}_N \right] = \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\xi_N(i,j)) \xrightarrow{\mathbb{P}} \eta h \,,$$

where convergence holds by assumption. First, note that $U_N(i) \leq \|h\|_\infty < \infty$ for all $i \in [\![1,N_1]\!]$ and $N \in \mathbb{N}^*$. Moreover, (**A1**) is trivially satisfied. Thus, consistency is established by showing that (**A2**) is satisfied, which is an immediate implication of Lemma 3.5.3 with $X_N = Y_N = 0$ and $V_N = \|h\|_\infty \max_{i \in [\![1,N_1]\!]} \Omega_N(i) / \sum_{i'=1}^{N_1} \Omega_N(i')$, which is $\mathcal{F}_N$-adapted and tends to zero in probability thanks to (**C2**). Hence, by Theorem 3.5.1, the series $\sum_{i=1}^{N_1} U_N(i)$ and $\sum_{i=1}^{N_1} \mathbb{E} \left[U_N(i) \mid \mathcal{F}_N \right]$ have the same limit ηh in probability. This completes the proof.

3.6.5 Proof of Theorem 3.3.5

We may bound the quantity of interest according to

$$\max_{i\in[\![1,N_1]\!]}\left|\frac{1}{N_2}\sum_{j=1}^{N_2}h(\xi_N(i,J_N(i,j)))-\eta h\right| \\ \leq \max_{i\in[\![1,N_1]\!]}\frac{1}{N_2}\left|\sum_{j=1}^{N_2}\delta_N(i,j)\right| + \max_{i\in[\![1,N_1]\!]}\left|\sum_{j=1}^{N_2}\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}h(\xi_N(i,j))-\eta h\right| ,$$

where we have set

$$\delta_N(i,j)\stackrel{\text{def}}{=} h(\xi_N(i,J_N(i,j)))-\sum_{j'=1}^{N_2}\frac{\omega_N(i,j')}{\sum_{j''=1}^{N_2}\omega_N(i,j'')}h(\xi_N(i,j')) \quad ((i,j)\in[\![1,N_1]\!]\times[\![1,N_2]\!]) . \tag{3.6.8}$$

By Lemma 3.2.3, the second term on the right hand side satisfies

$$\mathbb{P}\left(\max_{i\in[\![1,N_1]\!]}\left|\sum_{j=1}^{N_2}\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}h(\xi_N(i,j))-\eta h\right|\geq\varepsilon/2\right)\leq 2c_1N_1\exp\left(-c_2N_2\frac{\varepsilon^2\varrho^2}{16\left\|h\right\|_\infty^2}\right) .$$

For each $i\in[\![1,N_1]\!]$, the variables $\{\delta_N(i,j)\}_{j=1}^{N_2}$ are, conditionally on $\mathcal{F}_N$, independent and identically distributed with zero mean; moreover, as $|\delta_N(i,j)|\leq 2\left\|h\right\|_\infty$ for all $(i,j)\in[\![1,N_1]\!]\times[\![1,N_2]\!]$, Hoeffding's inequality implies that for all $\varepsilon>0$,

$$\mathbb{P}\left(\max_{i\in[\![1,N_1]\!]}\frac{1}{N_2}\left|\sum_{j=1}^{N_2}\delta_N(i,j)\right|\geq\varepsilon/2\mid\mathcal{F}_N\right)\leq 2N_1\exp\left(-N_2\frac{\varepsilon^2}{8\left\|h\right\|_\infty^2}\right) . \tag{3.6.9}$$

Combining the previous two displays show that (**D**) is satisfied with the choice of $\tilde{c}_1$ and $\tilde{c}_2$ given in the theorem.

3.6.6 Proof of Theorem 3.3.6

We start with (**AN1**). In order to apply Theorem 3.5.2, define the array

$$U_N(i)\stackrel{\text{def}}{=}\sqrt{\frac{N_1}{N_2}}\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\sum_{j=1}^{N_2}\{h(\xi_N(i,J_N(i,j)))-\eta h\} \quad ((i,j)\in[\![1,N_1]\!]\times[\![1,N_2]\!]) , \tag{3.6.10}$$

equipped with the usual filtration $\{\mathcal{F}_N\}_{N\in\mathbb{N}}$ given by Equation 3.6.1. We first note that $U_N(i)\leq 2\sqrt{N}\left\|h\right\|_\infty<\infty$ for all $i\in[\![1,N_1]\!]$ and $N\in\mathbb{N}^*$. In order to check (**B1**), write, following the arguments of the proof of Theorem 3.3.4,

$$\sum_{i=1}^{N_1}\mathbb{E}\left[U_N^2(i)\mid\mathcal{F}_N\right]=N_1\sum_{i=1}^{N_1}\left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\right)^2\sum_{j=1}^{N_2}\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\{h(\xi_N(i,j))-\eta h\}^2 \\ +N_1(N_2-1)\sum_{i=1}^{N_1}\left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\right)^2\left(\sum_{j=1}^{N_2}\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\{h(\xi_N(i,j))-\eta h\}\right)^2 .$$

Moreover, since

$$\sum_{i=1}^{N_1} \mathbb{E}^2\left[U_N(i) \mid \mathcal{F}_N\right] = N \sum_{i=1}^{N_1} \left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \right)^2 \left(\sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{h(\xi_N(i,j)) - \eta h\} \right)^2 ,$$

we obtain

$$\begin{aligned} &\sum_{i=1}^{N_1} \left\{ \mathbb{E}\left[U_N^2(i) \mid \mathcal{F}_N\right] - \mathbb{E}^2\left[U_N(i) \mid \mathcal{F}_N\right] \right\} \\ &= N_1 \sum_{i=1}^{N_1} \left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \right)^2 \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{h(\xi_N(i,j)) - \eta h\}^2 \\ &- N_1 \sum_{i=1}^{N_1} \left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \right)^2 \left(\sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{h(\xi_N(i,j)) - \eta h\} \right)^2 . \end{aligned} \tag{3.6.11}$$

Since the ancestor archipelago satisfies (3.2.1) and is consistent for η, we deduce that

$$N_1 \sum_{i=1}^{N_1} \left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \right)^2 \left(\sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{h(\xi_N(i,j)) - \eta h\} \right)^2 \xrightarrow[N \to +\infty]{\mathbb{P}} 0 .$$

Then, since the ancestor archipelago also satisfies (**AN3**) we conclude that the variance (3.6.11) tends in probability to $\mu_1\{(h-\eta h)^2\}$. Consequently, the triangular array satisfies Assumption (**B1**) with limit $\mu_1\{(h-\eta h)^2\}$. In order to check Assumption (**B2**) we may apply Lemma 3.5.3 by bounding

$$\max_{i \in [\![1, N_1]\!]} |U_N(i)| \leq V_N + X_N Y_N^2 \quad (N \in \mathbb{N}^*) , \tag{3.6.12}$$

with, for $N \in \mathbb{N}^*$,

$$\begin{cases} V_N = \sqrt{N} \max\limits_{i \in [\![1, N_1]\!]} \dfrac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \max\limits_{i \in [\![1, N_1]\!]} \left| \sum\limits_{j=1}^{N_2} \dfrac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{h(\xi_N(i,j)) - \eta h\} \right| , \\ X_N = N_1 \max\limits_{i \in [\![1, N_1]\!]} \dfrac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} , \\ Y_N^2 = \sqrt{\dfrac{N_2}{N_1}} \max\limits_{i \in [\![1, N_1]\!]} \left| \dfrac{1}{N_2} \sum\limits_{j=1}^{N_2} \delta_N(i,j) \right| , \end{cases}$$

where the δ_Ns are defined in (3.6.8). Here $\{V_N\}_{N \in \mathbb{N}^*}$ is $\mathcal{F}_N$-adapted and tends to zero in probability by (**AN6**) and Lemma 3.2.3. In addition, $\{X_N\}_{N \in \mathbb{N}^*}$ is $\mathcal{F}_N$-adapted and, by (**AN6**), tight. Moreover, for all $N \in \mathbb{N}^*$, Y_N has, by (3.6.9), a tail of the type

$$\mathbb{P}(Y_N \geq \varepsilon \mid \mathcal{F}_N) \leq 2 N_1 \exp\left(-N_1 \frac{\varepsilon^4}{2 \|h\|_\infty^2} \right) .$$

Thus, Lemma 3.5.3 applies, which establishes (**B2**). Finally, we may conclude, by using Lemma 3.5.5, the proof of (**AN1**) to obtain that $\tilde{\sigma}^2(h) = \sigma^2(h) + \mu_1\{(h-\eta h)^2\}$.

The conditions (**AN2**–**6**) are checked using similar techniques. We refer the reader to subsection 4.1.2 for all details.

3.6.7 Proof of Theorem 3.3.7

First, note that

$$\sum_{i=1}^{N_1} \frac{\tilde{\Omega}_N(i)}{\sum_{i'=1}^{N_1} \tilde{\Omega}_N(i')} \sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \tilde{\omega}_N(i,j')} h(\tilde{\xi}_N(i,j)) \\ = \frac{\sum_{i'=1}^{N_1} \Omega_N(i')}{\sum_{i''=1}^{N_1} \tilde{\Omega}_N(i'')} \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\tilde{\xi}_N(i,j)) \;, \quad (3.6.13)$$

using the definition of $\{\tilde{\Omega}_N(i)\}_{i=1}^{N_1}$ in Algorithm 9. In order to determine the limit in probability of this quantity we apply Theorem 3.5.1 to the array

$$U_N(i) \stackrel{\text{def}}{=} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\tilde{\xi}_N(i,j)) \quad (i \in [\![1, N_1]\!], N \in \mathbb{N}^*)$$

associated with the filtration $\{\mathcal{F}_N\}_{N\in\mathbb{N}^*}$ given in (3.6.1). For each $(i,j) \in [\![1,N_1]\!] \times [\![1,N_2]\!]$, the conditional distribution of $\tilde{\xi}_N(i,j)$ given $\mathcal{F}_N$ is $R(\xi_N(i,j),\cdot)$; thus,

$$\begin{aligned} \mathbb{E}\left[\tilde{\omega}_N(i,j) h(\tilde{\xi}_N(i,j)) \mid \mathcal{F}_N\right] &= \omega_N(i,j) \int w(\xi_N(i,j),\tilde{x}) h(\tilde{x})\, R(\xi_N(i,j), \mathrm{d}\tilde{x}) \\ &= \omega_N(i,j) Q h(\xi_N(i,j)) \;, \end{aligned} \quad (3.6.14)$$

implying that

$$\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right] = \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} Q h(\xi_N(i,j)) \xrightarrow{\mathbb{P}} \eta Q h \;, \quad (3.6.15)$$

where convergence holds since the ancestor archipelago satisfies Assumption (**C1**). This implies (**A1**). To check also the condition (**A2**) we apply Lemma 3.5.3 with $X_N = Y_N = 0$ and $V_N = \|w\|_\infty \|h\|_\infty \max_{i\in[\![1,N_1]\!]} \Omega_N(i) / \sum_{i'=1}^{N_1} \Omega_N(i')$, where V_N is $\mathcal{F}_N$-adapted and tends to zero in probability by the assumption (**C2**). Hence, Theorem 3.5.1 ensures that the two series $\sum_{i=1}^{N_1} U_N(i)$ and $\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right]$ have the same limit $\eta Q h$ in probability. Moreover, by setting h is equal to the constant function $\mathbb{1}_{\tilde{\mathsf{X}}}$ we deduce that

$$\frac{\sum_{i=1}^{N_1} \Omega_N(i)}{\sum_{i'=1}^{N_1} \tilde{\Omega}_N(i')} \xrightarrow{\mathbb{P}} \frac{1}{\eta Q \mathbb{1}_{\tilde{\mathsf{X}}}} \;, \quad (3.6.16)$$

which allows us to complete the proof of (**C1**) using Slutsky's lemma.

Finally, Assumption (**C2**) is checked straightforwardly by just noting that

$$\max_{i\in[\![1,N_1]\!]} \frac{\tilde{\Omega}_N(i)}{\sum_{i'=1}^{N_1} \tilde{\Omega}_N(i')} \leq \|w\|_\infty \max_{i\in[\![1,N_1]\!]} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \frac{\sum_{i'=1}^{N_1} \Omega_N(i')}{\sum_{i''=1}^{N_1} \tilde{\Omega}_N(i'')} \;,$$

where the right hand side tends to zero in probability by (3.6.16) and the fact that the ancestor archipelago satisfies (**C1**).

3.6.8 Proof of Theorem 3.3.8

Note that $\tilde{\varrho} \times \tilde{\eta} h = \varrho \times \eta Q h$ and bound the quantity of interest according to

$$\max_{i \in [\![1, N_1]\!]} \left| \frac{1}{N_2} \sum_{j=1}^{N_2} \tilde{\omega}_N(i,j) h(\tilde{\xi}_N(i,j)) - \varrho \times \eta Q h \right| \leq \max_{i \in [\![1, N_1]\!]} \frac{1}{N_2} \left| \sum_{j=1}^{N_2} \tilde{\delta}_N(i,j) \right| + \max_{i \in [\![1, N_1]\!]} \frac{1}{N_2} \left| \sum_{j=1}^{N_2} \omega_N(i,j) Q h(\xi_N(i,j)) - \varrho \times \eta Q h \right| , \quad (3.6.17)$$

where

$$\tilde{\delta}_N(i,j) \stackrel{\text{def}}{=} \tilde{\omega}_N(i,j) h(\tilde{\xi}_N(i,j)) - \omega_N(i,j) Q h(\xi_N(i,j)) \quad ((i,j) \in [\![1, N_1]\!] \times [\![1, N_2]\!]) . \quad (3.6.18)$$

Since the input archipelago satisfies (**D**) it holds that

$$\mathbb{P}\left(\max_{i \in [\![1, N_1]\!]} \frac{1}{N_2} \left| \sum_{j=1}^{N_2} \{ \omega_N(i,j) Q h(\xi_N(i,j)) - \varrho \times \eta Q h \} \right| \geq \varepsilon/2 \right) \leq N_1 c_1 \exp\left(-c_2 N_2 \frac{\varepsilon^2}{4 \left\| Q \mathbb{1}_{\mathsf{X}} \right\|_\infty^2 \left\| h \right\|_\infty^2} \right) .$$

For each $i \in [\![1, N_1]\!]$, the random variables $\{\tilde{\delta}_N(i,j)\}_{j=1}^{N_2}$ are, conditionally on $\mathcal{F}_N$, independent and, by (3.6.14), zero mean. Moreover, since for all $(i,j) \in [\![1, N_1]\!] \times [\![1, N_2]\!]$, $|\tilde{\delta}_N(i,j)| \leq \delta \left\| h \right\|_\infty$, where δ is defined in the statement of theorem, Hoeffding's inequality implies that for all $\varepsilon > 0$,

$$\mathbb{P}\left(\max_{i \in [\![1, N_1]\!]} \frac{1}{N_2} \left| \sum_{j=1}^{N_2} \tilde{\delta}_N(i,j) \right| \geq \varepsilon/2 \mid \mathcal{F}_N \right) \leq 2 N_1 \exp\left(-N_2 \frac{\varepsilon^2}{2\delta^2 \left\| h \right\|_\infty^2} \right) . \quad (3.6.19)$$

By combining the two previous displays we may conclude that (**D**) is satisfied with $\tilde{c}_1$ and $\tilde{c}_2$ defined as in the theorem statement.

3.6.9 Proof of Theorem 3.3.9

We preface the proof by the following auxiliary result, which is obtained as a straightforward extension of the generalized hoeffding inequality in [Douc et al., 2011, Lemma 4].

Lemma 3.6.1. *Let the assumptions of Theorem 3.3.8 hold. Then for all $N_1 \in \mathbb{N}^*$, $N_2 \in \mathbb{N}^*$, and $\varepsilon > 0$,*

$$\mathbb{P}\left(\max_{i \in [\![1, N_1]\!]} \left| \frac{\tilde{\Omega}_N(i)}{\Omega_N(i)} - \eta Q \mathbb{1}_{\tilde{\mathsf{X}}} \right| \geq \varepsilon \right) \leq N_1 \check{c}_1 \exp\left(-\check{c}_2 N_2 \varepsilon^2 \right) , \quad (3.6.20)$$

where $\check{c}_1 \stackrel{\text{def}}{=} 2(c_1 \vee \tilde{c}_1)$ and $\check{c}_2 \stackrel{\text{def}}{=} \{(c_2 / \left\| w \right\|_\infty^2) \wedge \tilde{c}_2\} \varrho^2 / 4$.

To check (**AN1**), take $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$ and assume without loss of generality that $\tilde{\eta} h = 0$ and, consequently, $\eta Q h = 0$. We again rewrite the estimator according to (3.6.13) and apply Theorem 3.5.2 to the second factor. For this purpose, define the array

$$U_N(i) \stackrel{\text{def}}{=} \sqrt{N} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\tilde{\xi}_N(i,j)) \quad (i \in [\![1, N_1]\!], N \in \mathbb{N}^*) , \quad (3.6.21)$$

and furnish the same with the filtration $\{\mathcal{F}_N\}_{N \in \mathbb{N}^*}$ defined in (3.6.1). We may now write

$$\sum_{i=1}^{N_1} U_N(i) = \sum_{i=1}^{N_1} \{ U_N(i) - \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N \right] \} + \sqrt{N} \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} Q h(\xi_N(i,j)) , \quad (3.6.22)$$

where, by assumption, since $\|Qh\|_\infty \leq \|h\|_\infty \|Q\mathbb{1}_{\tilde{\mathsf{X}}}\|_\infty$, the second term on the right hand side satisfies the CLT

$$\sqrt{N}\sum_{i=1}^{N_1}\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\sum_{j=1}^{N_2}\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}Qh(\xi_N(i,j)) \xrightarrow{\mathcal{D}} \mathcal{N}(0,\sigma^2(Qh)) \,.$$

Our main challenge will be to handle the first term on the right hand side of (3.6.22). Since all individuals of the mutated archipelago are conditionally independent given $\mathcal{F}_N$, we notice that for all $i \in [\![1, N_1]\!]$,

$$\begin{aligned}\mathbb{E}\left[\left(\sum_{j=1}^{N_2}\frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}h(\tilde{\xi}_N(i,j))\right)^2 \mid \mathcal{F}_N\right] &= \sum_{j=1}^{N_2}\left(\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\right)^2 R(w^2h^2)(\xi_N(i,j)) \\ &+ \left(\sum_{j=1}^{N_2}\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}Qh(\xi_N(i,j))\right)^2 - \sum_{j=1}^{N_2}\left(\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\right)^2 (Qh)^2(\xi_N(i,j)) \,.\end{aligned}$$

Using this, we turn to the variance and deduce the expression

$$\begin{aligned}&\sum_{i=1}^{N_1}\left\{\mathbb{E}\left[U_N^2(i) \mid \mathcal{F}_N\right] - \mathbb{E}^2\left[U_N(i) \mid \mathcal{F}_N\right]\right\} \\ &\qquad = N\sum_{i=1}^{N_1}\left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\right)^2\sum_{j=1}^{N_2}\left(\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\right)^2 R(w^2h^2)(\xi_N(i,j)) \\ &\qquad\quad - N\sum_{i=1}^{N_1}\left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\right)^2\sum_{j=1}^{N_2}\left(\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\right)^2 (Qh)^2(\xi_N(i,j)) \,, \qquad (3.6.23)\end{aligned}$$

which tends in probability to $\mu_2 R(w^2h^2) - \mu_2(Qh)^2$ as the input archipelago satisfies Assumption (**AN**4) . This implies that Assumption (**B**1) in Theorem 3.5.2 holds with the same limit. To verify the Lindeberg condition (**B**2) in Theorem 3.5.2, note that proceeding as in (3.6.17) yields

$$\max_{i\in[\![1,N_1]\!]}|U_N(i)| \leq V_N + X_N Y_N^2 \quad (N \in \mathbb{N}^*) \,, \qquad (3.6.24)$$

where, for $N \in \mathbb{N}^*$,

$$\begin{cases} V_N = \sqrt{N}\max\limits_{i\in[\![1,N_1]\!]}\dfrac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\max\limits_{i\in[\![1,N_1]\!]}\left|\sum\limits_{j=1}^{N_2}\dfrac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}Qh(\xi_N(i,j))\right| \,, \\ X_N = N_1\max\limits_{i\in[\![1,N_1]\!]}\dfrac{\Omega_N(i)}{\sum_{i=1}^{N_1}\Omega_N(i')} \,, \\ Y_N^2 = \sqrt{\dfrac{N_2}{N_1}}\max\limits_{i\in[\![1,N_1]\!]}\left|\dfrac{\sum_{j=1}^{N_2}\tilde{\delta}_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\right| \,. \end{cases}$$

Here $\{V_N\}_{N\in\mathbb{N}^*}$ is $\mathcal{F}_N$-adapted and tends to zero in probability by (**AN**6) and Lemma 3.2.3, $\{X_N\}_{N\in\mathbb{N}^*}$ is $\mathcal{F}_N$-adapted and tight by (**AN**6), and Y_N has, by (3.6.19), (**D**), and the extension of Hoeffding's inequality in [Douc et al., 2011, Lemma 4], a tail of the form (3.5.1) (with $\alpha = 2$). Thus, by Lemma 3.5.3, (**B**2) holds true, and we may conclude the proof of (**AN**1) using first Lemma 3.5.5 and then Slutsky's lemma.

We turn to (**AN2**) and decompose the quantity under consideration according to

$$N_2 \sum_{i=1}^{N_1} \frac{\tilde{\Omega}_N(i)}{\sum_{i'=1}^{N_1} \tilde{\Omega}_N(i')} \left(\sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \tilde{\omega}_N(i,j')} h(\tilde{\xi}_N(i,j)) \right)^2$$
$$= N_2 \frac{\sum_{i''=1}^{N_1} \Omega_N(i'')}{\sum_{i'=1}^{N_1} \tilde{\Omega}_N(i')} \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i''=1}^{N_1} \Omega_N(i'')} \left(\frac{\Omega_N(i)}{\tilde{\Omega}_N(i)} - \frac{1}{\eta Q \mathbb{1}_{\tilde{\mathbb{X}}}} \right) \left(\sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\tilde{\xi}_N(i,j)) \right)^2$$
$$+ N_2 \frac{1}{\eta Q \mathbb{1}_{\tilde{\mathbb{X}}}} \frac{\sum_{i''=1}^{N_1} \Omega_N(i'')}{\sum_{i'=1}^{N_1} \tilde{\Omega}_N(i')} \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i''=1}^{N_1} \Omega_N(i'')} \left(\sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\tilde{\xi}_N(i,j)) \right)^2 . \quad (3.6.25)$$

The convergence in probability of the second term on the right hand side will now to be established using Theorem 3.5.1. For this purpose, define the triangular array

$$U_N(i) \overset{\text{def}}{=} N_2 \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \left(\sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\tilde{\xi}_N(i,j)) \right)^2 \quad (i \in [\![1, N_1]\!], N \in \mathbb{N}^*) , \quad (3.6.26)$$

and associate the same with the σ-field $\mathcal{F}_N$ defined in (3.6.1). We now apply the previous machinery and study the convergence of the series

$$\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N \right] = N_2 \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \left(\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \right)^2 R(w^2 h^2)(\xi_N(i,j))$$
$$+ N_2 \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \left(\sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} Q h(\xi_N(i,j)) \right)^2$$
$$- N_2 \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \left(\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \right)^2 (Qh)^2(\xi_N(i,j)) ,$$

which tends in probability to $\nu^2(Qh) + \mu_3 R(w^2 h^2) - \mu_3(Q^2 h)$ as the ancestor archipelago satisfies (**AN2**) and (**AN5**). Thus, the condition (**A1**) in Theorem 3.5.1 is checked. In addition, (**A2**) is checked using Lemma 3.5.3, as

$$\max_{i \in [\![1, N_1]\!]} |U_N(i)| \leq V_N + X_N Y_N^2 \quad (N \in \mathbb{N}^*) ,$$

where for $N \in \mathbb{N}^*$, $V_N = 0$ and

$$\begin{cases} X_N = N_1 \max\limits_{i \in [\![1, N_1]\!]} \dfrac{\Omega_N(i)}{\sum_{i=1}^{N_1} \Omega_N(i')} , \\ Y_N = \sqrt{\dfrac{N_2}{N_1}} \max\limits_{i \in [\![1, N_1]\!]} \left| \sum\limits_{j=1}^{N_2} \dfrac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\tilde{\xi}_N(i,j)) \right| , \end{cases}$$

where $\{X_N\}_{N \in \mathbb{N}^*}$ is $\mathcal{F}_N$-adapted and tight by (**AN6**) and each Y_N has, by [Douc et al., 2011, Lemma 4], since the input and output archipelagos satisfy (**D**), a tail of the form (3.5.1) (with $\alpha = 1$). Thus, (**A1**) holds true, and we may conclude that the series $\sum_{i=1}^{N_1} U_N(i)$ and $\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N \right]$ tend to the same limit in probability.

We turn to the first term of (3.6.25) and show that this tends to zero in probability. Indeed, note that the absolute value of the same is, up to the factor $\sum_{i'=1}^{N_1} \Omega_N(i') / \sum_{i''=1}^{N_1} \tilde{\Omega}_N(i'')$, which converges in probability

by (3.6.16), bounded by

$$\max_{i' \in [\![1,N_1]\!]} \left| \frac{\Omega_N(i')}{\tilde{\Omega}_N(i')} - \frac{1}{\eta Q \mathbb{1}_{\tilde{\mathsf{X}}}} \right| N_2 \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i''=1}^{N_1} \Omega_N(i'')} \left(\sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\tilde{\xi}_N(i,j)) \right)^2 ,$$

where the first factor vanishes in probability by Lemma 3.6.1 and Lemma 3.5.4, and the convergence of the second factor was established above. This establishes (**AN2**).

To check Assumption (**AN6**), consider the bound

$$N_1 \max_{i \in [\![1,N_1]\!]} \frac{\tilde{\Omega}_N(i)}{\sum_{i'=1}^{N_1} \tilde{\Omega}_N(i')} \leq \|w\|_\infty N_1 \max_{i \in [\![1,N_1]\!]} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \left(\left| \frac{\sum_{i'=1}^{N_1} \Omega_N(i')}{\sum_{i''=1}^{N_1} \tilde{\Omega}_N(i'')} - \frac{1}{\eta Q \mathbb{1}_{\tilde{\mathsf{X}}}} \right| + \frac{1}{\eta Q \mathbb{1}_{\tilde{\mathsf{X}}}} \right) , \tag{3.6.27}$$

where the second factor on the right hand side is tight as the ancestor archipelago is assumed to satisfy (**AN6**). Moreover, as the third factor tends to $1/\eta Q \mathbb{1}_{\tilde{\mathsf{X}}}$ in probability by (3.6.16) we conclude that (**AN6**) holds true also for the output. The assumptions (**AN3**–**5**) are checked using similar techniques, and we refer the reader to subsection 4.1.3 for all details.

3.6.10 Proof of Corollary 3.4.5

For $\ell \in \mathbb{N}$, combining the identity

$$\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}} = \eta_\ell Q_\ell \mathbb{1}_{\mathsf{X}} \times \eta_{\ell+1} Q_{\ell+1} \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}$$

with the bound $\eta_\ell Q_{\ell-1} \mathbb{1}_{\mathsf{X}} \geq c_-$ (the latter implied by (**M**)(iii)) yields

$$\frac{\eta_\ell R_\ell \{ w_\ell^2 Q_{\ell+1} \cdots Q_{n-1} (h - \eta_n h)^2 \}}{(\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}})^2} \leq c_-^{-1} \|w_\ell\|_\infty \left\| \frac{Q_{\ell+1} \cdots Q_{n-1} (h - \eta_n h)}{\eta_{\ell+1} Q_{\ell+1} \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}} \right\|_\infty^2 .$$

However, since, by Vergé et al. [2014],

$$\left\| \frac{Q_{\ell+1} \cdots Q_{n-1} (h - \eta_n h)}{\eta_{\ell+1} Q_{\ell+1} \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}} \right\|_\infty \leq \frac{\rho^{n-\ell-1}}{1-\rho} \operatorname{osc}(h) , \tag{3.6.28}$$

it holds that

$$\frac{\eta_\ell R_\ell \{ w_\ell^2 Q_{\ell+1} \cdots Q_{n-1} (h - \eta_n h)^2 \}}{(\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}})^2} \leq w_+ \frac{\rho^{2(n-\ell-1)}}{(1-\rho)^2 c_-} \operatorname{osc}^2(h) .$$

Finally, the proof is concluded by summing up the terms. Additional details may be found in section 4.2.

3.6.11 Proof of Corollary 3.4.8

Since the ℓth terms of the asymptotic variances (3.4.4) and (3.4.5) differ only by the multiplicative constant $n - \ell$, the proof follows straightforwardly by direct inspection of the proof of the analogous result for the standard bootstrap particle filter given in [Douc et al., 2014a, Theorem 11] (which in turn is an adaptation of the proof of Theorem 10 in the same paper, providing the analogous result for the particle predictor). More specifically, the result is obtained by

- embedding, using a trivial extension of Kolmogorov's extension theorem, the stationary sequence $\{Z_p\}_{p \in \mathbb{N}}$ into a stationary process $\{Z_p\}_{p \in \mathbb{Z}}$ with doubly infinite time.
- bounding, for a given $n \in \mathbb{N}$, using [Douc et al., 2014a, Equations 34–35], $\sigma_n^2 \langle Z_{0:n} \rangle(h)$ by a quantity of form $c \sum_{\ell=0}^n (n-\ell) \Delta_{n-\ell} \langle h \rangle (Z_{-\infty:\ell-1}, Z_{\ell:n})$, where c is a $\mathbb{P}$-a.s. finite random variable and Êeach function $\Delta_m \langle h \rangle : \mathsf{Z}^\infty \to \mathbb{R}_+$, $m \in \mathbb{N}$, is of the same type as the terms of the sum in [Douc et al., 2014a, Equation 35].

- using the stationarity to conclude that $\sum_{\ell=0}^{n}(n-\ell)\Delta_{n-\ell}\langle h\rangle(Z_{-\infty:\ell-1}, Z_{\ell:n})$ has the same distribution as $\sum_{\ell=0}^{n}\ell\Delta_{\ell}\langle h\rangle(Z_{-\infty:-\ell-1}, Z_{-\ell:0})$.
- bounding, using [Douc et al., 2014a, Equation 39], each term of the sum as $\Delta_{\ell}\langle h\rangle(Z_{-\infty:-\ell-1}, Z_{-\ell:0}) \leq d\beta^{\ell}$, $\mathbb{P}$-a.s., where d is a $\mathbb{P}$-a.s. finite random variable and $\beta < 1$ is a constant. This shows that $\sigma_n^2\langle Z_{0:n}\rangle(h) \leq cd\sum_{\ell=0}^{\infty}\ell\beta^{\ell} < \infty$, $\mathbb{P}$-a.s., which concludes the proof.

Chapitre 4

Discussions autour des propriétés de convergence des îlots de particules

Sommaire

4.1 Additional details concerning the preservation of asymptotic normality 97
4.1.1 Checking the assumptions AN2-6 in the proof of Theorem 3.3.3 97
4.1.2 Checking the assumptions AN2-6 in the proof of Theorem 3.3.6 98
4.1.3 Checking the assumptions AN3-5 in the proof of Theorem 3.3.9 99
4.2 Some additional details for the proof of Corollary 3.4.5 101
4.3 Technical result in the proof of Lemma 3.5.3 102
4.4 Analysis of the double bootstrap (B^2) predictor 102
4.5 Some additional convergence result for the B^2 filter algorithm 105
4.5.1 Joint central limit theorem for the B^2 filter algorithm 105
4.5.2 Technical lemmas 108
4.6 Some additional convergence result for the B^2 filter algorithm 109
4.6.1 Joint central limit theorem for the B^2 filter algorithm 109
4.6.2 Technical lemmas 112

Ce chapitre complète essentiellement le chapitre 3. D'une part, il donne des détails et achève les vérifications techniques de certains résultats du chapitre 3. D'autre part, il fournit des nouveaux résultats sur l'analyse des modèles d'îlots de particules. En particulier, un TCL joint pour le B^2 est démontré.
Ce chapitre s'organise de la façon suivante. Dans la section 4.1, on complète les preuves des théorèmes 3.3.3, 3.3.6, 3.3.9 qui montrent que les transformations élémentaires des îlots (SIL, SiL, et la Mutation respectivement) propagent la normalité asymptotique. En effet, seules certaines hypothèses sont vérifiées dans le chapitre 3. Les hypothèses restantes, (**AN2–6**) dans le cas des sélections au niveaux des îlots et des particules, et (**AN3–5**) dans le cas de la mutation sont vérifiées dans la section 4.1. Ces preuves complémentaires utilisent les mêmes techniques que les preuves du chapitre 3, notamment les théorèmes limites pour les tableaux triangulaires de variables aléatoires obtenus dans [Douc and Moulines, 2008]. Des détails de la preuve du Corollary 3.4.5, qui établit une borne uniforme de la variance asymptotique de l'algorithme du B^2 sous des hypothèses de mélange fort, se trouvent dans la section 4.2. De plus, un résultat technique utilisé dans la preuve du Lemma 3.5.3 est démontré dans la section 4.3. Dans la section 4.4, on compare les résultats obtenus dans chapitre 2 et ceux obtenus comme conséquence du chapitre 3 pour l'estimateur de prédiction dans l'algorithme B^2. Enfin, un CLT joint est démontré dans la section 4.6. Le lecteur intéressé pourra se référer au chapitre 3 pour les détails concernant les notations, acronymes, etc.

Auteurs:

- Christelle Vergé (Onera, CNES, CMAP, Palaiseau)
- Eric Moulines (Telecom ParisTech, Paris)
- Jimmy Olsson (KTH Royal Institute of Technology, Stockholm, Suède)

Statut: Chapitre basé sur l'article *Supplementary to "Convergence properties of weighted particle islands with application to the double bootstrap algorithm"*, soumis aux *Annales de l'Institut Henri Poincaré*, en octobre 2014, [Vergé et al., 2014]

Introduction This supplementary material, providing details that were omitted from some proofs in chapter 3, is structured as follows. In section 4.1 we complement the proofs of Theorem 3.3.3, Theorem 3.3.6, and Theorem 3.3.9 by checking that archipelagos transformed by SIL, SiL, and Mutation operations, respectively, satisfy all properties of the definition of asymptotic normality given that these properties hold true for the ancestor archipelago. Since a few of these properties are checked in chapter 3, the remaining properties are (**AN2–6**) in the cases of selection on the island and individual levels and (**AN3–5**) in the case of Mutation. Our proofs, which rely on limit theorems for triangular arrays obtained in [Douc and Moulines, 2008], will be propelled by the same machinery as the proofs of chapter 3. In addition, section 4.2 provides some details of the proof of Corollary 3.4.5, establishing time uniform boundedness of the asymptotic variance of estimates produced by B^2 algorithm. We refer to the chapter 3 for all details concerning notation, acronyms, etc. Finally, we prove an additional convergence result in section 4.6 which is a joint CLT.

4.1 Additional details concerning the preservation of asymptotic normality

4.1.1 Checking the assumptions AN2-6 in the proof of Theorem 3.3.3

First, a prefatory lemma.

Lemma 4.1.1. *Assume (**D**) and (**S**). Then*

$$\lim_{\lambda\to\infty} \sup_{N\in\mathbb{N}^*} \mathbb{P}\left(\max_{(i,j)\in[\![1,N_1]\!]\times[\![1,N_2]\!]} N_2 \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')} \geq \lambda \right) = 0 \,.$$

Proof. Using the boundedness of the particle weights,

$$\max_{(i,j)\in[\![1,N_1]\!]\times[\![1,N_2]\!]} N_2 \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')} \leq |\omega|_\infty \max_{i\in[\![1,N_1]\!]} \left(\frac{1}{N_2}\sum_{j=1}^{N_2}\omega_N(i,j) \right)^{-1} ,$$

where the quantity on right hand side is tight as it converges in probability to $|\omega|_\infty/\varrho$ by (**D**), (**S**) and Lemma 3.5.4. □

We start off with Assumption (**AN2**) and apply Theorem 3.5.1, this time to the array $U_N^{'}(i) \overset{\text{def}}{=} U_N^2(i)$, $i \in [\![1,N_1]\!], N \in \mathbb{N}^*$, where $\{U_N(i)\}_{i=1}^{N_1}$ is defined by (3.6.3), and the filtration $\{\mathcal{F}_N\}_{N\in\mathbb{N}^*}$ is defined by (3.6.1). To prove that $\sum_{i=1}^{N_1} U_N^{'}(i)$ converges in probability, we first note that the sum $\sum_{i=1}^{N_1}\mathbb{E}[U_N^{'}(i) \mid \mathcal{F}_N]$ converges in probability to $\nu^2(h)$ by (3.6.5). Moreover, the two conditions (**A1**) and (**A2**) in Theorem 3.5.1 are straightforwardly satisfied for the array under consideration, the latter condition by Lemma 3.5.3 as $\max_{i\in[\![1,N_1]\!]}|U_N^{'}(i)| \leq V_N^2$, where V_N is defined in (3.6.7) and V_N^2 vanishes in probability by Lemma 3.2.3. Consequently, (**AN2**) is satisfied with $\tilde{\nu}^2 = \nu^2$.

In the case of multinomial island selection, (**AN3**) coincides with the consistency property, which is implied by (**AN1**); thus, $\tilde{\mu}_1 = \eta$.

To check (**AN4**) we apply Theorem 3.5.1 to the array

$$U_N(i) \overset{\text{def}}{=} \frac{N_2}{N_1}\sum_{j=1}^{N_2}\left(\frac{\omega_N(I_N(i),j)}{\sum_{j'=1}^{N_2}\omega_N(I_N(i),j')} \right)^2 h(\xi_N(I_N(i),j)) \quad (i\in[\![1,N_1]\!], N\in\mathbb{N}^*) \,.$$

associated with the same filtration as previously. Since the ancestor archipelago satisfies (**AN5**),

$$\sum_{i=1}^{N_1}\mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right] = N_2\sum_{i=1}^{N_1}\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\sum_{j=1}^{N_2}\left(\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')} \right)^2 h(\xi_N(i,j)) \xrightarrow{\mathbb{P}} \mu_3 h \,, \tag{4.1.1}$$

and we show that $\sum_{i=1}^{N_1} U_N(i)$ tends to the same limit by using Theorem 3.5.1.
First, $U_N(i) \leq {N_2}^2 \|h\|_\infty / N_1 < \infty$ for all $i \in [\![1, N_1]\!]$ and $N \in \mathbb{N}^*$; moreover, be reusing (4.1.1) for $|h|$ we check (**A1**). To check the Lindeberg condition (**A2**), we bound, using $\{I_N(i)\}_{i=1}^{N_1} \subset [\![1, N_1]\!]$,

$$\max_{i \in [\![1,N_1]\!]} |U_N(i)| \leq \max_{(i,j) \in [\![1,N_1]\!] \times [\![1,N_2]\!]} N_2 \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \times \max_{i \in [\![1,N_1]\!]} \frac{1}{N_1} \left| \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\xi_N(i,j)) \right| ,$$

where the first factor on the right hand side is tight by Lemma 4.1.1 and the second term is bounded by $\|h\|_\infty / N_1$, which tends to zero when N tends to infinity. Thus, Lemma 3.5.4 can be applied for checking (**A2**), which establishes that $\tilde{\mu}_2 = \mu_3$.

Finally, in the case of selection on the island level, (**AN5**) coincides with (**AN4**) and (**AN6**) is trivially satisfied.

4.1.2 Checking the assumptions AN2-6 in the proof of Theorem 3.3.6

To check (**AN2**) and prove that the series $\sum_{i=1}^{N_1} U_N(i)$, where

$$U_N(i) \overset{\text{def}}{=} \frac{\Omega_N(i)}{N_2 \sum_{i'=1}^{N_1} \Omega_N(i')} \left(\sum_{j=1}^{N_2} \{h(\xi_N(i, J_N(i,j))) - \eta h\} \right)^2 \quad (i \in [\![1, N_1]\!], N \in \mathbb{N}^*) ,$$

converges in probability as $N \to \infty$, let $\{\mathcal{F}_N\}_{N \in \mathbb{N}}$ be defined as in (3.6.1) and consider the sum

$$\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right] = \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{h(\xi_N(i,j)) - \eta h\}^2 + (N_2 - 1) \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \left(\sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \{h(\xi_N(i,j)) - \eta h\} \right)^2 , \quad (4.1.2)$$

where we used, as previously, that for each $i \in [\![1, N_1]\!]$, the variables $\{h(\xi_N(i, J_N(i,j)))\}_{j=1}^{N_2}$ are independent and identically distributed with common mean $\sum_{j=1}^{N_2} \omega_N(i,j) h(\xi_N(i,j)) / \sum_{j'=1}^{N_2} \omega_N(i,j')$, conditionally on $\mathcal{F}_N$. The first term of the right hand side of (4.1.2) tends in probability to $\eta\{(h - \eta h)^2\}$ by consistency, while the second term tends in probability to $\nu^2(h)$ by (**AN2**). Since this establishes the condition (**A1**) in Theorem 3.5.1, the series $\sum_{i=1}^{N_1} U_N(i)$ and $\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right]$ have the same limit $\eta\{(h - \eta h)^2\} + \nu^2(h)$ in probability as soon as the condition (**A2**) in the same theorem can be checked for the array in question. However, write

$$\max_{i \in [\![1,N_1]\!]} |U_N(i)| \leq V_N + X_N Y_N^2 \quad (N \in \mathbb{N}^*) ,$$

where, for $N \in \mathbb{N}^*$,

$$\begin{cases} V_N = 2N_2 \max\limits_{i \in [\![1,N_1]\!]} \dfrac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \left(\max\limits_{i \in [\![1,N_1]\!]} \left| \sum\limits_{j=1}^{N_2} \dfrac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\xi_N(i,j)) \right| \right)^2 , \\ X_N = 2 \max\limits_{i \in [\![1,N_1]\!]} N_1 \dfrac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} , \\ Y_N = \max\limits_{i \in [\![1,N_1]\!]} \dfrac{1}{\sqrt{N}} \left| \sum\limits_{j=1}^{N_2} \delta_N(i,j) \right| , \end{cases}$$

and the δ_Ns are defined in Equation 3.6.8; then, since V_N tends to zero in probability (by (**AN6**) and Lemma 3.2.3), X_N is tight, and Y_N has an exponential tail (by (3.6.9)), Lemma 3.5.3 applies, establishing that the array satisfies Assumption (**A2**). Consequently, we obtain that $\tilde{\nu}^2(h) = \eta\{(h - \eta h)^2\} + \nu^2(h)$.

To verify (**AN3**) we retain to the previous machinery and study the array

$$U_N(i) \overset{\text{def}}{=} \frac{N_1}{N_2} \left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \right)^2 \sum_{j=1}^{N_2} h(\xi_N(i, J_N(i,j))) \quad (i \in [\![1, N_1]\!], N \in \mathbb{N}^*)$$

associated with the filtration $\{\mathcal{F}_N\}_{N \in \mathbb{N}^*}$ defined in (3.6.1). To establish the convergence of $\sum_{i=1}^{N_1} U_N(i)$ we reapply Theorem 3.5.1 and consider

$$\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right] = N_1 \sum_{i=1}^{N_1} \left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \right)^2 \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\xi_N(i,j)) \xrightarrow{\mathbb{P}} \mu_1 h \,, \tag{4.1.3}$$

where convergence follows since the ancestor archipelago satisfies (**AN3**). By reusing (4.1.3) for $|h|$ we check that the condition (**A1**) in Theorem 3.5.1 is satisfied. Moreover, since

$$\max_{i \in [\![1, N_1]\!]} |U_N(i)| \leq \|h\|_\infty \left(\sqrt{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \right)^2 \,,$$

where the right hand side vanishes in probability by (**AN6**), Lemma 3.5.3 implies that the array satisfies (**A2**) as well. Thus, (**AN3**) holds true with $\tilde{\mu}_1 = \mu_1$.

In addition, since Assumption (**AN4**) coincides with (**AN3**) in the case of uniform particle weights, we obtain immediately that $\tilde{\mu}_2 = \mu_1$. Moreover, (**AN5**) coincides precisely with (**C1**),
which is satisfied as the output satisfies the stronger condition (**AN1**), and we obtain $\tilde{\mu}_3 = \eta$. Finally, (**AN6**) holds trivially true.

4.1.3 Checking the assumptions AN3-5 in the proof of Theorem 3.3.9

In order to check (**AN3**), pick $h \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})$ and decompose the quantity of interest according to

$$\begin{aligned}
&N_1 \sum_{i=1}^{N_1} \left(\frac{\tilde{\Omega}_N(i)}{\sum_{i'=1}^{N_1} \tilde{\Omega}_N(i')} \right)^2 \sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \tilde{\omega}_N(i,j')} h(\tilde{\xi}_N(i,j)) \\
&\quad = N_1 \left(\frac{\sum_{i=1}^{N_1} \Omega_N(i)}{\sum_{i'=1}^{N_1} \tilde{\Omega}_N(i')} \right)^2 \sum_{i=1}^{N_1} \left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \right)^2 \left(\frac{\tilde{\Omega}_N(i)}{\Omega_N(i)} - \eta Q \mathbb{1}_{\mathsf{X}} \right) \sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\tilde{\xi}_N(i,j)) \\
&\qquad + N_1 \eta Q \mathbb{1}_{\mathsf{X}} \left(\frac{\sum_{i=1}^{N_1} \Omega_N(i)}{\sum_{i'=1}^{N_1} \tilde{\Omega}_N(i')} \right)^2 \sum_{i=1}^{N_1} \left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \right)^2 \sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\tilde{\xi}_N(i,j)) \,.
\end{aligned} \tag{4.1.4}$$

In order to handle the second term of this decomposition, we apply Theorem 3.5.1 to the array

$$U_N(i) \overset{\text{def}}{=} N_1 \left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \right)^2 \sum_{j=1}^{N_2} \frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} h(\tilde{\xi}_N(i,j)) \quad (i \in [\![1, N_1]\!], N \in \mathbb{N}^*)$$

furnished with the filtration $\{\mathcal{F}_N\}_{N \in \mathbb{N}^*}$ given by (3.6.1). First, we observe that

$$\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right] = N_1 \sum_{i=1}^{N_1} \left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \right)^2 \sum_{j=1}^{N_2} \frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} Q h(\xi_N(i,j)) \xrightarrow{\mathbb{P}} \mu_1 Q h \,,$$

as the ancestor archipelago satisfies Assumption (**AN3**). Thus, the condition (**A1**) in Theorem 3.5.1 holds true. In addition, as

$$\max_{i\in[\![1,N_1]\!]}|U_N(i)|\le\|w\|_\infty\|h\|_\infty\left(\sqrt{N_1}\max_{i\in[\![1,N_1]\!]}\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\right)^2,$$

also (**A2**) is verified by Lemma 3.5.3 (applied with $X_N=Y_N=0$) and the fact that the input archipelago satisfies (**AN6**). Consequently, the also series $\sum_{i=1}^{N_1}U_N(i)$ tends in probability to the limit μ_1Qh, which, by (3.6.16), implies that the second term of (4.1.4) tends to $\mu_1Qh/\eta Q\mathbb{1}_{\tilde{\mathsf{X}}}$. To treat the first term of (4.1.4), note that this is, up to the factor $\sum_{i'=1}^{N_1}\Omega_N(i')/\sum_{i''=1}^{N_1}\tilde{\Omega}_N(i'')$, which converges in probability by (3.6.16), bounded by

$$N_1\max_{i\in[\![1,N_1]\!]}\left|\frac{\tilde{\Omega}_N(i)}{\Omega_N(i)}-\eta Q\mathbb{1}_{\tilde{\mathsf{X}}}\right|\sum_{i=1}^{N_1}\left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\right)^2\sum_{j=1}^{N_2}\frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}|h|(\tilde{\xi}_N(i,j)),$$

which tends to zero in probability by the previous computation and Lemma 3.6.1. This completes the proof of (**AN3**).

In order to prove (**AN4**), introduce the array

$$U_N(i)\overset{\text{def}}{=}N\left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\right)^2\sum_{j=1}^{N_2}\left(\frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\right)^2h(\tilde{\xi}_N(i,j))\quad(i\in[\![1,N_1]\!],N\in\mathbb{N}^*)$$

and equip the same with usual filtration $\{\mathcal{F}_N\}_{N\in\mathbb{N}^*}$. With this notation, the quantity of interest in (**AN4**) can be written as

$$\left(\frac{\sum_{i=1}^{N_1}\Omega_N(i)}{\sum_{i'=1}^{N_1}\tilde{\Omega}_N(i')}\right)^2\sum_{i=1}^{N_1}U_N(i),$$

where the first factor tends to $1/(\eta Q\mathbb{1}_{\tilde{\mathsf{X}}})^2$ by Lemma 3.6.1. Thus, it is enough to show that the second factor tends to $\mu_2R(w^2h)$ in probability, and for this purpose we use Theorem 3.5.1. As the ancestor archipelago satisfies (**AN4**), the quantity

$$\sum_{i=1}^{N_1}\mathbb{E}\left[U_N(i)\mid\mathcal{F}_N\right]=N\sum_{i=1}^{N_1}\left(\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\right)^2\sum_{j=1}^{N_2}\left(\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\right)^2R(w^2h)(\xi_N(i,j))$$

tends in probability to the desired limit $\mu_2R(w^2h)$. This implies the condition (**A1**) in Theorem 3.5.1. In addition, (**A2**) is checked using Lemma 3.5.3; indeed,

$$\max_{i\in[\![1,N_1]\!]}|U_N(i)|\le|\omega|_\infty\|w\|_\infty^2\|h\|_\infty\left(\sqrt{N_1}\max_{i\in[\![1,N_1]\!]}\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\right)^2\max_{i\in[\![1,N_1]\!]}\frac{N_2}{\sum_{j=1}^{N_2}\omega_N(i,j)},$$

where the the right hand side is adapted to $\{\mathcal{F}_N\}_{N\in\mathbb{N}^*}$ and vanishes in probability by Lemma 3.5.4, as the ancestor archipelago satisfies (**AN6**) and (**D**). This shows (**AN4**).

Finally, in order to prove (**AN5**) we decompose the quantity of interest according to

$$\begin{aligned}N_2\sum_{i=1}^{N_1}\frac{\tilde{\Omega}_N(i)}{\sum_{i'=1}^{N_1}\tilde{\Omega}_N(i')}\sum_{j=1}^{N_2}\left(\frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2}\tilde{\omega}_N(i,j')}\right)^2h(\tilde{\xi}_N(i,j))=\\ N_2\left(\frac{\sum_{i'=1}^{N_1}\Omega_N(i')}{\sum_{i''=1}^{N_1}\tilde{\Omega}_N(i'')}\right)\sum_{i=1}^{N_1}\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\left(\frac{\Omega_N(i)}{\tilde{\Omega}_N(i)}-\frac{1}{\eta Q\mathbb{1}_{\tilde{\mathsf{X}}}}\right)\sum_{j=1}^{N_2}\left(\frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\right)^2h(\tilde{\xi}_N(i,j))\\ +N_2\frac{1}{\eta Q\mathbb{1}_{\tilde{\mathsf{X}}}}\left(\frac{\sum_{i'=1}^{N_1}\Omega_N(i')}{\sum_{i''=1}^{N_1}\tilde{\Omega}_N(i'')}\right)\sum_{i=1}^{N_1}\frac{\Omega_N(i)}{\sum_{i'=1}^{N_1}\Omega_N(i')}\sum_{j=1}^{N_2}\left(\frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2}\omega_N(i,j')}\right)^2h(\tilde{\xi}_N(i,j)).\end{aligned}\tag{4.1.5}$$

To deal with the second term we reapply Theorem 3.5.1, this time to the array

$$U_N(i) = N_2 \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \left(\frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \right)^2 h(\tilde{\xi}_N(i,j)) \quad (i \in [\![1, N_1]\!], N \in \mathbb{N}^*) .$$

As usual, we study first the series

$$\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right] = N_2 \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \left(\frac{\omega_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \right)^2 R(w^2 h)(\xi_N(i,j)) \xrightarrow{\mathbb{P}} \mu_3 R(w^2 h) ,$$

where the limit is a consequence of the fact that the ancestor archipelago satisfies (**AN5**).
This establishes (**A1**) in Theorem 3.5.1. To check also (**A2**), consider the upper bound

$$\max_{i \in [\![1,N_1]\!]} |U_N(i)| \leq |\omega|_\infty \|w\|_\infty^2 \|h\|_\infty \max_{i \in [\![1,N_1]\!]} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \max_{i \in [\![1,N_1]\!]} \frac{N_2}{\sum_{j=1}^{N_2} \omega_N(i,j)} ,$$

which is $\{\mathcal{F}_N\}_{N \in \mathbb{N}^*}$-adapted and tends to zero in probability by Lemma 3.5.4, as the ancestor archipelago satisfies (**C2**) and (**D**). Now, Theorem 3.5.1 guarantees that $\sum_{i=1}^{N_1} U_N(i)$ and $\sum_{i=1}^{N_1} \mathbb{E}\left[U_N(i) \mid \mathcal{F}_N\right]$ have the same limit $\mu_3 R(w^2 h)$ in probability. Moreover, note that the second term of (4.1.5) is, up to the factor $\sum_{i'=1}^{N_1} \Omega_N(i') / \sum_{i''=1}^{N_1} \tilde{\Omega}_N(i'')$, bounded by

$$N_2 \max_{i \in [\![1,N_1]\!]} \left| \frac{\Omega_N(i)}{\tilde{\Omega}_N(i)} - \frac{1}{\eta Q \mathbb{1}_{\tilde{\mathsf{X}}}} \right| \sum_{i=1}^{N_1} \frac{\Omega_N(i)}{\sum_{i'=1}^{N_1} \Omega_N(i')} \sum_{j=1}^{N_2} \left(\frac{\tilde{\omega}_N(i,j)}{\sum_{j'=1}^{N_2} \omega_N(i,j')} \right)^2 |h|(\tilde{\xi}_N(i,j)) ,$$

which tends to zero in probability by (3.6.20) and Lemma 3.5.4. Thus, also (**AN5**) holds true.

4.2 Some additional details for the proof of Corollary 3.4.5

In subsection 3.4.3 we establish that the asymptotic variances, given in Corollary 3.4.3, of Monte Carlo estimates produced by the B^2 algorithm are uniformly bounded in the iteration index n. In the following we check (3.6.28) in the proof of Corollary 3.4.5. Write, for all $(\ell, n) \in \mathbb{N}^2$ such that $\ell \leq n$ and $x \in \mathsf{X}$,

$$\begin{aligned}
&\frac{Q_\ell \cdots Q_{n-1}(h - \eta_n h)(x)}{\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}} \\
&= \frac{Q_\ell \cdots Q_{n-1} h(x)}{\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}} - \frac{Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}(x)}{\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}} \eta_n h \\
&= \frac{Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}(x)}{\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}} \left[\frac{Q_\ell \cdots Q_{n-1} h(x)}{Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}(x)} - \frac{\eta_\ell Q_\ell \cdots Q_{n-1} h}{\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}} \right] .
\end{aligned} \tag{4.2.1}$$

Note that since $Q_\ell \ldots Q_{n-1} h(x) = \delta_x Q_\ell \ldots Q_{n-1} h$ (where δ_x denotes the Dirac mass located at x) we may, under (**M**), apply [Douc et al., 2014b, Proposition 10.20], yielding the uniform bound

$$\left| \frac{\delta_x Q_\ell \cdots Q_n h}{\delta_x Q_\ell \cdots Q_n \mathbb{1}_{\mathsf{X}}} - \frac{\eta_\ell Q_\ell \cdots Q_n h}{\eta_\ell Q_\ell \cdots Q_n \mathbb{1}_{\mathsf{X}}} \right| \leq \rho^{n-\ell} \operatorname{osc}(h) \quad (x \in \mathsf{X}) . \tag{4.2.2}$$

Combining (4.2.1) and (4.2.2) with the uniform bound

$$\frac{Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}(x)}{\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}} \leq \frac{\sigma_+}{\sigma_-} = \frac{1}{1-\rho} \quad (x \in \mathsf{X}) ,$$

where $\rho = 1 - \sigma_-/\sigma_+$, we obtain

$$\left\| \frac{Q_\ell \cdots Q_{n-1}(h - \eta_n h)}{\eta_\ell Q_\ell \cdots Q_{n-1} \mathbb{1}_{\mathsf{X}}} \right\|_\infty \leq \frac{\rho^{n-\ell}}{1-\rho} \operatorname{osc}(h) \ ,$$

showing (3.6.28).

4.3 Technical result in the proof of Lemma 3.5.3

We recall here two properties of the expectation that are used in the proof of Lemma 3.5.3.

Lemma 4.3.1. *Let X be a random variable. Then,*

$$\mathbb{E}\left[X \mathbb{1}_{X \geq 0}\right] = \int_0^{+\infty} \mathbb{P}(X > t) \, \mathrm{d}t \ .$$

Proof.

$$\begin{aligned} \int_0^{+\infty} \mathbb{P}(X > t) \, \mathrm{d}t &= \int_0^{+\infty} \int_t^{+\infty} \mathbb{P}_X(\mathrm{d}x) \, \mathrm{d}t = \int_{0 \leq t \leq x \leq +\infty} \mathbb{P}_X(\mathrm{d}x) \, \mathrm{d}t \\ &= \int_0^{+\infty} \left(\int_0^x \mathrm{d}t \right) \mathbb{P}_X(\mathrm{d}x) = \int_0^{+\infty} x \, \mathbb{P}_X(\mathrm{d}x) = \mathbb{E}\left[X \mathbb{1}_{X \geq 0}\right] \ . \end{aligned}$$

□

Proposition 4.3.2. *Let X be a random variable and $\varepsilon > 0$. Then,*

$$\mathbb{E}\left[X \mathbb{1}_{X \geq \varepsilon}\right] = \int_\varepsilon^{+\infty} \mathbb{P}(X > t) \, \mathrm{d}t + \varepsilon \, \mathbb{P}(X \geq \varepsilon)$$

Proof. $\mathbb{E}\left[X \mathbb{1}_{X \geq \varepsilon}\right] = \mathbb{E}\left[(X - \varepsilon) \mathbb{1}_{X - \varepsilon \geq 0}\right] + \varepsilon \, \mathbb{P}(X \geq \varepsilon)$. Write $U \overset{\text{def}}{=} X - \varepsilon$,

$$\begin{aligned} \mathbb{E}\left[(X - \varepsilon) \mathbb{1}_{X - \varepsilon \geq 0}\right] &= \mathbb{E}\left[U \mathbb{1}_{U \geq 0}\right] \\ &= \int_0^{+\infty} \mathbb{P}(U > t) \mathrm{d}t \quad \text{using Lemma 4.3.1} \\ &= \int_0^{+\infty} \mathbb{P}(X > t + \varepsilon) \mathrm{d}t \\ &= \int_\varepsilon^{+\infty} \mathbb{P}(X > y) \mathrm{d}y \ . \end{aligned}$$

□

4.4 Analysis of the double bootstrap (B^2) predictor

In this section, we apply the results proved in chapter 3 to derive the asymptotic variances of the B² predictor algorithm and compare them with the expressions found in chapter 2. As a consequence, we deduce the asymptotic variance for the standard bootstrap predictor. This section specifies Remark 4 in chapter 3.

Let M_n be a collection of normalized transition kernels $M_n : \mathsf{X} \times \mathcal{X} \to [0, 1]$ and g_n be a collection of nonnegative potential functions $g_n : \mathsf{X} \to \mathbb{R}_+^*$. The B² predictor algorithm approximates the sequence of distributions $\{\eta_n\}_{n \in \mathbb{N}}$, defined by recursion as $\eta_{n+1} h = \eta_n(g_n M_{n+1} h)/\eta_n g_n$. The B² predictor algorithm proceeds as follows. Start with an un-weighted archipelago $\{(1, \{(\xi_N^{(0)}(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}$, where $\xi_N^{(0)}(i,j)$ are i.i.d. according to η_0. Then, each iteration $n \in \mathbb{N}$ of the B² algorithm can be decomposed into 4 steps:

- The weighting of the archipelago: suppose that the archipelago $\{(1, \{(\xi_N^{(n)}(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ approximates η_n, then, the weighting of this archipelago consists in the mutation step introduced in Algorithm 9 with

$$\begin{cases} w(x,y) = g_n(x) \\ R(x, \mathrm{d}y) = \delta_x(\mathrm{d}y) \\ Q(x, \mathrm{d}y) = g_n(x)\delta_x(\mathrm{d}y) \end{cases}$$

  Hence, the weight $\omega_N^{(n)}(i,j) = g_n(\xi_N^{(n)}(i,j))$ is affected to each particle $\xi_N^{(n)}(i,j)$, and the island weights are defined by $\Omega_N^{(n)}(i) = N_2^{-1} \sum_{j=1}^{N_2} g_n(\xi_N^{(n)}(i,j))$. At the end of this step, as a consequence of Theorem 3.3.7 and Theorem 3.3.9, the archipelago $\{(\Omega_N^{(n)}(i), \{(\xi_N^{(n)}(i,j), \omega_N^{(n)}(i,j))\}$ is consistent for $\tilde{\eta}_n$ and asymptotically normal for
  $(\tilde{\sigma}_n^2, \tilde{\nu}_n^2, \{\tilde{\mu}_1^{(n)}, \tilde{\mu}_2^{(n)}, \tilde{\mu}_3^{(n)}\})$, where

$$\begin{cases} \tilde{\eta}_n h = \eta_n(g_n h)/\eta_n g_n \\ \tilde{\sigma}_n^2 h = \sigma_n^2\{g_n(h - \tilde{\eta}_n h)\}/(\eta_n g_n)^2 \\ \tilde{\nu}_n^2 h = \nu_n^2\{g_n(h - \tilde{\eta}_n h)\}/(\eta_n g_n)^2 \\ \tilde{\mu}_1^{(n)} h = \mu_1^{(n)}\{Q_n h\}/\eta_n g_n \\ \tilde{\mu}_2^{(n)} h = \mu_2^{(n)}\{g_n^2 h\}/(\eta_n g_n)^2 \\ \tilde{\mu}_3^{(n)} h = \mu_3^{(n)}\{g_n^2 h\}/(\eta_n g_n)^2 \end{cases}$$

  The new estimator $\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}(g_n h)(\xi_N^{(n)}(i,j))/\sum_{i=1}^{N_1}\sum_{j=1}^{N_2} g_n(\xi_N^{(n)}(i,j))$ approximates $\tilde{\eta}_n h$.

- The island selection : this is exactly the same step described in Algorithm 7. As a consequence of Theorem 3.3.1 and Theorem 3.3.3, the output archipelago
  $\{(1, \{(\xi_N^{(n)}(I_N(i), j), \omega_N^{(n)}(I_N(i), j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ is consistent for $\tilde{\eta}_n$ and asymptotically normal for $(\tilde{\sigma}_n^2 + \tilde{\nu}_n^2, \tilde{\nu}_n^2, \{\tilde{\eta}_n, \tilde{\mu}_3^{(n)}, \tilde{\mu}_3^{(n)}\})$. The output estimator
  $\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}(g_n h)(\xi_N^{(n)}(I_N(i), j))/N_1 \sum_{j=1}^{N_2} g_n(\xi_N^{(n)}(I_N(i), j))$ approximates $\tilde{\eta}_n h$.

- The individual selection: this is exactly the same step described in Algorithm 8. As a consequence of Theorem 3.3.4 and Theorem 3.3.6, the output archipelago
  $\{(1, \{(\xi_N^{(n)}(I_N(i), J_N(i,j)), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ is consistent for $\tilde{\eta}_n$ and asymptotically normal for $(\tilde{\sigma}_n^2 + \tilde{\nu}_n^2 + \tilde{\eta}_n\{(h - \tilde{\eta}_n h)^2\}, \tilde{\nu}_n^2 + \tilde{\eta}_n\{(h - \tilde{\eta}_n h)^2\}, \{\tilde{\eta}_n, \tilde{\eta}_n, \tilde{\eta}_n\})$. The output estimator
  $\sum_{i=1}^{N_1}\sum_{j=1}^{N_2} h(\xi_N^{(n)}(I_N(i), J_N(i,j)))/N_1 N_2$ approximates $\tilde{\eta}_n h$.

- The mutation: this step is an application of Algorithm 9 with

$$\begin{cases} w(x,y) = 1 \\ R = M_{n+1} \\ Q = M_{n+1} \end{cases}$$

  As a consequence of Theorem 3.3.7 and Theorem 3.3.9, the output archipelago
  $\{(1, \{(\xi_N^{(n+1)}(i,j), 1)\}_{j=1}^{N_2})\}_{i=1}^{N_1}$ is consistent for η_{n+1} and asymptotically normal for

$(\sigma^2_{n+1}, \nu^2_{n+1}, \{\eta_{n+1}, \eta_{n+1}, \eta_{n+1}\})$, where

$$\begin{cases} \eta_{n+1}h = \tilde{\eta}_n(M_{n+1}h) = \dfrac{\eta_n(g_n M_{n+1}h)}{\eta_n g_n} \ . \\ \\ \sigma^2_{n+1}h = \dfrac{\sigma^2_n\{g_n M_{n+1}(h-\eta_{n+1}h)\}}{(\eta_n g_n)^2} + \dfrac{\nu^2_n\{g_n M_{n+1}(h-\eta_{n+1}h)\}}{(\eta_n g_n)^2} + \eta_{n+1}\{(h-\eta_{n+1}h)^2)\} \\ \\ \nu^2_{n+1}h = \dfrac{\nu^2_n\{g_n M_{n+1}(h-\eta_{n+1}h)\}}{(\eta_n g_n)^2} + \eta_{n+1}\{(h-\eta_{n+1}h)^2\} \end{cases}$$

Define $\tilde{Q}_n : \mathsf{X} \times \mathsf{X} \to \mathbb{R}_+$ a possibly un-normalised transition kernel, as in (2.3.1) or (3.4.7), *i.e.* $\tilde{Q}_n(x, \mathrm{d}y) = g_{n-1}(x) M_n(x, \mathrm{d}y)$. Also recall the definition (2.3.6) of the finite kernel $P_{p,n}$ from $\mathsf{X} \times \mathsf{X}$ to $\mathbb{R}_+$,

$$P_{p,n} = \frac{\tilde{Q}_{p+1}\tilde{Q}_{p+2}\dots\tilde{Q}_n}{\eta_p \tilde{Q}_{p+1}\tilde{Q}_{p+2}\dots\tilde{Q}_n \mathbb{1}_{\mathsf{X}}} \quad (p < n) \ , \tag{4.4.1}$$

Then, one may write ν^2_{n+1} as

$$\nu^2_{n+1}h = \nu^2_n \left\{ \frac{\tilde{Q}_{n+1}(h - \eta_{n+1}h)}{\eta_n \tilde{Q}_{n+1}\mathbb{1}_{\mathsf{X}}} \right\} + \eta_{n+1}\{(h-\eta_{n+1}h)^2\} \ .$$

By iteration, one may show that

$$\nu^2_n h = \sum_{p=0}^{n} \eta_p \left\{ \left(\frac{\tilde{Q}_{p+1}\tilde{Q}_{p+2}\dots\tilde{Q}_n(h-\eta_n h)}{\eta_p \tilde{Q}_{p+1}\tilde{Q}_{p+2}\dots\tilde{Q}_n \mathbb{1}_{\mathsf{X}}} \right)^2 \right\} = \sum_{p=0}^{n} \eta_p \left\{ (P_{p,n}(h-\eta_n h))^2 \right\} \ , \tag{4.4.2}$$

with the convention $P_{p,p} = id$ for all $p \in \mathbb{N}$. From the variance increment for the B^2 predictor

$$\sigma^2_{n+1}h = \sigma^2_n \left\{ \frac{\tilde{Q}_{n+1}(h - \eta_{n+1}h)}{\eta_n \tilde{Q}_{n+1}\mathbb{1}_{\mathsf{X}}} \right\} + \nu^2_{n+1}h \ ,$$

one may deduce the explicit expression for the asymptotic variance

$$\begin{aligned} \sigma^2_n h &= \sum_{p=0}^{n} \nu^2_p \left\{ P_{p,n}(h - \eta_n h) \right\} \\ &= \sum_{p=0}^{n} \sum_{\ell=0}^{p} \eta_\ell \left\{ (P_{\ell,p} P_{p,n}(h-\eta_n h))^2 \right\} \\ &= \sum_{\ell=0}^{n} \sum_{p=\ell}^{n} \eta_\ell \left\{ (P_{\ell,n}(h-\eta_n h))^2 \right\} \\ &= \sum_{\ell=0}^{n} (n - \ell + 1)\, \eta_\ell \left\{ (P_{\ell,n}(h-\eta_n h))^2 \right\} \\ &= V_n(h) + \tilde{V}_n(h) \ , \end{aligned}$$

where $V_n(h)$ and $\tilde{V}_n(h)$ are defined in chapter 2 by

$$V_n(h) = \sum_{p=0}^{n} \eta_p \left\{ (P_{p,n}(h-\eta_n h))^2 \right\} \quad \text{and} \quad \tilde{V}_n(h) = \sum_{p=0}^{n} (n-p)\, \eta_p \left\{ (P_{p,n}(h-\eta_n h))^2 \right\} \ .$$

Remark 1. The standard bootstrap predictor algorithm can be viewed as a particular case of the B^2 predictor algorithm for $N_1 = 1$. In this case, there is necessarily no island selection. So, the variance increment for the standard bootstrap predictor becomes

$$\sigma_{n+1}^2 h = \sigma_n^2 \left\{ \frac{\tilde{Q}_{n+1}(h - \eta_{n+1} h)}{\eta_n \tilde{Q}_{n+1} \mathbb{1}_{\mathsf{X}}} \right\} + \eta_{n+1}\{(h - \eta_{n+1})^2\} \, . \tag{4.4.3}$$

It follows immediately that the explicit variance of the standard bootstrap predictor algorithm is $V_n(h)$, as shown in chapter 2.

In the light of the previous remark, the variance of the B^2 predictor algorithm is the sum of the variance of the standard bootstrap predictor algorithm ($V_n(h)$) and an additional variance due to the island interaction ($\tilde{V}_n(h)$).

4.5 Some additional convergence result for the B^2 filter algorithm

4.5.1 Joint central limit theorem for the B^2 filter algorithm

The following theorem extends property (**AN1**), giving a joint CLT. Denote by ϕ_n^N the filter estimator produced by the B^2 algorithm (cf Algorithm 4), where for all $p \geq 0$,

$$\phi_{3p}^N(h_{3p}) \stackrel{\text{def}}{=} \sum_{i=1}^{N_1} \frac{\Omega_N^{(p)}(i)}{\sum_{i'=1}^{N_1} \Omega_N^{(p)}(i')} \sum_{j=1}^{N_2} \frac{\omega_N^{(p)}(i,j)}{\sum_{j'=1}^{N_2} \omega_N^{(p)}(i,j')} h_{3p}(\xi_N^{(p)}(i,j)) \, , \tag{4.5.1}$$

$$\phi_{3p+1}^N(h_{3p+1}) \stackrel{\text{def}}{=} \frac{1}{N_1} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \frac{\omega_N^{(p)}(I_N^{(p)}(i), j)}{\sum_{j'=1}^{N_2} \omega_N^{(p)}(I_N^{(p)}(i), j')} h_{3p+1}(\xi_N^{(p)}(I_N^{(p)}(i), j)) \, , \tag{4.5.2}$$

$$\phi_{3p+2}^N(h_{3p+2}) \stackrel{\text{def}}{=} \frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} h_{3p+2}(\xi_N^{(p)}(I_N^{(p)}(i), J_N^{(p)}(i,j))) \, , \tag{4.5.3}$$

and by $\overline{h}_n$ the centered functions, where for all $p \in \mathbb{N}$,

$$\begin{aligned} \overline{h}_{3p} &\stackrel{\text{def}}{=} h_{3p} - \eta_p h_{3p} \, , \\ \overline{h}_{3p+1} &\stackrel{\text{def}}{=} h_{3p+1} - \eta_p h_{3p+1} \, , \\ \overline{h}_{3p+2} &\stackrel{\text{def}}{=} h_{3p+2} - \eta_p h_{3p+2} \, . \end{aligned}$$

We establish the joint convergence in distribution of the vector $(\sqrt{N}\phi_k^N(\overline{h}_k))_{0 \leq k \leq n}$ for all $n \in \mathbb{N}$.

Theorem 4.5.1. *For the B^2 filter, for all $n \in \mathbb{N}$ and all functions $(h_0, \ldots, h_n) \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})^{n+1}$,*

$$\left(\sqrt{N}\phi_k^N(\overline{h}_k)\right)_{0 \leq k \leq n} \xrightarrow[N \to +\infty]{\mathcal{D}} \mathcal{N}(0, \Gamma_n(h_0, \ldots, h_n)) \, ,$$

where the asymptotic covariance matrix Γ_n is recursively given by $\Gamma_0(\overline{h}_0, \overline{h}_0) \stackrel{\text{def}}{=} \sigma_0^2 h_0 = \eta_0(\overline{h}_0^2)$ and for all $p \geq 0$,

$$\Gamma_{3(p+1)}(h_0, \ldots, h_{3(p+1)}) = \left(\begin{array}{c|c} \Gamma_{3p+2}(h_0, \ldots, h_{3p+2}) & \begin{array}{c} \dfrac{\Gamma_{3p+2}(Q_p \overline{h}_{3(p+1)}, \overline{h}_0)}{\eta_p Q_p \mathbb{1}_{\mathsf{X}}} \\ \vdots \\ \dfrac{\Gamma_{3p+2}(Q_{3p+2} \overline{h}_{3(p+1)}, \overline{h}_{3p+2})}{\eta_p Q_p \mathbb{1}_{\mathsf{X}}} \end{array} \\ \hline * & \dfrac{\Gamma_{3p+2}(Q_p \overline{h}_{3(p+1)}, Q_p \overline{h}_{3(p+1)}) + \varsigma_{3(p+1)}^2(\overline{h}_{3(p+1)})}{(\eta_p Q_p \mathbb{1}_{\mathsf{X}})^2} \end{array} \right) \, ,$$

where

$$\varsigma^2_{3(p+1)}(\overline{h}_{3(p+1)}) = \eta_p\{R_p(w_p^2\overline{h}^2_{3(p+1)}) - (Q_p\overline{h}_{3(p+1)})^2\}\ , \tag{4.5.4}$$

$$\Gamma_{3p+2}(h_0,\dots,h_{3p+2}) = \left(\begin{array}{c|c} \Gamma_{3p+1}(h_0,\dots,h_{3p+1}) & \begin{array}{c}\Gamma_{3p+1}(\overline{h}_{3p+2},\overline{h}_0)\\ \vdots \\ \Gamma_{3p+1}(\overline{h}_{3p+2},\overline{h}_{3p+1})\end{array} \\ \hline * & \Gamma_{3p+1}(\overline{h}_{3p+2},\overline{h}_{3p+2}) + \varsigma^2_{3p+2}(\overline{h}_{3p+2}) \end{array}\right)\ ,$$

where

$$\varsigma^2_{3p+2}(\overline{h}_{3p+2}) = \eta_p\{\overline{h}^2_{3p+2}\}\ , \tag{4.5.5}$$

$$\Gamma_{3p+1}(h_0,\dots,h_{3p+1}) = \left(\begin{array}{c|c} \Gamma_{3p}(h_0,\dots,h_{3p}) & \begin{array}{c}\Gamma_{3p}(\overline{h}_{3p+1},\overline{h}_0)\\ \vdots \\ \Gamma_{3p}(\overline{h}_{3p+1},\overline{h}_{3p})\end{array} \\ \hline * & \Gamma_{3p}(\overline{h}_{3p+1},\overline{h}_{3p+1}) + \varsigma^2_{3p+1}(\overline{h}_{3p+1}) \end{array}\right)\ ,$$

where

$$\varsigma^2_{3p+1}(\overline{h}_{3p+1}) = \nu_p^2(h_{3p+1})\ . \tag{4.5.6}$$

*In the previous formulas, for all $(m,\ell)\in\mathbb{N}^2$, $\Gamma_n(\overline{h}_m,\overline{h}_\ell)$ denotes the asymptotic covariance between $\sqrt{N}\phi^N_m(\overline{h}_m)$ and $\sqrt{N}\phi^N_\ell(\overline{h}_\ell)$ and * is the transpose of the top right vector, as a covariance matrix is symmetric.*

Proof. We make a recursive proof. Define the filtration $(\mathcal{G}^N_k)_{k\geq 0}$, where for all $p\in\mathbb{N}$,

$$\begin{aligned}\mathcal{G}^N_{3p} &\stackrel{\text{def}}{=} \sigma\left(\{(\Omega^{(p)}_N(i),\{(\xi^{(p)}_N(i,j),\omega^{(p)}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}\right)\\ \mathcal{G}^N_{3p+1} &\stackrel{\text{def}}{=} \mathcal{G}^N_{3p}\vee\sigma\left(\{I^{(p)}_N(i)\}_{i=1}^{N_1}\right)\\ \mathcal{G}^N_{3p+2} &\stackrel{\text{def}}{=} \mathcal{G}^N_{3p+1}\vee\sigma\left(\{\{J^{(p)}_N(i,j))\}_{j=1}^{N_2}\}_{i=1}^{N_1}\right)\ .\end{aligned}$$

By definition, for all $k\in\mathbb{N}$, ϕ^N_k is $\mathcal{G}^N_k$-measurable. Let n be a non negative integer.

- If $n+1\equiv 0\,[3]$, there exists some $p\in\mathbb{N}$ such that $n+1=3(p+1)$. Write

$$\sqrt{N}\phi^N_{3(p+1)}(\overline{h}_{3(p+1)}) = \frac{\sqrt{N}\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\omega^{(p+1)}_N(i,j)\overline{h}_{3(p+1)}(\xi^{(p+1)}_N(i,j))/N_1N_2}{\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\omega^{(p+1)}_N(i,j)/N_1N_2}\ ,$$

  where the denominator converges in probability to $\eta_pQ_p\mathbb{1}_{\mathsf{X}}$. Write the numerator

$$\sum_{i=1}^{N_1}U_N(i) = \sum_{i=1}^{N_1}\{U_N(i)-\mathbb{E}\left[U_N(i)\mid\mathcal{G}^N_{3p+2}\right]\} + \sqrt{N}\phi^N_{3p+2}(Q_p\overline{h}_{3(p+1)})\ ,$$

  where

$$U_N(i)\stackrel{\text{def}}{=}\frac{1}{\sqrt{N_1N_2}}\sum_{j=1}^{N_2}\omega^{(p+1)}_N(i,j)\overline{h}_{3(p+1)}(\xi^{(p+1)}_N(i,j))\ .$$

  One may apply Corollary 4.6.2 in order to prove that

$$\left(\begin{array}{c}(\sqrt{N}\phi^N_k(\overline{h}_k))_{0\leq k\leq 3p+2}\\ \sum_{i=1}^{N_1}U_N(i)\end{array}\right)\xrightarrow[N\to+\infty]{\mathcal{D}}\mathcal{N}(0,\hat{\Gamma}_{3(p+1)}(h_0,\dots,h_{3(p+1)}))\ ,$$

where

$$\hat{\Gamma}_{3(p+1)}(h_0,\dots,h_{3(p+1)}) = \left(\begin{array}{c|c} \Gamma_{3p+2}(h_0,\dots,h_{3p+2}) & \begin{array}{c}\Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)},\overline{h}_0)\\ \vdots \\ \Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)},\overline{h}_{3p+2})\end{array} \\ \hline * & \Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)},Q_p\overline{h}_{3(p+1)}) + \varsigma^2_{3(p+1)}(\overline{h}_{3(p+1)}) \end{array}\right),$$

and $\varsigma^2_{3(p+1)}(\overline{h}_{3(p+1)})$ is defined in (4.6.4). First, (**B1**) and (**B2**) are respectively checked using the same lines as (3.6.23) and (3.6.24). Then, in order to check (**B4**), take $a = (a_0,\dots,a_{3(p+1)})^T \in \mathbb{R}^{3p+4}$, and notice that $\sum_{k=0}^{3p+2} a_k\sqrt{N}\phi^N_k(\overline{h}_k) + a_{3(p+1)}\sqrt{N}\phi^N_{3p+2}(Q_p\overline{h}_{3(p+1)}) = \sum_{k=0}^{3p+1} a_k\sqrt{N}\phi^N_k(\overline{h}_k) + \sqrt{N}\phi^N_{3p+2}(a_{3p+2}\overline{h}_{3p+2} + a_{3(p+1)}Q_p\overline{h}_{3(p+1)})$ converges in distribution using the Cramer'-Wold device and the recursive hypothesis. Hence, we have

$$\begin{pmatrix}(\sqrt{N}\phi^N_k(\overline{h}_k))_{0\le k\le 3p+2}\\ \sqrt{N}\phi^N_{3p+2}(Q_p\overline{h}_{3(p+1)})\end{pmatrix} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}(0,\tilde{\Gamma}_{3(p+1)}(h_0,\dots,h_{3(p+1)}))\,, \tag{4.5.7}$$

where the covariance matrix is

$$\tilde{\Gamma}_{3(p+1)}(h_0,\dots,h_{3(p+1)}) = \left(\begin{array}{c|c} \Gamma_{3p+2}(h_0,\dots,h_{3p+2}) & \begin{array}{c}\Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)},\overline{h}_0)\\ \vdots \\ \Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)},\overline{h}_{3p+2})\end{array} \\ \hline * & \Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)},Q_p\overline{h}_{3(p+1)}) \end{array}\right).$$

Finally, one may apply the generalization of Slutsky's Lemma 4.6.3 to conclude the case $n+1 \equiv 0\,[3]$.

- If $n+1 \equiv 2\,[3]$, there exists some $p \ge 0$ such that $n+1 = 3p+2$. Write

$$\sqrt{N}\phi^N_{3p+2}(\overline{h}_{3p+2}) = \sum_{i=1}^{N_1}\{U_N(i) - \mathbb{E}\left[U_N(i)\mid \mathcal{G}^N_{3p+1}\right]\} + \sqrt{N}\phi^N_{3p+1}(\overline{h}_{3p+2})$$

where

$$U_N(i) \overset{\text{def}}{=} \frac{1}{\sqrt{N_1N_2}}\sum_{j=1}^{N_2}\overline{h}_{3p+2}(\xi^{(p)}_N(I^{(p)}_N(i),J^{(p)}_N(i,j)))\,.$$

(**B1**) and (**B2**) are respectively checked using the same lines as (3.6.11) and (3.6.12). Assumption (**B4**) holds as an immediate consequence of the recursive hypothesis

$$\begin{pmatrix}(\sqrt{N}\phi^N_k(\overline{h}_k))_{0\le k\le 3p+1}\\ \sqrt{N}\phi^N_{3p+1}(\overline{h}_{3p+2})\end{pmatrix} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}\left(0,\left(\begin{array}{c|c} \Gamma_{3p+1}(h_0,\dots,h_{3p+1}) & \begin{array}{c}\Gamma_{3p+1}(\overline{h}_{3p+2},\overline{h}_0)\\ \vdots \\ \Gamma_{3p+1}(\overline{h}_{3p+2},\overline{h}_{3p+1})\end{array} \\ \hline * & \Gamma_{3p+1}(\overline{h}_{3p+2},\overline{h}_{3p+2}) \end{array}\right)\right).$$

Hence, we apply Corollary 4.6.2 to conclude the case $n+1 \equiv 2\,[3]$.

- If $n+1 \equiv 1\,[3]$, there exists some $p \ge 0$ such that $n+1 = 3p+1$. Write

$$\sqrt{N}\phi^N_{3p+1}(\overline{h}_{3p+1}) = \sum_{i=1}^{N_1}\{U_N(i) - \mathbb{E}\left[U_N(i)\mid \mathcal{G}^N_{3p}\right]\} + \sqrt{N}\phi^N_{3p}(\overline{h}_{3p+1})$$

where

$$U_N(i) \overset{\text{def}}{=} \frac{\sqrt{N}}{N_1} \sum_{j=1}^{N_2} \frac{\omega_N^{(p)}(I_N^{(p)}(i), j)}{\sum_{j'=1}^{N_2} \omega_N^{(p)}(I_N^{(p)}(i), j')} \overline{h}_{3p+1}(\xi_N^{(p)}(I_N^{(p)}(i), j)) \, .$$

(**B1**) and (**B2**) are respectively checked using the same lines as (3.6.5), (3.6.6), and (3.6.7). Assumption (**B4**) holds as an immediate consequence of the recursive hypothesis

$$\begin{pmatrix} (\sqrt{N}\phi_k^N(\overline{h}_k))_{0\le k\le 3p} \\ \sqrt{N}\phi_{3p}^N(\overline{h}_{3p+1}) \end{pmatrix} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}\left(0, \left(\begin{array}{c|c} \Gamma_{3p}(h_0, \ldots, h_{3p}) & \begin{matrix} \Gamma_{3p}(\overline{h}_{3p+1}, \overline{h}_0) \\ \vdots \\ \Gamma_{3p}(\overline{h}_{3p+1}, \overline{h}_{3p}) \end{matrix} \\ \hline * & \Gamma_{3p}(\overline{h}_{3p+1}, \overline{h}_{3p+1}) \end{array}\right)\right) \, . \tag{4.5.8}$$

Hence, we apply Corollary 4.6.2 to conclude the case $n+1 \equiv 1\,[3]$.

□

4.5.2 Technical lemmas

First, we prove a corollary of Theorem 3.5.2 that is used in the proof of the joint CLT Theorem 4.6.1.

Corollary 4.5.2. *Let $(\Omega, \mathcal{A}, \{\mathcal{F}_N\}_{N\in\mathbb{N}}, \mathbb{P})$ be a filtered probability space. In addition, for a given sequence $\{M_N\}_{N\in\mathbb{N}}$ of integers such that $M_N \to \infty$ as $N \to \infty$, let $\{U_N(i)\}_{i=1}^{M_N}$, $N \in \mathbb{N}$, be a triangular array of random variables on $(\Omega, \mathcal{A}, \mathbb{P})$ such that for all $N \in \mathbb{N}$, the variables $\{U_N(i)\}_{i=1}^{M_N}$ are conditionally independent given $\mathcal{F}_N$ with $\mathbb{E}[U_N(i)^2 \mid \mathcal{F}_N] < \infty$, $\mathbb{P}$-a.s., for all $i \in [\![1, M_N]\!]$. Let $\{V_N(i)\}_{i=1}^{M_N}$ be a triangular array of random variables on $(\Omega, \mathcal{A}, \mathbb{P})$ such that $V_N(i)$ is $\mathcal{F}_N$-measurable, for all $i \in [\![1, M_N]\!]$. Moreover, assume that*
*(**B1**) For some constant $\varsigma^2 > 0$, as $N \to \infty$,*

$$\sum_{i=1}^{M_N} \left(\mathbb{E}[U_N(i)^2 \mid \mathcal{F}_N] - \mathbb{E}^2\left[U_N(i) \mid \mathcal{F}_N\right]\right) \xrightarrow{\mathbb{P}} \varsigma^2 \, .$$

*(**B2**) For all $\varepsilon > 0$, as $N \to \infty$,*

$$\sum_{i=1}^{M_N} \mathbb{E}\left[U_N(i)^2 \mathbb{1}\{|U_N(i)| \ge \varepsilon\} \mid \mathcal{F}_N\right] \xrightarrow{\mathbb{P}} 0 \, .$$

(B3) $$\begin{pmatrix} \sum_{k=1}^{M_N} V_N(k) \\ \sum_{k=1}^{M_N} \mathbb{E}\left[U_N(k) \mid \mathcal{F}_N\right] \end{pmatrix} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}(0, \Gamma)$$

Then,

$$\begin{pmatrix} \sum_{k=1}^{M_N} V_N(k) \\ \sum_{k=1}^{M_N} U_N(k) \end{pmatrix} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}\left(0, \Gamma + \begin{pmatrix} 0 & 0 \\ 0 & \varsigma^2 \end{pmatrix}\right) \, .$$

Proof. One may use the Cramer'-Wold device. For all $(a, b) \in \mathbb{R}^2$,

$$\begin{aligned} &\mathbb{E}\left[e^{i\{a\sum_{k=1}^{M_N} V_N(k) + b\sum_{k=1}^{M_N} U_N(k)\}}\right] \\ &= \mathbb{E}\left[e^{i\{a\sum_{k=1}^{M_N} V_N(k) + b\sum_{k=1}^{M_N} \mathbb{E}[U_N(k)|\mathcal{F}_N]\}} \, \mathbb{E}\left[e^{ib\{\sum_{k=1}^{M_N}\{U_N(k) - \mathbb{E}[U_N(k)|\mathcal{F}_N]\}\}} \mid \mathcal{F}_N\right]\right] \end{aligned}$$

which converges using Theorem 3.5.2 and Lemma 3.5.5. □

Then, recall a generalization of Slutsky's lemma for random vectors.

Lemma 4.5.3. *Let $X^N = (X_1^N, \dots, X_n^N)^T$ be a sequence of random vectors which converges in distribution to a random vector X, and D^N be a n-by-n random diagonal matrix which converges in probability to a deterministic n-by-n diagonal matrix $D \in \mathbb{D}_n(\mathbb{R})$. Then, $D^N X^N$ converges in distribution to DX.*

Proof. The proof is based on the Cramer'-Wold device. For all $a = (a_1, \dots, a_n)^T \in \mathbb{R}^n$, $a^T D^N X^N = \sum_{k=1}^n a_k D_k^N X_k^N \xrightarrow[N\to+\infty]{\mathcal{D}} \sum_{k=1}^n a_k D_k X_k = a^T D X$, using Slutsky's lemma and the Cramer'-Wold theorem applied to X^N. □

4.6 Some additional convergence result for the B² filter algorithm

4.6.1 Joint central limit theorem for the B² filter algorithm

The following theorem extends property (**AN1**), giving a joint CLT. Denote by ϕ_n^N the filter estimator produced by the B² algorithm (cf Algorithm 4), where for all $p \geq 0$,

$$\phi_{3p}^N(h_{3p}) \stackrel{\text{def}}{=} \sum_{i=1}^{N_1} \frac{\Omega_N^{(p)}(i)}{\sum_{i'=1}^{N_1} \Omega_N^{(p)}(i')} \sum_{j=1}^{N_2} \frac{\omega_N^{(p)}(i,j)}{\sum_{j'=1}^{N_2} \omega_N^{(p)}(i,j')} h_{3p}(\xi_N^{(p)}(i,j)) \;, \tag{4.6.1}$$

$$\phi_{3p+1}^N(h_{3p+1}) \stackrel{\text{def}}{=} \frac{1}{N_1} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \frac{\omega_N^{(p)}(I_N^{(p)}(i),j)}{\sum_{j'=1}^{N_2} \omega_N^{(p)}(I_N^{(p)}(i),j')} h_{3p+1}(\xi_N^{(p)}(I_N^{(p)}(i),j)) \;, \tag{4.6.2}$$

$$\phi_{3p+2}^N(h_{3p+2}) \stackrel{\text{def}}{=} \frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} h_{3p+2}(\xi_N^{(p)}(I_N^{(p)}(i), J_N^{(p)}(i,j))) \;, \tag{4.6.3}$$

and by $\overline{h}_n$ the centered functions, where for all $p \in \mathbb{N}$,

$$\begin{aligned} \overline{h}_{3p} &\stackrel{\text{def}}{=} h_{3p} - \eta_p h_{3p} \;, \\ \overline{h}_{3p+1} &\stackrel{\text{def}}{=} h_{3p+1} - \eta_p h_{3p+1} \;, \\ \overline{h}_{3p+2} &\stackrel{\text{def}}{=} h_{3p+2} - \eta_p h_{3p+2} \;. \end{aligned}$$

We establish the joint convergence in distribution of the vector $(\sqrt{N}\phi_k^N(\overline{h}_k))_{0\leq k\leq n}$ for all $n \in \mathbb{N}$.

Theorem 4.6.1. *For the B² filter, for all $n \in \mathbb{N}$ and all functions $(h_0, \dots, h_n) \in \mathsf{F}_{\mathrm{b}}(\mathsf{X})^{n+1}$,*

$$\left(\sqrt{N}\phi_k^N(\overline{h}_k)\right)_{0\leq k\leq n} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}(0, \Gamma_n(h_0, \dots, h_n)) \;,$$

where the asymptotic covariance matrix Γ_n is recursively given by $\Gamma_0(\overline{h}_0, \overline{h}_0) \stackrel{\text{def}}{=} \sigma_0^2 h_0 = \eta_0(\overline{h}_0^2)$ and for all $p \geq 0$,

$$\Gamma_{3(p+1)}(h_0, \dots, h_{3(p+1)}) = \left(\begin{array}{c|c} \Gamma_{3p+2}(h_0, \dots, h_{3p+2}) & \begin{array}{c} \frac{\Gamma_{3p+2}(Q_p \overline{h}_{3(p+1)}, \overline{h}_0)}{\eta_p Q_p \mathbb{1}_{\mathsf{X}}} \\ \vdots \\ \frac{\Gamma_{3p+2}(Q_{3p+2} \overline{h}_{3(p+1)}, \overline{h}_{3p+2})}{\eta_p Q_p \mathbb{1}_{\mathsf{X}}} \end{array} \\ \hline * & \frac{\Gamma_{3p+2}(Q_p \overline{h}_{3(p+1)}, Q_p \overline{h}_{3(p+1)}) + \varsigma_{3(p+1)}^2(\overline{h}_{3(p+1)})}{(\eta_p Q_p \mathbb{1}_{\mathsf{X}})^2} \end{array}\right) \;,$$

where

$$\varsigma_{3(p+1)}^2(\overline{h}_{3(p+1)}) = \eta_p\{R_p(w_p^2 \overline{h}_{3(p+1)}^2) - (Q_p \overline{h}_{3(p+1)})^2\} \;, \tag{4.6.4}$$

$$\Gamma_{3p+2}(h_0,\dots,h_{3p+2}) = \left(\begin{array}{c|c} \Gamma_{3p+1}(h_0,\dots,h_{3p+1}) & \begin{array}{c}\Gamma_{3p+1}(\overline{h}_{3p+2},\overline{h}_0)\\ \vdots \\ \Gamma_{3p+1}(\overline{h}_{3p+2},\overline{h}_{3p+1})\end{array} \\ \hline * & \Gamma_{3p+1}(\overline{h}_{3p+2},\overline{h}_{3p+2}) + \varsigma^2_{3p+2}(\overline{h}_{3p+2}) \end{array}\right) ,$$

where

$$\varsigma^2_{3p+2}(\overline{h}_{3p+2}) = \eta_p\{\overline{h}^2_{3p+2}\} , \tag{4.6.5}$$

$$\Gamma_{3p+1}(h_0,\dots,h_{3p+1}) = \left(\begin{array}{c|c} \Gamma_{3p}(h_0,\dots,h_{3p}) & \begin{array}{c}\Gamma_{3p}(\overline{h}_{3p+1},\overline{h}_0)\\ \vdots \\ \Gamma_{3p}(\overline{h}_{3p+1},\overline{h}_{3p})\end{array} \\ \hline * & \Gamma_{3p}(\overline{h}_{3p+1},\overline{h}_{3p+1}) + \varsigma^2_{3p+1}(\overline{h}_{3p+1}) \end{array}\right) ,$$

where

$$\varsigma^2_{3p+1}(\overline{h}_{3p+1}) = \nu^2_p(h_{3p+1}) . \tag{4.6.6}$$

*In the previous formulas, for all $(m,\ell)\in\mathbb{N}^2$, $\Gamma_n(\overline{h}_m,\overline{h}_\ell)$ denotes the asymptotic covariance between $\sqrt{N}\phi^N_m(\overline{h}_m)$ and $\sqrt{N}\phi^N_\ell(\overline{h}_\ell)$ and * is the transpose of the top right vector, as a covariance matrix is symmetric.*

Proof. We make a recursive proof. Define the filtration $(\mathcal{G}^N_k)_{k\geq 0}$, where for all $p\in\mathbb{N}$,

$$\begin{aligned}
\mathcal{G}^N_{3p} &\stackrel{\text{def}}{=} \sigma\left(\{(\Omega^{(p)}_N(i),\{(\xi^{(p)}_N(i,j),\omega^{(p)}_N(i,j))\}_{j=1}^{N_2})\}_{i=1}^{N_1}\right)\\
\mathcal{G}^N_{3p+1} &\stackrel{\text{def}}{=} \mathcal{G}^N_{3p}\vee\sigma\left(\{I^{(p)}_N(i)\}_{i=1}^{N_1}\right)\\
\mathcal{G}^N_{3p+2} &\stackrel{\text{def}}{=} \mathcal{G}^N_{3p+1}\vee\sigma\left(\{\{J^{(p)}_N(i,j))\}_{j=1}^{N_2}\}_{i=1}^{N_1}\right) .
\end{aligned}$$

By definition, for all $k\in\mathbb{N}$, ϕ^N_k is $\mathcal{G}^N_k$-measurable. Let n be a non negative integer.

- If $n+1\equiv 0\,[3]$, there exists some $p\in\mathbb{N}$ such that $n+1=3(p+1)$. Write

  $$\sqrt{N}\phi^N_{3(p+1)}(\overline{h}_{3(p+1)}) = \frac{\sqrt{N}\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\omega^{(p+1)}_N(i,j)\overline{h}_{3(p+1)}(\xi^{(p+1)}_N(i,j))/N_1N_2}{\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\omega^{(p+1)}_N(i,j)/N_1N_2} ,$$

  where the denominator converges in probability to $\eta_pQ_p\mathbb{1}_{\mathsf{X}}$. Write the numerator

  $$\sum_{i=1}^{N_1}U_N(i) = \sum_{i=1}^{N_1}\{U_N(i)-\mathbb{E}\left[U_N(i)\mid\mathcal{G}^N_{3p+2}\right]\}+\sqrt{N}\phi^N_{3p+2}(Q_p\overline{h}_{3(p+1)}) ,$$

  where

  $$U_N(i)\stackrel{\text{def}}{=}\frac{1}{\sqrt{N_1N_2}}\sum_{j=1}^{N_2}\omega^{(p+1)}_N(i,j)\overline{h}_{3(p+1)}(\xi^{(p+1)}_N(i,j)) .$$

  One may apply Corollary 4.6.2 in order to prove that

  $$\left(\begin{array}{c}(\sqrt{N}\phi^N_k(\overline{h}_k))_{0\leq k\leq 3p+2}\\ \sum_{i=1}^{N_1}U_N(i)\end{array}\right)\xrightarrow[N\to+\infty]{\mathcal{D}}\mathcal{N}(0,\hat{\Gamma}_{3(p+1)}(h_0,\dots,h_{3(p+1)})) ,$$

  where

  $$\hat{\Gamma}_{3(p+1)}(h_0,\dots,h_{3(p+1)}) = \left(\begin{array}{c|c} \Gamma_{3p+2}(h_0,\dots,h_{3p+2}) & \begin{array}{c}\Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)},\overline{h}_0)\\ \vdots \\ \Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)},\overline{h}_{3p+2})\end{array} \\ \hline * & \Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)},Q_p\overline{h}_{3(p+1)}) + \varsigma^2_{3(p+1)}(\overline{h}_{3(p+1)}) \end{array}\right) ,$$

and $\varsigma^2_{3(p+1)}(\overline{h}_{3(p+1)})$ is defined in (4.6.4). First, (**B1**) and (**B2**) are respectively checked using the same lines as (3.6.23) and (3.6.24). Then, in order to check (**B4**), take $a = (a_0, \dots, a_{3(p+1)})^T \in \mathbb{R}^{3p+4}$, and notice that $\sum_{k=0}^{3p+2} a_k \sqrt{N}\phi_k^N(\overline{h}_k) + a_{3(p+1)}\sqrt{N}\phi^N_{3p+2}(Q_p\overline{h}_{3(p+1)}) = \sum_{k=0}^{3p+1} a_k\sqrt{N}\phi_k^N(\overline{h}_k) + \sqrt{N}\phi^N_{3p+2}(a_{3p+2}\overline{h}_{3p+2} + a_{3(p+1)}Q_p\overline{h}_{3(p+1)})$ converges in distribution using the Cramer'-Wold device and the recursive hypothesis. Hence, we have

$$\begin{pmatrix} (\sqrt{N}\phi_k^N(\overline{h}_k))_{0\le k\le 3p+2} \\ \sqrt{N}\phi^N_{3p+2}(Q_p\overline{h}_{3(p+1)}) \end{pmatrix} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}(0, \tilde{\Gamma}_{3(p+1)}(h_0, \dots, h_{3(p+1)}))\,, \tag{4.6.7}$$

where the covariance matrix is

$$\tilde{\Gamma}_{3(p+1)}(h_0, \dots, h_{3(p+1)}) = \left(\begin{array}{c|c} \multirow{2}{*}{$\Gamma_{3p+2}(h_0, \dots, h_{3p+2})$} & \Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)}, \overline{h}_0) \\ & \vdots \\ & \Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)}, \overline{h}_{3p+2}) \\ \hline * & \Gamma_{3p+2}(Q_p\overline{h}_{3(p+1)}, Q_p\overline{h}_{3(p+1)}) \end{array}\right)\,.$$

Finally, one may apply the generalization of Slutsky's Lemma 4.6.3 to conclude the case $n+1 \equiv 0\,[3]$.

- If $n+1 \equiv 2\,[3]$, there exists some $p \ge 0$ such that $n+1 = 3p+2$. Write

$$\sqrt{N}\phi^N_{3p+2}(\overline{h}_{3p+2}) = \sum_{i=1}^{N_1}\{U_N(i) - \mathbb{E}\left[U_N(i) \mid \mathcal{G}^N_{3p+1}\right]\} + \sqrt{N}\phi^N_{3p+1}(\overline{h}_{3p+2})$$

where

$$U_N(i) \overset{\text{def}}{=} \frac{1}{\sqrt{N_1N_2}}\sum_{j=1}^{N_2}\overline{h}_{3p+2}(\xi_N^{(p)}(I_N^{(p)}(i), J_N^{(p)}(i,j)))\,.$$

(**B1**) and (**B2**) are respectively checked using the same lines as (3.6.11) and (3.6.12). Assumption (**B4**) holds as an immediate consequence of the recursive hypothesis

$$\begin{pmatrix} (\sqrt{N}\phi_k^N(\overline{h}_k))_{0\le k\le 3p+1} \\ \sqrt{N}\phi^N_{3p+1}(\overline{h}_{3p+2}) \end{pmatrix} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}\left(0, \left(\begin{array}{c|c} \multirow{2}{*}{$\Gamma_{3p+1}(h_0, \dots, h_{3p+1})$} & \Gamma_{3p+1}(\overline{h}_{3p+2}, \overline{h}_0) \\ & \vdots \\ & \Gamma_{3p+1}(\overline{h}_{3p+2}, \overline{h}_{3p+1}) \\ \hline * & \Gamma_{3p+1}(\overline{h}_{3p+2}, \overline{h}_{3p+2}) \end{array}\right)\right)\,.$$

Hence, we apply Corollary 4.6.2 to conclude the case $n+1 \equiv 2\,[3]$.

- If $n+1 \equiv 1\,[3]$, there exists some $p \ge 0$ such that $n+1 = 3p+1$. Write

$$\sqrt{N}\phi^N_{3p+1}(\overline{h}_{3p+1}) = \sum_{i=1}^{N_1}\{U_N(i) - \mathbb{E}\left[U_N(i) \mid \mathcal{G}^N_{3p}\right]\} + \sqrt{N}\phi^N_{3p}(\overline{h}_{3p+1})$$

where

$$U_N(i) \overset{\text{def}}{=} \frac{\sqrt{N}}{N_1}\sum_{j=1}^{N_2}\frac{\omega_N^{(p)}(I_N^{(p)}(i), j)}{\sum_{j'=1}^{N_2}\omega_N^{(p)}(I_N^{(p)}(i), j')}\overline{h}_{3p+1}(\xi_N^{(p)}(I_N^{(p)}(i), j))\,.$$

(**B1**) and (**B2**) are respectively checked using the same lines as (3.6.5), (3.6.6), and (3.6.7). Assumption (**B4**) holds as an immediate consequence of the recursive hypothesis

$$\begin{pmatrix} (\sqrt{N}\phi_k^N(\overline{h}_k))_{0\le k\le 3p} \\ \sqrt{N}\phi_{3p}^N(\overline{h}_{3p+1}) \end{pmatrix} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}\left(0, \left(\begin{array}{c|c} \Gamma_{3p}(h_0,\dots,h_{3p}) & \begin{array}{c}\Gamma_{3p}(\overline{h}_{3p+1},\overline{h}_0)\\ \vdots \\ \Gamma_{3p}(\overline{h}_{3p+1},\overline{h}_{3p})\end{array} \\ \hline * & \Gamma_{3p}(\overline{h}_{3p+1},\overline{h}_{3p+1}) \end{array}\right)\right) . \tag{4.6.8}$$

Hence, we apply Corollary 4.6.2 to conclude the case $n+1 \equiv 1\,[3]$.

□

4.6.2 Technical lemmas

First, we prove a corollary of Theorem 3.5.2 that is used in the proof of the joint CLT Theorem 4.6.1.

Corollary 4.6.2. *Let $(\Omega, \mathcal{A}, \{\mathcal{F}_N\}_{N\in\mathbb{N}}, \mathbb{P})$ be a filtered probability space. In addition, for a given sequence $\{M_N\}_{N\in\mathbb{N}}$ of integers such that $M_N \to \infty$ as $N \to \infty$, let $\{U_N(i)\}_{i=1}^{M_N}$, $N \in \mathbb{N}$, be a triangular array of random variables on $(\Omega, \mathcal{A}, \mathbb{P})$ such that for all $N \in \mathbb{N}$, the variables $\{U_N(i)\}_{i=1}^{M_N}$ are conditionally independent given $\mathcal{F}_N$ with $\mathbb{E}[U_N(i)^2 \mid \mathcal{F}_N] < \infty$, $\mathbb{P}$-a.s., for all $i \in [\![1, M_N]\!]$. Let $\{V_N(i)\}_{i=1}^{M_N}$ be a triangular array of random variables on $(\Omega, \mathcal{A}, \mathbb{P})$ such that $V_N(i)$ is $\mathcal{F}_N$-measurable, for all $i \in [\![1, M_N]\!]$. Moreover, assume that*
*(**B1**) For some constant $\varsigma^2 > 0$, as $N \to \infty$,*

$$\sum_{i=1}^{M_N} \left(\mathbb{E}[U_N(i)^2 \mid \mathcal{F}_N] - \mathbb{E}^2\left[U_N(i) \mid \mathcal{F}_N\right]\right) \xrightarrow{\mathbb{P}} \varsigma^2 .$$

*(**B2**) For all $\varepsilon > 0$, as $N \to \infty$,*

$$\sum_{i=1}^{M_N} \mathbb{E}\left[U_N(i)^2 \mathbb{1}\{|U_N(i)| \ge \varepsilon\} \mid \mathcal{F}_N\right] \xrightarrow{\mathbb{P}} 0 .$$

(**B4**) $$\begin{pmatrix} \sum_{k=1}^{M_N} V_N(k) \\ \sum_{k=1}^{M_N} \mathbb{E}\left[U_N(k) \mid \mathcal{F}_N\right] \end{pmatrix} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}(0, \Gamma)$$

Then,

$$\begin{pmatrix} \sum_{k=1}^{M_N} V_N(k) \\ \sum_{k=1}^{M_N} U_N(k) \end{pmatrix} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}\left(0, \Gamma + \begin{pmatrix} 0 & 0 \\ 0 & \varsigma^2 \end{pmatrix}\right) .$$

Proof. One may use the Cramer'-Wold device. For all $(a, b) \in \mathbb{R}^2$,

$$\begin{aligned} &\mathbb{E}\left[e^{i\{a\sum_{k=1}^{M_N} V_N(k) + b\sum_{k=1}^{M_N} U_N(k)\}}\right] \\ &= \mathbb{E}\left[e^{i\{a\sum_{k=1}^{M_N} V_N(k) + b\sum_{k=1}^{M_N} \mathbb{E}[U_N(k)|\mathcal{F}_N]\}}\, \mathbb{E}\left[e^{ib\{\sum_{k=1}^{M_N}\{U_N(k) - \mathbb{E}[U_N(k)|\mathcal{F}_N]\}\}} \mid \mathcal{F}_N\right]\right] \end{aligned}$$

which converges using Theorem 3.5.2 and Lemma 3.5.5. □

Then, recall a generalization of Slutsky's lemma for random vectors.

Lemma 4.6.3. *Let $X^N = (X_1^N, \dots, X_n^N)^T$ be a sequence of random vectors which converges in distribution to a random vector X, and D^N be a n-by-n random diagonal matrix which converges in probability to a deterministic n-by-n diagonal matrix $D \in \mathbb{D}_n(\mathbb{R})$. Then, $D^N X^N$ converges in distribution to DX.*

Proof. The proof is based on the Cramer'-Wold device. For all $a = (a_1, \ldots, a_n)^T \in \mathbb{R}^n$, $a^T D^N X^N = \sum_{k=1}^n a_k D_k^N X_k^N \xrightarrow[N \to +\infty]{\mathcal{D}} \sum_{k=1}^n a_k D_k X_k = a^T DX$, using Slutsky's lemma and the Cramer'-Wold theorem applied to X^N. □

Chapitre 5

Un exemple d'algorithme d'îlots de particules pour l'analyse d'évènements rares

Sommaire

5.1 Introduction . . . 119
5.2 General problem . . . 121
5.3 The Particle Markov chain Monte Carlo (PMCMC) algorithm . . . 122
5.3.1 The Metropolis-Hastings (MH) algorithm . . . 122
5.3.2 The splitting algorithm . . . 123
5.3.3 The PMCMC algorithm . . . 126
5.4 The *sequential Monte Carlo square* (SMC^2) algorithm . . . 127
5.4.1 Principle . . . 127
5.4.2 The *iterated batch importance sampling* (IBIS) algorithm . . . 128
5.4.3 The SMC^2 algorithm . . . 130
5.5 Application on different test cases . . . 132
5.5.1 Gaussian toy case . . . 132
5.5.2 Estimation of launch vehicle booster fallout zone . . . 133
5.5.3 Estimation of collision probability between orbiting objects . . . 137
5.6 Conclusion . . . 140

La sûreté et la fiabilité sont au centre des préoccupations dans plusieurs domaines tels que l'aérospatial [Morio, 2011a], l'aéronautique [Prandini and Watkins, 2005], la génétique [Rosenbluth and Rosenbluth, 1955], la finance [Richard and Zhang, 2007; Glasserman et al., 2000], l'assurance [Asmussen, 2003], les phénomènes géologiques [Asmussen, 2003; Kagan and Knopoff, 1977], les réseaux informatiques [Parekh and Walrand, 1989] ou le nucléaire [Sjöberg and Drottz-Sjöberg, 1991]. Les règles de sécurité et de prévention dans ces différents domaines sont tels que le risque de défaillance des systèmes doit être extrêmement faible. Ce risque doit être évalué et nécessite de pouvoir calculer des probabilités d'évènements rares.

La technique la plus connue pour estimer une probabilité p est la méthode de Monte-Carlo classique (abrégé CMC pour *crude Monte Carlo*) qui consiste à approcher la probabilité p par la moyenne empirique d'un échantillon de N variables aléatoires i.i.d. selon une loi de Bernoulli. Le CMC est simple à implémenter et fournit un estimateur sans biais et consistant de la probabilité d'intérêt p. L'erreur relative (qui est le rapport entre l'écart type de l'estimateur et sa moyenne) mesure la variabilité de l'estimation par rapport à la quantité que l'on cherche à estimer, et constitue un indicateur important de la qualité de l'estimation. L'erreur relative du CMC est de l'ordre de $1/\sqrt{Np}$. Ainsi, pour estimer une probabilité faible $p = 10^{-k}$ ($k \in \mathbb{N}$) avec une erreur relative de 10% en utilisant le CMC il faut un échantillon de grande taille $N = 10^{k+2}$. C'est pourquoi, dans le cadre des évènements rares, le CMC est inefficace et de nombreuses techniques de réduction de variance ont été proposées pour estimer des faibles probabilités depuis les années 1950. L'*échantillonnage préférentiel* [Engelund and Rackwitz, 1993; Orsak and Aazhang, 1995; Richard and Zhang, 2007; Kroese and Rubinstein, 2012] est à la base de plusieurs méthodes telles que la transformation d'Esscher pour les processus de Poisson composés [Asmussen, 2003; Glasserman, 2004; Blanchet et al., 2009], les transformations des vecteurs aléatoires Gaussiens [Jourdain and Lelong, 2009], les méthodes de Monte-Carlo séquentielles (SMC) [Rosenbluth and Rosenbluth, 1955; Gordon et al., 1993; Doucet et al., 2001], *etc.* Le *splitting* est un exemple de méthodes SMC et a été considérablement étudié et amélioré ces dernières décennies, voir les nombreuses références [Kahn and Harris, 1951], [Rubinstein, 1981], [Fishman, 1995], [Melas, 1997], [Glasserman et al., 1999], [Garvels, 2000], [Au and Beck, 2001], [Cérou et al., 2006], [Lagnoux, 2006; Lagnoux-Renaudie, 2009], [Cérou and Guyader, 2007; Cérou et al., 2012], [Au et al., 2007], [Gogate and Dechter, 2011]. Différentes techniques comme les méthodes de Monte-Carlo par chaînes de Markov (MCMC), FORM/SORM (pour *first/second order reliability methods*) [Bjerager, 1991; Madsen et al., 1986; Yan-Gang and Tetsuro, 1999], ou la théorie des valeurs extrêmes [Fisher and Tippett, 1928; Leadbetter et al., 1983; Leadbetter and Rootzén, 1988] sont aussi utilisées pour estimer des probabilités d'évènements rares. Leurs principes et leurs avantages et inconvénients ont également été largement étudiés, voir par exemple l'article [Morio et al., 2014].

Dans cette section, nous nous focalisons sur des évènements rares qui se modélisent sous la forme d'un dépassement de seuil de la sortie d'une fonction scalaire déterministe. Cette fonction entrée-sortie souvent vue comme une "boîte noire" possède des entrées aléatoires. Certains paramètres, notés dans la suite par un vecteur Θ, dans les fonctions de type "boîtes noires" peuvent être fixés, comme les paramètres du modèle ou les paramètres des lois des variables d'entrées du modèle, et leur valeur influence l'estimation de la probabilité de l'évènement rare. Ces hypothèses sont souvent faites pour des raisons de simplicité et de réduction de coût de calculs. Du point de vue de l'analyse de risques, il est intéressant de déterminer la variabilité des sorties du code par rapport aux incertitudes liées aux paramètres d'entrée Θ ou par rapport à un paramètre particulier, et de quantifier l'impact d'un tel paramétrage sur la réalisation de l'évènement critique. Bien sûr, différentes valeurs des paramètres Θ peuvent fortement modifier l'estimation de la probabilité de l'évènement critique. Il ne s'agit pas seulement de tester tous les paramétrages possibles, mais aussi de prendre en compte leur vraisemblance. Un sujet de préoccupation en sûreté est de ne pas sous-estimer le risque à cause d'un mauvais réglage des paramètres. C'est pourquoi, nous proposons dans cette section d'estimer les lois des paramètres conditionnellement à un évènement rare, et nous exposerons deux algorithmes à cette fin.

Des études sur des sujets proches ont été récemment proposées, principalement basées sur de l'analyse

de sensibilité. L'indice de fiabilité des méthodes FORM/SORM [Lemaire et al., 2009] peut être utilisé pour analyser l'influence des paramètres d'entrée sur la probabilité de défaillance. L'échantillonnage stratifié [Munoz Zuniga et al., 2011] et l'échantillonnage préférentiel [Morio, 2011a] ont été adapté dans le même but, voir [Beck and Arnold, 1977] pour une présentation détaillée des méthodes d'estimation de paramètres. L'analyse de sensibilité des paramètres concerne en particulier l'analyse phylogénétique [Wheeler, 1995], la biologie des systèmes [Cho et al., 2003], le design des bassins hydrologiques [Beven and Kirkby, 1979], *etc.*

L'algorithme SMC^2 introduit par Chopin et al. [2013] est un exemple d'algorithme SMC permettant de simuler selon les lois *a posteriori* des paramètres et des états latents dans les modèles de Markov cachés. Une approche idéale serait d'utiliser un système de particules sur l'espace des paramètres Θ (comme l'algorithme IBIS de Chopin [2002] pour l'inférence bayésienne). Mais les incréments de vraisemblances étant la plupart du temps incalculables, ils sont remplacés par des estimateurs sans biais, obtenus pour tout paramètre fixé par un autre algorithme SMC sur l'espace d'état X. Deux algorithmes SMC étant imbriqués (un avec N_θ particules en θ et l'autre avec N_x particules en X), cet algorithme s'appelle SMC^2. Malgré l'approximation des vraisemblances, l'algorithme SMC^2 cible les lois d'intérêt: les lois *a posteriori* au fur et à mesure que les observations arrivent. Les étapes de mise à jour (mutation), nécessaires à tout algorithme SMC appliqué aux problèmes statiques, sont assurées par des noyaux de Markov introduits récemment par les méthodes dites de Monte-Carlo à chaîne de Markov particulaires (PMCMC). La méthode n'est cependant pas en ligne puisque son coût computationnel augmente avec le nombre d'observations, en raison de l'étape de mise à jour. Cet algorithme peut être vu comme un algorithme SMC sur l'espace étendu $\Theta \times \mathsf{X}^{N_x}$. De cette façon, on prouve la consistance de l'estimateur induit par le SMC^2 quand le nombre N_θ de particules en θ tend vers l'infini. L'analyse de la complexité algorithmique du SMC^2 a été faite dans Chopin et al. [2013] sous des hypothèses fortes, qui sont rarement vérifiées en pratique. Une étude théorique de la variance de cet estimateur en fonction du nombre d'observations n'a pas encore été menée et viendrait justifier théoriquement l'utilisation de la méthode SMC^2 pour l'estimation des lois *a posteriori* dans les modèles de Markov cachés.

En analyse de sensibilité, un paramètre est influent quand ses variations augmentent ou diminuent la probabilité d'un évènement rare. Dans cette section, nous nous focalisons plus précisément sur les réglages des paramètres qui augmentent la probabilité de l'évènement critique, ce qui est plus intéressant en fiabilité et sûreté. La méthode que nous proposons est d'estimer et analyser les distributions des paramètres d'entrée conditionnellement à la réalisation de l'évènement critique considéré. Nous présentons deux méthodes pour estimer ce genre de lois d'intérêt, dérivées des algorithmes PMCMC et SMC^2, introduits respectivement par [Andrieu et al., 2010] et [Chopin et al., 2013] pour faire du filtrage sur des modèles de Markov cachés. Ils sont respectivement des versions bruitées des algorithmes MCMC et SMC où les quantités incalculables sont remplacées par des estimateurs sans biais produits par des filtres particulaires auxiliaires. La différence avec les algorithmes développés par [Andrieu et al., 2010] et [Chopin et al., 2013] est que notre problème nécessite l'estimation de probabilités d'évènements rares. Nous devons donc introduire une des méthodes spécifiques à l'estimation de tels évènements et nous avons choisi d'utiliser le *splitting*.

L'algorithme MCMC consiste à construire une chaîne de Markov dont les transitions laissent la loi d'intérêt invariante. Les exemples les plus courants sont l'algorithme de MH et l'échantillonneur de Gibbs. Ils nécessitent tous deux dans leur implémentation l'estimation de probabilité d'évènements rares pour différentes propositions de paramètres. Comme elles ne peuvent pas être calculées explicitement, elles doivent être estimées. Dans ce cas, l'algorithme s'appelle alors PMCMC.

L'algorithme SMC consiste à simuler un échantillon de particules définies sur l'espace des paramètres, et de sélectionner puis de mettre à jour les paramètres selon des étapes de sélection et de mutation, pour estimer séquentiellement les lois d'intérêt. Dans ce cas précis, pour chaque valeur de paramètre dans l'échantillon, il faut calculer une probabilité d'évènement rare, qui peut être estimée par un autre algorithme SMC (le *splitting*). On a donc développé un algorithme de type SMC pour l'analyse d'évènements rares. Cet algorithme

est aussi un exemple d'*îlots de particules* introduits au chapitre 2 où un îlot est composé d'un paramètre et du filtre particulaire (auxiliaire) associé.

A notre connaissance, le développement de ces méthodes pour l'analyse d'évènements rares et des applications en fiabilité n'a jamais été utilisée auparavant. Nous prouvons la validité de ces algorithmes dans les sections 5.3.3 et 5.4.3. Nous illustrons la convergence de ces deux algorithmes sur un cas test. Puis nous appliquons le SMC^2 pour estimer la zone de retombée d'un étage d'un lanceur spatial et le risque de collision en orbite entre un satellite et un débris spatial. Le principal avantage de l'algorithme SMC^2 est de contourner la convergence en temps long de l'algorithme PMCMC, en utilisant des îlots en interaction.

Auteurs:

- Christelle Vergé (Onera, CNES, CMAP, Palaiseau)
- Jérôme Morio (Onera, Toulouse)
- Pierre Del Moral (Université UNSW, Sydney)

Statut: Le présent chapitre est basé sur l'article "An island particle algorithm for rare event analysis", soumis à *Reliability Engineering & System Safety*, en mai 2015, [Vergé et al., 2015b]

Abstract Estimating rare event probability with accuracy is of great interest for safety and reliability applications. Nevertheless, it is not just evaluating a risk or a probability but estimating the law of random phenomena that leads to critical events. Some parameters of the model or density parameters of input random variables in the system, may be fixed by an experimenter for simplicity reasons. From a risk analysis point of view, it is interesting to determine the impact of such tuning of parameters on the realization of a critical event, because a bad estimation of these parameters can strongly modify rare event probability estimations. In the present section, we present an example of island particle algorithm (see [Vergé et al., 2015a]) referred to as SMC^2. This algorithm is adapted from an existing one, introduced in [Chopin et al., 2013] in order to do inference for hidden Markov chains. Our version concerns rare event analysis. This algorithm samples from the law of parameters of a system conditionally on a rare event, and allows to determine the parameters that, in case of bad estimation, tend to increase the rare event probability value. It also gives an estimate of the rare event maximum probability taking into account the prior of the parameters. The principles of this statistical technique are described throughout this section. The SMC^2 algorithm is compared to the PMCMC algorithm introduced by [Andrieu et al., 2010] and applied on a test case in order to illustrate its convergence, and to estimate the fallout zone of launch vehicle booster. Another application concerns the estimation of the probability of collision between orbit objects.

Keywords: Rare event, sequential Monte Carlo, island particle models, particle filtering, sensitivity analysis, reliability theory

5.1 Introduction

Safety and reliability have been an increasing attention in various fields as aerospace [Morio, 2011a], aeronautics [Prandini and Watkins, 2005], genetics [Rosenbluth and Rosenbluth, 1955], finance [Richard and Zhang, 2007; Glasserman et al., 2000], insurance [Asmussen, 2003], geological phenomena [Asmussen, 2003; Kagan and Knopoff, 1977], informatic networks [Parekh and Walrand, 1989] or nuclear domain [Sjöberg and Drottz-Sjöberg, 1991]. The recent security rules or constraints in all those areas are such that the failure of a complex system must be a rare event. It is thus essential to be able to compute small probabilities.

The most popular technique to estimate a probability p is the Crude Monte Carlo (CMC) which consists in approximating a probability by an empirical mean of N independent and i.i.d. samples. CMC is simple to implement and provides an unbiased and consistent estimate of the probability of interest p. The relative error (which is the ratio between the standard deviation of an estimator and its mean) measures the variability of the estimation with respect to (w. r. t.) the quantity to estimate. It is an indicator for the quality of the estimation. The relative error of the CMC is of order $1/\sqrt{Np}$, so that in order to estimate a low probability $p = 10^{-k}$ ($k \in \mathbb{N}$) with a relative error of 10%, a very large sample $N = 10^{k+2}$ is needed. In the rare event context, the CMC is thus inefficient and a lot of fast simulation and variance reduction techniques have been proposed to estimate small probabilities since the 1950s. Indeed, importance sampling [Engelund and Rackwitz, 1993; Orsak and Aazhang, 1995; Richard and Zhang, 2007; Kroese and Rubinstein, 2012] is one of the most famous methods and is the basics of a lot of other methods as the Esscher transformation for compound Poisson processes [Asmussen, 2003; Glasserman, 2004; Blanchet et al., 2009], the transformations for random Gaussian vectors [Jourdain and Lelong, 2009] or sequential Monte Carlo (SMC) methods [Rosenbluth and Rosenbluth, 1955; Gordon et al., 1993; Doucet et al., 2001]. Splitting (also called subset simulation) is an example of SMC methods and has been considerably studied and improved over the past decades, see the references [Kahn and Harris, 1951], [Rubinstein, 1981], [Fishman, 1995], [Melas, 1997], [Glasserman et al., 1999], [Garvels, 2000], [Au and Beck, 2001], [Cérou et al., 2006], [Lagnoux, 2006; Lagnoux-Renaudie, 2009], [Cérou and Guyader, 2007; Cérou et al., 2012], [Au et al., 2007], [Gogate and Dechter, 2011]. Different techniques as *Markov chain Monte Carlo* (MCMC), first order reliability methods (FORM)/second order reliability methods (SORM)

[Bjerager, 1991; Madsen et al., 1986; Yan-Gang and Tetsuro, 1999], or extreme value theory [Fisher and Tippett, 1928; Leadbetter et al., 1983; Leadbetter and Rootzén, 1988] are also notably well known simulation algorithms to estimate very small probability of failure. Their principles and advantages/drawbacks have also been deeply studied, see for example [Morio et al., 2014].

In this section, we focus on rare event which can be modeled by a threshold exceedence of a deterministic output function. This "input-output" function can be seen as a "black-box" with random inputs. Some parameters, denoted by a vector Θ, in black-box functions are implicitly set, such as parameters of the model or of the input parametric model density, and their value influences the rare event probability estimation. These hypotheses are often assumed for simplification and computational reasons. From a risk analysis point of view, it is interesting to determine the variability of the output of a code w. r. t. the uncertainty on theses input parameters Θ or w. r. t. one particular parameter, and to quantify the impact of such tuning in the realization of a critical event. Of course, different values of Θ can strongly modify rare event probability estimation and sometimes miss a risk situation. The issue of concern in safety would be to underestimate a risk because of a bad tuning of model parameters Θ. That is why in this section we propose to estimate the law of the parameters conditionally on a critical event, and to present different algorithms for this purpose.

Some studies on close topics have been proposed recently, mostly based on sensitivity analysis methods. Monte Carlo filtering [Saltelli et al., 2004] consists in determining the differences between a "safe" sample and a "faulty" sample *via* standard statistical tests. The reliability index resulting from FORM/SORM [Lemaire et al., 2009] can also be used to analyze the influence of input parameters on the failure probability. Stratified sampling [Munoz Zuniga et al., 2011] and importance sampling [Morio, 2011a] have been adapted with the same purpose. See [Beck and Arnold, 1977] and the references wherein for an in-depth presentation of parameter estimation methods. The analysis of parameter sensitivity is challenging in numerous domains such as molecular phylogenetic sequence analysis [Wheeler, 1995], systems biology or postgenomic era [Cho et al., 2003], basin hydrology design [Beven and Kirkby, 1979], etc.

The methods proposed in this section are rather different. Indeed, in sensitivity analysis, a parameter is influent if its variation increases or decreases the rare event probability. It is not the case here since one focuses more precisely on the settings of the parameters that increase the rare event probability, which is of major interest for safety and reliability. The proposed approach is to estimate and analyze the distribution of these input parameters conditionally on the rare event of interest. We present two methods to estimate this kind of targeted laws derived from the PMCMC and the SMC^2 algorithms, respectively introduced by [Andrieu et al., 2010] and [Chopin et al., 2013] to do filtering on hidden Markov models. There are respectively noisy versions of MCMC and SMC algorithms where intractable quantities are replaced by unbiased estimators produced by an auxiliary SMC algorithm.

MCMC consists in constructing a Markov chain whose transition kernels leave the target density of interest invariant. The most common examples are the MH algorithm and the Gibbs sampler. In their implementation, they both require the evaluation of the probability of the rare event for different proposals of parameters. As it cannot be computed, it must be estimated by one method designed for rare event estimation, and the algorithm becomes the PMCMC. We have chosen to implement splitting in order to estimate rare event probabilities in this section.

SMC algorithms consist in running a batch of particles defined on the parameter space, and to select and update these parameters through recursive selection and mutation steps, in order to sequentially approximate the targeted laws of interest. In this special framework, for each value of parameter in the sample we need to compute the probability of the rare event given this parameter, which can be estimated by another SMC algorithm (as the splitting). We then have two embedded SMC algorithms that justify the terminology SMC^2 algorithm. It is also an example of island particle algorithm introduced by [Vergé et al., 2015a].

To our knowledge, the development of these methods for rare event case and safety application has not

been proposed yet. We proved the validity of these algorithms in subsection 5.3.3 and subsection 5.4.3. We illustrate the convergence of these two algorithms on a test case. The main advantage of the SMC^2 algorithm is to bypass the long time convergence issue of the PMCMC algorithm using interacting island Monte Carlo spatial integrations. When there is a single individual in each island, the SMC^2 coincides with the conventional SMC algorithm.

In the section, we firstly raise the fundamental question : how to estimate the law of Θ conditionally on a rare event? Two algorithms, PMCMC and SMC^2, are proposed to solve this issue. Their principle and algorithmic implementation are notably described, and the performance of these algorithms are discussed on a test case. We also apply the SMC^2 algorithm to two realistic simulations in the aerospace domain : the estimation of fallout position of a stage of a launch vehicle and the estimation of a space collision between a satellite and a space debris.

5.2 General problem

In this section, we underline the challenge of estimating conditional distribution of the parameters of a model given a rare event event in reliability, and we set notations.

Let us consider a d-dimensional random variable X defined on a measurable space $(\mathsf{X}, \mathcal{X})$, ϕ a continuous positive scalar function $\phi : \mathbb{R}^d \to \mathbb{R}$ and S a given critical threshold. The function ϕ is static, *i.e.* does not depend on time, and represents for instance an input-output model. This kind of model is notably used in various applications [Keane and Nair, 2005; Banerjee et al., 2010; Grancharova et al., 2011; Kreinovich and Ferson, 2004; Worden et al., 2007]. We assume that the output $Y = \phi(X)$ is also a positive random variable (typically a distance). The quantity of interest on the output Y is the probability of exceedance

$$\mathbb{P}(Y > S) = \mathbb{P}(\phi(X) > S) \ .$$

When the event $\{Y > S\}$ is rare relatively to the available simulation budget (which is often the case in safety and reliability issues), different algorithms described in [Sobol, 1994; Bucklew, 2004; Rubinstein and Kroese, 2004; Zhang, 1996; Bjerager, 1991; Botev and Kroese, 2012; Cérou et al., 2012] have notably been proposed to estimate accurately its probability.

In the present section, one focuses on the case where the law of X is uncertain and depends upon unknown parameters. We assume that X is distributed according to a well known parametric model and its parameters, denoted by a random vector Θ, have a probability density ν. We also suppose that Θ has a density f_Θ w. r. t. a dominating measure of reference λ, that is

$$\nu(\mathrm{d}\theta) = f_\Theta(\theta) \ \lambda(\mathrm{d}\theta) \ .$$

For instance, in the applications considered in this section, X is a random vector with a multivariate normal distribution, and Θ may describe the mean or the covariance matrix of X. It corresponds notably to realistic applications where it is not always possible to evaluate accurately the density of input parameters. This formalism enables thus to consider a large range of input probability density function.

The probability of interest $\mathbb{P}(Y > S)$ depends of course on Θ and thus on the distribution ν. In safety applications, it is important to estimate a superior bound of the rare event probability $\mathbb{P}(Y > S)$ taking also into account the prior on Θ. The prior on Θ is important since unrealistic bad tuning values of Θ which lead to high probabilities $\mathbb{P}(Y > S)$ are not relevant. The idea of this section is thus to determine the distribution of Θ conditionally on the fact that Y exceeds the threshold S. This distribution, denoted by π, will be referred to in the sequel as the *target law*.

In the further development, when there is no confusion, we sometimes write $\mathbb{P}(Y > S|\theta)$ instead of $\mathbb{P}(Y > S|\Theta = \theta)$.

Note that using the Bayes' formula, the target law can be written

$$\pi(\mathrm{d}\theta) = \frac{1}{\mathbb{P}(Y > S)} \mathbb{P}(Y > S|\theta)\nu(\mathrm{d}\theta) \, . \tag{5.2.1}$$

We propose in this section two techniques in order to estimate π. The first one is based on the construction of a Markov chain that evolves according to proposition and acceptance/rejection steps, and which approximates π when the number of iterations gets large. The second method uses an SMC algorithm which evolves according to iterative selection and mutation steps, and which approximates the target law when the number of particles gets large. These two methods are detailed in the following sections.

5.3 The PMCMC algorithm

We assume that all Markov kernels and probability measures considered in the sequel have a density w. r. t. some common dominating measure λ. In the following, we will make a slight abuse of notations writing it in the same way. MCMC methods consists in constructing an ergodic Markov chain $(\theta_n)_{n\in\mathbb{N}}$ with π as stationary law. Ergodic theorem ensures that the occupation measure of the Markov chain approaches the target law π as the number of iterations gets large. The key point of these methods is to build a Markov kernel that leaves invariant the target law. The most common methods are the MH sampler (see [Metropolis et al., 1953; Hastings, 1970]) and the Gibbs sampler (see [Geman and Geman, 1984; Robert and Casella, 2004; Andrieu et al., 2010] for a complete introduction).

5.3.1 The MH algorithm

The MH algorithm consists in proposing a new candidate and accepting it with a probability ensuring the reversibility of the MH kernel w. r. t. the target law π. In order to apply the MH algorithm we only need to know the density of the target law up to a normalizing constant. According to (5.2.1),

$$\pi(\mathrm{d}\theta) \propto \mathbb{P}(Y > S|\theta) f_\Theta(\theta)\lambda(\mathrm{d}\theta) \, .$$

The Markov chain $(\theta_n)_n$ is arbitrarily initialized. Denote by π_0 its initial distribution. Then, it evolves according to successive proposition and acceptance-rejection steps. The proposition step allows to explore the state space Θ, proposing a new candidate using a Markov kernel Q. The acceptance-rejection step consists in accepting the previous proposal with some probability α, called *acceptance rate*; if the move is rejected, the chain goes back to the state before the proposition. For all $\theta \in \Theta$, set $q(\theta, \cdot)$ the probability density of $Q(\theta, \cdot)$ w. r. t. the dominating measure λ. The MH algorithm proceeds as follows.

Data: π_0, Q, q, n: the number of iterations of the algorithm, fixed by the experimenter
Result: $(\theta_p)_{0\leq p\leq n}$
/* Initialization */
Sample $\theta_0 \sim \pi_0$;
/* Transition */
for $p \leftarrow 0$ **to** n **do**
  Sample $\theta'_p \sim Q(\theta_p, \cdot)$;
  Compute the acceptance probability

$$\alpha(\theta_p, \theta'_p) \stackrel{\text{def}}{=} 1 \wedge \frac{\mathbb{P}(Y > S|\theta'_p)}{\mathbb{P}(Y > S|\theta_p)} \frac{f_\Theta(\theta'_p) q(\theta'_p, \theta_p)}{f_\Theta(\theta_p) q(\theta_p, \theta'_p)} ,$$

  with the convention $\alpha(\theta_p, \theta'_p) = 0$ if the denominator is zero ;
  Sample $U \sim \mathcal{U}([0,1])$;
  if $U \leq \alpha(\theta_p, \theta'_p)$ **then**
    $\theta_{p+1} = \theta'_p$,
  else
    $\theta_{p+1} = \theta_p$.
  end
end

Algorithm 11: The MH algorithm

Notice that the acceptance rate α can be simplified when the proposal kernel Q is chosen reversible w. r. t. the prior ν. For example the Cranck-Nicholson shaker that proposes $\theta' \sim \sqrt{1-a}\theta + \sqrt{a}Z$, where $a \in]0;1[$ and Z is sampled from a standard Gaussian random variable, is reversible w. r. t. the standard Gaussian distribution. In this case, $q(\theta, \theta') = \exp\{-(\theta' - \sqrt{1-a}\theta)^2/2a\}/\sqrt{2\pi a}$.

The MH transition is a Markov kernel denoted by M, defined for all $(x, z) \in \Theta^2$ by

$$M(x, \mathrm{d}z) = \alpha(x, z)\, Q(x, \mathrm{d}z) + \left(1 - \int_\Theta \alpha(x, y)\, Q(x, \mathrm{d}y)\right) \delta_x(\mathrm{d}z).$$

The MH transition M is reversible w. r. t. π, so that the target law π is invariant by M, *i.e.*

$$\pi = \pi M . \tag{5.3.1}$$

5.3.2 The splitting algorithm

Unless in trivial cases, the probabilities $\mathbb{P}(\phi(X) > S|\Theta = \theta)$ are mostly not computable, but can be estimated for different settings of parameter θ. In the following, one value of θ is fixed and we discuss the estimation of $\mathbb{P}(\phi(X) > S|\Theta = \theta)$. One possibility is to use the *splitting algorithm* [Kahn and Harris, 1951; Cérou et al., 2006; Cérou et al., 2006]. The idea of the splitting algorithm is to decompose the probability of a rare event into a product of probabilities that are not rare. For that purpose, introduce a sequence of increasing thresholds

$$0 \stackrel{\text{def}}{=} S_0 < S_1 < \cdots < S_m \stackrel{\text{def}}{=} S .$$

Using Bayes' formula,

$$\mathbb{P}(\phi(X) > S|\Theta = \theta) = \prod_{p=0}^{m-1} \mathbb{P}(\phi(X) > S_{p+1}|\phi(X) > S_p, \Theta = \theta). \tag{5.3.2}$$

SMC methods allow to construct an unbiased estimator of $\mathbb{P}(\phi(X) > S|\Theta = \theta)$. The key idea is to interpret $\mathbb{P}(\phi(X) > S|\Theta = \theta)$ defined in (5.3.2) as the normalizing constant of a sequence of new targeted distributions

$\eta_{n,\theta}$ (which depends upon a parameter θ). Let $(\eta_{p,\theta})_{p\in\mathbb{N}}$ be a sequence of probability measures defined on the measurable space $(\mathsf{X},\mathcal{X})$. For all bounded measurable function h on X, and all $p\in\mathbb{N}$, $\eta_{p,\theta}$ is defined by

$$\begin{aligned}\eta_{p,\theta}h &\overset{\text{def}}{=} \frac{1}{\mathbb{P}(\phi(X)>S_p|\Theta=\theta)}\int_{\mathsf{X}} \mathbb{1}_{\phi(x)>S_p}\, h(x)\, p(x|\theta)\lambda(\mathrm{d}x)\\ &\propto \int_{\mathsf{X}} h(x) g_{p-1,\theta}(x)\, p(x|\theta)\lambda(\mathrm{d}x)\end{aligned}$$

where $g_{p-1,\theta}(x)=\mathbb{1}_{\phi(x)>S_p}$ and $p(x|\theta)\lambda(\mathrm{d}x)$ is the conditional distribution of X given $\Theta=\theta$.

As the thresholds are increasing, *i.e.* $g_{p+1,\theta}=g_{p+1,\theta}\times g_{p,\theta}$, one obtains

$$\eta_{p+1,\theta}h=\frac{\eta_{p,\theta}(g_{p,\theta}h)}{\eta_{p,\theta}(g_{p,\theta})}.$$

If $M_{p,\theta}$ is a Markov kernel that leaves $\eta_{p,\theta}$-invariant ($\eta_{p,\theta}\ M_{p,\theta}=\eta_{p,\theta}$), one gets

$$\eta_{p+1,\theta}h=\frac{\eta_{p,\theta}(g_{p,\theta}\times M_{p+1,\theta}h)}{\eta_{p,\theta}(g_{p,\theta})}\ , \tag{5.3.3}$$

One way of constructing such a Markov kernel $M_{p+1,\theta}$ is to consider a MH kernel that targets $\eta_{p+1,\theta}$. Assume a new particle is proposed using a Markov kernel T. For all $x\in\mathsf{X}$, set $t(x,\cdot)$ the probability density of $T(x,\cdot)$ w. r. t. the dominating measure λ. Then the proposal is accepted with probability

$$\alpha_{p,\theta}(x,x')=1\wedge\frac{g_{p,\theta}(x')p(x'|\theta)t(x',x)}{g_{p,\theta}(x)p(x|\theta)t(x,x')},$$

so that the MH kernel is $M_{p+1,\theta}(x,\mathrm{d}z)=\alpha_{p,\theta}(x,z)T(x,\mathrm{d}z)+\left(1-\int_{\mathsf{X}}\alpha_{p,\theta}(x,y)\,T(x,\mathrm{d}y)\right)\ \delta_x(\mathrm{d}z)$. This acceptance-rejection procedure can be repeated N_{app}^x times in order to decrease the correlation between particles.

In the light of (5.3.3), one can interpret $\eta_{p,\theta}$ as the Feynman-Kac probability measure associated to the couple $(g_{p,\theta},M_{p,\theta})$. Hence, it can be estimated by an interacting particle system that evolves recursively according to selection and mutation steps.

Consider $(\xi_{p,\theta}^1,...,\xi_{p,\theta}^{N_x})$ an N_x-particle system which evolves respectively according to selection steps (with $g_{p,\theta}$) and mutation steps (with $M_{p,\theta}$). The Feynman-Kac theory (see [Del Moral, 2004]) ensures that

$$\prod_{p=0}^{m-1}\left(\frac{1}{N_x}\sum_{i=1}^{N_x}\mathbb{1}\{\phi(\xi_{p,\theta}^i)\geq S_{p+1}\}\right)\ ,$$

is an unbiased estimator of $\mathbb{P}(\phi(X)>S|\Theta=\theta)$. In other words, $\mathbb{P}(\phi(X)>S|\Theta=\theta)$ is estimated by the products of proportions of particles which exceed each threshold up to the last one.

Data: $S_0, S_1, \ldots, S_m$, T, t, N^x_{app} the number of applications of the Markov kernels $(M_{n,\theta})_{n\in\mathbb{N}}$.

/* Initialization */

Sample $(\xi^1_{0,\theta}, ..., \xi^{N_x}_{0,\theta})$ i.i.d. according to the conditional density $p(x|\theta)$;

for $k \leftarrow 0$ **to** $m-1$ **do**

  /* Selection */

  Sample $(\hat{\xi}^1_{k,\theta}, ..., \hat{\xi}^{N_x}_{k,\theta})$ randomly and independently among the particles $(\xi^1_{k,\theta}, ..., \xi^{N_x}_{k,\theta})$ which have reached the next threshold S_{k+1} ;

  /* Mutation */

  for $r \leftarrow 1$ **to** N^x_{app} **do**

    for $i \leftarrow 1$ **to** N_x **do**

      Sample $x \sim T(\hat{\xi}^i_{k,\theta}, \cdot)$;

      Compute the acceptance probability

$$\alpha_{k,\theta}(\hat{\xi}^i_{k,\theta}, x) = 1 \wedge \mathbb{1}_{\phi(x)>S_{k+1}} \frac{p(x|\theta)t(x, \hat{\xi}^i_{k,\theta})}{p(\hat{\xi}^i_{k,\theta}|\theta)t(\hat{\xi}^i_{k,\theta}, x)} ,$$

      Sample $U \sim \mathcal{U}([0,1])$;

      if $U \leq \alpha_{k,\theta}(\hat{\xi}^i_{k,\theta}, x)$ **then**

        $\xi^i_{k+1,\theta} = x$,

      else

        $\xi^i_{k+1,\theta} = \hat{\xi}^i_{k,\theta}$.

      end

      while $r < N^x_{app}$ **do**

        $\hat{\xi}^i_{k,\theta} = \xi^i_{k+1,\theta}$

      end

    end

  end

end

Algorithm 12: The splitting algorithm

Notice that only proposals that are above the next threshold are considered in the acceptance-rejection step.

Instead of fixing the thresholds in advance, which is difficult in practice and requires experience accumulated by practitioners, the thresholds can be adaptively set. Ideally, the multilevel decomposition should be chosen in such a way that the probability $\mathbb{P}(Y > S_{p+1}|Y > S_p, \Theta = \theta)$ of hitting the $(p+1)$-umpteenth level set given that the p-umpteenth level has already been reached, is independent of the index p, denoted by γ. Based on this remark, adaptive strategies have been recently proposed in [L'Ecuyer et al., 2006; Cérou and Guyader, 2007], and limit theorems [Cérou et al., 2005] have been proved in the special case of a one-dimensional state space, in which the levels are chosen online. By construction, these algorithms guarantee a probability of success of γ for the transition from one virtual level to the next virtual level, whereas the number of virtual levels that should be crossed to finally hit the critical set is not known in advance, and is learned by the algorithm as well. Alternatively, an empirical method called RESTART (see [Villén-Altamirano and Villén-Altamirano, 1994; Tuffin and Trivedi, 2000]) can also be used to compute rare transient events and the probability of rare events in steady state, not only the probability to reach the target before coming back to a recurrent set.

5.3.3 The PMCMC algorithm

PMCMC may be understood as *exact Monte Carlo approximations* of an ideal algorithm, as it samples exactly from the distribution of interest, in the sense that for a fixed number of particles $N_x \geq 1$, the transition kernels leave the target density of interest invariant. PMCMC is a noisy version of an ideal algorithm where some intractable quantity of the ideal algorithm is replaced by an unbiased Monte Carlo estimator produced by an interacting particle system.

This is for instance the case for the Particle Metropolis-Hastings (PMH) algorithm which is a noisy version of the MH presented in [Andrieu et al., 2010]. The adaptation of PMH algorithm to our problem of rare event estimation is described in Algorithm 13.

Data: π_0, Q, q, n: the number of iterations of the algorithm, $S_0, \ldots, S_m$: the intermediary thresholds, fixed by the experimenter
Result: $(\theta_p)_{0 \leq p \leq n}$
/* Initialization */
Sample $\theta_0 \sim \pi_0$;
Estimate $\mathbb{P}(Y > S|\theta_0)$ using Algorithm 12;
/* Transition */
for $p \leftarrow 0$ **to** n **do**
  Sample $\theta'_p \sim Q(\theta_p, \cdot)$;
  Estimate $\mathbb{P}(Y > S|\theta'_p)$ using Algorithm 12;
  Compute the acceptance probability

$$\alpha^{N_x}(\theta_p, \theta'_p) \stackrel{\text{def}}{=} 1 \wedge \left(\prod_{k=0}^{m-1} \frac{\sum_{i=1}^{N_x} \mathbb{1}\{\phi(\xi^i_{k,\theta'_p}) \geq S_{k+1}\}/N_x}{\sum_{i=1}^{N_x} \mathbb{1}\{\phi(\xi^i_{k,\theta_p}) \geq S_{k+1}\}/N_x} \times \frac{f_\Theta(\theta'_p) q(\theta'_p, \theta_p)}{f_\Theta(\theta_p) q(\theta_p, \theta'_p)} \right) ,$$

  with the convention $\alpha^{N_x}(\theta_p, \theta'_p) = 0$ if the denominator is zero ;
  Sample $U \sim \mathcal{U}([0,1])$;
  if $U \leq \alpha^{N_x}(\theta_p, \theta'_p)$ **then**
    $\theta_{p+1} = \theta'_p$,
  else
    $\theta_{p+1} = \theta_p$.
  end
end

Algorithm 13: The PMH algorithm

[Andrieu et al., 2010, Theorem 4] establishes that the PMH update leaves π invariant, reformulating the PMH as a MH on an extended state space. Moreover, as the indicator functions are upper bounded, the acceptance probability α^{N_x} of PMH converges to the one of MH as $N_x \to +\infty$ (see [Andrieu et al., 2010, Theorem 1]). On the same way, our version of the PMH algorithm for rare event estimation allows to estimate the target law π.

Similarly, the Particle Gibbs algorithm which has been introduced by [Andrieu et al., 2010] is a noisy version of the Gibbs sampler. The Particle Gibbs algorithm has notably been studied in [Lindsten et al., 2014], where the authors show that the Particle Gibbs Markov kernel is uniformly ergodic under rather general assumptions.

5.4 The SMC^2 algorithm

5.4.1 Principle

The SMC^2 algorithm introduced by [Chopin et al., 2013] is a noisy version of an SMC algorithm, the IBIS algorithm introduced in [Chopin, 2002]. These two algorithms are designed for exploring the parameter posterior distribution. IBIS algorithm is a kind of idealized algorithm where the importance weights are often not computable, and are replaced in the SMC^2 algorithm by unbiased estimators produced by an auxialiary SMC. We adapt the SMC^2 algorithm from [Chopin et al., 2013] for rare event analysis in order to sample from the target distribution π. In SMC^2, two sets of particles are embedded. The first set of particles is defined on the parameter space Θ, the second one is defined on the state space X and is useful to produce unbiased estimates of the probabilities $\mathbb{P}(Y > S|\theta)$.

Define a sequence of increasing thresholds

$$0 \stackrel{\text{def}}{=} S_0 < S_1 < \cdots < S_m \stackrel{\text{def}}{=} S \, . \tag{5.4.1}$$

For all $n \in [\![0, m]\!]$, denote by π_n the conditional distribution of the parameters Θ given the exceedence of the threshold S_n. The target law is of course $\pi_m = \pi$, but this distribution will be estimated sequentially using the auxiliary distributions $(\pi_n)_{0 \leq n \leq m-1}$. Thanks to the Bayes' theorem, one can rewrite π_n such as :

$$\pi_n(\mathrm{d}\theta) \propto \mathbb{P}(Y > S_n|\theta) \; \nu(\mathrm{d}\theta) \quad (n \in [\![0, m]\!]) \, . \tag{5.4.2}$$

Notice that for all $n \in [\![0, m]\!]$,

$$\mathbb{P}(Y > S_n|\Theta = \theta) = \prod_{p=0}^{n-1} \mathbb{P}(Y > S_{p+1}|Y > S_p, \Theta = \theta) \, . \tag{5.4.3}$$

Denote by $G_p(\theta)$ the conditional probability

$$G_p(\theta) \stackrel{\text{def}}{=} \mathbb{P}(Y > S_{p+1}|Y > S_p, \Theta = \theta) \quad (p \in [\![0, m-1]\!]) \, . \tag{5.4.4}$$

As the thresholds are increasing, the target distributions may be defined by recursion: for all bounded measurable function h on Θ, one has

$$\pi_{n+1} h = \frac{\pi_n(G_n h)}{\pi_n(G_n)} \, ,$$

If M_{n+1} is a Markov kernel that leaves π_{n+1} invariant, then

$$\pi_{n+1} h = \frac{\pi_n(G_n \times M_{n+1} h)}{\pi_n G_n} \, . \tag{5.4.5}$$

For example, one can consider the MH kernel that targets π_{n+1}. Consider a Markov kernel Q and for all $\theta \in \Theta$, set $q(\theta, \cdot)$ the probability density of $q(\theta, \cdot)$ w. r. t. the dominating measure λ. Assume a new particle is proposed using a Markov kernel Q, then this proposal is accepted with probability

$$\alpha_n(\theta, \theta') = 1 \wedge \frac{\prod_{p=0}^{n} G_p(\theta')}{\prod_{p=0}^{n} G_p(\theta)} \frac{f_\Theta(\theta') q(\theta', \theta)}{f_\Theta(\theta) q(\theta, \theta')} \, ,$$

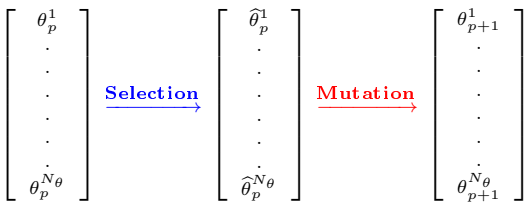
so that

$$M_{n+1}(\theta, \mathrm{d}z) = \alpha_n(\theta, z) Q(\theta, \mathrm{d}z) + \left(1 - \int_\Theta \alpha_n(\theta, \theta') Q(\theta, \mathrm{d}\theta')\right) \delta_\theta(\mathrm{d}z). \tag{5.4.6}$$

Equation 5.4.5 may be cast in the Feynman-Kac framework and then, each measure π_p can be approximated by an interacting particle system which evolves in accordance with selection steps related to the so-called potential functions G_n and mutation steps related to the Markov kernel M_n.

5.4.2 The IBIS algorithm

Denote by $\{(\theta_p^1, \dots, \theta_p^{N_\theta})\}_{p\geq 0}$ a system of N_θ particles defined on the state space Θ^{N_θ}.



Initially, $(\theta_0^1, \dots, \theta_0^{N_\theta})$ are i.i.d. according to the prior ν. Then, for $p \geq 1$, the transition between the samples $(\theta_p^1, \dots, \theta_p^{N_\theta})$ and $(\theta_{p+1}^1, \dots, \theta_{p+1}^{N_\theta})$ is made up for a selection and a mutation step.

- **Selection stage:** selection consists in choosing randomly and independently N_θ particles among $\{\theta_p^i\}_{i=1}^{N_\theta}$ with probabilities proportional to their weights $\{G_p(\theta_p^i)\}_{i=1}^{N_\theta}$, according to the probability measure $\pi_p(G_p \cdot)/\pi_p(G_p)$. Thus, the particles with low weights are eliminated whereas those with large weights are duplicated. The number of particles is kept constant in this stage and a new set of particles $\{\widehat{\theta}_p^i\}_{i=1}^{N_\theta}$ can be defined.
  The importance weights $G_k(\theta_p^i) = \mathbb{P}(Y > S_{p+1} | Y > S_p, \Theta = \theta_p^i)$, defined in (5.4.4), have to be computed.

- **Mutation stage:** mutation consists in updating the selected particles conditionally independently using a Markov kernel M_{p+1} that leaves π_{p+1} invariant using (5.4.6). This step enables to increase the diversity of the selected sample without changing its probability law, which is already close to π_{p+1}. This acceptance/rejection procedure can be repeated N_{app}^θ times in order to decrease the correlation between particles. At the end of this stage, a new set of particles $\{\theta_{p+1}^i\}_{i=1}^{N_\theta}$ is defined.

Mutation and selection stages are applied m times until reaching the target threshold S_m. At the end of the algorithm, the particles $\{\theta_m^i\}_{i=1}^{N_\theta}$ provide an estimate of π :

$$\widehat{\pi}_m^{N_\theta} = \frac{1}{N_\theta} \sum_{i=1}^{N_\theta} \delta_{\theta_m^i} \ .$$

The Feynman-Kac theory [Del Moral et al., 2012] ensures that at each transition stage $p \in [\![0, m]\!]$,

$$\widehat{\pi}_p^{N_\theta} \xrightarrow[N_\theta \to +\infty]{\mathbb{P}} \pi_p \ .$$

Data: $S_0, \dots, S_m$, the number of applications N_{app}^{θ} of Markov kernels M_n, Q, q

/* Initialization */

Sample $(\theta_0^1, \dots \theta_0^{N_\theta}) \sim_{i.i.d.} \nu$;

/* Transition */

for $k \leftarrow 0$ **to** $m-1$ **do**

  /* Selection */

  Compute the weights $\{G_k(\theta_k^i)\}_{i=1}^{N_\theta}$;

  Sample $\left(I^1, \dots, I^{N_\theta}\right) \sim_{i.i.d.} \mathsf{Mult}\left(\left\{G_k\left(\theta_k^i\right)\right\}_{i=1}^{N_\theta}\right)$;

  Set $\left(\widehat{\theta}_k^1, \dots, \widehat{\theta}_k^{N_\theta}\right) = \left(\theta_k^{I^1}, \dots, \theta_k^{I^{N_\theta}}\right)$;

  /* Mutation */

  for $r \leftarrow 1$ **to** N_{app}^{θ} **do**

    for $i \leftarrow 1$ **to** N_θ **do**

      Sample $\widehat{\theta}_k^{i\prime} \sim Q(\theta_k^i, \cdot)$;

      Compute the weights $G_\ell(\widehat{\theta}_k^i)$ and $G_\ell(\widehat{\theta}_k^{i\prime})$ for $\ell \in [\![0, k]\!]$;

      Compute the acceptance probability

$$\alpha_k(\widehat{\theta}_k^i, \widehat{\theta}_k^{i\prime}) = 1 \wedge \frac{\prod_{\ell=0}^k G_\ell(\widehat{\theta}_k^{i\prime})}{\prod_{\ell=0}^k G_\ell(\widehat{\theta}_k^i)} \frac{f_\Theta(\widehat{\theta}_k^{i\prime}) q(\widehat{\theta}_k^{i\prime}, \widehat{\theta}_k^i)}{f_\Theta(\widehat{\theta}_k^i) q(\widehat{\theta}_k^i, \widehat{\theta}_k^{i\prime})} ,$$

      Sample $U \sim \mathcal{U}([0,1])$;

      if $U < \alpha_k(\widehat{\theta}_k^i, \widehat{\theta}_k^{i\prime})$ **then**

        set $\theta_{k+1}^i = \widehat{\theta}_k^{i\prime}$

      else

        set $\theta_{k+1}^i = \widehat{\theta}_k^i$.

      end

    end

    while $r < N_{app}^{\theta}$ **do**

      set $\widehat{\theta}_k^i = \theta_{k+1}^i$

    end

  end

end

/* Estimation */

Estimate π with $\widehat{\pi}_m^{N_\theta} \stackrel{\text{def}}{=} \sum_{i=1}^{N_\theta} \delta_{\theta_m^i} / N_\theta$.

Algorithm 14: The IBIS algorithm

One may notice that at iteration $k \in [\![0, m-1]\!]$, for each particle $\{\theta_k^i\}_{i=1}^{N_\theta}$, the IBIS needs to compute the weights $G_\ell(\theta_k^i)$ for $\ell \in [\![1, N_\theta]\!]$. In practice, these weights are often not analytically computable but can be estimated using a splitting algorithm for the different parameters of the sample. The IBIS is thus an idealized algorithm, and we present in the following its practical counterpart : the SMC^2 algorithm.

5.4.3 The SMC^2 algorithm

For $i \in [\![1, N_\theta]\!]$ and $k \in [\![0, m-1]\!]$, the point is to compute the probability $G_\ell(\theta_k^i)$. If they cannot be analytically calculated, they can be estimated with another interacting particle system (also called, in that case, SMC, importance splitting, subset simulation or subset sampling). It is a rare event estimation technique detailed in subsection 5.3.2, which consists in estimating several conditional probabilities that are easier to evaluate than estimating only one probability through a very tough simulation. Its principle is also based on iterative selection and mutation stages.

For each $\Theta = \theta_k^i$, one can construct a sample $\{(\xi_p^{i,1}, \dots, \xi_p^{i,N_x})\}_{0 \leq p \leq k}$, using the splitting algorithm (**Algorithm** 12) with the thresholds $S_0, \dots, S_m$ defined in (5.4.1), such that for all $\ell \in [\![0, k]\!]$, $G_\ell(\theta_k^i) = \mathbb{P}(\phi(X) > S_{\ell+1} | \phi(X) > S_\ell, \Theta = \theta_k^i)$ is approximated by $\sum_{j=1}^{N_x} \mathbb{1}\{\phi(\xi_\ell^{i,j}) > S_{\ell+1}\}/N_x$. Moreover, for all $\ell \in [\![0, k]\!]$,

$$\frac{1}{N_x} \sum_{j=1}^{N_x} \mathbb{1}\{\phi(\xi_\ell^{i,j}) > S_{\ell+1}\} \xrightarrow[N_x \to +\infty]{\mathbb{P}} G_\ell(\theta_k^i) \, .$$

For a given particle θ_k^i, a complete set of particles $\{\xi_\ell^{i,j}\}_{1 \leq \ell \leq k}^{1 \leq j \leq N_x}$ is thus generated. A single particle θ_k^i and its associated $\{\xi_\ell^{i,j}\}_{1 \leq \ell \leq k}^{1 \leq j \leq N_x}$ particle set, can be considered as an island. The SMC^2 is thus an example of *island particle model* introduced in [Vergé et al., 2015a]. The SMC^2 algorithm is described in **Algorithm** 15.

Data: $S_0, \dots, S_m$, the number of applications N^{θ}_{app} of Markov kernels M_n, the number of applications N^{x}_{app} of the Markov kernels $M_{n,\theta}$, Q, q, T, t

/* Initialization */

Sample $(\theta_0^1, \dots \theta_0^{N_\theta}) \sim_{i.i.d.} \nu$;

for $i \leftarrow 1$ **to** N_θ **do**

  Sample $(\xi_0^{i,1}, \dots, \xi_0^{i,N_x})$ i.i.d. according to the conditional density $p(x|\theta_0^i)$;

end

/* Transition */

for $k \leftarrow 0$ **to** $m-1$ **do**

  Run a particle system $\{(\xi_\ell^{i,1}, \dots, \xi_\ell^{i,N_x})\}_{1 \le \ell \le k}$ associated to each θ_k^i in order to estimate $G_\ell(\theta_k^i)$ for $\ell \in [\![0, k]\!]$ with **Algorithm** 12;

  /* Selection of the θ-particles */

  Sample $\left(I^1, \dots, I^{N_\theta}\right) \sim_{i.i.d.} \mathsf{Mult}\left(\left\{\sum_{j=1}^{N_x} \mathbb{1}\{\phi(\xi_k^{i,j} > S_{k+1})\}\right\}_{i=1}^{N_\theta}\right)$;

  Set $\left(\widehat{\theta}_k^1, \dots, \widehat{\theta}_k^{N_\theta}\right) = \left(\theta_k^{I^1}, \dots, \theta_k^{I^{N_\theta}}\right)$;

  /* Mutation of the θ-particles */

  for $r \leftarrow 1$ **to** N^{θ}_{app} **do**

    for $i \leftarrow 1$ **to** N_θ **do**

      Sample $\widehat{\theta}_k^{i\prime} \sim Q(\theta_k^i, \cdot)$;

      Run a particle system $\{(\tilde{\xi}_\ell^{i,1}, \dots, \tilde{\xi}_\ell^{i,N_x})\}_{1 \le \ell \le k}$ associated to each $\widehat{\theta}_k^{i\prime}$ in order to estimate $G_\ell(\widehat{\theta}_k^{i\prime})$ for $\ell \in [\![0, k]\!]$ with **Algorithm** 12;

      Compute the acceptance probability

$$\alpha_k^{N_x}(\widehat{\theta}_k^i, \widehat{\theta}_k^{i\prime}) = 1 \wedge \left(\prod_{\ell=0}^{k} \frac{\sum_{j=1}^{N_x} \mathbb{1}\{\phi(\tilde{\xi}_\ell^{i,j}) > S_{\ell+1}\}}{\sum_{j=1}^{N_x} \mathbb{1}\{\phi(\xi_\ell^{i,j}) > S_{\ell+1}\}} \frac{f_\Theta(\widehat{\theta}_k^{i\prime}) q(\widehat{\theta}_k^{i\prime}, \widehat{\theta}_k^i)}{f_\Theta(\widehat{\theta}_k^i) q(\widehat{\theta}_k^i, \widehat{\theta}_k^{i\prime})} \right) ,$$

      Sample $U \sim \mathcal{U}([0,1])$;

      if $U < \alpha_k^{N_x}(\widehat{\theta}_k^i, \widehat{\theta}_k^{i\prime})$ **then**

        set $\theta_{k+1}^i = \widehat{\theta}_k^{i\prime}$

      else

        set $\theta_{k+1}^i = \widehat{\theta}_k^i$.

      end

    end

    while $r < N^{\theta}_{app}$ **do**

      set $\widehat{\theta}_k^i = \theta_{k+1}^i$

    end

  end

end

/* Estimation */

Estimate π with $\widehat{\pi}_m^{N_\theta} \stackrel{\text{def}}{=} \sum_{i=1}^{N_\theta} \delta_{\theta_m^i} / N_\theta$.

Algorithm 15: The SMC^2 algorithm

The SMC^2 algorithm can be reformulated as an SMC algorithm on the extended state space $\Theta \times \mathsf{X}^{N_x}$. This interpretation allows to establish the consistency of the empirical estimate induced by the SMC^2, as the number N_θ of θ-particles gets large.

This algorithm is however not online since its computational cost increases at each iteration, with the number of intermediary thresholds.

In most applications, the potential functions G_ℓ are strictly positive, so that the interacting island model described in **Algorithm** 14 is well defined at any time. In the reverse angle, the evolution of the particles within each island described in **Algorithm** 12 is based on a selection mechanism associated with indicator functions. When all particles in a given island, cannot enter into the desired event, their evolution is stopped and the corresponding level crossing probability is estimated by 0. To bypass this technical difficulty, we can use adaptive multiple levels evolution strategies [Cérou et al., 2012].

5.5 Application on different test cases

In this section, we apply the SMC^2 algorithm on a Gaussian toy case and on two realistic aerospace examples.

5.5.1 Gaussian toy case

In this part, we illustrate the convergence of the PMCMC and the SMC^2 algorithms presented in this section in order to sample from the conditional distribution π of model parameters given a rare event. We consider a simple example where one is able to compute exactly the density of π. We assume that ϕ is the identity function, and X and Θ are normal random variables

$$X \sim \mathcal{N}(\Theta, 1) \qquad \text{and} \qquad \Theta \sim \mathcal{N}(0, 1) \; . \tag{5.5.1}$$

In the light of (5.5.1), X can be written as $X = \Theta + Z$ where $(\Theta, Z) \sim_{i.i.d.} \mathcal{N}(0, 1)$. Hence, $X \sim \mathcal{N}(0, 2)$. The probability of interest $\mathbb{P}(\phi(X) > S)$ can be explicitly computed in this particular case, since

$$\mathbb{P}(\phi(X) > S) = F_Z(-S/\sqrt{2}) \; ,$$

where F_Z is the cumulative distribution function (c.d.f.) of the standard normal distribution. Moreover, in order to determine the conditional distribution π, it is required to compute the following expectation for all bounded measurable function h on Θ,

$$\begin{aligned}
\mathbb{E}\left[h(\Theta)\mathbb{1}\{\phi(X) > S\}\right] = \mathbb{E}\left[h(\Theta)\mathbb{1}\{\Theta + Z > S\}\right] &= \int_{\mathbb{R}} \int_{S-\theta}^{+\infty} h(\theta) \frac{1}{2\pi} e^{-\theta^2/2} e^{-z^2/2} \mathrm{d}\theta \mathrm{d}z \\
&= \int_{\mathbb{R}} h(\theta) \frac{e^{-\theta^2/2}}{\sqrt{2\pi}} \left(\int_{S-\theta}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-z^2/2} \mathrm{d}z \right) \mathrm{d}\theta \\
&= \int_{\mathbb{R}} h(\theta) \frac{e^{-\theta^2/2}}{\sqrt{2\pi}} F_Z(\theta - S) \mathrm{d}\theta \; .
\end{aligned}$$

Thus,

$$\mathbb{E}\left[h(\Theta) | \phi(X) > S\right] = \frac{\mathbb{E}\left[h(\Theta)\mathbb{1}\{\phi(X) > S\}\right]}{\mathbb{P}(\phi(X) > S)} = \int_{\mathbb{R}} h(\theta) \frac{e^{-\theta^2/2}}{\sqrt{2\pi}} \frac{F_Z(\theta - S)}{F_Z(-S/\sqrt{2})} \mathrm{d}\theta \; ,$$

so that, we deduce the conditional density of π is

$$\frac{e^{-\theta^2/2}}{\sqrt{2\pi}} \frac{F_Z(\theta - S)}{F_Z(-S/\sqrt{2})} \; .$$

We plot the exact density of π in red for $S = 5$. In this case, $\mathbb{P}(\phi(X) > S) \simeq 2\ 10^{-4}$. The intermediate thresholds are between [0.5,5] with a step of 0.1.

We run the SMC^2 algorithm for the settings $N_\theta = 2000, N_x = 20, N_{app}^x = 3, N_{app}^\theta = 10, m = 46$. With the notations used in subsection 5.4.3, π is approximated by the empirical distribution $\sum_{i=1}^{N_\theta} \delta_{\theta_m^i}/N_\theta$. So, we plot in green the density of this last sample, using a kernel density estimator.

We also apply PMCMC with $n = 100000$ and $N_x = 500$. According to the ergodic theorem, the occupation measure of the Markov chain approaches π. We plot its density, after a warm-up time of 10000 iterations, in blue.

These densities are represented in Figure 5.1. We observe that the two algorithms converge to the

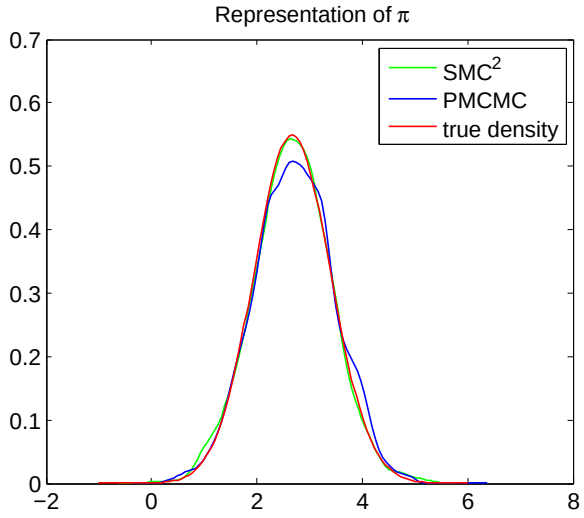


Figure 5.1: Estimation of π with PMCMC and SMC^2 on a toy case, compared to exact density.

theoretical density. However, the convergence of the PMCMC is slow, so that the PMCMC necessitates a large number of iterations n. We can compare the number of calls to the function ϕ. It is 1.8 10^8 for the PMCMC and 1.9 10^9 for the SMC^2. Considering this present example and the conditions of the convergence, we favor the utilization of the SMC^2 in the following applications.

5.5.2 Estimation of launch vehicle booster fallout zone

Spatial launch vehicle fall-back safety zone estimation is a very important problem in space applications since the consequences of a mistake can be dramatic for the population. We consider a solid rocket booster that is the first stage of a launch vehicle. Its mass is about 35000 kilograms and the launch point is at an altitude of 112 kilometers with a slope of 15 degrees. At the end of its mission, the rocket booster falls into the sea at some distance of a predicted position. Similar models have already been analyzed in [Morio, 2011b].
The launch vehicle stage fall-back is thus modeled as an input-output model with 4 Gaussian inputs X and one output $Y = \phi(X)$, representing the distance between the estimated launch stage fall-back position and the predicted one. In this study case, the aim is to estimate the probability that the distance to the predicted impact position exceeds 0.72 km: $\mathbb{P}(\phi(X) > 0.72)$. The different inputs (components of X) that can influence the impact position of the stage of the launch are:

- meteorological conditions (2 inputs).

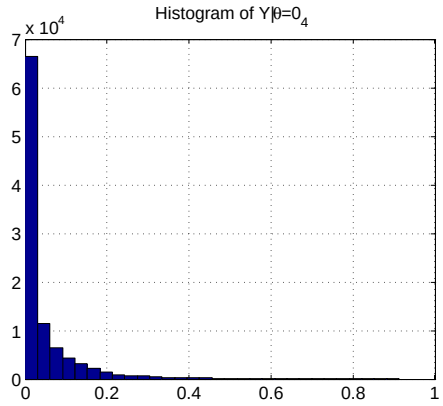


Figure 5.2: Law of Y given $\Theta = 0_4$ estimated with 100000 runs of Monte Carlo simulations.

- launch vehicle mass (1 input).
- the slope angle between the vertical axis and the velocity vector (1 input).

Indeed, the wind variations during the fall-back can influence the impact position. In addition, the mass of the different parts of the launch vehicle is also slightly random during the fall-back.

The uncertainty on X is modeled by a Gaussian random vector with mean $\Theta = (\Theta_1, \Theta_2, \Theta_3, \Theta_4)^t$ and a covariance matrix equals to the identity matrix. Its mean vector Θ is also uncertain since it is difficult to estimate accurately these quantities. We thus assume that each component of Θ has a standard normal distribution. The input-output code includes the transformation that allows to switch from the standard space of the input to the physical space where evolve the different input parameters.

The law of Y given $\Theta = 0_4$ is represented in Figure 5.2 with 100000 runs of Monte Carlo simulations. The probability $\mathbb{P}(\phi(X) > 0.72|\Theta = 0_4)$ is estimated to 8.5 10^{-4}, with a relative error of 11%.

The SMC^2 algorithm has been applied on this example with the following parameters: $N_\theta = 250$, $N_x = 500$, $N^\theta_{app} = 3$, $N^x_{app} = 3$. The intermediate thresholds expressed in kilometers are $\{0, 0.2, 0.4, 0.6, 0.72\}$. The estimators of the different marginals of π, obtained with the SMC^2 algorithm, are given in Figure 5.3, where the first marginal is related to the first parameter and so on. Table 5.4 gathers estimations of the probability $\mathbb{P}(Y > 0.72|\Theta = \theta)$ for different values of θ. The interpretation based on Table 5.4 is that parameters Θ_1 and Θ_4 have to be tuned accurately in order to not underestimate $\mathbb{P}(\phi(X) > 0.72)$. Indeed, in the light of Figure 5.3, distributions of the θ-particles at the end of the SMC^2 algorithm are very different from initial probability distribution π_0 (standard normal distributions), particularly for parameters Θ_1 and Θ_4. It means that, if the assumption "Θ_1 and Θ_4 have zero means" is not valid, then, there is a high risk to underestimate $\mathbb{P}(\phi(X) > 0.72|\Theta = \theta)$ (see Table 5.4) and it can be very problematic for obvious safety reasons. At the opposite, the initial assumption on parameters Θ_2 and Θ_3 already maximizes the trade-off between prior of Θ and the probability $\mathbb{P}(\phi(X) > 0.72)$. Thus, if Θ_2 and Θ_3 are not equal to 0, the probability $\mathbb{P}(\phi(X) > 0.72)$ will not increase too much or even decrease which is positive for safety.

The estimated probabilities $\widehat{\mathbb{P}}(Y > 0.72|\Theta = \theta^i_m)$, where $\{\theta^i_m\}_{i=1}^{N_\theta}$ are the particles at the end of the SMC^2, are given in Figure 5.5. Their values can be much higher than $\widehat{\mathbb{P}}(Y > 0.72|\Theta = 0_4)$, that is when Θ is equal to its mean value. For instance, when $\theta = \sum_{i=1}^{N_\theta} \theta^i_m / N_\theta$, the probability $\widehat{\mathbb{P}}(Y > 0.72|\Theta = \sum_{i=1}^{N_\theta} \theta^i_m / N_\theta)$ is equal to 0.034. This increase of the probability, compared to the initial situation where $\Theta = 0_4$, has to be taken into account in order not to underestimate the rare event probability.

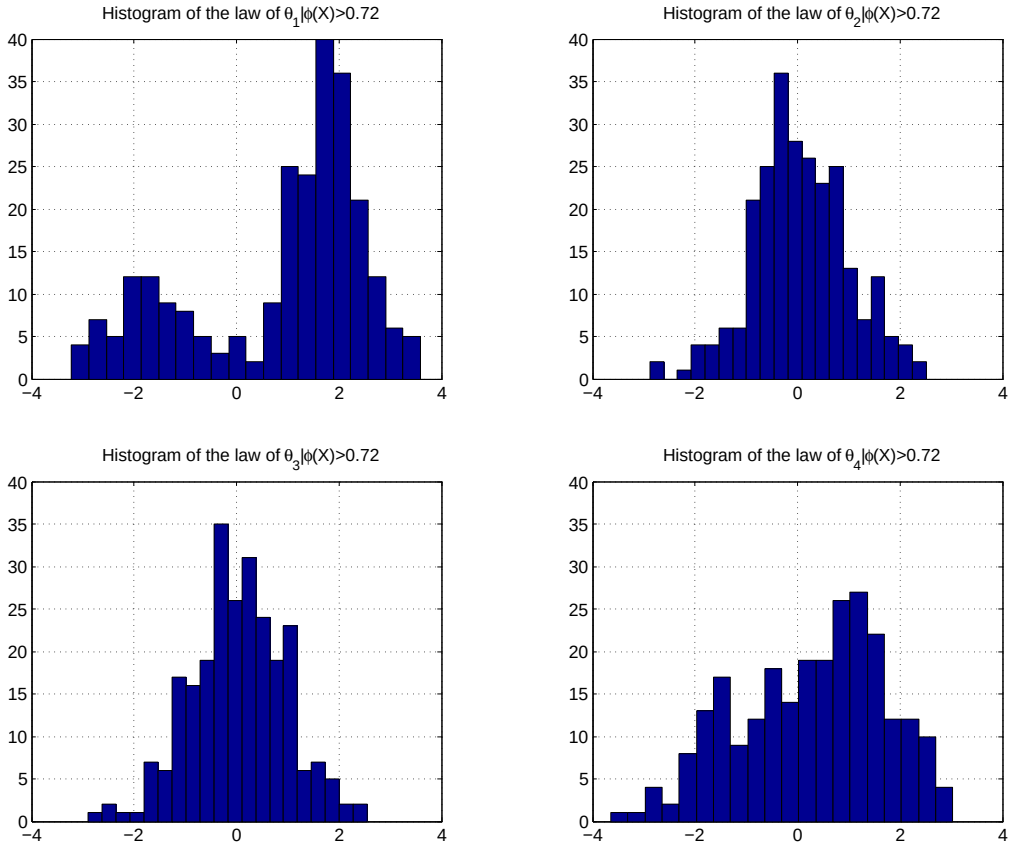


Figure 5.3: Estimations of the marginals of π using the SMC^2 algorithm.

θ_1	θ_2	θ_3	θ_4	$\hat{\mathbb{P}}(Y > 0.72\|\theta)$
0	0	0	0	$8.5\ 10^{-4}$
1	0	0	0	$3.5\ 10^{-3}$
1	0	0	1	$1.05\ 10^{-2}$
-1	0	0	1	$1.02\ 10^{-2}$
-1	0	0	-1	$1.14\ 10^{-2}$
0	-1	0	0	$9.4\ 10^{-4}$
0	-1	1	0	$9.8\ 10^{-4}$
0	1	1	0	$9.7\ 10^{-4}$
0	1	-1	0	$1.0\ 10^{-3}$
0	0	-1	0	$7.2\ 10^{-4}$

Table 5.4: Estimates of the probability $\mathbb{P}(Y > 0.72|\theta)$ for different values of parameter θ.

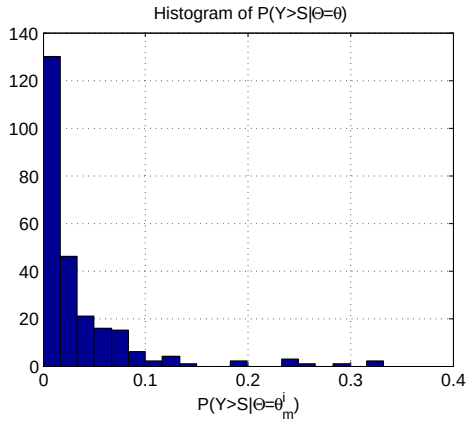


Figure 5.5: Estimates of $\mathbb{P}(Y > 0.72|\Theta = \theta)$ for Θ following π.

For large values of the threshold S the probabilities $\mathbb{P}(Y > S|\theta)$ can be extremely small and the computational cost rather high. For critical values of interest, which can be evaluated in a reasonable time, the SMC^2 outperforms the conventional Monte Carlo method, and bypasses the long time equilibrium convergence of the PMCMC.

5.5.3 Estimation of collision probability between orbiting objects

In this last use case, we consider two spatial objects (a debris and a satellite) orbiting around an Earth centered inertial reference frame. The orbital motion of the spatial objects is simulated using a simplified deterministic dynamical model that may be considered as an input-output function. Their geometry is assumed spherical (*i.e.* the objects have a high tumbling motion when compared with their orbital period) and we assume that we perfectly know the radius of such sphere and the mass of the objects. We wonder whether the distance between the two objects could be smaller than a conflict distance T during a given time span I. SGP4 deterministic model [Miura, 2009] is used to propagate the trajectories of debris and satellite according to the time. This use case may be modeled by an input-output function where:

- the input X represents the position and the speed of the debris space (the position and the speed of the satellite are assumed to be known). X is a 6-dimensional multivariate normal random vector of mean $(\Theta_1, \Theta_2, \Theta_3, \Theta_4, \Theta_5, \Theta_6)^t$ and covariance matrix is equal to the identity matrix. The means corresponds to the debris measurement errors on its position and speed.

- the input-output function ϕ enables to propagate the debris and satellite trajectories with the SGP4 model during I. The input-output code includes the transformation that allows to switch from the standard space of the input to the physical space in which evolve the satellite and debris position and speed.

- The error on the drag coefficient which is considered inside the function ϕ is also random and follows a normal distribution with mean Θ_7 and variance 1.

- the output Y is the minimum distance between the debris and the satellite during I.

The threshold T is set to 20 meters. The probability of interest is in that test case $\mathbb{P}(\phi(X) < T)$. The parameter vector Θ is thus defined by $\Theta = (\Theta_1, \Theta_2, \Theta_3, \Theta_4, \Theta_5, \Theta_6, \Theta_7)^t$. We assume that each component of Θ has a standard normal distribution.

The SMC^2 algorithm has been applied on this example in order to estimate π, the conditional law of Θ given $\phi(X) < T$, with the following parameters: $N_\theta = 1000$, $N_x = 50$, $N^\theta_{app} = 1$, $N^x_{app} = 1$. The intermediate thresholds on the output distance are expressed in meters with $\{200, 100, 66, 50, 40, 33, 28, 25, 22, 20\}$. The estimators of the different marginals of π, obtained with the SMC^2 algorithm, are given in Figure 5.6, where the first marginal is related to the first parameter and so on.

The estimated probabilities $\widehat{\mathbb{P}}(Y < T|\Theta = \theta)$, when Θ follows ν and π are represented in Figure 5.7. The mean probability $\widehat{\mathbb{P}}(Y < T|\Theta = \theta)$ when Θ follows ν is estimated to 3.9 10^{-4}. When $\theta = \sum_{i=1}^{N_\theta} \theta^i_m / N_\theta$, the probability $\widehat{\mathbb{P}}(Y > 0.72|\Theta = \sum_{i=1}^{N_\theta} \theta^i_m / N_\theta)$ is equal to 0.034.

The question is how to analyze the estimated density of π for the tuning of Θ. A possible approach is to consider the Kullback-Leibler distance between the estimated marginal density of π for the parameter Θ_i and the initial marginal density of ν for parameter Θ_i. If the Kullback-Leibler distance is significant for Θ_i, then one can assume that Θ_i has to be finely tuned and conversely. In that case, a misestimation of Θ_i will indeed tend to increase the failure probability. Table 5.8 summaries the different Kullback-Leibler distance obtained for the different components of Θ. The first error component Θ_1 of the position vector seems to be the most influent parameter on $\mathbb{P}(\phi(X) < T)$. On the contrary, the second error component of position

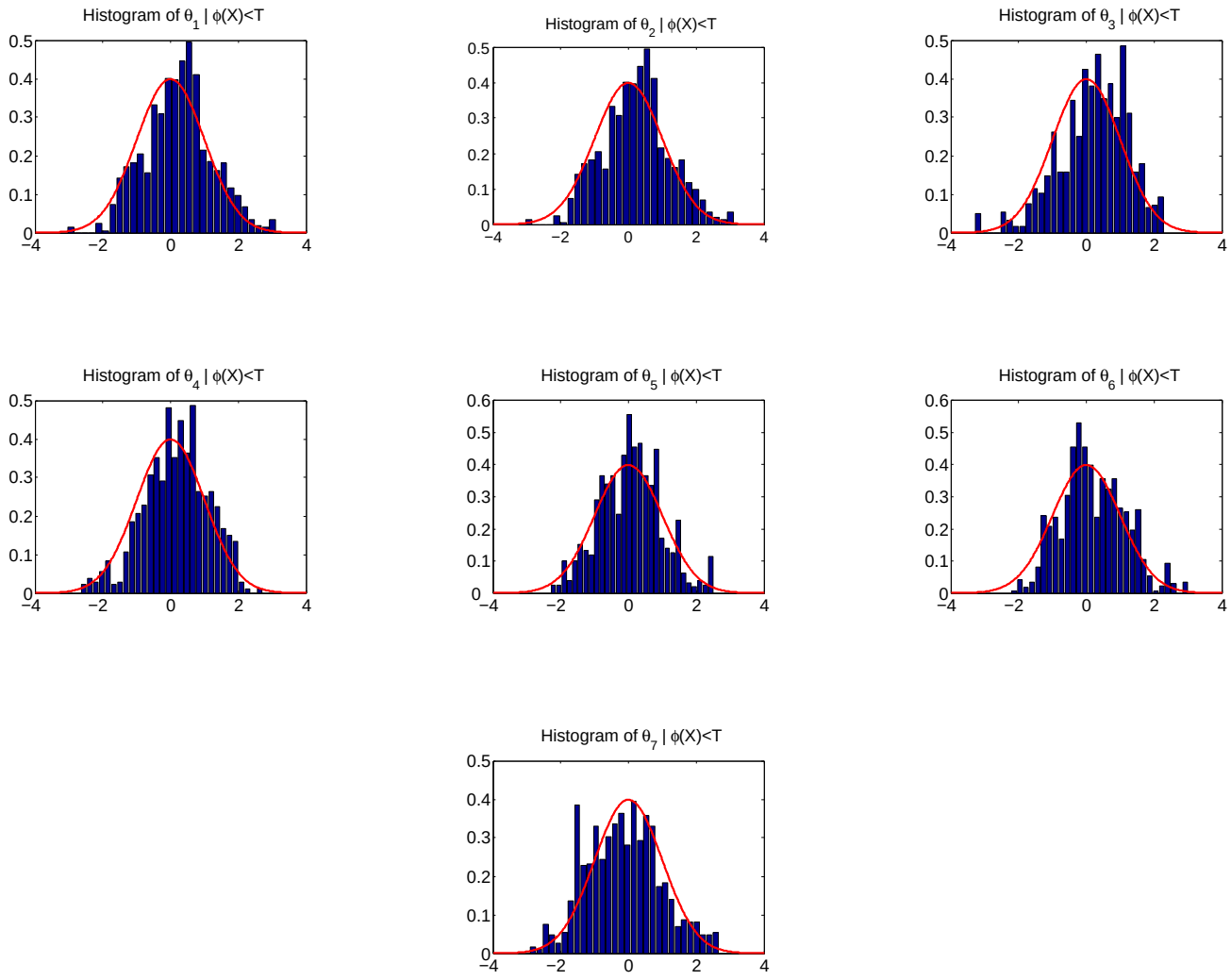


Figure 5.6: Estimations of the marginals of π using the SMC^2 algorithm. The red curve corresponds to the standard normal density.

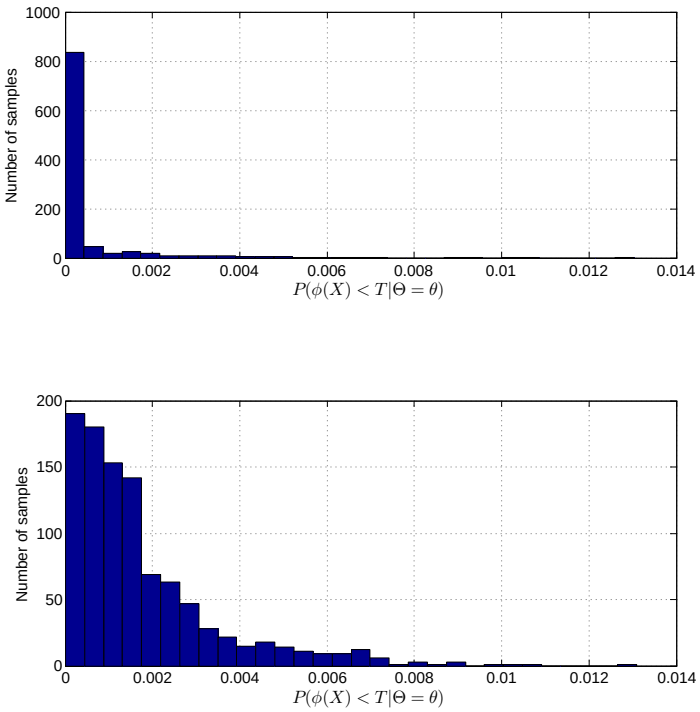


Figure 5.7: Estimates of $\mathbb{P}(Y < T|\theta)$ with Θ following ν and π.

Component of Θ	Kullback-Leibler distance
Θ_1	0.46
Θ_2	0.13
Θ_3	0.30
Θ_4	0.24
Θ_5	0.11
Θ_6	0.25
Θ_7	0.10

Table 5.8: Kullback-Leibler distance between marginal density π and ν for parameters Θ_i.

and speed vector, that are Θ_2 and Θ_5 require a low accuracy since an error on these parameters will tend to decrease the failure probability. In the same way, the density parameter Θ_7 of the drag coefficient does not require also a too fine tuning.

5.6 Conclusion

In this article, we have proposed original methodologies to analyze the influence of the input probability density function (p.d.f.) parameters on a rare failure probability. The proposed methods based on the PMCMC and SMC^2 algorithms have been described in the case of a general problem where the model is a black-box system. Applications illustrate the convergence of these algorithms. One of them deals with the application of this algorithm on the estimation of a launch vehicle booster fallout zone. We firstly show that it is important to estimate input p.d.f. parameters since they influence strongly the value of the output model probability. The conditional law π is estimated with success in that realistic case.
A possible perspective to this work could be the analysis of the particles obtained by the SMC^2 algorithm with Sobol indices [Sobol and Kuchereko, 1993]. A possible ranking of the most influent input p.d.f. parameters could then be derived.

Acknowledgments The authors gratefully acknowledge Juan Carlos Dolado Perez (CNES) for the spatial collision model used in the final application.

Conclusion et perspectives

Nous avons introduit dans cette thèse les modèles d'îlots de particules. La motivation de ce travail était la parallélisation des méthodes SMC, qui n'est pas aisée à cause de l'étape de ré-échantillonnage. Nous avons donc proposé de diviser la population totale de $N = N_1 N_2$ particules en N_1 sous-ensembles appelés *îlots*, chacun de taille réduite, N_2, par rapport à un SMC standard. Ces îlots peuvent par exemple être traités par différents processeurs. Chaque îlot évolue selon un SMC standard en alternant ré-échantillonnage (sélection) et échantillonnage préférentiel (mutation) au niveau des particules. C'est un moyen de paralléliser les méthodes SMC. En revanche, lorsque le nombre N_2 de particules à l'intérieur de chaque îlot est faible, le biais est important et l'interaction des îlots permet de réduire ce biais. Nous avons donc donné des exemples d'interaction des îlots :

- l'échantillonnage systématique (comme dans le *double bootstrap* (B^2)),
- des échantillonnages adaptatifs qui pénalisent moins la parallélisation:
  - le premier est basé sur le coefficient de variation des poids des îlots (le *double bootstrap with adaptive selection on the island level* (B^{2}ASIL)),
  - le second ne ré-échantillonne que les îlots qui ont de faibles poids (comme l'ε-bootstrap).

Nous avons validé ces algorithmes en montrant que les estimateurs induits sont consistants et ciblent les mêmes lois qu'un SMC classique. Ces convergences ont été illustrées sur des exemples en filtrage (cf Chapitre 2). Afin de déterminer quand il est bénéfique de faire interagir les îlots, nous avons établi un critère basé sur l'erreur quadratique moyenne, en fonction des nombres de particules N_2 et d'îlots N_1. Enfin, nous avons prolongé l'étude des propriétés de convergence du B^{2}ASIL. Cette étude est difficile due à la forte dépendance des particules pendant les ré-échantillonnages des îlots et des particules. Mais, la propriété de propagation du chaos (voir [Del Moral, 2004]) suggérant que les particules deviennent asymptotiquement indépendantes quand le nombre total de particules tend vers l'infini, permet d'espérer établir un théorème central limite (TCL) avec un taux de $\sqrt{N}$. Ainsi, nous avons établi la consistance (loi faible des grands nombres), la normalité asymptotique (TCL) et une inégalité de déviation exponentielle pour l'algorithme B^{2}ASIL, en faisant une preuve par récurrence sur les itérations de l'algorithme. Pour ce faire, nous avons décomposé une itération du B^{2}ASIL en différentes opérations élémentaires (la sélection au niveau des îlots, la sélection au niveau des particules, et la mutation des particules) et montré que chacune de ces opérations élémentaires préservent les trois propriétés sus-nommées. Ces preuves se basent sur la définition de tableaux triangulaires au niveau des îlots, l'application des théorèmes de Lindeberg pour les tableaux triangulaires et des inégalités de déviation exponentielle de type Hoeffding. L'intérêt de faire cette décomposition est d'avoir une preuve très générale qui permet d'établir la consistance, la normalité asymptotique et une inégalité de déviation exponentielle pour n'importe quel algorithme qui se décompose en opérations élémentaires (sélection au niveau des îlots, sélection au niveau des particules, mutation). En particulier, on déduit de l'étude du B^{2}ASIL, la consistance,

la normalité asymptotique et une inégalité de déviation exponentielle pour le B^2. Nous montrons la stabilité du B^2 en bornant sa variance asymptotique du B^2, sous des hypothèses souvent utilisées dans la littérature, typiquement vérifiées pour des espaces d'états compacts. Nous étendons ce résultat à des hypothèses plus faibles, comme dans [Douc et al., 2014a]. Dans le chapitre 5, nous avons donné un exemple d'algorithme d'îlots de particules afin d'estimer la loi de paramètres aléatoires conditionnellement à la réalisation d'un évènement rare. Cet algorithme est l'adaptation du SMC^2 introduit par [Chopin et al., 2013] dans le cadre du filtrage bayésien. Sa dénomination s'explique par le fait que deux niveaux de particules imbriquées (de natures diverses) interviennent. Cet algorithme est une version approchée d'un SMC standard où les vraisemblances et les poids d'importance sont incalculables et approchés par des estimateurs sans biais dérivés d'un autre algorithme SMC (comme le *splitting* dans le cas des évènements rares). Nous illustrons la convergence de cet algorithme et nous l'appliquons à deux cas critiques en aérospatiale: l'estimation de la position de retombée d'un étage d'un lanceur et l'estimation de collision entre un satellite et un débris spatial.

Voici quelques perspectives de recherche dans la continuité de ce travail de thèse.

Analyse de sensibilité
Il serait intéressant de faire une étude de sensibilité sur les deux exemples multidimensionnels proposés en fiabilité pour essentiellement deux raisons: d'une part, dans le but de confirmer notre interprétation suite à l'application du SMC^2; d'autre part, afin de réduire la dimension des variables d'entrées, ce qui permettrait d'avoir des codes simplifiés et donc plus rapides d'exécution.

Application du SMC^2 en fiabilité
Les modèles étudiés au chapitre 5 sont de types "boîte noire". Le temps d'exécution de la fonction ϕ dans ce modèle est un critère important pour l'application du SMC^2 en fiabilité en des temps raisonnables. En effet, en considérant N_1 particules en θ et N_2 particules auxiliaires, $2MN_1N_2^M$ appels à la fonction ϕ sont effectués pendant M itérations du SMC^2. Lorsque plusieurs minutes sont nécessaires pour évaluer la fonction ϕ en une seule valeur, le temps d'exécution de l'algorithme devient très grand et n'est pas satisfaisant. Dans ce cas, l'utilisation de modèles de substitution (ou méta-modèles) qui approchent la fonction ϕ pourrait être envisagée, tels que le krigeage ([Kaymaz, 2005]), les machines à vecteurs de support (SVN, [Basudhar et al., 2012; Bourinet et al., 2011]), les méthodes à base de surface de réponse ([Faravelli, 1989; Hurtado, 2004]) ou les réseaux de neurones ([Hurtado and Alvarez, 2001]). Pour le krigeage notamment, l'incertitude liée à l'erreur du méta-modèle peut être quantifiée. Il serait intéressant d'appliquer des modèles de substitution à nos exemples en fiabilité et de quantifier l'impact de l'incertitude du modèle sur l'estimation finale.

Étude des algorithmes d'îlots de particules faisant intervenir l'ε-bootstrap
Dans le chapitre 2, nous avons montré que l'algorithme ε-bootstrap généralise le *bootstrap* dans le cas où $\varepsilon_n^N = 0$ pour tout $n \in \mathbb{N}$, et a une variance plus faible que celle du *bootstrap*. Considérer ce type de sélection au niveau des îlots permet de diminuer l'interaction entre les îlots. En revanche, un TCL et la variance asymptotique en N_1N_2 n'ont pas été établi pour cet algorithme. Cette étude peut se mener de la même manière que celle du B^{2}ASIL. En effet, pour l'ε-bootstrap, avec une certaine probabilité ε_n^N, les îlots ne sont pas sélectionnés et avec la probabilité $1-\varepsilon_n^N$, l'étape multinomial selection on the island level (SIL) est appliquée. Si ε_n^N converge en probabilité quand N tend vers l'infini vers une constante ε_n (ce qui est le cas pour l'algorithme 5 par exemple), la preuve se déroulera de la même manière que celle du théorème 3.4.1.

Variantes des algorithmes d'îlots de particules Dans le chapitre 2, nous avons établi un critère permettant de déterminer que l'interaction des îlots est bénéfique quand

$$N_2 < \frac{B_n^2}{\tilde{V}_n} N_1 \ .$$

Malgré leurs expressions explicites, les quantités B_n^2 et $\tilde{V}_n$ ne sont pas calculables. On pourrait imaginer que ces quantités soient estimées en ligne à chaque itération de l'algorithme, et adapter les nombres d'îlots et de particules, N_1 et N_2 respectivement, de manière à ne jamais ré-échantillonner les îlots. L'inconvénient étant que ces estimations nécessitent la communication des îlots ce qui est contraire à la volonté de les paralléliser. Une autre idée serait de modifier la sélection au niveau des îlots et de dupliquer uniquement les meilleures particules des îlots, au lieu de dupliquer entièrement des îlots.

Table des figures et des tableaux

1.1 Recuit simulé 23

2.1 Comparaison du biais et de la variance 36
2.2 Comparison of different interactions across the islands with bootstrap within each island for the LGM (1) Bootstrap/independent; (2) B²ASIL; (3) B²; (4) Bootstrap/$(1/\|g_n\|_\infty)$)-bootstrap; (5) Bootstrap/essup$_{\eta_p^{N_2}}(g_n)$-bootstrap 53
2.3 Island interactions number using bootstrap within ϵ_p-bootstrap and B² for the LGM. 54
2.4 Island interactions number using B²ASIL and B² for the LGM. 54
2.5 Percentage of the variance gain using bootstrap within ϵ_p-bootstrap on the left side and B²ASIL on the right side, compared to the B², in the LGM example. 54
2.6 Comparison of different interactions across the islands with bootstrap within each island for the Stochastic volatility model (1) Bootstrap/independent; (2) B²ASIL ; (3) B²; (4) Bootstrap/$(1/\|g_n\|_\infty)$)-bootstrap; (5) Bootstrap/essup$_{\eta_p^{N_2}}(g_n)$-bootstrap 55
2.7 Island interactions number using bootstrap within ϵ_p-bootstrap and B² for the Stochastic volatility model. 56
2.8 Island interactions number using B²ASIL and B² for the Stochastic volatility example. 56
2.9 Percentage of the variance gain using bootstrap within ϵ_p-bootstrap on the left side and B²ASIL on the right side, compared to the B², in the Stochastic volatility example. 56

5.1 Estimation of π with PMCMC and SMC² on a toy case, compared to exact density. 133
5.2 Law of Y given $\Theta = 0_4$ estimated with 100000 runs of Monte Carlo simulations. 134
5.3 Estimations of the marginals of π using the SMC² algorithm. 135
5.4 Estimates of the probability $\mathbb{P}(Y > 0.72|\theta)$ for different values of parameter θ. 136
5.5 Estimates of $\mathbb{P}(Y > 0.72|\Theta = \theta)$ for Θ following π. 136
5.6 Estimations of the marginals of π using the SMC² algorithm. The red curve corresponds to the standard normal density. 138
5.7 Estimates of $\mathbb{P}(Y < T|\theta)$ with Θ following ν and π. 139
5.8 Kullback-Leibler distance between marginal density π and ν for parameters Θ_i. 140

Références

Anderson, B. D. O. and Moore, J. B. (1979). *Optimal Filtering*. Prentice-Hall. 15

Andrieu, C., Doucet, A., and Holenstein, R. (2010). Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, 72(3):269–342. 22, 23, 117, 119, 120, 122, 126

Arnaud, E. and Le Gland, F. (2009). SMC with adptive resampling:large sample asymptotics. pages 481–484. 16, 28, 34, 49

Asmussen, S. (2003). *Applied probability and queues*, volume 51. Springer. 21, 116, 119

Au, S., Ching, J., and Beck, J. (2007). Application of subset simulation methods to reliability benchmark problems. *Structural Safety*, 29:183–193. DOI :10.1016/j.strusafe.2006.07.008. 116, 119

Au, S.-K. and Beck, J. (2001). Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics, Elsevier*, 16:263–277. 116, 119

Banerjee, I., Pal, S., and Maiti, S. (2010). Computationally efficient black-box modeling for feasibility analysis. *Computers & Chemical Engineering*, 34(9):1515 – 1521. 121

Basudhar, A., Dribusch, C., Lacaze, S., and Missoum, S. (2012). Constrained efficient global optimization with support vector machines. *Structural and Multidisciplinary Optimization*, 46(2):201–221. 142

Beck, J. V. and Arnold, K. J. (1977). *Parameter estimation in engineering and science*. James Beck. 117, 120

Beven, K. and Kirkby, M. (1979). A physically based, variable contributing area model of basin hydrology/un modèle à base physique de zone d'appel variable de l'hydrologie du bassin versant. *Hydrological Sciences Journal*, 24(1):43–69. 117, 120

Billio, M., Casarin, R., Ravazzolo, F., and van Dijk, H. K. (2013). Time-varying combinations of predictive densities using nonlinear filtering. *Journal of Econometrics*, 177(2):213 – 232. 16, 34

Bjerager, R. (1991). *Methods for structural reliability computation*, pages 89–136. Springer Verlag, New York. 116, 120, 121

Blanchet, J. H., Leder, K., and Glynn, P. W. (2009). Efficient simulation of light-tailed sums: an old-folk song sung to a faster new tune... In *Monte Carlo and Quasi-Monte Carlo Methods 2008*, pages 227–248. Springer. 21, 116, 119

Botev, Z. I. and Kroese, D. P. (2012). Efficient Monte-Carlo simulation via the generalized splitting method. *Statistics and Computing*, 22(1):1–16. 121

Bourinet, J.-M., Deheeger, F., and Lemaire, M. (2011). Assessing small failure probabilities by combined subset simulation and support vector machines. *Structural Safety*, 33(6):343–353. 142

Bréhier, M., Lelièvre, T., and Rousset, M. (2014). Analysis od adaptive multilevel splitting algorithms in an idealized case. *To appear in ESAIM P&S*. 17

Brockwell, A., Del Moral, P., and Doucet, A. (2010). Sequentially interacting Markov chain Monte Carlo methods. *Ann. Statist.*, 38(6):3387–3411. 17, 35

Bucklew, J. A. (2004). *Introduction to Rare Event Simulation*. Springer. 121

Cappé, O. and Moulines, E. (2005). On the use of particle filtering for maximum likelihood parameter estimation. In *European Signal Processing Conference (EUSIPCO)*, Antalya, Turkey. 16, 37, 66

Cappé, O., Moulines, E., and Rydén, T. (2005). *Inference in Hidden Markov Models*. Springer. 24, 26, 74, 80

Casarin, R., Grassi, S., Ravazzolo, F., and Van Dijk, H. K. (2013). Parallel Sequential Monte Carlo for efficient Density Combination : the deco matlab toolboox. *Tinbergen Institute Discussion Paper*, 055/III. 16, 34, 38

Cérou, F., Del Moral, P., Furon, T., and Guyader, A. (2012). Sequential Monte Carlo for rare event estimation. *Stat. Comput.*, 22(3):795–808. 17, 26, 66, 74, 116, 119, 121, 132

Cérou, F., Del Moral, P., Guyader, A., LeGland, F., Lezaud, P., and Topart, H. (2006). Some recent improvements to importance splitting. In *preliminary version), Proceedings of 6th International Workshop on Rare Event Simulation, Bamberg, Germany*. 17, 123

Cérou, F., Del Moral, P., Le Gland, F., and Lezaud, P. (2006). Genetic genealogical models in rare event analysis. 1. 17, 116, 119, 123

Cérou, F. and Guyader, A. (2007). Adaptive multilevel splitting for rare event analysis. *Stochastic Analysis and Applications*, 25(2):417–443. DOI: 10.1080/07362990601139628. 17, 116, 119, 125

Cérou, F., LeGland, F., Del Moral, P., and Lezaud, P. (2005). Limit theorems for the multilevel splitting algorithm in the simulation of rare events. In *Simulation Conference, 2005 Proceedings of the Winter*, pages 10–pp. IEEE. 17, 125

Cho, K.-H., Shin, S.-Y., Kolch, W., and Wolkenhauer, O. (2003). Experimental design in systems biology, based on parameter sensitivity analysis using a monte carlo method: A case study for the tnfα-mediated nf-κ b signal transduction pathway. *Simulation*, 79(12):726–739. 117, 120

Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika*, 89(3):539–552. 16, 17, 37, 66, 117, 127

Chopin, N. (2004). Central limit theorem for sequential Monte Carlo methods and its application to Bayesian inference. *Ann. Statist.*, 32(6):2385–2411. 16, 28, 64, 67

Chopin, N., Jacob, P. E., and Papaspiliopoulos, O. (2013). SMC^2: an efficient algorithm for sequential analysis of state space models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 75(3):397–426. 17, 40, 66, 117, 119, 120, 127, 142

Crisan, D. and Doucet, A. (2002). A survey of convergence results on particle filtering methods for practitioners. IEEE Transactions on Signal Processing. 64

Crisan, D. and Rozovskii, B. L., editors (2011). *The Oxford handbook of nonlinear filtering*. Oxford N.Y. Oxford University Press. 66

Del Moral, P. (2004). *Feynman-Kac Formulae. Genealogical and Interacting Particle Systems with Applications*. Springer. 13, 16, 18, 25, 28, 34, 35, 37, 38, 42, 44, 46, 48, 58, 64, 65, 67, 68, 78, 80, 124, 141

Del Moral, P., Doucet, A., and Jasra, A. (2012). On adaptive resampling strategies for sequential Monte Carlo methods. *Bernoulli*, 18(1):252–278. 16, 28, 34, 49

Del Moral, P. and Garnier, J. (2005). Genealogical particle analysis of rare events. *Ann. Appl. Probab.*, 15(4):2496–2534. 15, 26, 66, 74

Del Moral, P. and Guionnet, A. (1999). Central limit theorem for nonlinear filtering and interacting particle systems. *Ann. Appl. Probab.*, 9(2):275–297. 16, 28, 64, 67

Del Moral, P. and Guionnet, A. (2001). On the stability of interacting processes with applications to filtering and genetic algorithms. *Annales de l'Institut Henri Poincaré*, 37:155–194. 65, 68, 80

Del Moral, P., Hu, P., and Wu, L. (2012). On the concentration properties of interacting particle processes. *Foundations and Trends in Machine Learning*, 3(3-4):225–289. 16, 28, 37, 38, 42, 128

Del Moral, P. and Miclo, L. (2001). Genealogies and increasing propagation of chaos for feynman-kac and genetic models. *Ann. Appl. Probab.*, 11:1166–1198. 17

Douc, R., Garivier, A., Moulines, E., and Olsson, J. (2011). Sequential Monte Carlo smoothing for general state space hidden Markov models. *Ann. Appl. Probab.*, 21(6):2109–2145. 31, 69, 90, 91, 92

Douc, R. and Moulines, E. (2008). Limit theorems for weighted samples with applications to sequential Monte Carlo methods. *Ann. Statist.*, 36(5):2344–2376. 16, 18, 28, 34, 49, 64, 65, 67, 70, 81, 82, 96, 97

Douc, R., Moulines, E., and Olsson, J. (2009). Optimality of the auxiliary particle filter. *Probab. Math. Statist.*, 29(1):1–28. 79, 80

Douc, R., Moulines, E., and Olsson, J. (2014a). Long-term stability of sequential Monte Carlo methods under verifiable conditions. *Ann. Appl. Probab.*, 24(5):1767–1802. 65, 68, 80, 81, 93, 94, 142

Douc, R., Moulines, E., and Stoffer, D. (2014b). *Nonlinear Time Series: Theory, Methods and Applications with R Examples*. Chapman & Hall/CRC Texts in Statistical Science. 13, 80, 101

Doucet, A., De Freitas, N., and Gordon, N., editors (2001). *Sequential Monte Carlo Methods in Practice*. Springer, New York. 13, 15, 16, 25, 37, 66, 116, 119

Durham, G. and Geweke, J. (2011). Massively parallel sequential monte carlo for bayesian inference. *Manuscript, URL http://www. censoc. uts. edu. au/pdfs/geweke_papers/gp_ working_9. pdf. Nalan Bastürk, Lennart Hoogerheide, Anne Opschoor, Herman K. van Dijk*, 29. 17, 34, 38

Engelund, S. and Rackwitz, R. (1993). A benchmark study on importance sampling techniques in structural reliability. *Structural Safety*, 12(4):255–276. 116, 119

Faravelli, L. (1989). Response-surface approach for reliability analysis. *Journal of Engineering Mechanics*, 115(12):2763–2781. 142

Fisher, R. and Tippett, L. (1928). On the estimation of the frequency distributions of the largest or smallest member of a sample. Proceedings of the Cambridge Philosophical Society. 116, 120

Fishman, G. (1995). *Monte Carlo*. Springer. 116, 119

Garvels, M. J. and Kroese, D. P. (1998). A comparison of restart implementations. In *Simulation Conference Proceedings, 1998. Winter*, volume 1, pages 601–608. IEEE. 17

Garvels, M. J. J. (2000). *The splitting method in rare event simulation*. PhD thesis. 17, 116, 119

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 6:721–741. 22, 122

Gilks, W. R. and Berzuini, C. (2001). Following a moving target - monte carlo inference for dynamic bayesian models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(1):127–146. 23

Glasserman, P. (2004). *Monte Carlo methods in financial engineering*, volume 53 of *Applications of Mathematics (New York)*. Springer-Verlag, New York. Stochastic Modelling and Applied Probability. 21, 116, 119

Glasserman, P., Heidelberger, P., and Shahabuddin, P. (2000). Variance reduction techniques for estimating value-at-risk. 46. 116, 119

Glasserman, P., Heidelberger, P., Shahabuddin, P., and Zajic, T. (1999). Multilevel splitting for estimating rare event probabilities. *Informs*, pages 585–600. 17, 116, 119

Gogate, V. and Dechter, R. (2011). Samplesearch: Importance sampling in presence of determinism. *Artificial Intelligence*, 175(2):694 – 729. 116, 119

Gordon, N., Salmond, D., and Smith, A. F. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc. F, Radar Signal Process.*, 140:107–113. 15, 16, 27, 116, 119

Grancharova, A., Kocijan, J., and Johansen, T. A. (2011). Explicit output-feedback nonlinear predictive control based on black-box models. *Engineering Applications of Artificial Intelligence*, 24(2):388–397. 121

Handschin, J. and Mayne, D. (1969). Monte Carlo techniques to estimate the conditionnal expectation in multi-stage non-linear filtering. In *Int. J. Control*, volume 9, pages 547–559. 15

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their application. *Biometrika*, 57:97–109. 22, 122

Hull, J. and White, A. (1987). The pricing of options on assets with stochastic volatilities. *J. Finance*, 42:281–300. 24

Hurtado, J. E. (2004). An examination of methods for approximating implicit limit state functions from the viewpoint of statistical learning theory. *Structural Safety*, 26(3):271–293. 142

Hurtado, J. E. and Alvarez, D. A. (2001). Neural-network-based reliability analysis: a comparative study. *Computer methods in applied mechanics and engineering*, 191(1):113–132. 142

Jasra, A., Doucet, A., Stephens, D. A., and Holmes, C. C. (2008). Interacting sequential Monte Carlo samplers for trans-dimensional simulation. *Comp. Stat. Data Anal.*, 52:1765–1791. 17, 35, 38

Jazwinski, A. (1970). *Stochastic processes and filtering theory*. Academic Press, New York. 15

Jourdain, B. and Lelong, J. (2009). Robust adaptive importance sampling for normal random vectors. *Ann. Appl. Probab.*, 19(5):1687–1718. 21, 116, 119

Kagan, Y. and Knopoff, L. (1977). Earthquake risk prediction as a stochastic process. *Physics of the Earth and Planetary Interiors*, 14(2):97–108. 116, 119

Kahn, H. and Harris, T. (1951). Estimation of particle transmission by random sampling. *Appl. Math. Series*, 12:27–30. 17, 116, 119, 123

Kalman, R. E. and Bucy, R. (1961). New results in linear filtering and prediction theory. *J. Basic Eng., Trans. ASME, Series D*, 83(3):95–108. 15

Kaymaz, I. (2005). Application of kriging method to structural reliability problems. *Structural Safety*, 27(2):133–151. 142

Keane, A. J. and Nair, P. B. (2005). *Computational Approaches for Aerospace Design*. John Wiley & Sons, Ltd. 121

Kitagawa, G. (1987). Non-Gaussian state space modeling of nonstationary time series. *Journal of the American Statistical Association*, 82(400):1023–1063. 15

Kong, A., Liu, J. S., and Wong, W. (1994). Sequential imputation and Bayesian missing data problems. *J. Am. Statist. Assoc.*, 89(278-288):590–599. 64, 75

Kreinovich, V. and Ferson, S. (2004). A new Cauchy-based black-box technique for uncertainty in risk analysis. *Reliability Engineering & System Safety*, 85(1-3):267–279. 121

Kroese, D. P. and Rubinstein, R. Y. (2012). Monte-Carlo methods. *Wiley Interdisciplinary Reviews: Computational Statistics*, 4(1):48–58. 116, 119

Künsch, H. R. (2000). State space and hidden Markov models. In Barndorff-Nielsen, O. E., Cox, D. R., and Kluppelberg, C., editors, *Complex Stochastic Systems*. CRC Press. 15

Künsch, H. R. (2005). Recursive Monte-Carlo filters: algorithms and theoretical analysis. *Ann. Statist.*, 33(5):1983–2021. 28, 64, 67

Lagnoux, A. (2006). Rare event simulation. *Probability in the Engineering and Informational Sciences*, 20(1). 116, 119

Lagnoux-Renaudie, A. (2009). A two-step branching splitting model under cost constraint for rare event analysis. *Journal of Applied Probability*, 46(2):429–452. 116, 119

Leadbetter, M., Lindgren, G., and Rootzén, H. (1983). *Extremes and related properties of random sequences and series*. Springer Verlag, New York, USA. 116, 120

Leadbetter, M. and Rootzén, H. (1988). Extremal theory for stochastic processes. *Annals of Probability*, 16:431–478. 116, 120

L'Ecuyer, P., Demers, V., and Tuffin, B. (2006). Splitting for rare-event simulation. In *Simulation Conference, 2006. WSC 06. Proceedings of the Winter*, pages 137–148. IEEE. 125

L'Ecuyer, P., Demers, V., and Tuffin, B. (2007). Rare events, splitting, and quasi-monte carlo. *ACM Transactions on Modeling and Computer Simulation*, 17(2). 17, 21

Lee, A. and Whiteley, N. (2014). Forest resampling for distributed sequential monte carlo. *arXiv preprint arXiv:1406.6010*. 35

Lee, A., Yau, C., Giles, M. B., Doucet, A., and Holmes, C. C. (2010). On the utility of graphics cards to perform massively parallel simulation of advanced monte carlo methods. *Journal of computational and graphical statistics*, 19(4):769–789. 17, 34

LeGland, F. and Oudjane, N. (2004). Stability and uniform approximation of nonlinear filters using hilbert metric and aplication to particle filters. *Ann. Appl. Probab.*, 14(1):144 –187. 64

Lemaire, M., Chateauneuf, A., and Mitteau, J. (2009). *Structural reliability*. Wiley Online Library. 117, 120

Lindsten, F., Douc, R., and Moulines, E. (2014). Uniform ergodicity of the particle gibbs sampler. *arXiv preprint arXiv:1401.0683*. 126

Liu, J. (2008). *Monte Carlo Strategies in Scientific Computing*. Springer, New York. 16, 28, 34, 49, 64, 75

Liu, J. and Chen, R. (1995). Blind deconvolution via sequential imputations. *J. Am. Statist. Assoc.*, 90(420):567–576. 16, 28, 34, 49

Liu, J. S. (1996). Metropolized independent sampling with comparisons to rejection sampling and importance sampling. *Stat. Comput.*, 6:113–119. 64, 75

MacDonald, I. and Zucchini, W. (2009). *Hidden Markov models for time series: an introduction using R*. CRC Press. 24

Madsen, H., Krenk, S., and Lind, N. C. (1986). *Methods of structural safety*. Springer-Verlag, Englewood Cliffs. 116, 120

Melas, V. B. (1997). On the efficiency of the splitting and roulette approach for sensitivity analysis. In *Proceedings of the 29th Conference on Winter Simulation*, pages 269–274. IEEE Computer Society. 116, 119

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 23(2):1087–1092. 22, 122

Miura, N. Z. (2009). *Comparison and design of Simplified General Perturbation Models (SGP4) and code for NASA Johnson Space Center, Orbital Debris Program Office*. PhD thesis, Faculty of California Polytechnic State University. 137

Morio, J. (2011a). Influence of input pdf parameters of a model on a failure probability estimation. *Simulation Modelling Practice and Theory*, 19(10):2244–2255. 116, 117, 119, 120

Morio, J. (2011b). Non-parametric adaptive importance sampling for the probability estimation of a launcher impact position. *Reliability Engineering and System Safety*, 96(1):178 – 183. 133

Morio, J., Balesdent, M., Jacquemart, D., and Vergé, C. (2014). A survey of rare event estimation methods for statistic input-output models. *Simulation Modelling Practice and Theory*, 49:287 – 304. 116, 120

Munoz Zuniga, M., Garnier, J., Remy, E., and de Rocquigny, E. (2011). Adaptive directional stratification for controlled estimation of the probability of a rare event. *Reliability Engineering & System Safety*, 96(12):1691 – 1712. 117, 120

Olsson, J., Cappé, O., Douc, R., and Moulines, E. (2008). Sequential Monte Carlo smoothing with application to parameter estimation in non-linear state space models. *Bernoulli*, 14(1):155–179. arXiv:math.ST/0609514. 80

Olsson, J. and Rydén, T. (2004). The bootstrap particle filtering bias. 28

Orsak, G. C. and Aazhang, B. (1995). A class of optimum importance sampling strategies. *Information Sciences*, 84(1-2):139–160. 116, 119

Parekh, S. and Walrand, J. (1989). A quick simulation method for excessive backlogs in networks of queues. *Automatic Control, IEEE Transactions on*, 34(1):54–66. 116, 119

Pitt, M. K. and Shephard, N. (1999). Filtering via simulation: Auxiliary particle filters. *J. Am. Statist. Assoc.*, 94(446):590–599. 78, 79

Prandini, M. and Watkins, O. (2005). Probabilistic aircraft conflict detection. *HYBRIDGE, IST-2001*, 32460. 116, 119

Richard, J.-F. and Zhang, W. (2007). Efficient high-dimensional importance sampling. *Journal of Econometrics*, 141(2):1385–1411. 116, 119

Ristic, B., Arulampalam, M., and Gordon, A. (2004). *Beyond Kalman Filters: Particle Filters for Target Tracking*. Artech House. 66

Robert, C. P. and Casella, G. (2004). *Monte Carlo Statistical Methods*. Springer, New York, 2nd edition. 17, 22, 35, 122

Rosenbluth, M. and Rosenbluth, A. (1955). Monte-Carlo calculation of the average extension of molecular chains. *J. Chem. Phys*, 356:356–359. 27, 116, 119

Rubino, G., Tuffin, B., et al. (2009). *Rare event simulation using Monte Carlo methods*. Wiley Online Library. 21

Rubinstein, R. Y. (1981). *Simulation and the Monte-Carlo method*. 116, 119

Rubinstein, R. Y. and Kroese, D. P. (2004). *The Cross-Entropy Method: a unified approach to combinatorial optimization, Monte Carlo Simulation and Machine Learning*. Springer. 121

Saltelli, A., Tarantola, S., Campolongo, F., and Ratto, M. (2004). *Sensitivity analysis in practice: A guide to assessing scientific models*. John Wiley & Sons, Chichester, England. 120

Sjöberg, L. and Drottz-Sjöberg, B. (1991). Knowledge and risk perception among nuclear power plant employees. *Risk Analysis*, 11:607–618. 116, 119

Sobol, I. and Kuchereko, S. (1993). Sensitivity estimates for non linear mathematical models. *Mathematical Modelling and Computationnal Experiments*, 1:407–414. 140

Sobol, I. M. (1994). *A Primer for the Monte Carlo Method*. CRC Press, Boca Raton, Fl. 121

Stratonovich, R. L. (1960). Conditional Markov processes. *Theory Probab. Appl.*, 5(2):156–178. 15

Suchard, M. A. and Rambaut, A. (2009). Many-core algorithms for statistical phylogenetics. *Bioinformatics*, 25(11):1370–1376. 16, 34

Tuffin, B. and Trivedi, K. S. (2000). Implementation of importance splitting techniques in stochastic petri net package. In *Computer Performance Evaluation. Modelling Techniques and Tools*, pages 216–229. Springer. 125

Vergé, C., Del Moral, P., Moulines, E., and Olsson, J. (2014). Convergence properties of weighted particle islands with application to the double bootstrap algorithm. Preprint. 13, 18, 19, 35, 65

Vergé, C., Del Moral, P., Moulines, E., and Olsson, J. (2014). Supplementary paper on convergence properties of weighted particle islands with application to the double bootstrap algorithm. Preprint. 18, 93, 96

Vergé, C., Dubarry, C., Del Moral, P., and Moulines, E. (2015a). On parallel implementation of sequential Monte Carlo methods: the island particle model. *Statistics and Computing*, 25(2):243–260. 13, 18, 35, 36, 64, 66, 67, 75, 78, 79, 119, 120, 130

Vergé, C., Morio, J., and Del Moral, P. (2015b). An island Particle Markov Chain Monte Carlo algorithm for safety analysis. Preprint. 14, 18, 118

Villén-Altamirano, M. and Villén-Altamirano, J. (1994). Restart: a straightforward method for fast simulation of rare events. In *Simulation Conference Proceedings, 1994. Winter*, pages 282–289. IEEE. 17, 125

Wang, F. and Landau, D. P. (2001). Efficient, multiple-range random walk algorithm to calculate the density of states. *Phys. Rev. Lett.*, 86:2050–2053. 17, 35

Wheeler, W. C. (1995). Sequence alignment, parameter sensitivity, and the phylogenetic analysis of molecular data. *Systematic Biology*, 44(3):321–331. 117, 120

Whiteley, N., Lee, A., and Heine, K. (2013). On the role of interaction in sequential monte carlo algorithms. *arXiv preprint arXiv:1309.2918*. 35

Worden, K., Wong, C., Parlitz, U., Hornstein, A., Engster, D., Tjahjowidodo, T., Al-Bender, F., Rizos, D., and Fassois, S. (2007). Identification of pre-sliding and sliding friction dynamics: Grey box and black-box models. *Mechanical Systems and Signal Processing*, 21(1):514–534. 121

Xu, K., Wang, Y., Wang, F., Liao, Y., Zhang, Q., Li, H., and Zheng, X. (2014). Neural decoding using a parallel sequential Monte Carlo method on point processes with ensemble effect. *BioMed research international*. 16, 34

Yan-Gang, Z. and Tetsuro, O. (1999). A general procedure for first/second-order reliability method (FORM/SORM). *Structural Safety*, 21(2):95–112. 116, 120

Zhang, P. (1996). Nonparametric importance sampling. *Journal of the American Statistical Association*, 91(434):1245–1253. 121

Modèle d'îlots de particules et application en fiabilité

Les modèles de Feynman-Kac (généralisant les modèles de Markov cachés) sont aujourd'hui très largement utilisés afin de modéliser une grande diversité de séries temporelles dans différents domaines tels que l'aéronautique, l'analyse d'évènements rares, le traitement du signal, les mathématiques financières, la biologie, etc. La complexité croissante de ces modèles a conduit au développement d'approximations via différentes méthodes de Monte-Carlo, dont les méthodes de Monte-Carlo par chaînes de Markov (MCMC) et les méthodes de Monte-Carlo séquentielles (SMC). Les méthodes SMC appliquées au filtrage particulaire sont au centre de cette thèse. Elles consistent à approcher la loi d'intérêt à l'aide d'une population de particules en interaction définies séquentiellement. De nombreux algorithmes ont déjà été développés et étudiés dans la littérature. Nous proposons des techniques de parallélisation des méthodes SMC, en considérant des sous-populations de particules appelées îlots qui peuvent également interagir entre elles. Nous étudions les propriétés de convergence de ces algorithmes d'îlots de particules. En particulier, nous démontrons un théorème central limite (TCL) et la stabilité de la variance des estimateurs induits, grâce à des inégalités de déviation exponentielle et des tableaux triangulaires définis au niveau des îlots. Nous proposons également un nouvel algorithme de type îlots de particules en interaction pour estimer la loi de paramètres aléatoires conditionnellement à la réalisation d'un évènement rare. Nous illustrons sa convergence et nous l'appliquons à deux cas critiques en aérospatiale.

Mots-clés : APPROXIMATION PARTICULAIRE DE MESURES DE FEYNMAN-KAC ; MODELES D'ILOTS DE PARTICULES ; 6 CALCUL EN PARALLELE ; METHODES DE MONTE-CARLO SEQUENTIELLES ; FILTRAGE

Island particle models and their application in reliability

Feynman-Kac models (which generalize hidden Markov models) are nowadays widely used as they allow to model a large variety of time series in several fields such as aeronautics, rare event analysis, signal processing, finance, biology, and so on. Different approximations based on Monte Carlo principles have been developed as Markov chain Monte Carlo (MCMC) and sequential Monte Carlo (SMC). In this thesis, we focus on SMC methods. They consist in approximating a targeted law through an interacting particle system sequentially defined. Numerous algorithms have been developed and studied in the literature.
We propose techniques of parallelization of such SMC methods, considering subpopulations of particles referred to by us as islands which can also interact.We study convergence properties of these island particle algorithms.
Especially, we prove a central limit theorem (CLT) and the stability of the variance, thanks to exponential deviation inequality and triangular arrays de ned on the island level. We also propose a novel algorithm of interacting island particles to estimate the law of random parameters conditionally to a rare event. We illustrate its convergence and we apply it to two critical cases in aerospace.

Keywords : PARTICLE APPROXIMATION OF FEYNMAN-KAC FLOW ; ISLAND PARTICLE MODELS ; PARALLEL IMPLEMENTATION ; SEQUENTIAL MONTE CARLO METHODS ; FILTERING

Office National d'Études et de Recherches Aérospatiales
BP 72 - 29 avenue de la Division Leclerc
92322 CHATILLON CEDEX
Tél. : +33 1 46 73 40 40 - Fax : +33 1 46 73 41 41
http://www.onera.fr