Sequential Bayesian Computation

Hai Dang Dau

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2022IPPAG006

Thèse de doctorat

ENSAE

IP PARIS

Sequential Bayesian Computation

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à l'Ecole nationale de la statistique et de l'administration économique

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

Thèse présentée et soutenue à Palaiseau, le 29 septembre 2022, par

HAI-DANG DAU

Rapporteurs :

Anthony Lee
Professor, University of Bristol

Pierre del Moral
Directeur de recherche, INRIA (Bordeaux Research Center)

Composition du Jury :

Christian Robert Professeur, Université Paris-Dauphine (UMR 7534)	Président
Anthony Lee Professor, University of Bristol	Rapporteur
Stéphanie Allassonnière Professeur, Université de Paris (UMRS 1138 team 22)	Examinatrice
Randal Douc Professeur, Institut Polytechnique de Paris (Télécom Sudparis)	Examinateur
Arnaud Doucet Professor, University of Oxford	Examinateur
Nicolas Chopin Professeur, Institut Polytechnique de Paris (ENSAE)	Directeur de thèse


Fondation mathématique
FMJH
Jacques Hadamard


CREST
CENTER FOR RESEARCH
IN ECONOMICS AND STATISTICS

Remerciements

Tout d'abord, je voudrais adresser mes remerciements les plus sincères à mon directeur de thèse Nicolas Chopin. De vos connaissances techniques profondes et votre culture générale très étendue (en statistique, mais aussi dans la vie), à votre disponibilité pour les étudiants (en particulier les doctorants), vous m'avez beaucoup aidé et vous m'avez beaucoup appris. J'ai également apprécié votre effort pour organiser des séminaires comme Laplace's Demon ou Monte Plateau qui ont énormément élargi ma connaissance et créé de belles occasions pour échanger. La thèse s'est déroulée sous le signe du covid; cette mémoire a été rédigée dans une condition particulière à cause du Brexit. Face à tout cela, je n'aurais pas pu m'en sortir sans un directeur de thèse toujours à l'écoute et toujours prêt à agir.

Je voudrais remercier l'X de m'avoir attribué la bourse AMX. Le laboratoire CREST est l'endroit où j'ai passé mes trois années. Je remercie ses deux directeurs que j'ai pu connaître au cours de mon séjour, Francis Kramarz et Arnak Dalalyan. Je suis reconnaissant envers les chercheurs et doctorants du laboratoire qui contribuent tous à son ambiance très agréable. J'ai bénéficié aussi de l'aide du personnel administratif du CREST qui a su résoudre efficacement mes divers problèmes ; merci à vous.

Je voudrais remercier l'ensemble du jury et les rapporteurs d'avoir accepté d'évaluer ma thèse. I would like to thank the jury members and the reviewers for accepting to evaluate my thesis.

Merci à Charles-Henry, Martine et Antoine d'être à mon côté depuis mon arrivée en France. Je consacre les derniers mots à mes parents :

Con cám ơn bố mẹ đã luôn ở bên con trong suốt thời gian qua. Luận án này xin dành tặng bố mẹ.

Contents

INTRODUCTION EN FRANÇAIS

Résumé. Après une présentation générale de l'inférence bayésienne et ses méthodes, nous faisons une introduction aux deux sujets principaux de la thèse, à savoir les échantillonneurs séquentiels par Monte-Carlo et les algorithmes de lissage, suivie d'un résumé de nos contributions pour chacun.

1. L'inférence bayésienne et ses méthodes

1.1. **La formulation du problème.** Cette section présente de manière très succincte le cadre de la statistique bayésienne, en mettant l'accent sur ce qui est nécessaire pour la suite. Pour une introduction plus complète, nous conseillons vivement le livre de Robert (2007).

L'estimation de paramètres est un problème fondamental de la statistique. Un paramètre θ (dont l'espace auquel il appartient est souvent noté par Θ) capture une explication possible de la dynamique derrière la génération des données y. Plus formellement, à chaque θ est associée une mesure de probabilité (notée soit par $\mathbb{P}_\theta$, soit par p_θ ou encore par $p(\cdot|\theta)$) définie sur l'espace des observations $\mathcal{Y}$. Très souvent, il existe une mesure $\lambda(\mathrm{d}y)$ qui domine toutes les $(\mathbb{P}_\theta)_{\theta\in\Theta}$ au sens de Radon-Nikodym, ce qui implique, pour chaque θ, l'existence d'une densité de $\mathbb{P}_\theta$ par rapport à λ. Par abus de notation, la lettre p est recyclée et ladite densité est notée $p_\theta(y)$ ou encore $p(y|\theta)$.

Supposons que les données y aient été générées par un paramètre inconnu θ^*. Deux approches principales existent pour extraire des informations sur θ^* à partir des données y : l'approche fréquentiste et l'approche bayésienne. Le principe phare de la première consiste à maximiser la vraisemblance, c'est-à-dire à chercher

$$\hat{\theta} := \arg\max_\theta p_\theta(y).$$

Ainsi, les techniques d'optimisation jouent un rôle capital dans l'approche fréquentiste. L'incertitude est ensuite communiquée aux utilisateurs à l'aide d'un intervalle de confiance $[\hat{\theta}_\ell, \hat{\theta}_h]$, par ex. de niveau 95% :

$$\mathbb{P}_{\theta^*}\left([\hat{\theta}_\ell, \hat{\theta}_h] \ni \theta^*\right) = 0.95.$$

Malheureusement, il n'existe pas de technique universelle pour construire $\hat{\theta}_\ell$ et $\hat{\theta}_h$. On peut se baser sur des fonctions dites « pivotales » ou des théorèmes de normalité asymptotique, mais le détail exact varie d'un problème à l'autre.

La statistique bayésienne exige une structure supplémentaire sur Θ, celle d'une distribution de probabilité à priori $\pi_0(\mathrm{d}\theta)$. Elle admet plusieurs interprétations, mais la plus courante est qu'elle représente une information dont on dispose avant le recueil des données sur une région probable de θ^*. Si une telle information n'est pas disponible, on prend souvent une loi à priori très générique, par ex. une loi normale de grande variance ou une loi uniforme sur Θ si celui-ci est borné. Le

point crucial est que désormais, (θ, y) peut être vu comme un couple de variables aléatoires dont la loi jointe est donnée par

$$p(\mathrm{d}\theta, \mathrm{d}y) := \pi_0(\mathrm{d}\theta)p(\mathrm{d}y|\theta)$$

où l'on acceptera à nouveau un abus de notation. Si le modèle est dominé par une mesure λ, la formule de Bayes implique alors

$$p(\mathrm{d}\theta|y) = \frac{p(y|\theta)\pi_0(\mathrm{d}\theta)}{Z} \tag{1}$$

où

$$Z = \int p(y|\theta)\pi_0(\mathrm{d}\theta)$$

est une constante de normalisation que l'on ne connaît pas en général. A la différence de l'inférence fréquentiste, le résultat de l'estimation n'est pas communiqué à l'usager à travers un certain $\hat{\theta}$, mais est donné sous la forme d'une distribution aléatoire. Celle-ci s'appelle la distribution a posteriori et est en l'occurrence $p(\mathrm{d}\theta|y)$. Elle a l'avantage de prendre en compte, par construction, l'incertitude. De plus, une fois que la distribution a priori a été choisie, la distribution a posteriori (1) est automatiquement définie. Ainsi, la démarche d'inférence (au moins en principe) ne varie pas d'un problème à l'autre, contrairement à la statistique fréquentiste. Toute la question est donc comment échantillonner à partir de, ou calculer une espérance par rapport à, la mesure (1).

1.2. **Les méthodes d'inférence.** Pour ce faire, deux familles de méthodes existent : les méthodes exactes et les méthodes approchées. Les algorithmes de la deuxième famille sont plus rapides, mais les garanties théoriques sont moins solides. Nous renvoyons à Chopin and Ridgway (2017) pour un survol de la discipline du calcul bayésien. Nous nous contentons ici de présenter deux méthodes exactes qui sont à la base des échantillonneurs séquentiels : l'échantillonnage pondéré et la méthode de Monte-Carlo par chaîne de Markov (MCMC). Toutes les deux permettent de calculer une espérance par rapport à une densité de probabilité calculable uniquement à une constante de normalisation près. Ainsi, elles sont capables d'attaquer le problème posé par (1).

Etant donnée une distribution cible $\mathbb{Q}$, l'échantillonnage pondéré simule un échantillon $X^1, \ldots, X^N$ suivant une autre distribution $\mathbb{M}$ plus facile d'accès et approche l'espérance d'une fonction φ sous $\mathbb{Q}$ (notée par $\mathbb{Q}(\varphi) := \mathbb{E}_{\mathbb{Q}}[\varphi(X)]$) via la quantité

$$\frac{\omega(X^1)\varphi(X^1) + \cdots + \omega(X^N)\varphi(X^N)}{\omega(X^1) + \ldots + \omega(X^N)}$$

où $\omega(x) \propto \frac{\mathrm{d}\mathbb{Q}}{\mathrm{d}M}$ est, à une constante de normalisation près, la dérivée Radon-Nikodym de $\mathbb{Q}$ par rapport à $\mathbb{M}$. On peut alors dire que l'échantillon $X^1, \ldots, X^N$ *avec des poids* $\omega(X^1), \ldots, \omega(X^N)$ permet d'approcher la loi $\mathbb{Q}$. (Si les poids ne somment pas à 1, on fait référence implicitement aux poids normalisés obtenus en divisant chaque terme avec la bonne constante de normalisation.) L'échantillonnage pondéré est un algorithme standard dans toutes les références sur la méthode de Monte-Carlo (voir par ex. Robert and Casella, 2004, Chap. 3.3 ou Chopin and Papaspiliopoulos, 2020, Chap. 8). Ces références mentionnent systématiquement l'effondrement de sa performance quand la dimension augmente (fléau de la dimension). En revanche, son avantage principal est qu'il fournit un estimateur très simple et sans biais de la constante de normalisation $\int \omega(x)\mathbb{M}(\mathrm{d}x)$.

La méthode MCMC permet de briser le fléau de la dimension en explorant la loi $\mathbb{Q}$ de manière plus intelligente que par une loi de proposition globale $\mathbb{M}$. Elle consiste à simuler une chaîne de Markov ayant $\mathbb{Q}$ comme la mesure invariante et se justifie par des théorèmes ergodiques. Là encore, elle est incluse dans toutes les références sur la méthode de Monte Carlo. L'article Roberts and Rosenthal (2004) survole les techniques pour étudier la convergence. Nous présentons dans l'Algorithm 1 la méthode la plus basique (Metropolis-Hastings via marche aléatoire, ou RWMH).

Algorithme 1 : Metropolis-Hastings via marche aléatoire (RWMH)

Entrées : Densité cible $q : \mathbb{R}^d \to \mathbb{R}_+$; fonction $\varphi : \mathbb{R}^d \to \mathbb{R}$; point de départ $x \in \mathbb{R}^d$; matrice de covariance Σ de taille $d \times d$ de la distribution de proposition ; nombre d'itérations N

Assigner $X_1 \leftarrow x$

pour $n \leftarrow 2$ **à** N **faire**

  Simuler $X_n^* \sim \mathcal{N}(X_{n-1}, \Sigma)$

  Simuler $U_n \sim \text{Uniform}[0, 1]$

  si $U_n \leq \min\left(1, q(X_n^*)/q(X_{n-1})\right)$ **alors**

    Assigner $X_n \leftarrow X_n^*$

  sinon

    Assigner $X_n \leftarrow X_{n-1}$

Output : Estimateur $N^{-1} \sum_{n=1}^{N} \varphi(X_n)$ de $\mathbb{E}_{\mathbb{Q}}[\varphi(X)]$

Aujourd'hui, on utilise des algorithmes plus avancés que l'Algorithme 1, tels que ceux basés sur la diffusion de Langevin (Roberts and Tweedie, 1996), la dynamique hamiltonienne (Duane et al., 1987; Neal, 2011; Hoffman and Gelman, 2014) ou encore les processus en temps continu (Fearnhead et al., 2018). L'Algorithme 1 semble donc bien daté, mais il souligne déjà des problèmes importants avec MCMC qui ne sont pas encore complètement résolus. En effet, ses trois paramètres x, Σ et N correspondent aux trois problèmes difficiles pour les utilisateurs de MCMC : le rodag (*burn-in*), l'échelle optimale des paramètres (*scaling*) et le diagnostic de la convergence. Nous renvoyons aux références plus spécialisées (par ex. Brooks et al., 2011) mais nous souhaitons signaler d'ores et déjà que les échantillonneurs séquentiels permettront d'atténuer certaines de ces difficultés. Pour terminer, nous attirons l'attention sur une nouvelle direction dans le domaine de MCMC. Elle consiste à chercher des algorithmes qui présentent des compromis intéressants entre performance et robustesse, voir par ex. Livingstone and Zanella (2022); Riou-Durand and Vogrinc (2022).

2. Les echantillonneurs séquentiels par Monte-Carlo

2.1. **Les suites de distributions.** Au moins deux situations en inférence bayésienne conduisent à produire des échantillons venant d'une suite de distributions :

— quand les données $y_1, \ldots, y_D$ arrivent en temps réel (par ex. une donnée par seconde) et il est nécessaire par conséquent de mettre à jour l'estimation du paramètre θ continuellement ; et

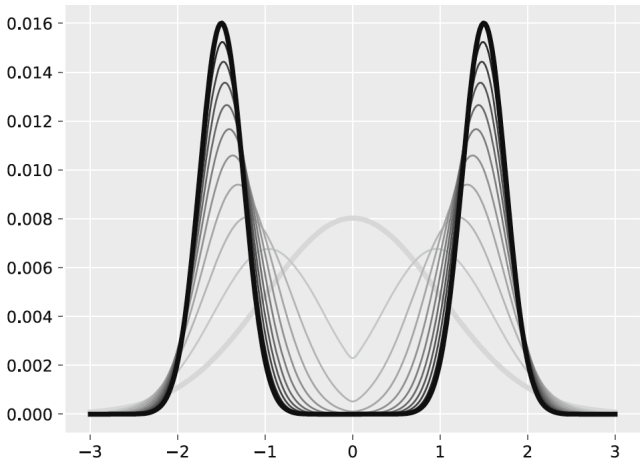

FIGURE 1. Une suite de distribution menant d'une simple gaussienne à une loi multimodale. Figure extraite de Chopin and Papaspiliopoulos (2020, Chapitre 3.3)

— quand on s'intéresse uniquement à une certaine loi a posteriori de θ sachant les données y, mais le fait de passer par de distributions intermédiaires se justifie par une meilleure performance.

Nous développons maintenant le deuxième cas. Etant donnée la distribution a posteriori souhaitée

$$\pi(\theta|y) \propto \pi_0(\theta)L(y|\theta)$$

où π_0 est la distribution a priori et L est la fonction de vraisemblance, une suite fréquemment utilisée dans la littérature (par ex. Neal, 2001) est

$$\pi_{\lambda_t}(\theta|y) \propto \pi_0(\theta)L(y|\theta)^{\lambda_t} \tag{2}$$

où $0 < \lambda_0 < \ldots < \lambda_T = 1$ est une suite de nombres réels. Plusieurs motivations sont à l'origine de l'introduction de telles distributions intermédiaires. D'abord, elles pourraient mieux se comporter que la loi cible (Figure 1). Ensuite, elles permettent aux utilisateurs d'adapter les échantillonneurs petit à petit. Une multitude (si ce n'est pas la totalité) d'algorithmes dépendent de paramètres d'ajustement qui, lorsqu'ils sont mal choisis, peuvent entraîner une réduction drastique de la performance. On pense notamment aux algorithmes de MCMC pour lesquels une littérature entière se consacre à la recherche d'une « échelle optimale » des paramètres (par ex. Roberts and Rosenthal, 2001; Beskos et al., 2013 ; voir aussi la discussion en fin de la Section 1.2).

De plus, en observant de petits changements d'une distribution à l'autre, on peut produire des estimations très précises de la constante de normalisation. On reviendra à ce point dans la suite.

2.2. **Le principe de base.** Les échantillonneurs dits de Monte-Carlo séquentiel (les échantillonneurs SMC) permettent d'attaquer la simulation à partir d'une suite de distributions de manière particulièrement efficace. Suite à leur formalisation (Del Moral et al., 2006), ils ont été appliqués avec succès dans plusieurs domaines

et sur des problèmes réputés difficiles (Ridgway, 2016; Beskos et al., 2017; Buchholz et al., 2020).

Pour apprécier les échantillonneurs SMC, il faut d'abord bien comprendre ses deux ingrédients : l'échantillonnage pondéré et la méthode MCMC. Nous renvoyons à la Section 1.2 et aux références qui y figurent. Nous allons maintenant décrire précisément les étapes d'un échantillonneur SMC. A partir d'un échantillon $X_{t-1}^1, \ldots, X_{t-1}^N$ avec des poids normalisés $W_{t-1}^1, \ldots, W_{t-1}^N$ qui approche $\pi_{\lambda_{t-1}}$, on procède à un ré-échantillonnage qui aboutit à un nouvel échantillon $X_{t-1}^{A_t^1}, \ldots, X_{t-1}^{A_t^N}$ sans poids, qui suit toujours à peu près $\pi_{\lambda_{t-1}}$. Plus précisément, les indices A_t^n sont tirées conditionnellement i.i.d. à partir de la loi discrète à support $\{1, 2, \ldots, N\}$ et à poids $W_{t-1}^1, \ldots, W_{t-1}^N$. Cette loi sera notée $\mathcal{M}(W_{t-1}^{1:N})$. Le ré-échantillonnage a pour avantage d'initialiser les poids, mais son inconvénient est de créer des groupes de particules identiques. Par conséquent, les particules ré-échantillonnées $X_{t-1}^{A_t^n}$ sont ensuite injectées dans un noyau M_t laissant invariante la loi $\pi_{\lambda_{t-1}}$, afin de produire un échantillon de meilleure qualité. Ce noyau peut consister en plusieurs itérations d'un noyau MCMC K_t, au bout desquelles sortent les nouvelles particules qu'on nommera $X_t^1, \ldots, X_t^N$. C'est à ce moment que l'échantillonnage pondéré intervient pour redonner des poids aux particules X_t^n, en vue d'en faire une approximation de π_{λ_t}. La formulation complète est donnée dans l'Algorithme 2.

Algorithme 2 : Echantillonneur SMC

Entrées : Distribution a priori π_0 ; Fonction de vraisemblance L ; Exposants $0 < \lambda_0 < \ldots < \lambda_T = 1$; Noyaux MCMC $K_1, \ldots K_T$ laissant invariant respectivement $\pi_{\lambda_0}, \ldots, \pi_{\lambda_{T-1}}$; Nombre de particules N ; Nombre d'itérations MCMC k

pour $n \leftarrow 1$ **à** N **faire**
- Simuler $X_0^n \sim \pi_0$
- Calculer $\omega_0^n \leftarrow L(X_0^n)^{\lambda_0}$

Assigner $\ell_0^N \leftarrow 1/N \sum_n \omega_0^n$

Assigner $W_0^n \leftarrow \omega_0^n / N\ell_0^N$ pour $n = 1, \ldots, N$

pour $t \leftarrow 1$ **à** T **faire**
- **pour** $n \leftarrow 1$ **à** N **faire**
  - Simuler $A_t^n \sim \mathcal{M}(W_{t-1}^{1:N})$
  - Simuler $X_t^n \sim K_t^k(X_{t-1}^{A_t^n}, \cdot)$
  - Calculer $\omega_t^n \leftarrow L(X_t^n)^{\lambda_t - \lambda_{t-1}}$
- Assigner $\ell_t^N \leftarrow 1/N \sum_n \omega_t^n$
- Assigner $W_t^n \leftarrow \omega_t^n / N\ell_t^N$ pour $n = 1, \ldots, N$

Output : Estimateur $\sum W_T^n \varphi(X_T^n)$ pour $\pi(\varphi) := \mathbb{E}_\pi[\varphi(X)]$; estimateur $\prod_{t=1}^T \ell_t^N$ pour $\int L(x)\pi_0(\mathrm{d}x)$

2.3. **L'aspect théorique.** La convergence de l'Algorithme 2 quand $N \to \infty$ est garantie par une construction théorique qui s'appelle le modèle de Feynman-Kac dont il est un cas particulier. Il s'agit d'un cadre général qui englobe les échantillonneurs SMC mais aussi une grande variété des algorithmes dits génétiques comprenant des

opérations de mutation ou sélection. Historiquement, ces algorithmes étaient motivés par des applications en physique ou en chimie (Rosenbluth and Rosenbluth, 1955; Hetherington, 1984) et faisaient leur apparition plus tard dans la communauté statistique (Gordon et al., 1993; Kitagawa, 1996). De premiers résultats rigoureux (estimation sans biais de la constante de normalisation, convergence presque sûre, etc.) ont été publiés dans Del Moral (1997); Crisan et al. (1998), suivis par des théorèmes de la limite centrale (Del Moral and Guionnet, 1999; Del Moral and Miclo, 2000; Del Moral and Ledoux, 2000; Chopin, 2004).

Le lecteur intéressé trouvera une présentation complète du formalisme de Feynman-Kac dans le livre Del Moral (2004). On se content ici de résumer les points importants : l'Algorithme 2 converge presque surement, admet des théorèmes de la limite centrale et fournit un estimateur sans biais de la constante de normalisation $\int L(x)\pi_0(\mathrm{d}x)$.

L'analyse théorique des échantillonneurs SMC doit prendre en compte, en complément de la structure de Feynman-Kac, les propriétés des noyaux MCMC utilisés. L'interaction entre l'échantillonnage pondéré et le noyau MCMC n'est pas encore totalement comprise, mais des travaux ont été effectués pour étudier son implication sur le comportement de l'algorithme en grande dimension (Beskos et al., 2014) et en régime non-asymptotique (Giraud and Del Moral, 2017).

2.4. **L'aspect pratique.** Deux précautions très importantes sont à prendre pour que les échantillonneurs SMC aient de bonne performance. D'abord, il est essentiel d'adapter les paramètres des noyaux MCMC à la distribution invariante. Cette tâche, normalement ardue faute d'information suffisante sur la distribution cible, est grandement facilitée dans le cadre des échantillonneurs SMC grâce à la disponibilité d'un échantillon pondéré suivant à peu près la loi en question. Pour dire la même chose avec des notations, avant de déclencher le noyau K_t ayant pour distribution invariante $\pi_{\lambda_{t-1}}$, on a déjà accès à un échantillon $X_{t-1}^1, \ldots, X_{t-1}^N$ pondéré de $W_{t-1}^1, \ldots, W_{t-1}^N$ qui approche cette loi.

Le second point crucial est le choix de la suite $0 < \lambda_0 < \ldots < \lambda_T = 1$. Le bon sens dit qu'il ne faut pas une croissance trop rapide. Sa traduction mathématique est une distance contrôlée entre deux distributions successives $\pi_{\lambda_{t-1}}$ et π_{λ_t}. Si l'on choisit la distance de chi-deux, une estimation empirique est donnée par

$$N \sum_{n=1}^{N} [(W_t)^n]^2 - 1.$$

Cet estimateur est très proche du concept de la « taille effective » de l'échantillon (ESS, Kong et al., 1994). La recommandation est donc de choisir les λ de manière adaptative telle que cet estimateur prend toujours une valeur fixée, par ex. 1. Dans le cadre des échantillonneurs SMC, cette manière de faire est devenue populaire après l'article Jasra et al. (2011).

Pour ceux qui souhaitent mettre la main à la pâte, nous conseillons le logiciel `particles` accompagnant le livre de Chopin and Papaspiliopoulos (2020) et téléchargeable sur `https://github.com/nchopin/particles` sous la forme d'une librairie pour Python.

Pour terminer, nous notons que malgré sa technicité, les échantillonneurs SMC restent une alternative intéressante à la méthode MCMC pour des raisons suivantes (Chopin and Papaspiliopoulos, 2020, Chap. 17 ; Dai et al., 2022) :

— ils peuvent être parallélisés ;

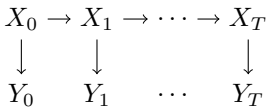

FIGURE 2. Représentation graphique d'un modèle à espace d'états

— ils permettent d'estimer la constante de normalisation (qui joue le rôle de la vraisemblance du modèle) ; et
— ils facilitent l'adaptation des paramètres d'ajustement des noyaux MCMC.

3. Les algorithmes de lissage

3.1. **Les modèles à espace d'états.** Un modèle à espace d'états est composé d'un processus de Markov $(X_t)_{t=0}^T$ non observé et de données (Y_t) qui sont *grosso modo* des observations bruitées des X_t. Plus précisément, la loi de Y_t sachant $X_{0:T}$ ne dépend que de X_t et est supposée connue. Figure 2 décrit la relation probabiliste entre les variables dans le langage des modèles graphiques (Bishop, 2006, Chap. 8). Cette définition étant énoncée, le problème dit de « lissage » correspond à la question la plus naturelle : retrouver les états cachés à partir des observations.

Il est clair que ce genre de modèle a une portée applicative très large (en biologie, économie, finance, ingénierie, etc.) : en effet, on s'y ramène dès qu'un phénomène quelconque peut être modélisé de manière markovienne (possiblement non homogène) et qu'on est capable de l'observer, ne serait-ce que de manière partielle et bruitée ! De plus, même si un phénomène $(X_t)_{t=0}^T$ ne se passe pas forcément de manière markovienne, on peut se permettre de mettre une loi à priori markovienne au profit de l'inférence bayésienne $p(x_{0:T}|y_{0:T})$. Nous renvoyons au Chap. 2 de Chopin and Papaspiliopoulos (2020) pour une revue détaillée des applications.

3.2. **Le lissage par remontée de généalogie.** Il est aisé de vérifier la récurrence suivante

$$p(x_{0:t}|y_{0:t}) \propto p(x_{0:t-1}|y_{0:t-1})\{p(x_t|x_{t-1})\}\,[p(y_t|x_t)] \tag{3}$$

où le terme dans l'accolade correspond à la dynamique markovienne et celui dans le crochet à la pondération. Le filtre de Bootstrap, présenté dans l'Algorithme 3, devrait ainsi paraître très naturelle une fois que l'échantillonnage pondéré (expliqué brièvement dans la Section 2.2) a été compris. (Voir la Sous-section 2.3 pour le contexte historique.) On notera $\mathcal{X}_t$ l'espace dans lequel vit l'état x_t.

A chaque itération, l'Algorithme 3 « tue » certaines trajectoires et prolonge celles encore vivantes, mais à aucun moment il ne modifie les états antérieurs d'une trajectoire. Ainsi, à mesure que $T \to \infty$ et N reste fixe, toutes les trajectoires finissent par avoir le même état au point 0. Ce phénomène s'appelle *l'appauvrissement* et on dit que les trajectoires sont devenues *dégénérées*.

D'un point de vue légèrement différent, on peut ajouter une dernière étape de ré-échantillonnage à l'Algorithme 3 au temps $t = T$ pour produire N trajectoires $\hat{Z}_T^1, \ldots, \hat{Z}_T^N$ sans poids qui approchent $p(x_{0:T}|y_{0:T})$. Dans ce cas, on peut dire que l'Algorithme 3 produit des suites d'*indices*, chacune de la forme $(b_0, \ldots, b_T)$ qui est

Algorithme 3 : Une itération du filtre de Bootstrap pour approcher (3)

Entrées : N trajectoires $Z_{t-1}^1, \dots, Z_{t-1}^N$ pondérées de $W_{t-1}^1, \dots, W_{t-1}^N$ qui approchent $p(x_{0:t-1}|y_{0:t-1})$. Chacune a pour longueur t et vit dans $\mathcal{X}_0 \times \dots \mathcal{X}_{t-1}$.

pour $n \leftarrow 1$ **à** N **faire**

- Simuler $A_t^n \sim \mathcal{M}(W_{t-1}^{1:N})$
- Simuler X_t^n à partir de la dernière composante de $Z_{t-1}^{A_t^n}$ en suivant la dynamique markovienne du temps $t-1$ au temps t
- Prolonger Z_{t-1}^n en Z_t^n par l'ajout de X_t^n à sa fin
- Assigner $\omega_t^n \leftarrow p(y_t|X_t^n)$

Assigner $\ell_t^N \leftarrow 1/N \sum_n \omega_t^n$

Assigner $W_t^n \leftarrow \omega_t^n / N\ell_t^N$, pour $n = 1, \dots, N$

Output : N trajectoires $Z_t^1, \dots, Z_t^N$ pondérées de $W_t^1, \dots, W_t^N$ qui approchent $p(x_{0:t}|y_{0:t})$

une représentation compacte de la trajectoire $z_t = (X_0^{b_0}, \dots, X_T^{b_t})$. Si on définit les indices B_t^n par

$$\hat{Z}_T^n = (X_0^{B_0^n}, \dots, X_T^{B_T^n})$$

alors il est facile de démontrer que

$$B_{t-1}^n = A_t^{B_t^n} \tag{4}$$

ce qui justifie le terme « remontée de généalogie ».

3.3. **Le lissage statique.** Pour contrer l'appauvrissement, il est nécessaire de renouveler d'une manière ou d'une autre les états antérieurs d'une trajectoire. Pour ce faire, il est naturel de considérer

$$p(x_{t-1}|x_t, y_{0:T}) \propto \{p(x_{t-1}|y_{0:t-1})\} \left[p(x_t|x_{t-1})\right].$$

Le terme en accolade peut être approché par l'échantillon $X_{t-1}^1, \dots, X_{t-1}^N$ pondéré par $W_{t-1}^1, \dots, W_{t-1}^N$ et le terme en crochet joue désormais le rôle d'une pondération. Ainsi, au lieu d'obtenir B_{t-1}^n par (4), on peut simuler B_{t-1}^n à partir d'une loi à support discret $\{1, 2, \dots N\}$ dont la probabilité de l'élément i est

$$\frac{W_{t-1}^i p(X_t^{B_t^n}|X_{t-1}^i)}{\sum_{j=1}^N W_{t-1}^j p(X_t^{B_t^n}|X_{t-1}^j)}. \tag{5}$$

La procédure qui consiste à tourner d'abord l'Algorithme 3 puis simuler les B_t^n par (5) a été proposée par Godsill et al. (2004) sous le nom FFBS (filtrage en aval – lissage en amont). Sa complexité est $\mathcal{O}(N^2)$. Le point important est qu'elle forme des trajectoires en piochant à chaque temps t une particule parmi les $X_t^1, \dots, X_t^N$. L'ensemble des $N \times (T+1)$ particules $(X_t^n)_{t=0,\dots,T}^{n=1,\dots,N}$ qu'on notera par commodité $X_{0:T}^{1:N}$ constitue ainsi la *squelette* de lissage.

Ensuite, Del Moral et al. (2010); Douc et al. (2011) étudient en grand détail l'algorithme FFBS et démontrent ses convergence et stabilité – terme faisant référence à sa capacité de produire des trajectoires non-dégénérées. De plus, sous l'hypothèse $p(x_t|x_{t-1}) \leq C$ pour une certaine constante C, Douc et al. (2011) proposent de

simuler (5) par l'algorithme de rejet (Robert and Casella, 2004, Chap. 2.3) en utilisant pour distribution de proposition celle ayant les poids W_{t-1}^i. Le coût de cet algorithme (judicieusement appelé FFBS-Rejet) est alors aléatoire mais peut être contrôlé si la densité $p(x_t|x_{t-1})$ est aussi bornée inférieurement par une constante $c > 0$.

Enfin, remarquons que indépendamment de l'utilisation ou pas de l'échantillonnage par rejet, le coût de cet algorithme augmente linéairement avec T, d'où la dénomination « statique ». C'est-à-dire qu'il n'est pas convenable de l'utiliser quand les données sont obtenues en temps réel.

3.4. **Le lissage en ligne.** Le lissage en ligne ne s'intéresse pas à la construction des trajectoires *stricto sensu* mais à l'estimation d'une espérance par rapport à la loi de lissage. Pour qu'elle soit faisable sans que l'on soit obligé de tout garder en mémoire, la fonction en question doit être *additive*. On s'intéressera donc à la quantité

$$\mathbb{E}[\psi_0(X_0) + \psi_1(X_0, X_1) + \ldots + \psi_t(X_{t-1}, X_t)|Y_0, Y_1, \ldots, Y_t] \tag{6}$$

dont on limite la discussion au cas où seule la fonction ψ_0 est non-triviale et les autres ψ_t sont égales à la fonction nulle. Compte tenu de notre discussion ci-dessus, l'espérance $\mathbb{E}[\psi_0(X_0)|Y_0, \ldots, Y_T]$ admet comme estimateur

$$\mu_T^N := \mathbb{E}[\psi_0(X_0^{B_0})|X_{0:T}^{1:N}]$$

où, sachant $X_{0:T}^{1:N}$, l'indice B_T suit la loi $\mathcal{M}(W_T^{1:N})$ et les autres sont définis récursivement par

$$B_{t-1}|B_{t:T}, X_{0:T}^{1:N} \sim \mathcal{M}\left(\frac{W_{t-1}^i p(X_t^{B_t}|X_{t-1}^i)}{\sum_{j=1}^N W_{t-1}^j p(X_t^{B_t}|X_{t-1}^j)}\right)$$

à l'instar de (5). On voit que conditionnellement à $X_{0:T}^{1:N}$, la suite $B_T, \ldots, B_0$ est donc une chaîne de Markov non-homogène dont les matrices de transition $\hat{B}_T^N, \ldots, \hat{B}_1^N$ sont explicitement calculables. Ainsi, la quantité μ_T^N s'écrit

$$\mu_T^N = \begin{bmatrix} W_T^1 & \ldots & W_T^N \end{bmatrix} \hat{B}_T^N \ldots \hat{B}_1^N \begin{bmatrix} \psi_0(X_0^1) \\ \ldots \\ \psi_0(X_0^N) \end{bmatrix}.$$

Il est désormais aisé de comprendre comment elle peut se calculer en ligne à travers les quantités S_t^N (communément appelées les statistiques représentatives ou *summary statistics*) définies par la récursion

$$\begin{cases} S_t^N & := \hat{B}_t^N S_{t-1}^N \\ S_0^N & := \begin{bmatrix} \psi_0(X_0^1) \\ \ldots \\ \psi_0(X_0^N) \end{bmatrix} \end{cases} \tag{7}$$

et la réécriture

$$\mu_T^N = \begin{bmatrix} W_T^1 & \ldots & W_T^N \end{bmatrix} S_T^N.$$

Cette façon d'approcher en ligne (6) a été présentée pour la première fois dans Del Moral et al. (2010). Elle a pour coût $\mathcal{O}(N^2)$, ce qui a motivé Olsson and Westerborn (2017) à proposer une version à complexité $\mathcal{O}(N)$ sous réserve que $p(x_t|x_{t-1}) \geq c > 0$. Cet algorithme (baptisé l'algorithme de PaRIS) se base sur l'échantillonnage par rejet, dans les mêmes lignes que Douc et al. (2011). Nous

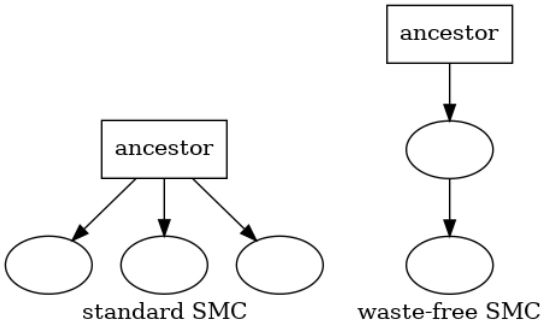

FIGURE 3. Représentation graphique de la réduction de corrélation grâce à l'algorithme WFSMC (voir texte). Figure extraite de Dau and Chopin (2022).

n'entrerons pas dans les détails, mais nous contentons d'évoquer l'idée principale : le calcul de (7) est coûteux car la matrice $\hat{B}_t^N$ est pleine. Une estimation sans biais et creuse $\hat{B}_t^{N,\mathrm{PaRIS}}$ est alors proposée avec des simulations de Monte-Carlo supplémentaire, qui font appel à la méthode de rejet.

4. Les contributions de la thèse

4.1. **Pour un échantillonneur séquentiel sans gaspillage.** La performance des échantillonneurs SMC dépend grandement de k, le nombre d'itérations de noyaux MCMC. Des expériences numériques (par ex. Chopin and Papaspiliopoulos, 2020, Chap. 17) font l'état de résultats calamiteux quand k est choisi trop faible. On observe que les valeurs optimales de k sont souvent très grandes, de l'ordre d'une centaine (Buchholz et al., 2020) ; alors qu'une grande variété de recettes empiriques existent pour choisir k sans qu'une procédure véritablement fiable se dégage. Au bout des itérations, les valeurs intermédiaires des chaînes de Markov ne sont pas prises en compte. Or les itérations MCMC sont très coûteuses en temps de calcul.

Nous proposons l'échantillonneur SMC sans gaspillage (WFSMC, Dau and Chopin, 2022), un algorithme libéré de ces contraintes. Au lieu de ré-échantillonner N particules à chaque étape, on n'en conserve que M. Le reste est régénéré par N/M itérations de M chaînes MCMC en parallèle.

Une motivation alternative pour WFSMC réside dans l'étape de ré-échantillonnage, à l'issue de laquelle une particule peut être sélectionnées plusieurs fois et donne lieu donc à plusieurs descendants. Si le nombre de pas de MCMC est faible, ces descendants seront fortement corrélés. En revanche, s'ils sont générés par des itérations successives du noyau, on peut espérer que l'échantillon ainsi obtenu sera moins dégénéré (Figure 3).

Au cours de la révision du papier, nous avons constaté que Tan (2015) a déjà introduit un algorithme équivalent au nôtre. Toutefois, nous sommes les premiers à effectuer une analyse théorique rigoureuse de ses propriétés. En construisant une mesure appropriée sur un espace étendu, nous avons proprement exprimé l'algorithme dans le langage des modèles de Feynman-Kac. Les résultats théoriques tels que le caractère sans biais de l'estimateur de la constante de normalisation en découlent alors naturellement. Cependant, nos expériences numériques suggèrent

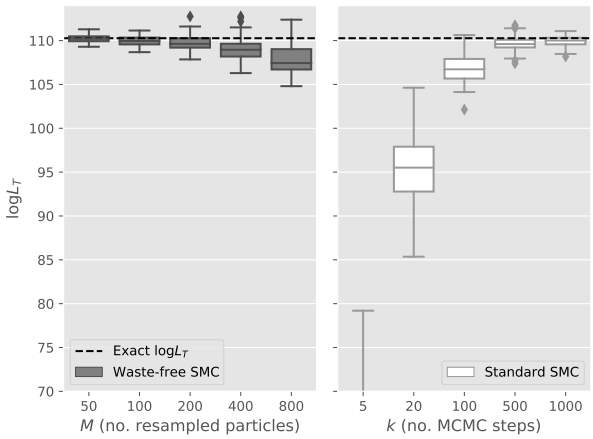

FIGURE 4. Comparaison de la performance de l'échantillonneur SMC standard et notre algorithme WFSMC dans le comptage des carrés latins (Dau and Chopin, 2022).

que le régime optimal est atteint quand M est petit et fixe alors que $N \to \infty$. Notre deuxième contribution est de démontrer la convergence et les théorèmes centraux limites dans ce régime, ce qui nécessite de faire intervenir l'ergodicité et pas seulement l'invariance des noyaux MCMC. Les particules intermédiaires jouent ainsi un rôle central dans notre algorithme, alors qu'elles ne sont pas incorporées dans l'échantillon final dans l'algorithme classique.

Plus les chaînes MCMC sont longues, plus les variances des estimateurs qu'elles donnent peuvent être estimées avec fiabilité. Basée sur les travaux précédents évaluant l'erreur des moyennes ergodiques issues des algorithmes MCMC (Geyer, 1992; Flegal and Jones, 2010), notre troisième contribution est de proposer des estimateurs de la variance pour l'algorithme WFSMC. Nous observons qu'ils sont plus précis que l'estimateur générique (Lee and Whiteley, 2018) qui ne prend en compte que la structure de Feynman-Kac du modèle et passe donc à côté des propriétés spécifiques aux noyaux MCMC. Nous utilisons également les estimateurs de la variance pour adapter le nombre de particules N notamment quand les noyaux deviennent de plus en plus mauvais à mesure que la température descend.

Notre quatrième contribution est de quantifier l'amélioration apportée par l'algorithme WFSMC en comparaison avec les échantillonneurs SMC standard. Nous avons démontré rigoureusement la supériorité de WFSMC dans un exemple très simple. Nous l'avons constatée également de manière empirique à travers plusieurs exemples numériques, mettant en jeu l'espace $\mathbb{R}^d$ classique mais aussi l'espace discret des carrés latins (Figure 4) ainsi qu'un modèle hors du cadre des mesures tempérées (celui du calcul de la probabilité qu'une loi gaussienne en grande dimension se trouve dans un pavé donné).

4.2. **Pour de nouveaux algorithmes de lissage plus puissants.** L'échantillonnage par rejet est censé réduire le temps d'exécution des algorithmes de lissage tout en le rendant aléatoire. Cet aléa est contrôlé si $p(x_t|x_{t-1}) \geq c > 0$ pour tout x_{t-1} et

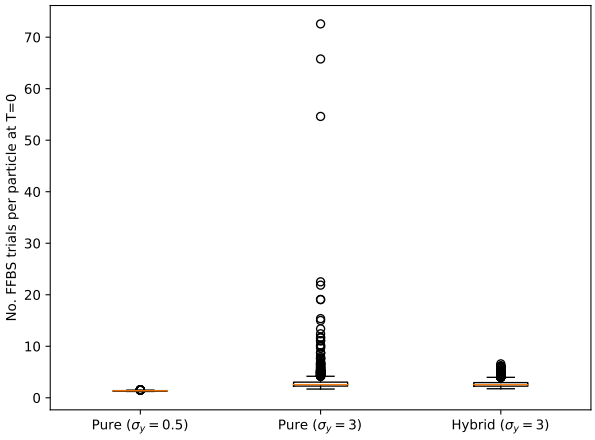

FIGURE 5. Box plots, pour les différents modes d'échantillonnage par rejet, des nombres d'évaluations de la densité de transition divisés par N dans un modèle linéaire gaussien. La performance de la méthode classique (rejet pur) dépend fortement d'un certain paramètre σ_y du modèle.

x_t, mais la plupart des modèles en pratique (y compris les modèles linéaires gaussiens) ne satisfont pas cette hypothèse. Notre première contribution est d'étudier précisément ce qui se passe alors. Nous montrons que le temps d'exécution peut devenir si aléatoire qu'il finit par ne pas avoir d'espérance, ou pas de moment d'ordre k pour une valeur assez faible de k. Les expériences numériques montrent des temps de calcul très variables (même après avoir étés divisés par $N \times T$) qui dépendent de manière incontrôlable des paramètres du modèle et même le type de filtrage utilisé (le filtre Bootstrap ou le filtre guidé).

Nous proposons alors deux solutions à ce problème. La première consiste à alterner habilement entre l'échantillonnage par rejet et l'algorithme classique ; la deuxième remplace la simulation exacte de (5), qui est chère en terme de temps de calcul, par un pas de MCMC gardant cette distribution invariante. Ces deux algorithmes ont été présentés dans la littérature dans un cadre restreint et sans résultat théorique (Taghavi et al., 2013; Bunch and Godsill, 2013). Nous les avons analysés rigoureusement et nous les avons formulées pour à la fois le scénario statique et le scénario en ligne. Notre deuxième contribution est de démontrer que l'algorithme hybride qui abandonne la méthode de rejet au bon moment a une performance intermédiaire entre $\mathcal{O}(N)$ et $\mathcal{O}(N^2)$. De plus, dans des modèles linéaires gaussiens, la complexité est ramenée à $\mathcal{O}(N)$ à un facteur de log près. Nos expériences numériques indiquent que l'algorithme hybride affiche un coût de calcul presque déterministe, en comparaison avec l'imprévisibilité frappante de l'algorithme de rejet classique (Figure 5).

Notre troisième contribution est de démontrer rigoureusement la convergence et la stabilité quand le lissage s'effectue avec MCMC. Si la chaîne de MCMC gardant invariante (5) démarre au bon endroit (i.e. l'ancêtre généré pendant l'étape de filtrage), nous montrons qu'un seul pas de MCMC suffit pour avoir un algorithme de lissage convergent et stable, que ce soit statique ou en ligne.

Notre quatrième contribution concerne des modèles dont la densité de transition n'est pas calculable, mais la dynamique markovienne peut toujours être simulée. On pense notamment aux diffusions discrétisées ou aux processus markoviens de saut. Toutes les méthodes de lissage mentionnées jusqu'alors devenant caduques, nous proposons une procédure originale basée sur le couplage. L'échec de la méthode de remontée généalogique (Section 3.2) tient au fait que chaque particule a une seul ancêtre. Ainsi, comme le nombre de particules N est fini, au bout d'un certain nombre de remontées, les deux lignées ancestrales vont forcément se rencontrer, moment à partir duquel elles se coïncident jusqu'au temps 0. En revanche, si l'on peut, d'une manière ou d'une autre, considérer une particule comme ayant deux ancêtres (à l'instar des êtres humains !), ledit appauvrissement n'aura pas lieu. Bien évidemment, dans la mesure de probabilité sur les indices engendrée par l'algorithme FFBS, une particule peut avoir plusieurs ancêtres, mais le défi est de les générer en temps de calcul réduit. Le problème devient encore plus dur dans le scénario où la densité de transition est inaccessible et par conséquent, le noyau FFBS l'est également. Dans le cadre des modèles ordinaires, l'idée de trouver deux parents a été réalisée par l'algorithme PaRIS (Olsson and Westerborn, 2017), notamment quand $\tilde{N} = 2$. Le point original, c'est que nous l'avons transportée aux modèles ayant une densité de transition incalculable en utilisant le couplage pendant l'étape de filtrage. Nous avons démontré la convergence et la stabilité grâce à un cadre général qui admet comme cas particuliers tous les algorithmes de lissage basé sur le squelette de filtrage, y compris celui-ci. Sur le plan numérique, nous avons réussi à mettre en œuvre le couplage des deux diffusions – tâche non triviale car la construction théorique (Lindvall and Rogers, 1986) est rendue délicate par la discrétisation.

Références

Beskos, A., Crisan, D., and Jasra, A. (2014). On the stability of sequential Monte Carlo methods in high dimensions. *Ann. Appl. Probab.*, 24(4) :1396–1445.

Beskos, A., Jasra, A., Law, K., Tempone, R., and Zhou, Y. (2017). Multilevel sequential Monte Carlo samplers. *Stochastic Process. Appl.*, 127(5) :1417–1440.

Beskos, A., Pillai, N., Roberts, G., Sanz-Serna, J.-M., and Stuart, A. (2013). Optimal tuning of the hybrid Monte Carlo algorithm. *Bernoulli*, 19(5A) :1501–1534.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Information Science and Statistics. Springer, New York.

Brooks, S., Gelman, A., Jones, G., and Meng, X.-L. (2011). *Handbook of Markov Chain Monte Carlo*. CRC press.

Buchholz, A., Chopin, N., and Jacob, P. E. (2020). Adaptive tuning of Hamiltonian Monte Carlo within sequential Monte Carlo. *Bayesian Anal.* Advance publication.

Bunch, P. and Godsill, S. (2013). Improved particle approximations to the joint smoothing distribution using Markov Chain Monte Carlo. *IEEE Transactions on Signal Processing*, 61(4) :956–963.

Chopin, N. (2004). Central limit theorem for sequential Monte Carlo methods and its application to Bayesian inference. *Ann. Statist.*, 32(6) :2385–2411.

Chopin, N. and Papaspiliopoulos, O. (2020). *An Introduction to Sequential Monte Carlo*. Springer Series in Statistics. Springer.

Chopin, N. and Ridgway, J. (2017). Leave Pima Indians alone : binary regression as a benchmark for Bayesian computation. *Statistical Science*, 32(1) :64–87.

Crisan, D., Del Moral, P., and Lyons, T. (1998). *Discrete filtering using branching and interacting particle systems*. Citeseer.

Dai, C., Heng, J., Jacob, P. E., and Whiteley, N. (2022). An invitation to sequential Monte Carlo samplers. *Journal of the American Statistical Association*, 0(ja) :1–38.

Dau, H.-D. and Chopin, N. (2022). Waste-free sequential Monte Carlo. *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, 84(1) :114–148.

Del Moral, P. (1997). Nonlinear filtering : Interacting particle resolution. *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*, 325(6) :653–658.

Del Moral, P. (2004). *Feynman-Kac formulae. Genealogical and interacting particle systems with applications*. Probability and its Applications. Springer Verlag, New York.

Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential Monte Carlo samplers. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 68(3) :411–436.

Del Moral, P., Doucet, A., and Singh, S. S. (2010). A backward particle interpretation of Feynman-Kac formulae. *M2AN Math. Model. Numer. Anal.*, 44(5) :947–975.

Del Moral, P. and Guionnet, A. (1999). Central limit theorem for nonlinear filtering and interacting particle systems. *Ann. Appl. Probab.*, 9(2) :275–297.

Del Moral, P. and Ledoux, M. (2000). Convergence of empirical processes for interacting particle systems with applications to nonlinear filtering. *Journal of Theoretical Probability*, 13(1) :225–257.

Del Moral, P. and Miclo, L. (2000). Branching and interacting particle systems approximations of Feynman-Kac formulae with applications to non-linear filtering. In Azéma, J., Emery, M., Ledoux, M., and Yor, M., editors, *Séminaire de Probabilités, XXXIV*, volume 1729 of *Lecture Notes in Math.*, pages 1–145. Springer, Berlin.

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.

Duane, S., Kennedy, A., Pendleton, B. J., and Roweth, D. (1987). Hybrid monte carlo. *Physics Letters B*, 195(2) :216–222.

Fearnhead, P., Bierkens, J., Pollock, M., and Roberts, G. O. (2018). Piecewise Deterministic Markov Processes for Continuous-Time Monte Carlo. *Statistical Science*, 33(3) :386 – 412.

Flegal, J. M. and Jones, G. L. (2010). Batch means and spectral variance estimators in Markov chain Monte Carlo. *Ann. Statist.*, 38(2) :1034–1070.

Geyer, C. J. (1992). Practical Markov Chain Monte Carlo. *Statistical science*, 7(4) :473–483.

Giraud, F. and Del Moral, P. (2017). Nonasymptotic analysis of adaptive and annealed Feynman-Kac particle models. *Bernoulli*, 23(1) :670–709.

Godsill, S. J., Doucet, A., and West, M. (2004). Monte Carlo smoothing for nonlinear times series. *J. Amer. Statist. Assoc.*, 99(465) :156–168.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc. F, Comm., Radar, Signal Proc.*, 140(2) :107–113.

Hetherington, J. (1984). Observations on the statistical iteration of matrices. *Physical Review A*, 30(5) :2713.

Hoffman, M. D. and Gelman, A. (2014). The no-U-turn sampler : adaptively setting path lengths in Hamiltonian Monte Carlo. *J. Mach. Learn. Res.*, 15 :1593–1623.

Jasra, A., Stephens, D., Doucet, and Tsagaris, T. (2011). Inference for Lévy driven stochastic volatility models via Sequential Monte Carlo. *Scand. J. of Statist.*, 38(1).

Kitagawa, G. (1996). Monte Carlo filter and smoother for non-gaussian state space models. *Journal of Computational and Graphical Statistics*, 5(1) :1–25.

Kong, A., Liu, J. S., and Wong, W. H. (1994). Sequential imputation and Bayesian missing data problems. *J. Am. Statist. Assoc.*, 89 :278–288.

Lee, A. and Whiteley, N. (2018). Variance estimation in the particle filter. *Biometrika*, 105(3) :609–625.

Lindvall, T. and Rogers, L. C. G. (1986). Coupling of multidimensional diffusions by reflection. *Ann. Probab.*, 14(3) :860–872.

Livingstone, S. and Zanella, G. (2022). The Barker proposal : Combining robustness and efficiency in gradient-based MCMC. *Journal of the Royal Statistical Society Series B*, 84(2) :496–523.

Neal, R. M. (2001). Annealed importance sampling. *Stat. Comput.*, 11(2) :125–139.

Neal, R. M. (2011). MCMC using Hamiltonian dynamics. In Brooks, S., Gelman, A., Jones, G. L., and Meng, X. L., editors, *Handbook of Markov chain Monte Carlo*, Chapman & Hall/CRC Handb. Mod. Stat. Methods, pages 113–162. CRC Press, Boca Raton, FL.

Olsson, J. and Westerborn, J. (2017). Efficient particle-based online smoothing in general hidden Markov models : the PaRIS algorithm. *Bernoulli*, 23(3) :1951–1996.

Ridgway, J. (2016). Computation of Gaussian orthant probabilities in high dimension. *Stat. Comput.*, 26(4) :899–916.

Riou-Durand, L. and Vogrinc, J. (2022). Metropolis Adjusted Langevin Trajectories : a robust alternative to Hamiltonian Monte Carlo. *arXiv preprint arXiv :2202.13230*.

Robert, C. P. (2007). *The Bayesian choice : from decision-theoretic foundations to computational implementation*. Springer Verlag.

Robert, C. P. and Casella, G. (2004). *Monte Carlo Statistical Methods, 2nd ed.* Springer-Verlag, New York.

Roberts, G. O. and Rosenthal, J. S. (2001). Optimal scaling for various Metropolis-Hastings algorithms. *Statist. Sci.*, 16(4) :351–367.

Roberts, G. O. and Rosenthal, J. S. (2004). General state space Markov chains and MCMC algorithms. *Probab. Surv.*, 1 :20–71.

Roberts, G. O. and Tweedie, R. L. (1996). Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli*, 2(4) :341–363.

Rosenbluth, M. N. and Rosenbluth, A. W. (1955). Monte Carlo calculation of the average extension of molecular chains. *The Journal of Chemical Physics*, 23(2) :356–359.

Taghavi, E., Lindsten, F., Svensson, L., and Schön, T. B. (2013). Adaptive stopping for fast particle smoothing. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6293–6297.

Tan, Z. (2015). Resampling Markov chain Monte Carlo algorithms : basic analysis and empirical comparisons. *J. Comput. Graph. Statist.*, 24(2) :328–356.

Waste-free Sequential Monte Carlo

This article has been published as:

Dau, H.-D. and Chopin, N. (2022). Waste-free sequential Monte Carlo. J. R. Stat. Soc. Ser. B. Stat. Methodol., 84(1) :114-148.

WASTE-FREE SEQUENTIAL MONTE CARLO

HAI-DANG DAU & NICOLAS CHOPIN

Abstract. A standard way to move particles in an SMC sampler is to apply several steps of an MCMC (Markov chain Monte Carlo) kernel. Unfortunately, it is not clear how many steps need to be performed for optimal performance. In addition, the output of the intermediate steps are discarded and thus wasted somehow. We propose a new, waste-free SMC algorithm which uses the outputs of all these intermediate MCMC steps as particles. We establish that its output is consistent and asymptotically normal. We use the expression of the asymptotic variance to develop various insights on how to implement the algorithm in practice. We develop in particular a method to estimate, from a single run of the algorithm, the asymptotic variance of any particle estimate. We show empirically, through a range of numerical examples, that waste-free SMC tends to outperform standard SMC samplers, and especially so in situations where the mixing of the considered MCMC kernels decreases across iterations (as in tempering or rare event problems).

1. Introduction

1.1. **Background.** Sequential Monte Carlo (SMC) methods are iterative stochastic algorithms that approximate a sequence of probability distributions through successive importance sampling, resampling, and Markov steps. Historically, they were mainly used to approximate the filtering distributions of a state-space model. More recently, they have been extended to an arbitrary sequence of probability distributions (Neal, 2001; Chopin, 2002; Del Moral et al., 2006); in such applications, they are often called "SMC samplers".

As an illustrative example, consider the tempering sequence:

$$\pi_t(\mathrm{d}x) \propto \nu(\mathrm{d}x)L(x)^{\gamma_t} \tag{1}$$

based on increasing exponents, $0 = \gamma_0 < \ldots < \gamma_T = 1$. This sequence may be used to interpolate between a distribution $\nu(\mathrm{d}x)$, which is easy to sample from, and a distribution of interest, $\pi(\mathrm{d}x) \propto \nu(\mathrm{d}x)L(x)$ (e.g. a Bayesian posterior distribution), which may be difficult to simulate directly. Other sequences of interest will be discussed later.

When used to sample from a fixed distribution (as in tempering), SMC samplers present several advantages over MCMC (Markov chain Monte Carlo). First, they provide an estimate of the normalising constant of the target distribution at no extra cost; this quantity is of interest in several cases, in particular in Bayesian model choice (e.g. Zhou et al., 2016). Second, they are easy to parallelise, as the bulk of the computation treats the N particles independently (Lee et al., 2010). Third, it is easy to make SMC samplers "adaptive"; that is, to use the current particle sample to automate the choice of most of its tuning parameters. This is often crucial for good performance.

To elaborate on the third point, a common strategy to move the particles is to apply a $k-$fold MCMC kernel that leaves the current distribution π_t invariant. One may use for instance a random walk Metropolis kernel, with the covariance of the proposal set to a small multiple of the empirical covariance of the particle sample. In that way, the algorithm automatically scales to the current distribution.

However, one tuning parameter of SMC samplers that is often overlooked in the literature is the number k of MCMC steps that should be applied to move the particles. For instance, Chopin and Ridgway (2017) set $k = 3$ arbitrarily in their numerical experiments, but it turns out that this value is very sub-optimal, as we show in our first numerical example.

A second issue with k is that there is no reason to set it to a fixed value across iterations. In application such as tempering, π_t may become more and more difficult to explore through MCMC; thus k should be increased accordingly, and may become very large.

To deal with these two issues, one could set k adaptively; that is, iterate MCMC steps until a certain stability criterion is met (Drovandi and Pettitt, 2011; Kantas et al., 2014; Ridgway, 2016; Salomone et al., 2018; Buchholz et al., 2020). However, in our experience, these approaches are not always entirely reliable. There seems to be a fundamental difficulty in determining, after k steps have been performed, that this value of k is optimal, without performing several extra steps.

A third, and perhaps more essential issue, is that, if indeed large values of k are required for good performance, the intermediate output of these k MCMC steps are not used directly, and seems somehow wasted.

1.2. **Motivation and plan.** These issues motivated us to develop a waste-free SMC algorithm that exploits the intermediate outputs of these MCMC steps; see Section 2. The basic idea is to resample only $M = N/P$ out of the N previous particles, for some $P \geq 2$. Then each resampled particle is moved $P - 1$ times through the chosen MCMC kernel. The resampled particles and their $P-1$ iterates are gathered to form a new sample of size N.

Standard results on the convergence of SMC estimates cannot be applied directly to this new algorithm. We establish the consistency and asymptotic normality of the output of waste-free SMC in Section 3. We also establish that wasteless SMC dominates standard SMC in terms of asymptotic variance in a simplified scenario.

These theoretical results (in particular the expression of the asymptotic variance) gives us various insights on how to implement waste-free SMC in practice; see Section 4. In particular, we are able to derive variance estimates and confidence intervals for any particle estimate, which may be computed from a single run.

To assess the performance and versatility of waste-free SMC, we perform numerical experiments in three different scenarios where SMC samplers already give state-of-the-art performance: logistic regression with a large number of predictors; the enumeration of Latin squares; and the computation of Gaussian orthant probabilities; see Section 5. In each case, waste-free SMC performs at least as well as properly tuned SMC samplers, while requiring considerably less tuning effort.

Proofs are delegated to the appendix.

1.3. **Related work.** We focus on SMC samplers based on invariant (MCMC) kernels. These algorithms have proved popular recently in a variety of applications,

such as rare events (Johansen et al., 2006; Cérou et al., 2012); experimental designs (Amzal et al., 2006); cross-validation (Bornn et al., 2010); variable selection (Schäfer and Chopin, 2013); graphical models (Naesseth et al., 2014); PAC-Bayesian classification (Ridgway et al., 2014); Gaussian orthant probabilities (Ridgway, 2016); Bayesian model choice in hidden Markov models (Zhou et al., 2016), and un-normalised models (Everitt et al., 2017); among others.

We note in passing that SMC samplers may be generalised to non-invariant kernels, as shown in Del Moral et al. (2006); see also Heng et al. (2020) for how to calibrate such kernels. On the other hand, it is also possible to add MCMC steps to various SMC algorithms that are not SMC samplers; the idea goes back to Berzuini et al. (1997). In particular, SMCMC (Sequential MCMC, Septier et al., 2009; Septier and Peters, 2016; Finke et al., 2020) algorithms approximate recursively the filtering distribution of a state-space model: each iteration t runs a MCMC chain that leaves invariant a certain (partly discrete) approximation of the current filter. It is not clear however how to derive a waste-free version of these algorithms, and thus we do not consider them further.

Several improvements proposed for standard SMC samplers might be also adapted to waste-free SMC, such as methods to combine the output of the intermediate steps, see Beskos et al. (2017) and South et al. (2019).

Finally, Tan (2015) proposes several algorithms presented as variations of the resample-move algorithm of Gilks and Berzuini (2001); one of them (generalized resample-move) is essentially equivalent to waste-free SMC in the context of tempering (with fixed temperatures). Note however that this paper gives little theoretical guidance on why the algorithm should work better than alternatives, on how to choose M and P (for a given N), and so on. Also, its numerical experiment does not use calibrated MCMC kernels, although, in our experience, using such kernels has a dramatic impact on performance.

2. Proposed algorithm

2.1. **Notations.** Throughout the paper, $(\mathcal{X}, \mathbb{X})$ stands for a measurable space, and $\varphi : \mathcal{X} \to \mathbb{R}$ for a measurable function; let $\|\varphi\|_\infty := \sup_{x\in\mathcal{X}} |\varphi(x)|$ (supremum norm). The expectation of $\varphi(X)$ when $X \sim \pi(\mathrm{d}x)$ is denoted by $\pi(\varphi)$; i.e. $\pi(\varphi) := \int \varphi(x)\pi(\mathrm{d}x)$. Recall that a Markov kernel $K(x, \mathrm{d}y)$ is a map $K : \mathcal{X} \times \mathbb{X} \to [0, 1]$ such that $x \to K(x, A)$ is measurable in x, for any $A \in \mathbb{X}$; and $A \to K(x, A)$ is a probability measure (on $(\mathcal{X}, \mathbb{X})$), for any $x \in \mathcal{X}$. We use the following standard notations for the integral operators associated to Markov kernel K: πK is the distribution such that $\pi K(A) = \int_{\mathcal{X}} \pi(\mathrm{d}x)K(x, A)$, and $K(\varphi)$ is the function $x \to \int_{\mathcal{X}} K(x, \mathrm{d}y)\varphi(y)$, for $\varphi : \mathcal{X} \to \mathbb{R}$.

Symbol $\Rightarrow$ means convergence in distribution, and $\|\cdot\|_{\mathrm{TV}}$ stands for the total variation norm, $\|\mu - \nu\|_{\mathrm{TV}} = \sup_{A\in\mathbb{X}} |\mu(A) - \nu(A)|$.

2.2. **A generic SMC sampler.** We consider a generic sequence of target probability distributions of the form (for $t = 0, 1, \ldots, T$):

$$\pi_t(\mathrm{d}x) = \frac{1}{L_t}\gamma_t(x)\nu(\mathrm{d}x) \tag{2}$$

where $\nu(\mathrm{d}x)$ is a probability measure, with respect to measurable space $(\mathcal{X}, \mathbb{X})$, γ_t is a measurable, non-negative function, and $L_t := \int_{\mathcal{X}} \gamma_t(x)\nu(\mathrm{d}x)$, the normalising constant, is assumed to be properly defined, i.e. $0 < L_t < \infty$. In the tempering

scenario mentioned in the introduction, $\gamma_t(x) = L(x)^{\gamma_t}$, for certain exponents γ_t. Other interesting scenarios include data tempering (sequential learning), where x represents a parameter, $\nu(\mathrm{d}x)$ its prior distribution, and $\gamma_t(x)$ is the likelihood of data-points $y_0, \ldots, y_t$; rare-event simulation (and likelihood-free inference), where $\gamma_t(x) = \mathbb{1}_{\mathcal{E}_t}(x)$, the indicator function of nested sets $\mathcal{E}_0 \supset \mathcal{E}_1 \supset \ldots$; among others. See e.g. Chapter 3 of Chopin and Papaspiliopoulos (2020) for a review of common applications of SMC samplers, and the sequence of target distributions arising in these applications.

One way to track the sequence π_t would be to perform sequential importance sampling: sample particles (random variates) from the initial distribution $\nu(\mathrm{d}x)$, then reweight them sequentially according to weight function $G_t(x) := \gamma_t(x)/\gamma_{t-1}(x)$ (for $t \geq 1$, and $G_0(x) := \gamma_0(x)$). In most applications however, the weights degenerate quickly, making this naive approach useless.

SMC samplers alternate such reweighting steps with resampling and Markov steps. For the latter, we introduce Markov kernels $M_t(x_{t-1}, \mathrm{d}x_t)$ which leave invariant the target distributions: $\pi_{t-1}M_t = \pi_{t-1}$ for $t \geq 1$. It is easy to check that the sequence of Feynman-Kac distributions (for $t = 0, \ldots, T$) defined as:

$$\mathbb{Q}_t(\mathrm{d}x_{0:t}) = \frac{1}{L_t}\nu(\mathrm{d}x_0)\prod_{s=1}^{t} M_s(x_{s-1}, \mathrm{d}x_s)\prod_{s=0}^{t} G_s(x_s) \tag{3}$$

is such that the marginal distribution of variable X_t (with respect to $\mathbb{Q}_t$) is π_t. We call Feynman-Kac model the set of the components that define this sequence of distributions, that is, the initial distribution ν, the kernels M_t, $t = 1, \ldots, T$, and the functions G_t, $t = 0, \ldots, T$. For more background on Feynman-Kac distributions, see e.g. Del Moral (2004).

Algorithm 1 recalls the structure of an SMC sampler that corresponds to this Feynman-Kac model; and in particular which targets at each iteration t distribution π_t. It takes as inputs: N, the number of particles, the considered Feynman-Kac model, and the chosen resampling scheme (function `resample`). Several resampling schemes exist. In this paper, we focus for simplicity on multinomial resampling, which generates ancestor variables A_t^n independently from the categorical distribution that generates label m with probability W_t^m.

At any iteration t, quantity $\sum_{n=1}^{N} W_t^n \varphi(X_t^n)$ is an estimate of the expectation $\pi_t(\varphi)$, for $\varphi : \mathcal{X} \to \mathbb{R}$, and quantity $L_t^N := \prod_{s=0}^{t} \ell_s^N$, where $\ell_s^N := N^{-1}\sum_{n=1}^{N} w_s^n$, is an estimate of the normalising constant L_t. These estimates are consistent and asymptotically normal (as $N \to +\infty$) under general conditions.

2.3. **Note on the generality of Algorithm 1.** While generic, Algorithm 1 is a simplified version of most practical SMC samplers. In particular, we have stressed in the introduction the importance of making SMC samplers adaptive; that is, to adapt both the distributions π_t and the Markov kernels M_t on the fly. This means that these quantities may depend on the current particle sample. For simplicity, our notations do not account for this. We will see later that similar adaptation tricks may be developed for waste-free SMC.

Another interesting generalisation is when the state space $\mathcal{X}$ evolves over time; in particular when its dimension increases. This happens for instance when performing sequential inference on a model involving latent variables. The ideas developed in

Algorithm 1: Generic SMC sampler

Input: Integer $N \geq 1$, a Feynman-Kac model (initial distribution $\nu(\mathrm{d}x)$, functions G_t, Markov kernels M_t)

for $t \leftarrow 0$ **to** T **do**
  if $t = 0$ **then**
    for $n = 1$ **to** N **do**
      $X_0^n \sim \nu(\mathrm{d}x_0)$
  else
    $A_t^{1:N} \sim \texttt{resample}(N, W_{t-1}^{1:N})$
    for $n = 1$ **to** N **do**
      $X_t^n \sim M_t(X_{t-1}^{A_t^n}, \mathrm{d}x_t)$
  for $n \leftarrow 1$ **to** N **do**
    $w_t^n \leftarrow G_t(X_t^n)$
  for $n \leftarrow 1$ **to** N **do**
    $W_t^n \leftarrow w_t^n / \sum_{m=1}^N w_t^m$

this paper may easily be adapted to this scenario, as we shall see in our third numerical example. For the sake of exposition, however, we focus on the fixed state space case.

2.4. **Proposed algorithm: waste-free SMC.** The idea behind waste-free SMC is to resample only M ancestors, with $M \ll N$. Then each of these ancestors is moved $P - 1$ times through Markov kernel M_t. The resulting M chains of length P are then put together to form a new particle sample, of size $N = MP$. See Algorithm 2.

The output of the algorithm may be used exactly in the same way as for standard SMC: e.g. $\sum_{n=1}^N W_t^n \varphi(X_t^n)$ is an estimate of $\pi_t(\varphi)$.

To get some intuition why waste-free SMC may be a valid and interesting alternative to standard SMC, consider at time $t-1$ a fictitious particle X_{t-1}^n, whose weight W_{t-1}^n is large. In a standard SMC sampler, this particle is selected many times as an ancestor for the Markov step. Then, if M_t mixes poorly, its many children will be strongly correlated.

On the other hand, in waste-less SMC, provided that $M \ll N$, the particle X_{t-1}^n is selected a much smaller number of times; each time it is selected, P successive variables are introduced in the sample. By construction, two such variables should be less correlated than if they had the same ancestor (as in standard SMC); see Figure 1 for a graphical representation of this idea.

Another insight is provided by chaos propagation theory (Del Moral, 2004, Chap. 8), which says that, when $M \ll N$, M resampled particles behave essentially like M independent variables that follows the current target distribution. Thus, in a certain asymptotic regime, we expect the particle sample to behave like the variables of M independent, stationary Markov chains, of length P.

Before backing these intuitions with a proper analysis, we provide a last insight regarding the underlying structure of waste-free SMC.

Algorithm 2: Waste-free SMC sampler

Input: Integers $M, P \geq 1$ (let $N \leftarrow MP$), a Feynman-Kac model (initial distribution $\nu(\mathrm{d}x)$, functions G_t, Markov kernels M_t)

for $t \leftarrow 0$ **to** T **do**
- **if** $t = 0$ **then**
  - **for** $n \leftarrow 1$ **to** N **do**
    - $X_0^n \sim \nu(\mathrm{d}x_0)$
- **else**
  - $A_t^{1:M} \sim \texttt{resample}(M, W_{t-1}^{1:N})$
  - **for** $m \leftarrow 1$ **to** M **do**
    - $\tilde{X}_t^{m,1} \leftarrow X_{t-1}^{A_t^m}$
    - **for** $p \leftarrow 2$ **to** P **do**
      - $\tilde{X}_t^{m,p} \leftarrow M_t(\tilde{X}_t^{m,p-1}, \mathrm{d}x_t)$
  - Gather variables $\tilde{X}_t^{m,p}$ so as to form new sample $X_t^{1:N}$
- **for** $n \leftarrow 1$ **to** N **do**
  - $w_t^n \leftarrow G_t(X_t^n)$
- **for** $n \leftarrow 1$ **to** N **do**
  - $W_t^n \leftarrow w_t^n / \sum_{m=1}^N w_t^m$

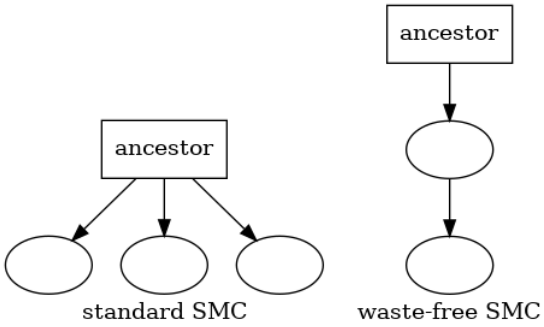

FIGURE 1. Pictorial representation of dependencies in standard SMC and waste-free SMC. Left: in standard SMC, an ancestor generates 3 children for the next iteration. Right: in waste-free SMC, the same ancestor generates itself, one child, and one grandchild. Each arrow corresponds to one transition through kernel M_t.

2.5. **Feynman-Kac model associated with waste-free SMC.** Algorithm 2 may be cast as a standard SMC sampler that propagates and reweights particles that are Markov chains of length P. The components of the corresponding Feynman-Kac model may be defined as follows. Assume $P \geq 1$ is fixed. Let $\mathcal{Z} = \mathcal{X}^P$, and, for $z \in \mathcal{Z}$, denote component p as $z[p]$: $z = (z[1], \ldots, z[P])$. Then

define the potential functions as:

$$G_t^{\mathrm{wf}}(z) := \frac{1}{P}\sum_{p=1}^{P} G_t(z[p]) \tag{4}$$

the initial distribution as: $\nu^{\mathrm{wf}}(\mathrm{d}z) := \prod_{p=1}^{P} \nu(\mathrm{d}z[p])$, and the Markov kernels as:

$$M_t^{\mathrm{wf}}(z_{t-1}, \mathrm{d}z_t) := \left\{\sum_{p=1}^{P} \frac{G_{t-1}(z_{t-1}[p])}{\sum_{q=1}^{P} G_{t-1}(z_{t-1}[q])} \times M_t(z_{t-1}[p], \mathrm{d}z_t[1])\right\} \prod_{p=2}^{P} M_t(z_t[p-1], \mathrm{d}z_t[p]). \tag{5}$$

The following proposition explains how this waste-free Feynman-Kac model relates to the initial Feynman-Kac model of Algorithm 1.

Proposition 1. *The Feynman-Kac model associated with initial distribution ν^{wf}, Markov kernels M_t^{wf}, and functions G_t^{wf}, that is, the sequence of distributions:*

$$\mathbb{Q}_t^{\mathrm{wf}}(dz_{0:t}) = \frac{1}{L_t^{\mathrm{wf}}} \nu^{\mathrm{wf}}(\mathrm{d}z_0) \prod_{s=1}^{t} M_s^{\mathrm{wf}}(z_{s-1}, \mathrm{d}z_s) \prod_{s=0}^{t} G_s^{\mathrm{wf}}(z_s)$$

where L_t^{wf} is a normalising constant, is such that:

- *$L_t^{\mathrm{wf}} = L_t$, the normalising constant of (2) and (3);*
- *$\mathbb{Q}_t^{\mathrm{wf}}(dz_t)$ is the distribution of a stationary Markov chain of size P whose Markov kernel is M_t (and thus whose initial distribution is π_t):*

$$\mathbb{Q}_t^{\mathrm{wf}}(\mathrm{d}z_t) = \pi_t(\mathrm{d}z_t[1]) \prod_{p=2}^{P} M_t(z_t[p-1], \mathrm{d}z_t[p]).$$

We can now interpret Algorithm 2 as an instance of Algorithm 1 where the number of particles is M, and the underlying Feynman-Kac model is defined as above. In particular, consider how Algorithm 1 would operate if applied to that Feynman-Kac model. At time t, it would select randomly an ancestor z_{t-1} (a chain of length P), with probability $\propto \sum_{p=1}^{P} G_{t-1}(z_{t-1}[p])$. Then, when kernel M_t^{wf} is applied to this chain, one component would be selected randomly, with probability $G_{t-1}(z_{t-1}[p]) / \sum_{q=1}^{P} G_{t-1}(z_{t-1}[q])$. Thus, this particular component would be used as a starting point of the subsequent chain with probability $\propto G_{t-1}(z_{t-1}[p])$. This is precisely what is done in Algorithm 2.

This interpretation of waste-free SMC as a standard SMC sampler makes it easy to derive several of its properties; for instance, regarding its estimates of the normalising constants.

Proposition 2. *At iteration $t \geq 0$ of Algorithm 2 , the quantity*

$$L_t^N := \prod_{s=0}^{t} \ell_s^N, \quad \text{where } \ell_s^N := \frac{1}{N}\sum_{n=1}^{N} G_s(X_s^n) \tag{6}$$

is an unbiased estimate of L_t, the normalising constant of target distribution π_t, as defined in (2).

For a proof, see Section A.1 in the Appendix.

This proposition is a small variation over the well known result of (Del Moral, 1996) that, in a standard SMC sampler, the estimate of the normalising constant estimate is unbiased.

We can also use the interpretation of waste-free SMC as a standard SMC sampler to derive asymptotic results.

Proposition 3. *For $P \geq 1$ fixed, and $\varphi : \mathcal{X} \to \mathbb{R}$ measurable and bounded, the output of Algorithm 2 at time $t \geq 0$ is such that*

$$\sqrt{N}\left(\frac{1}{N}\sum_{n=1}^{N}\varphi(X_t^n) - \pi_{t-1}(\varphi)\right) \Rightarrow \mathcal{N}\left(0, \tilde{\mathcal{V}}_t^P(\varphi)\right) \tag{7}$$

$$\sqrt{N}\left(\sum_{n=1}^{N}W_t^n\varphi(X_t^n) - \pi_t(\varphi)\right) \Rightarrow \mathcal{N}\left(0, \mathcal{V}_t^P(\varphi)\right) \tag{8}$$

as $M \to +\infty$, $N = MP$, where π_{t-1} means ν in (7) when $t = 0$, $\tilde{\mathcal{V}}_0^P(\varphi) := \mathrm{Var}_\nu(\varphi)$,

$$\tilde{\mathcal{V}}_t^P(\varphi) := \mathcal{V}_{t-1}^P(\bar{M}_t^P\varphi) + v_P(M_t, \varphi), \quad t \geq 1, \tag{9}$$

$$\mathcal{V}_t^P(\varphi) := \tilde{\mathcal{V}}_t^P\left(\bar{G}_t(\varphi - \pi_t\varphi)\right), \quad t \geq 0, \tag{10}$$

$\bar{G}_t := G_t/\ell_t$, $\bar{M}_t^P = P^{-1}\sum_{p=1}^{P} M_t^{p-1}$,

$$v_P(M_t, \varphi) := \mathrm{Var}\left(\frac{1}{\sqrt{P}}\sum_{p=0}^{P-1}\varphi(Y_p)\right)$$

and $(Y_p)_{p\geq 0}$ stands for a stationary Markov chain with kernel M_t (i.e. $Y_0 \sim \pi_t$).

This proposition is stated without proof, as it amounts to applying known central limit theorems (see Chapter 11 of Chopin and Papaspiliopoulos, 2020, and references therein) for SMC estimates to the waste-free Feynman-Kac model mentioned above. Notice how the asymptotic variances depend on P in a non-trivial way. This suggests that the fixed P regime is not very convenient; in particular it is not clear how to choose P for optimal performance. If we take $P \to +\infty$, we expect the first term of (9) to go to zero, and the second term to converge to the asymptotic variance of kernel M_t. This suggests, at the very least, that taking P large may often be reasonable. The next section studies the asymptotic behaviour of the algorithm as $P \to +\infty$.

3. Convergence as $P \to +\infty$

3.1. Assumptions.

This section is concerned with the behaviour of waste-free SMC in the "long-chain" regime, that is, when $P \to +\infty$, while M is either fixed or may grow with P at some rate. We start by remarking that this regime requires some assumption on the mixing of the Markov kernels M_t. Indeed, assume that M_t is the identity kernel: $M_t(x_{t-1}, \mathrm{d}x_t) = \delta_{x_{t-1}}(\mathrm{d}x_t)$. In that case, at time 1, one has:

$$\frac{1}{N}\sum_{n=1}^{N}\varphi(X_1^n) = \frac{1}{M}\sum_{m=1}^{M}\varphi(X_0^{A_0^m})$$

since the P particles $\tilde{X}_t^{m,p}$ are identical for a given m. The variance of this quantity should be $\mathcal{O}(M^{-1})$, and cannot go to zero if M is kept fixed.

We thus consider the following assumptions.

Assumption (M)**.** *The Markov kernels M_t are uniformly ergodic, that is, there exist constants $C_t \geq 0$ and $\rho_t \in [0, 1[$ such that,*

$$\left\| M_t^k(x_{t-1}, \mathrm{d}x_t) - \pi_{t-1}(\mathrm{d}x_t) \right\|_{\mathrm{TV}} \leq C_t \rho_t^k, \quad \forall x_{t-1} \in \mathcal{X}, k \geq 1.$$

Assumption (G)**.** *The functions G_t are upper-bounded, $G_t(x) \leq D_t$ for some $D_t > 0$ and all $x \in \mathcal{X}$.*

Ergodic Markov kernels in an SMC sampler was also considered in Beskos et al. (2014) in order to study the behaviour of the algorithm as the dimension of the state space gets high.

3.2. **Non-asymptotic bound.** We first state a non-asymptotic result.

Proposition 4. *Under Assumptions (M) and (G), there exist constants c_t and c'_t such that the following inequalities apply to the output of iteration $t \geq 0$ of Algorithm 2, for any $M, P \geq 1$, and any bounded function $\varphi : \mathcal{X} \to \mathbb{R}$:*

$$\mathbb{E}\left\{ \frac{1}{N} \sum_{n=1}^{N} \varphi(X_t^n) - \pi_{t-1}(\varphi) \right\}^2 \leq c_t \frac{\|\varphi\|_\infty^2}{N} \tag{11}$$

$$\mathbb{E}\left\{ \sum_{n=1}^{N} W_t^n \varphi(X_t^n) - \pi_t(\varphi) \right\}^2 \leq c'_t \frac{\|\varphi\|_\infty^2}{N} \tag{12}$$

where π_{t-1} means ν in (11) *at time $t = 0$.*

For a proof, see Section A.2 of the Appendix.

The constants c_t and c'_t are not sharp. However, this result remains interesting, in that it shows that waste-free SMC is consistent (in L^2 norm, and thus in probability) whenever $N = MP \to +\infty$, that is, whenever $P \to +\infty$, or $M \to +\infty$, or both simultaneously, possibly at different rates.

3.3. **Central limit theorems.** We now state a central limit theorem for the long chain regime.

Theorem 1. *Under Assumptions (M) and (G), for $M = M(P) = \mathcal{O}(P^\alpha)$, $\alpha \geq 0$ (i.e. M is either fixed or grows with P at a certain rate) and $\varphi : \mathcal{X} \to \mathbb{R}$ measurable and bounded, one has at any time $t \geq 0$*

$$\sqrt{N}\left(\frac{1}{N} \sum_{n=1}^{N} \varphi(X_t^n) - \pi_{t-1}(\varphi) \right) \Rightarrow \mathcal{N}\left(0, \tilde{\mathcal{V}}_t(\varphi)\right) \tag{13}$$

$$\sqrt{N}\left(\sum_{n=1}^{N} W_t^n \varphi(X_t^n) - \pi_t(\varphi) \right) \Rightarrow \mathcal{N}\left(0, \mathcal{V}_t(\varphi)\right) \tag{14}$$

as $P \to \infty$ (or equivalently as $N \to \infty$, since $N = MP$), where π_{t-1} in (13) *means ν at time $t = 0$, $\tilde{\mathcal{V}}_0(\varphi) = \mathrm{Var}_\nu(\varphi)$,*

$$\tilde{\mathcal{V}}_t(\varphi) := v_\infty(M_t, \varphi) := \mathrm{Var}\left(\varphi(Y_0)\right) + 2 \sum_{p=1}^{\infty} \mathrm{Cov}\left(\varphi(Y_0), \varphi(Y_p)\right), \quad t \geq 1, \tag{15}$$

$$\mathcal{V}_t(\varphi) := \tilde{\mathcal{V}}_t\left(\bar{G}_t(\varphi - \pi_t\varphi)\right), \quad t \geq 0, \tag{16}$$

and $(Y_p)_{p \geq 0}$ stands for a stationary Markov chain with kernel M_t (hence $Y_0 \sim \pi_t$).

For a proof, see Section A.3 in the Appendix.

The most striking feature of the asymptotic variances above is that they depend only on the current time step t; in standard CLTs for SMC algorithms, these quantities are a sum of terms depending on all the previous time steps. More precisely, $v_\infty(M_t, \varphi)$ is the asymptotic variance of an average $P^{-1}\sum_{p=1}^{P}\varphi(Y_p)$ obtained from a single stationary Markov chain with kernel M_t. Equation (15) shows that the N particles X_t^n behave like M independent, 'long' Markov chains. This simple interpretation will make it possible to construct estimates of the asymptotic variances above; see Section 4.3. We also note that these asymptotic variances do not depend on M (when M is fixed), or its growth rate (when $M = \mathcal{O}(P^\alpha)$, $\alpha > 0$). This suggests that the performance of the algorithm should depend weakly on the actual value of M, provided $M \ll N$.

We now consider a similar result for the normalising constant estimates that may be obtained from Algorithm 2.

Theorem 2. *Under Assumptions (M) and (G), for $M = \mathcal{O}(P^\alpha)$, $\alpha \in [0,1)$ (i.e. either M is fixed, or M grows sub-linearly with P), and $\varphi : \mathcal{X} \to \mathbb{R}$ measurable and bounded, one has at time $t \geq 0$:*

$$\sqrt{N}\left(\log L_t^N - \log L_t\right) \Rightarrow \mathcal{N}\left(0, \sum_{s=0}^{t} v_\infty(M_s, \bar{G}_s)\right) \tag{17}$$

as $P \to \infty$ (or equivalently as $N \to \infty$ since $N = MP$).

For a proof, see Section A.4 in the Appendix.

The theorem above puts a stronger constraint on M; i.e. it requires $M \ll P$, and thus $M \ll N^{1/2}$ (while Theorem 1 requires only $M \ll N$).

Note that

$$\log L_t^N - \log L_t = \sum_{s=0}^{t}\left(\log \ell_s^N - \log \ell_s\right), \quad \text{where } \ell_s^N = \frac{1}{N}\sum_{n=1}^{N} G_s(X_s^n),$$

and we could already deduce from (13) and the delta-method that

$$\sqrt{N}\left(\log \ell_s^N - \log \ell_s\right) \Rightarrow \mathcal{N}\left(0, v_\infty(M_s, \bar{G}_s)\right).$$

Thus, (17) suggests that the error terms in this decomposition are nearly independent. Again, we shall use this interpretation to derive an estimate of the asymptotic variance of L_t^N.

3.4. **Comparing the asymptotic variances of standard and waste-free SMC.** In this sub-section, we use the previous results to compare formally the performance of standard SMC and waste-free SMC in an artificial example.

Let A_t, $t = 0, 1, \ldots$ be a sequence of subsets of $\mathcal{X}$ such that $A_0 \supset A_1 \supset \ldots$ and $\nu(A_t) = r^t$ for some $r < 1$, and some initial distribution ν. Consider the Feynman-Kac distributions such that $G_t(x_t) = \mathbb{1}_{A_t}(x_t)$ and $M_t = K_t^k$, i.e. the $k-$fold kernel such that $K_t(x, B) = (1-p)\mathbb{1}_B(x) + p\pi_{t-1}(B)$ for some $0 < p < 1$. (In words, with probability p, do not move, with probability $1-p$, sample exactly from the current target.)

A standard SMC sampler applied to this problem will fulfil a CLT of the form:

$$\sqrt{N}\left(\sum_{n=1}^{N} W_t^n \varphi(X_t^n) - \pi_t(\varphi)\right) \Rightarrow \mathcal{N}\left(0, \mathcal{V}_t^{\mathrm{std},k}(\varphi)\right);$$

see (31) in the proof of Proposition 5 for an expression for $\mathcal{V}_t^{\mathrm{std},k}(\varphi)$ and e.g. Chapter 11 of Chopin and Papaspiliopoulos (2020) for more details. Define the 'inflation factor' (relative error) for standard SMC to be:

$$\mathrm{IF}_t^{\mathrm{std},k}(\varphi) := \frac{\mathcal{V}_t^{\mathrm{std},k}(\varphi)}{\mathrm{Var}_{\pi_t}(\varphi)}.$$

For waste-free SMC, we take $k = 1$, i.e. $M_t = K_t$, and define similarly its inflation factor to be $\mathrm{IF}_t^{\mathrm{wf}}(\varphi) := \frac{\mathcal{V}_t^{\mathrm{wf}}(\varphi)}{\mathrm{Var}_{\pi_t}(\varphi)}$, where $\mathcal{V}_t^{\mathrm{wf}}(\varphi)$ is the asymptotic variance defined in Theorem 1.

Proposition 5. *For the model considered above, let $k_0 := \log r / 2\log(1-p)$, then*

1. *The quantities $\mathrm{IF}_t^{std,k}(\varphi)$ and $\mathrm{IF}_t^{wf}(\varphi)$ do not depend on φ.*
2. *For the standard SMC sampler, the inflation factor $\mathrm{IF}_t^{std,k}$ is stable with respect to t if and only if $k \geq k_0$. If $k < k_0$ however, $\mathrm{IF}_t^{std,k}$ explodes exponentially with t.*
3. *For the waste-free SMC sampler, IF_t^{wf} is stable with respect to t and is always equal to $\frac{1}{r}\left(\frac{2}{p} - 1\right)$.*
4. *For any choice of k, we have*

$$\lim_{t\to\infty} \frac{\mathrm{IF}_t^{wf}}{k\,\mathrm{IF}_t^{std,k}} \leq 4.$$

For a proof, see Section A.5 in the Appendix.

In words, the performance of standard SMC may deteriorate very quickly whenever the number of MCMC steps, k, is set to a too small value. On the other hand, up to small factor, waste-free SMC provides the same level of performance as standard SMC based on a well chosen value for k.

Of course, these statements are proven here for a specific example; however, our numerical experiments (Section 5) suggest they apply more generally.

4. Practical considerations

4.1. **Choice of M.** By default, we recommend to take $M \ll N$, first, because our previous results indicate that, within this regime, performance should be robust to the precise value of M; and, second, because we observe empirically that this regime usually leads to best performance (i.e. lowest variance for a given CPU budget). See our numerical experiments in Section 5.

On parallel hardware, we recommend to take M equal to, or larger than the number of processors, as it is easy to divide the computational load of each iteration of Algorithm 2 into M independent tasks.

4.2. **Choice of kernels M_t.** As discussed in the introduction, a standard practice is to set M_t to be a $k-$fold Metropolis kernel, whose proposal is calibrated on the current particle sample; e.g. for a random walk proposal, set the covariance matrix of the proposal to a certain fraction of the empirical covariance matrix of the particles.

This type of recipe may be used within waste-free SMC, with one important twist. Contrary to standard SMC, we recommend to always take $k = 1$. This recommendation is based on the following thinning argument. We know from

MCMC theory that thinning (subsampling) an MCMC chain is generally detrimental: Geyer (1992, Theorem 3.3) shows that $kv_\infty(M_t^k, \varphi) > v_\infty(M_t, \varphi)$ (provided M_t is reversible and irreducible). In words, between two estimates computed from the same long chain, one using all the samples, and the other using only one every other k-sample, the former will have a lower variance (asymptotically, as the length of the chain goes to infinity).

The same remark applies to waste-free SMC: if we compare a waste-free SMC sampler with N particles, and Markov kernels $M_t = K_t^k$, for a certain K_t, with the same algorithm with kN particles, and kernels $M_t = K_t$, then the latter will have (asymptotically) lower variance, given the expression of the asymptotic variances in Theorem 1.

As announced in the introduction, we see therefore that waste-free SMC is indeed more economical than standard SMC, as it is able to exploit all the intermediate steps of a given MCMC kernel (while standard SMC often requires to take $k \gg 1$ for optimal performance).

4.3. **Variance estimation from a single run.** As explained below Theorem 1, the output of waste-free SMC at time t behaves asymptotically like M independent, stationary chains of size P. Thus, to estimate the asymptotic variance $\tilde{V}_t(\varphi) = v_\infty(M_t, \varphi)$ in (13), we propose the following 'M-chain estimate'. Denote by $\gamma_{t,q}^{M,P}$ the empirical autocovariance of order $q \in \{0, 1, \dots, p-1\}$ computed from the M chains:

$$\gamma_{t,q}^{M,P} := \frac{1}{MP} \sum_{m=1}^{M} \sum_{p=1}^{P-q} \left[\varphi(\tilde{X}_t^{m,p}) - \mu_t^{M,P}(\varphi)\right] \left[\varphi(\tilde{X}_t^{m,p+q}) - \mu_t^{M,P}(\varphi)\right]$$

where $\mu_t^{M,P}(\varphi) := N^{-1} \sum_{m=1}^{M} \sum_{p=1}^{P} \varphi(\tilde{X}_t^{m,p})$ is the empirical mean. Then, the estimator is defined as

$$\tilde{V}_t^{M,P}(\varphi) := \psi_P\left(\gamma_{t,0}^{M,P}(\varphi), \dots, \gamma_{t,P-1}^{M,P}(\varphi)\right)$$

where $\psi_P : \mathbb{R}^P \to \mathbb{R}$ is a certain estimator of the asymptotic variance $v_\infty(M_t, \varphi)$ based on the autocorrelations of a single chain of length P.

Several such single-chain estimators ψ_P have been proposed in the literature, see e.g. the introduction of Flegal and Jones (2010). In our experiments, we found the initial monotone sequence estimator of Geyer (1992) to be a convenient default, as it is simple to use (no tuning parameter), and it seems to work well. Note however that this estimator is based on a property which is specific to reversible kernels (namely that sums of adjacent pairs of autocovariance form a decreasing sequence). When the chosen kernels M_t are not reversible, one may consider an alternative estimator; see our third numerical experiment (Section 5) for more discussion on this point.

To estimate $V_t(\varphi) = \tilde{V}_t(G_t(\varphi - \mathbb{Q}_t(\varphi)))$, we use the same approach with φ replaced by $G_t(\varphi - \mathbb{Q}_t^N(\varphi))$, $\mathbb{Q}_t^N(\varphi) = \sum_{n=1}^{N} W_t^n \varphi(X_t^n)$. Similarly, to estimate each term in the asymptotic variance of the log normalising constant, (17), we replace $\bar{G}_t = G_t/\ell_t$ by G_t/ℓ_t^N.

We note that there is an alternative approach to obtain variance estimates from a single run of waste-free SMC. It consists in (a) casting waste-free SMC as a standard SMC sampler, as we did in Section 2.5 (taking P fixed); and (b) to apply the method of Lee and Whiteley (2018), see also Chan and Lai (2013), Olsson

and Douc (2019) and Du and Guyader (2019), for obtaining variance estimates from SMC outputs. This method relies on genealogy tracking (i.e. tracking the ancestors at time 0 of each current particle).

This alternative approach has two drawbacks however. First, it relies on the fixed P regime, while, as already said, we recommend by default to run waste-free SMC in the $P \to +\infty$ regime, i.e. by taking $M \ll N$. Second, the method of Lee and Whiteley (2018) degenerates as soon as the number of common ancestors of the N particle drops to one; something which tends to occur quickly as t increases.

One may mitigate the degeneracy by tracking the genealogy only up to time $t-l$, for a certain lag value l, as recommended by Olsson and Douc (2019). However this introduces a bias, and choosing l is non-trivial.

We will compare both approaches in the numerical experiments of Section 5.

4.4. **On-line adaptation of P.** In certain applications, the mixing of kernels M_t may vary wildly with t; for instance, for a tempering sequence, the mixing of M_t may deteriorate over time. The second numerical example in Section 5 illustrates this phenomenon.

In such a case, it makes sense to adjust the computational effort to the mixing of the chain. That is, at time t, take $P = P_t$ so that the variance of estimates computed at time t stay of the same order of magnitude. In practice, we found the following strategy to work reasonably well: at iteration t, adjust P_t so that it exceeds κ times the auto-correlation time of kernel M_t, i.e. the quantity $v_\infty(M_t, \varphi)/2\mathrm{Var}_{\pi_t}(\varphi)$ for a certain constant $\kappa \geq 1$, and a certain function φ, as estimated from the current sample (which consists of M chains of length P_t). In our simulations, we took $\varphi = \log G_t$, and κ between 2 and 10. To adjust P_t, we set it to an initial value, then we doubled it until the requirement was met.

The main drawback of this adaptive approach is that it makes the CPU time of the algorithm random, which is less convenient for the user. On the other hand, it seems to present two advantages, as observed in our experiments (see second example in Section 5): (a) it avoids the poor performance one obtains by taking a value for P that is too small for certain iterations t; and (b) it makes the variance estimates more robust in this type of scenario.

5. Numerical experiments

In this section, we evaluate the performance of waste-free SMC in a variety of challenging scenarios, covering different types of state-spaces (continuous or discrete, with a fixed or an increasing dimension), of sequence of target distributions (based on tempering or something else), and of MCMC kernels (Metropolis or Gibbs). In each example, standard SMC is known to be a competitive approach, and we assess in particular how waste-free SMC may improve on the performance of standard SMC.

5.1. **Logistic regression.** We consider the problem of sampling from, and computing the normalising constant of, the posterior distribution of a logistic regression model, based on data $(y_i, z_i) \in \{-1, 1\} \times \mathbb{R}^p$, parameter $x \in \mathbb{R}^p$, and likelihood

$$L(x) = \prod_{i=1}^{n_\mathcal{D}} F(y_i x^T z_i), \quad F(x) = \frac{1}{1 + e^{-x}}.$$

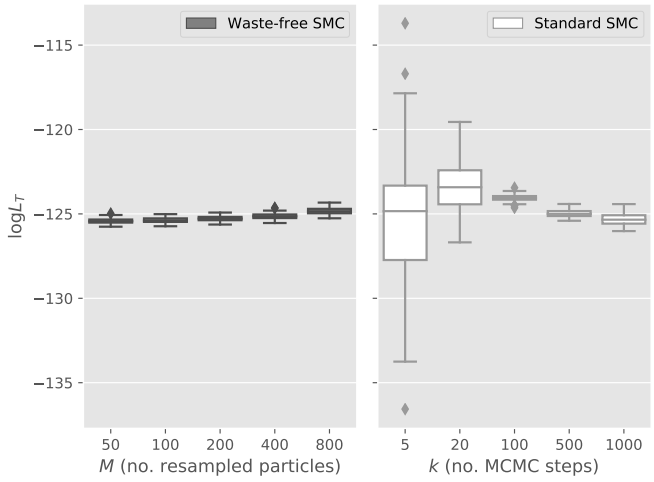

FIGURE 2. Logistic regression: estimates of the normalising constant obtained from waste-free SMC ($N = 2 \times 10^5$) and standard SMC ($N = 2 \times 10^5/k$).

We consider the sonar dataset (available in the UCI machine learning repository), which is one of the more challenging datasets considered in Chopin and Ridgway (2017), and for which SMC tempering is one of the competitive alternatives (and the only one that may be used to estimate the marginal likelihood). Following standard practice, each predictor is rescaled to have mean 0 and standard deviation 0.5; an intercept is added; the dimension of $\mathcal{X}$ is then $p = 63$. The prior is an independent product of centred normal distributions, with standard deviation 20 for the intercept, 5 for other coordinates.

We compare the performance of standard SMC and waste-free SMC when applied to the tempering sequence $\pi_t(\mathrm{d}x) \propto \nu(\mathrm{d}x)L(x)^{\gamma_t}$. In both cases, the tempering exponents are set automatically (using Brent's method) so that the ESS of each importance sampling step equals αN, and the Markov kernel M_t is a k-fold random walk Metropolis kernel calibrated to the resampled particles (see Section 4.2). For waste-free, we always take $k = 1$ (as per the thinning argument of the same Section). We take $\alpha = 1/2$ here; see the supplement for results with other values of α.

Figure 2 plots box-plots of estimates of the log of the normalising constant of the posterior obtained from 100 independent runs of standard SMC, for $k = 5$, 20, 100, 500, and 1000 and waste-free SMC for $k = 1$, and $M = 50$, 100, 200, 400 and 800. The number of particles is set to $N = N_0/k$, with $N_0 = 2 \times 10^5$, so that all algorithms have roughly the same CPU cost. (For waste-free, P is adjusted accordingly, i.e. $P = N/M$, with $N = 2 \times 10^5$.) Figure 3 does the same for the estimate of the posterior expectation of the mean of all components of x, namely $\pi_T(\varphi)$ with $\varphi(x) := p^{-1} \sum_{s=1}^{p} x_s$ for $x \in \mathbb{R}^p$.

These figures deserve several comments. First, waste-free seems to perform best in the "long chain" regime, when $M \ll N$. Second, within this regime, the performance seems robust to the choice of M; notice how the same level of performance is obtained whether $M = 50$ or $M = 400$ (similar performance is also obtained for

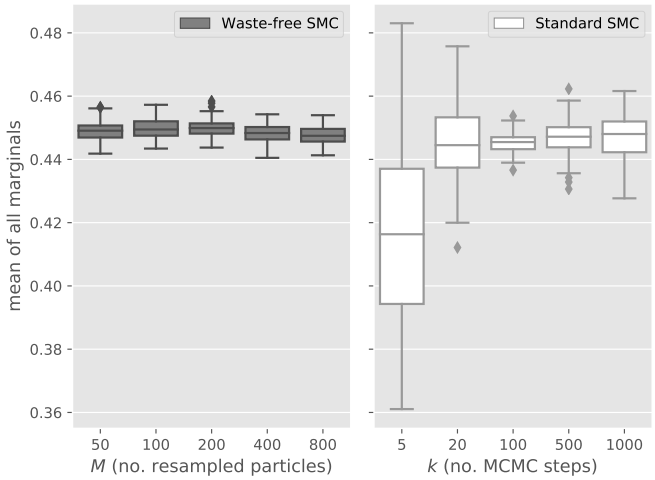

FIGURE 3. Same plot as Figure 2 for the estimate of the posterior expectation of the mean of all coordinates.

$M < 50$, results not shown. We focused on $M \geq 50$ for reasons related to parallel hardware as discussed in Section 4.1.) Third, in contrast, it seems difficult to choose k to obtain optimal performance; notice in particular that Figure 3 suggests to take $k = 100$, but, for this value of k, the estimate of the log-normalising constant seems biased, see Figure 2. (Interestingly, we observed such an upward bias for all values of k when we ran standard SMC for a smaller value of N, $N = 10^5$; hence standard SMC seems also slightly less robust to the choice of N; results not shown.) Fourth, and perhaps most importantly, we are able to obtain better performance from waste-free SMC for a given CPU budget.

We now evaluate the performance of the variance estimates discussed in Section 4.3. Figure 4 shows box-plots of these estimates obtained from 100 runs of waste-free SMC, for $N = 2 \times 10^5$ and $M = 50$: the $M-$chain estimate advocated in Section 4.3; the estimate of Olsson and Douc (2019), with a lag of 3 (the biased, but more stable version of Lee and Whiteley (2018), as explained in Section 4.3) and finally, the empirical variance over 10 independent runs. All these variance estimates are re-scaled by the same factor, such that the empirical variance over the 100 runs equals one. (Other values for the lag in the method of Olsson and Douc (2019) did not seem to give better results.)

Clearly, the $M-$chain estimator is more satisfactory, as it performs better (especially for the normalising constant, left plot) than the empirical variance, although being computed from a single run. On the other hand, the approach of Lee and Whiteley (2018) performs poorly. To be fair, this approach works more reasonably if we increase significantly M (results not shown), but since taking M too large decreases the performance of the algorithm, it seems fair to state that this approach is not useful for waste-free SMC, at least in this example.

5.2. **Latin squares.** Our second example concerns the enumeration of Latin squares of size d; that is, $d \times d$ matrices with entries in $\{0, \ldots, d-1\}$, and such that each integer in that range appears exactly once in each row and in each column; see

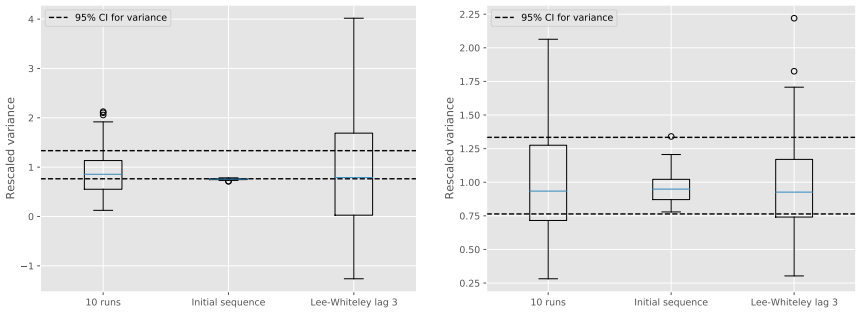

FIGURE 4. Logistic regression: box-plots of variance estimates over 100 runs obtained with waste-free SMC. Left: variance of the log-normalising constant estimate. Right: variance of the mean of all coefficients estimate. The variance estimates are re-scaled so that the empirical variance over the 100 runs equals one; see text for more details.

1	5	0	3	7	8	9	6	2	4
0	4	5	8	6	9	1	7	3	2
2	8	7	0	9	4	5	3	1	6
3	7	4	1	5	2	8	0	6	9
6	0	9	5	1	3	2	8	4	7
8	2	1	9	4	0	6	5	7	3
9	6	3	2	0	5	7	4	8	1
5	1	6	4	3	7	0	2	9	8
4	9	2	7	8	6	3	1	5	0
7	3	8	6	2	1	4	9	0	5

TABLE 1. A Latin square of size 10

Table 5.2 for an example. The number $l(d)$ of Latin squares of size d increases very quickly with d, and is larger than 10^{43} for $d = 11$, the largest value for which it is known; see sequence A002860 of the OEIS database (OEIS Foundation Inc., 2020).

Let $\mathcal{X}$ be the set of permutation squares of size d, that is, $d \times d$ matrices such that each row is a permutation of $\{0, \ldots, d-1\}$, and let $p(d)$ its cardinal, $p(d) = (d!)^d$. We consider the following sequence of tempered distributions: $\pi_t(\mathrm{d}x) = \nu(\mathrm{d}x) \exp\{-\lambda_t V(x)\}/L_t$, where $\nu(\mathrm{d}x)$ stands for the uniform distribution over $\mathcal{X}$, and V is a certain score function such that $V(x) = 0$ if x is a Latin square, $V(x) \geq 1$ otherwise. Specifically, denoting the entries of matrix x by $x[i, j]$, we take

$$V(x) = \sum_{j=1}^{d} \left\{ \sum_{l=1}^{d} \left(\sum_{i=1}^{d} \mathbb{1}(x[i, j] = l) \right)^2 - d \right\}.$$

The quantity $L_t \times p(d)$ will be at distance ε of $l(d)$, the number of Latin squares, as soon as $\lambda_t \geq \log(p(d)/\varepsilon)$. Thus, we select adaptively the successive exponents λ_t (as in the previous example), and stop the algorithm at the first iteration t such that this condition is fulfilled, for $\varepsilon = 10^{-16}$.

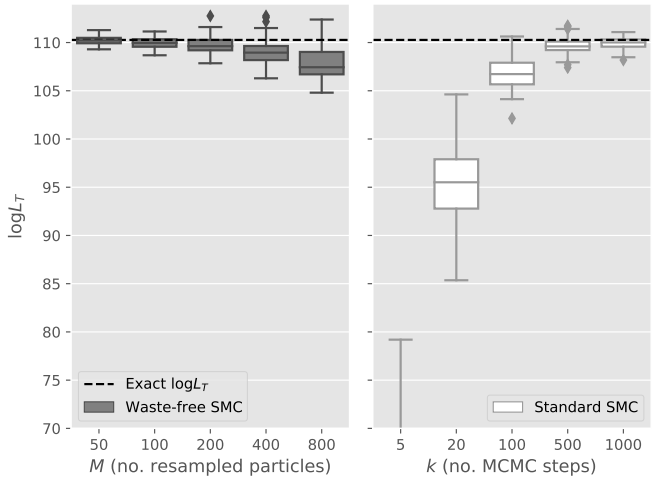

FIGURE 5. Latin squares: box-plots of estimates of $\log L_T$ (log of number of Latin squares) obtained from 100 independent runs of the following algorithms: waste-free SMC ($N = 2 \times 10^5$, different values of M, the number of resampled particles), and standard SMC ($N = 2 \times 10^5/k$, different values for k, the number of MCMC steps).

We set the Markov kernel M_t to be a k-fold Metropolis kernel based on the following proposal distribution: given x, select randomly a row i, two columns j, j', and swap components $x[i, j]$ and $x[i, j']$.

Figure 5 compares the performance of standard SMC and waste-free SMC for evaluating the log of the normalising constant L_T, that is (up to a small error as explained above), the log of the number of Latin squares $l(d)$; we take $d = 11$ since this is the largest value of d for which $l(d)$ is known exactly.

As in the previous example, the compared algorithms are given (roughly) the same CPU budget: $N = 2 \times 10^5/k$ for standard SMC, while $N = 2 \times 10^5$ for waste-free (and $k = 1$, as already discussed). We make the same observations as in the previous example: best performance is obtained from waste-free SMC in the long chain regime ($M \ll N$), and, within this regime, performance does not seem to depend strongly on M.

One distinctive feature of this example is that the mixing of the Metropolis kernel used to move the particles significantly decreases over time; see Figure 6, which plots the acceptance rate of that kernel at each iteration t of a waste-free SMC run.

It is interesting to note that waste-free SMC seems to work well despite this. Unfortunately, it does seem to affect the performance of our $M-$chain variance estimate. The left panel of Figure 7 makes the same comparison as Figure 4 in our first example. This time, however, the $M-$chain estimator seems to be biased downward, by a factor of two. This bias seems to originate from the terms of for the last values of t; these terms are both larger, and more difficult to estimate if P is not large enough.

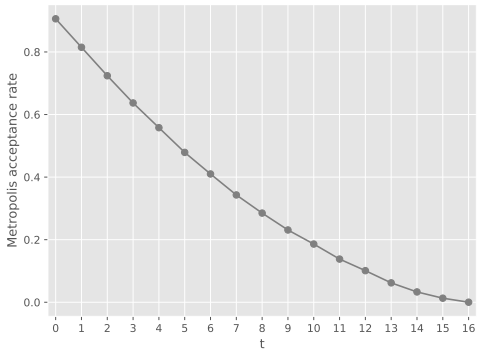

FIGURE 6. Latin squares: acceptance rate of the Metropolis kernel described in the text at each iteration t of a run of waste-free SMC.

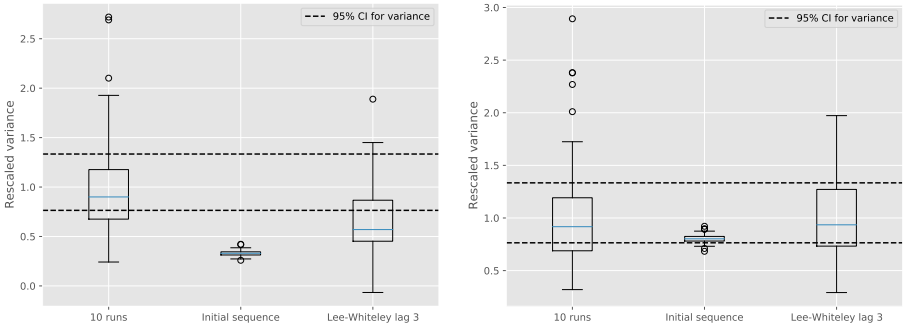

FIGURE 7. Latin squares: same plot as Figure 4, for the estimate of log-normalising constant $\log L_T$. Left: non-adaptive version ($M = 50$, $N = 2 \times 10^5$); Right: adaptive version ($M = 50$, $N_0 = 5$). See text for more details.

These results showcase the interest of adapting P across time, as discussed in Section 4.4. We re-run waste-free SMC for the same problem, with $M = 50$, and $\kappa = 5$; that is, at each iteration t, P_t is adjusted to be close to κ times the auto-correlation time, for function $\bar{G}_t$. The right side of Figure 7 repeats the comparison of the variance estimates, but for the adaptive P algorithm. This time, our $M-$chain estimate seems to perform satisfactorily.

In addition, Figure 8 compares the CPU vs error trade-off for both variants of waste-free SMC. In both cases, we set $M = 50$; "CPU time" on the x-axis is measured by the number of calls to the score function, re-scaled so that the smallest observed value is 1. (Both axis use a $\log 2$-scale.) Each dot corresponds to an average over 100 runs. For the vanilla version, we set $N = 6250$, 2.5×10^4, 10^5, 4×10^5 and 8×10^5. For the adaptive version, we set $\kappa = 2$, 5 and 10. The dotted lines have slope -1. For high CPU time both algorithms show the same level of performance. If N is set to too low a value for vanilla waste-free (e.g. $N = 6250$), then one obtains a very large MSE, because $P = N/M = 125$ is too small relative to the auto-correlation time of the kernels M_t for large t. Note that for the adaptive

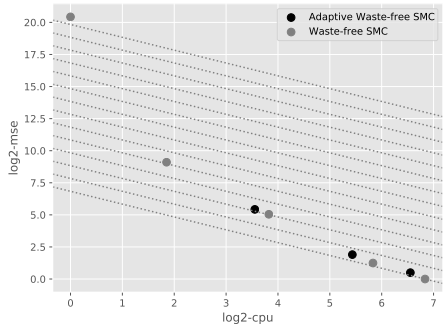

FIGURE 8. Latin squares: MSE (mean square error) vs CPU time (number of calls to score function), averaged over 100 independent runs of vanilla waste-free (grey dots, $N = 6250$, 2.5×10^4, 10^5, 4×10^5, 8×10^5), and adaptive waste-free (black dots, $\kappa = 2$, 5, 10). Both axes use a $\log 2-$scale; parallel dotted lines have slope -1.

version, it does not make sense to take $\kappa \ll 2$, as one cannot properly estimate the auto-correlation time of a chain without running it for a length commensurate with its auto-correlation time. In a sense, the adaptive version of waste-free prevents us from setting P to too low a value, where performance becomes sub-optimal.

By and large, in any problem when there is some evidence that the mixing of kernels M_t may decrease significantly over time, we recommend to use the adaptive P strategy. It is a bit less practical to use, as it gives less control to the user on the running time of the algorithm; but on the other hand it seems to provide more reliable variance estimates in this kind of scenario.

5.3. **Orthant probabilities.** Finally, we consider the problem of evaluating Gaussian orthant probabilities, i.e. $p(a, \Sigma) := \mathbb{P}(Z \geq a)$, where $a \in \mathbb{R}^d$, $Z \sim \mathcal{N}_d(0, \Sigma)$, and Σ is a covariance matrix of size $d \times d$.

Ridgway (2016) developed the following SMC approach for evaluating such probabilities. Let Γ be the lower triangle in the Cholesky decomposition of Σ: $\Sigma = \Gamma\Gamma^T$; $\Gamma = (\gamma_{ij})$ and $\gamma_{ii} > 0$ for all i. The orthant probability $p(a, \Sigma)$ may be rewritten as the joint probability that $X_t \geq f_t(X_{1:t-1})$ for $t = 1, \ldots, d$, where $f_t(x_{1:t-1}) = (a_t - \sum_{s<t} \gamma_{st} x_s)/\gamma_{tt}$, and the X_t's are IID $\mathcal{N}(0, 1)$ variables. (At time 1, $f_1(x_{1:0})$ is simply a_1, i.e. the constraint is $X_1 \geq a_1$.)

The SMC algorithm of Ridgway (2016) applies the following operations to particles $X^n_{1:t}$, from time 1 to time $T = d$. (We change notations slightly and start at time 1, for the sake of readability.) (a) At time t, particles $X^n_{1:t-1}$ are extended by sampling an extra component, X^n_t, from a univariate truncated Gaussian distribution (the distribution of $X_t \sim N(0, 1)$ conditional on $X_t \geq f_t(x_{0:t-1})$); (b) particles $X^n_{1:t}$ are then reweighted according to function $\Phi(-f_t(X^n_{1:t-1}))$, where Φ is the $\mathcal{N}(0, 1)$ cumulative distribution function; and (c) when the ESS (effective sample size) of the weights gets too low, the particles are moved through k iterations of a certain MCMC kernel that leaves invariant π_t, the distribution that corresponds to $X_{1:t} \sim N_t(0, I_t)$ constrained to $X_s \geq f_s(X_{1:s-1})$ for $s = 1, \ldots, t$. Based on numerical experiments, Ridgway (2016) recommended to use for the MCMC kernel

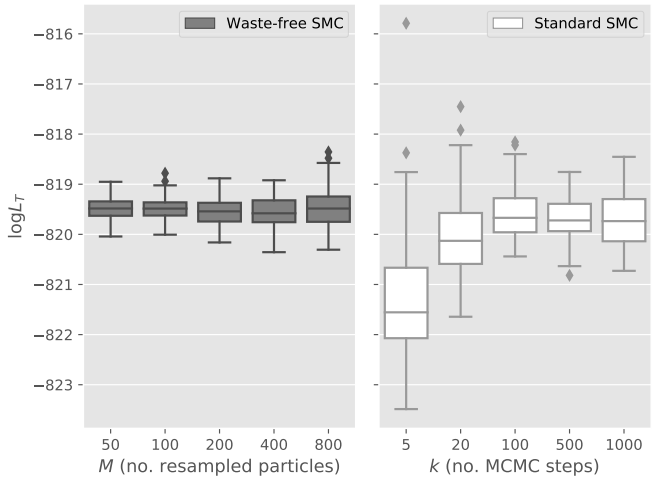

FIGURE 9. Orthants: estimates of the log normalizing constant obtained from waste-free SMC ($N = 2 \times 10^5$) and standard SMC ($N = 2 \times 10^5/k$).

at time t a Gibbs sampler that leaves π_t invariant. (the update of each variable amounts to sampling from a univariate truncated normal distribution.)

This SMC algorithm does not fit in the framework of Algorithm 1; in particular the dimension of the state-space $\mathcal{X} = \mathbb{R}^t$ increases over time. However, we can easily generalise waste-free SMC to this setting: whenever an MCMC rejuvenation step is applied, resample $M \ll N$ particles, apply $P-1$ steps of the chosen MCMC kernels to these M resampled particles, and gather the $N = MP$ so obtained values to form the new particle sample.

To make the problem challenging, we take $d = 150$, $a = (1.5, 1.5, \ldots)$, and Σ a random correlation matrix with eigenvalues uniformly distributed in the simplex $\{x_1 + \cdots + x_d = 150, x_i \geq 0\}$, which we simulated using the algorithm of Davies and Higham (2000). As in Ridgway (2016), before the computation we re-order the variables according to the heuristic of Gibson et al. (1994).

Figures 9 and 10 do the same comparison of standard SMC and waste-free SMC as in the two previous examples: $N = 2 \times 10^5$ for waste-free, $N = 2 \times 10^5/k$ for standard SMC, and M (resp. k) varies over a range of values. Figure 9 plots boxplots of estimates of $\log L_T$ (the log of the orthant probability), while Figure 10 does the same for $\mathbb{Q}_T(\varphi)$, with $\varphi(x_{0:T}) = (\sum_{t=0}^{T} x_t)/T$; i.e. the expectation of φ with respect to the corresponding truncated Gaussian distribution.

We observe again that waste-free SMC outperforms standard SMC, at least whenever $M \ll N$. In addition, the greater robustness of waste-free is quite striking in this example.

Finally, Figure 11 compares $M-$chain estimators of the variance of the orthant probability estimate based on two single-chain estimators: the initial sequence estimator we recommended by default in Section 4.3, and we used in the two previous examples; and a spectral estimator based on the Tukey-Hanning window (see e.g.

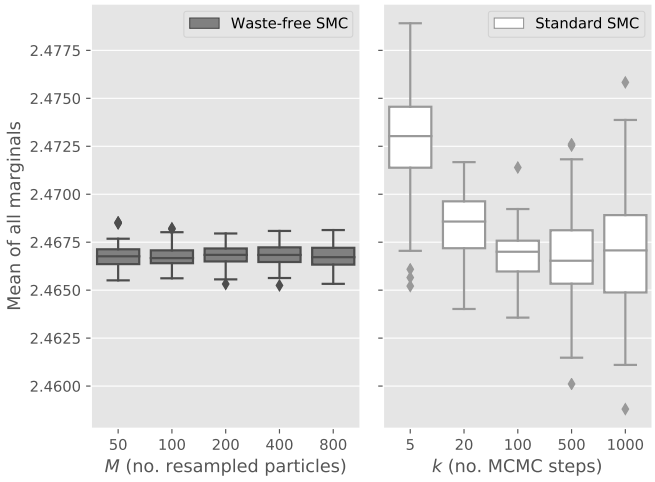

FIGURE 10. Orthants: Same plot as Figure 9 for $\mathbb{Q}_T(\varphi)$, the expectation of function $\varphi(x_{0:T}) = (\sum_{t=0}^{T} x_t)/T$ with respect to truncated Gaussian distribution $\mathcal{N}_{>0}(a, \Sigma)$.

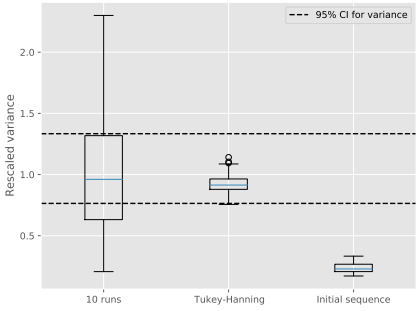

FIGURE 11. Orthants: box-plots of variance estimates over 100 runs for the estimate of the log orthant probability. The variance estimates are re-scaled so that the empirical variance over the 100 runs equals one; see text for more details.

Flegal and Jones, 2010). In this example, the kernels M_t are Gibbs kernels, and are therefore not reversible. This seems to explain the poor performance of the former.

(As in previous plots, Figures 4 and 7, we include for comparison the variance estimator obtained by taking an empirical variance over 10 runs; however we do not include, for the sake of readability, the estimator based on Lee and Whiteley (2018), but note simply it performs poorly in this case too.)

6. Concluding remarks

6.1. **Connection with nested sampling.** In our definition of waste-free SMC, we took $N = MP$, with $P \geq 2$; thus M divides N. We may generalise the algorithm to any pair (M, N), $M < N$: at time t, resample M particles, generate M chains of

length $k := \lfloor N/M \rfloor$ (using kernel M_t, and the resampled particles as the starting points); then select (without replacement) $N - Mk$ chains and extend them to have length $k+1$. The total number of particles is then N.

One interesting special case is $M = N - 1$. In that case, $N-1$ particles are resampled (thus at least one particle is discarded), and, among these $N-1$ resampled particles, only one particle is moved through kernel M_t. In addition, if the target distributions π_t are of the form $\pi_t(\mathrm{d}x) \propto \nu(\mathrm{d}x)\mathbb{1}\{L(x) \geq l_t\}$, where ν is a prior distribution, and L a likelihood function, then one recovers essentially the nested sampling algorithm of Skilling (2006).

This raises the question whether the regime $M = N - 1$ is useful, either for such a sequence of distributions, or more generally. For the former, the numerical experiments of Salomone et al. (2018) seem to indicate than standard SMC, when applied to this type of sequence, may perform as well as nested sampling. This suggests waste-free SMC should also perform at least as well as nested sampling, although we leave that point to further investigation. For the latter, we note that taking $M = N - 1$ is not very convenient, as this means we move only one particle at each iteration, although each iteration costs $\mathcal{O}(N)$. (In nested sampling, the cost of a single iteration may be reduced to $\mathcal{O}(1)$ by using the fact that weights are either 0 or 1.)

6.2. **Further work.** Our convergence results assume that the kernels M_t are uniformly ergodic. However, many practical MCMC kernels are not uniformly ergodic, hence it seems worthwhile to extend these results to, say, geometrically ergodic kernels. Another result we would like to establish is that waste-free SMC dominates standard SMC in terms of asymptotic variance, at least under certain conditions on the mixing of the kernels M_t.

In terms of applications, we wish to explore how waste-free may be implemented in various SMC schemes, in particular in the SMC^2 algorithm of Chopin et al. (2013). This algorithm is an SMC sampler with expensive Markov kernels (as a single step amounts to propagate a large number of particles in a "local" particle filter), hence the benefits brought by waste-free SMC may be particularly valuable in this type of scenario.

The original implementation of the numerical examples may be found at `https://github.com/hai-dang-dau/waste-free-smc`. Waste-free SMC is also now implemented in the `particles` library, see `https://github.com/nchopin/particles`.

Acknowledgements

The first author acknowledges a CREST PhD scholarship via AMX funding. The second author acknowledges partial support from Labex Ecodec (Ecodec/ANR-11-LABX-0047). We are grateful to Chris Drovandi, Pierre Jacob and two anonymous referees for helpful comments on a preliminary version of the paper.

Appendix A. Proofs

A.1. **Proof of Proposition 2.** We may rewrite (6) as:

$$\prod_{s=0}^{t} \left\{ \frac{1}{M} \sum_{m=1}^{M} \left(\frac{1}{P} \sum_{p=1}^{P} G_s(z_s^m[p]) \right) \right\}$$

where $z_s^m[p]$ stands for variable $\tilde{X}_s^{m,p}$ which is defined inside Algorithm 2.

We recognise the normalising constant estimate of a standard SMC sampler, Algorithm 1, when applied to the waste-free Feynman-Kac model defined in Proposition 1. The expectation of this quantity is therefore the normalising constant $L_t^{\mathrm{wf}} = L_t$ (Proposition 1), since such estimates are unbiased (Del Moral, 1996).

A.2. **Proof of Proposition 4.** We start by establishing two technical lemmas regarding a uniformly ergodic Markov chain $(X_p)_{p\geq 0}$, (X_p) for short, on probability space $(\mathcal{X}, \mathbb{X})$; i.e. $\left\|K^k(x, \mathrm{d}x') - \pi(\mathrm{d}x')\right\|_{\mathrm{TV}} \leq C\rho^k$ for certain constants $C > 0$ and $\rho < 1$ and a certain probability distribution π, where $K^k(x, \mathrm{d}x)$ stands for the $k-$fold Markov kernel that defines the distribution of X_{p+k} given X_p. Then $\pi(\mathrm{d}x)$ is its stationary distribution.

Lemma 1. *Assume that (X_p) is stationary, i.e. $X_0 \sim \pi(\mathrm{d}x)$, and therefore $X_p \sim \pi(\mathrm{d}x)$ for all $p \geq 0$. Then there exists a constant $C_1 > 0$ such that:*

$$\mathrm{Var}\left(\varphi(X_0)\right) + 2\sum_{k=1}^{\infty} \left|\mathrm{Cov}\left(\varphi(X_0), \varphi(X_k)\right)\right| \leq C_1 \left\|\varphi\right\|_{\infty}^2$$

for any measurable bounded function $\varphi : \mathcal{X} \to \mathbb{R}$.

Proof. One has

$$\begin{aligned}
\left|\mathrm{Cov}\left(\varphi(X_0), \varphi(X_k)\right)\right| &= \left|\mathbb{E}[\varphi(X_0)\varphi(X_k)] - \mathbb{E}[\varphi X_0]\mathbb{E}[\varphi X_k]\right| \\
&= \left| \int \left\{ \int \varphi(x_k) K^k(x_0, \mathrm{d}x_k) - \int \varphi(x_k)\pi(\mathrm{d}x_k) \right\} \varphi(x_0)\pi(\mathrm{d}x_0) \right| \\
&\leq 2\rho^k C \left\|\varphi\right\|_{\infty}^2
\end{aligned}$$

from which the result follows. □

In the second lemma, the distribution of the initial state X_0 is arbitrary, and therefore the chain is not necessarily stationary.

Lemma 2. *There exists a constant $C_2 > 0$ (which does not depend on the initial distribution of the chain, i.e. the distribution of X_0), such that*

$$\mathrm{Var}\left(\frac{1}{P}\sum_{p=1}^{P} \varphi(X_p)\right) \leq C_2 \frac{\left\|\varphi\right\|_{\infty}^2}{P}$$

for any $P \geq 1$ and any bounded measurable function $\varphi : \mathcal{X} \to \mathbb{R}$.

Proof. The proof relies on a standard coupling argument, see e.g. Chapter 19 of Douc et al. (2018). We introduce an arbitrary integer R, $1 \leq R \leq P$, and a Markov chain $(X_p^\star)$ constructed as follows: (a) $X_0^\star \sim \pi(\mathrm{d}x)$, the stationary distribution of (X_p); (b) variables X_R, $X_R^\star$ are maximally coupled, which implies that:

$$\mathbb{P}(X_R \neq X_R^\star) = \left\| \int \mu(\mathrm{d}x_0) K(x_0, \mathrm{d}x_p) - \pi(\mathrm{d}x_p) \right\|_{\mathrm{TV}} \leq C\rho^R \tag{18}$$

where $\mu(\mathrm{d}x_0)$ denotes the probability distribution of X_0, and the inequality stems from the uniform ergodicity of the chain; (c) if $X_R = X_R^\star$, the two chains remain equal until time P, otherwise they are independent; (d) the distribution of $X_1^\star, \ldots, X_{R-1}^\star$ given $X_0^\star$, $X_R^\star$ is the conditional distribution of these states induced by $K(x, \mathrm{d}x')$, the Markov kernel of (X_p). For more details on maximal coupling of two probability distributions, see e.g. Chap. 19 of Douc et al. (2018).

Using the inequality $\mathrm{Var}(X+Y+Z) \leq 3(\mathrm{Var}(X)+\mathrm{Var}(Y)+\mathrm{Var}(Z))$, we have:

$$\begin{aligned}\mathrm{Var}\left(\frac{1}{P}\sum_{p=1}^{P}\varphi(X_p)\right) &\leq 3\mathrm{Var}\left(\frac{1}{P}\sum_{p=1}^{P}\varphi(X_p^\star)\right) + 3\mathrm{Var}\left(\frac{1}{P}\sum_{p=1}^{R}\left\{\varphi(X_p)-\varphi(X_p^\star)\right\}\right) \\ &\quad + 3\mathrm{Var}\left(\mathbb{1}\{X_R \neq X_R^\star\}\frac{1}{P}\sum_{p=R}^{P}\left\{\varphi(X_p)-\varphi(X_p^\star)\right\}\right) \\ &\leq \frac{3C_1\left\|\varphi\right\|_\infty^2}{P} + \frac{12R^2\left\|\varphi\right\|_\infty^2}{P} + C\rho^R\left\|\varphi\right\|_\infty^2\end{aligned}$$

where we have applied Lemma 1 to the first term, and (18) to the third term. We conclude by taking $R = \lceil\sqrt{P}\rceil$.

□

We now prove Proposition 4 by induction. Clearly, (11) holds at time 0. The implication (11) ⇒ (12) at time t follows the same lines as for a standard SMC sampler, see e.g. Section 11.2.2 in Chopin and Papaspiliopoulos (2020). Now assume that (12) holds at time $t-1 \geq 0$, and let $\bar{\varphi} = \varphi - \mathbb{Q}_{t-1}(\varphi)$, $\mathcal{F}_{t-1} = \sigma(X_{t-1}^{1:N})$ (the σ-field generated by variables X_{t-1}^n, $n = 1, \ldots, N$). Then

$$\begin{aligned}&\mathbb{E}\left[\left(\frac{1}{N}\sum_{n=1}^{N}\varphi(X_t^n) - \mathbb{Q}_{t-1}(\varphi)\right)^2\middle|\mathcal{F}_{t-1}\right] \\ &\quad = \mathbb{E}\left[\left(\frac{1}{M}\sum_{m=1}^{M}\frac{1}{P}\sum_{p=1}^{P}\bar{\varphi}(\tilde{X}_t^{m,p})\right)^2\middle|\mathcal{F}_{t-1}\right] \\ &\quad = \left(\mathbb{E}\left[\frac{1}{P}\sum_{p=1}^{P}\bar{\varphi}(\tilde{X}_t^{1,p})\middle|\mathcal{F}_{t-1}\right]\right)^2 + \frac{1}{M}\mathrm{Var}\left(\frac{1}{P}\sum_{p=1}^{P}\bar{\varphi}(X_t^{1,p})\middle|\mathcal{F}_{t-1}\right)\end{aligned}$$

since the blocks of variables $X_t^{m,1:P}$ are IID (independent and identically distributed) conditional on $\mathcal{F}_{t-1}$.

The expectation of the first term may be bounded by $c'_{t-1}\left\|\varphi\right\|_\infty^2/N$ by applying (12) to function $P^{-1}\sum_{p=0}^{P-1}M_t^p\varphi$. The second term may be bounded by $C_2\left\|\varphi\right\|_\infty^2/N$ using Lemma 2.

A.3. **Proof of Theorem 1.** We start by proving a few basic lemmas. The first one concerns product measures. We use symbol $\bigotimes$ throughout to represent the product of two probability measures.

Lemma 3 (Total variation distance for product measure)**.** *Let $\mu_{1:N}$ and $\nu_{1:N}$ be $2N$ probability measures on $(\mathcal{X}, \mathscr{X})$. Then the following inequality holds:*

$$\left\|\bigotimes_{n=1}^{N}\mu_n - \bigotimes_{n=1}^{N}\nu_n\right\|_{\mathrm{TV}} \leq \sum_{n=1}^{N}\left\|\mu_n - \nu_n\right\|_{\mathrm{TV}}.$$

Proof. Take $N = 2$. Then

$$\left\|\mu_1\otimes\mu_2 - \nu_1\otimes\nu_2\right\|_{\mathrm{TV}} \leq \left\|\mu_1\otimes\mu_2 - \mu_1\otimes\nu_2\right\|_{\mathrm{TV}} + \left\|\mu_1\otimes\nu_2 - \nu_1\otimes\nu_2\right\|_{\mathrm{TV}}$$

and we may bound the first term as follows:

$$\begin{aligned}
&\|\mu_1 \otimes \mu_2 - \mu_1 \otimes \nu_2\|_{\mathrm{TV}} \\
&\quad = \sup_{f:\mathcal{X}^2 \to [0,1]} \left| \int \left(\int f(x,y) \mu_1(\mathrm{d}x) \right) \mu_2(\mathrm{d}y) - \int \left(\int f(x,y) \mu_1(\mathrm{d}x) \right) \nu_2(\mathrm{d}y) \right| \\
&\quad \leq \sup_{g:\mathcal{X} \to [0,1]} \left| \int g(y) \mu_2(\mathrm{d}y) - \int g(y) \nu_2(\mathrm{d}y) \right| = \|\mu_2 - \nu_2\|_{\mathrm{TV}} .
\end{aligned}$$

The result follows by bounding the second term similarly. For $N \geq 3$, proceed recursively. □

The two next lemmas concern the behaviour of $M \geq 1$ independent, stationary, Markov chains, $(Y_p^m)_{p \geq 0}$ on $(\mathcal{X}, \mathbb{X})$, $m = 1, \ldots, M$ with uniformly ergodic Markov kernel K, and invariant distribution π: $\|\delta_x K^p - \pi\|_{\mathrm{TV}} \leq C\rho^k$ for constants $C \geq 0$ and $\rho \in [0, 1)$.

Lemma 4. *The product kernel*

$$K^{\otimes M}(x_{1:M}, \mathrm{d}x'_{1:M}) = \prod_{m=1}^{M} K(x_m, \mathrm{d}x'_m)$$

is uniformly ergodic, with stationary distribution $\pi^{\otimes M}$.

Proof. This is a direct consequence of Lemma 3:

$$\begin{aligned}
\left\| \delta_{x_{1:M}} \left(K^{\otimes M} \right)^p - \pi^{\otimes M} \right\|_{\mathrm{TV}} &\leq \sum_{m=1}^{M} \|\delta_{x_m} K^p - \pi\|_{\mathrm{TV}} \\
&\leq CM\rho^p.
\end{aligned}$$

□

Lemma 5. *For* $\varphi : \mathcal{X} \to \mathbb{R}$ *measurable and bounded, one has:*

$$\sqrt{MP} \left(\frac{\sum_{m=1}^{M} \sum_{p=1}^{P} \varphi(Y_p^m)}{MP} - \pi(\varphi) \right) \Rightarrow \mathcal{N}\left(0, v_\infty(K, \varphi)\right)$$

as $P \to +\infty$, *whether* $M \geq 1$ *is fixed, or* M *grows with* P; *i.e.* $M = M(P) \to +\infty$ *as* $P \to +\infty$.

Proof. For $M = 1$, this is simply the classical central limit theorem for uniformly ergodic Markov chains, see e.g. Theorem 23 in Roberts and Rosenthal (2004) and references therein. For $M \geq 2$ fixed, we may apply the same theorem to the Markov chain $(Y_p^{1:M})_p$ in $(\mathcal{X}^M, \mathbb{X}^M)$, which is also uniformly ergodic (Lemma 4) and to test function $\varphi_M(y^{1:M}) = M^{-1} \sum_{m=1}^{M} \varphi(y^m)$.

Assume now $M = M(P)$ grows with P. Let $\bar{\varphi} = \varphi - \pi(\varphi)$ and let S_P denote a variable with the same distribution as $S_P^m := P^{-1/2} \sum_{p=1}^{P} \bar{\varphi}(Y_p^m)$ for $m = 1, \ldots, M$. (These M variables are IID.) By the formula (19) of Roberts and Rosenthal (2004), we have $\mathbb{E}[S_P^2] \to v_\infty(K, \varphi)$. Therefore, fixing $u \in \mathbb{R}$, we wish to prove that $\Delta_P \to 0$, where

$$\Delta_P := \left| \left(\mathbb{E} e^{iuS_P/\sqrt{M}} \right)^M - \left(1 - \frac{u^2}{2M} \mathbb{E}(S_P^2) \right)^M \right| .$$

Let $M_0 \geq 1$ be fixed such that $u^2\mathbb{E}[S_P^2]/2M_0 < 1$ for all $P > M_0$. Since $\left|a^M - b^M\right| \leq M\left|a-b\right|$ for $|a|, |b| \leq 1$ and $\left|e^{ix} - 1 - ix + x^2/2\right| \leq \min(x^2, |x^3|/6)$ for $x \in \mathbb{R}$, we have, for any $M \geq M_0$:

$$\Delta_P \leq \mathbb{E}\min\left(u^2 S_P^2, \frac{\left|u^3 S_P^3\right|}{6\sqrt{M}}\right) \leq \mathbb{E} f_{M_0}(S_P) \tag{19}$$

where $f_m(x) := \min\left(u^2x^2, \left|u^3x^3/6\sqrt{m}\right|\right) = f_m^1(x) + f_m^2(x)$, $f_m^1(x) := u^2x^2$ and $f_m^2(x) := \mathbb{1}_{|x| \leq 6\sqrt{m}/|u|}\left(\left|u^3x^3\right|/6\sqrt{m} - u^2x^2\right)$. Then, if G is a Gaussian variable with variance $v_\infty(K, \varphi)$, we have $\mathbb{E}f_{M_0}^1(S_P) \to \mathbb{E}f_{M_0}^1(G)$ as $P \to +\infty$. Moreover, $\mathbb{E}f_{M_0}^2(S_P) \to \mathbb{E}f_{M_0}^2(G)$ by Theorem 23 of Roberts and Rosenthal (2004) and the fact that $f_{M_0}^2$ is a bounded function and is only discontinuous on a set of measure zero with respect to a Gaussian distribution. Thus (19) implies that $\limsup_{P\to\infty} \Delta_P \leq \mathbb{E}f_{M_0}(G)$. But $\mathbb{E}f_{M_0}(G) \to 0$ as $M_0 \to \infty$ by the dominated convergence theorem, hence $\Delta_P \to 0$ and the lemma is proved. □

We now prove Theorem 1. We proceed by induction: (13) at time 0 is simply the standard central limit theorem for IID variables. The implication (13) ⇒ (14) at time t may be established exactly as in other proofs for central limit theorems for SMC algorithms; see e.g. Section 11.3 of Chopin and Papaspiliopoulos (2020).

We now assume that (14) holds at time $t-1 \geq 0$, and we wish to show that (13) holds at time t, or, equivalently, that:

$$\frac{1}{\sqrt{P}}\sum_{p=1}^{P} \varphi_M(Z_p) \Rightarrow \mathcal{N}\left(0, v_\infty(M_t, \varphi)\right) \tag{20}$$

where (dropping the dependence on t as it is fixed) $Z_p := (\tilde{X}_t^{1,p}, \ldots, \tilde{X}_t^{M,p})$ is a Markov chain on $\mathcal{X}^M$, which is uniformly ergodic (Lemma 4), and $\varphi_M(z) = M^{-1/2}\sum_{m=1}^{M} \bar{\varphi}(z[m])$.

We apply the coupling construction we used in the proof of Lemma 2 to this Markov chain: we introduce a stationary Markov chain, $(Z_p^\star)$, with the same Markov kernel as (Z_p), i.e. $M_t^{\otimes M}$, which is coupled to (Z_p) at time R, $1 \leq R \leq P$, with maximum coupling probability:

$$\mathbb{P}(Z_R \neq Z_R^\star) = \left\|\mathcal{L}(Z_1)(M_t^{\otimes M})^R - \pi_{t-1}^{\otimes M}\right\|_{\mathrm{TV}} \leq MC\rho^R \tag{21}$$

If the two chains are successfully coupled at time R, they remain equal at times $R+1, \ldots, P$.

We decompose the left-hand side of (20) as:

$$\begin{aligned}\frac{1}{\sqrt{P}}\sum_{p=1}^{P} \varphi_M(Z_p) &= \frac{1}{\sqrt{P}}\sum_{p=1}^{P} \varphi_M(Z_p^\star) + \frac{1}{\sqrt{P}}\sum_{p=1}^{R} \varphi_M(Z_p) \\ &\quad - \frac{1}{\sqrt{P}}\sum_{p=1}^{R} \varphi_M(Z_p^\star) + \frac{1}{\sqrt{P}}\mathbb{1}\{Z_R \neq Z_R^\star\}\sum_{p=R+1}^{P}\left(\varphi_M(Z_p) - \varphi_M(Z_p^\star)\right).\end{aligned} \tag{22}$$

The first terms converges to $\mathcal{N}\left(0, v_\infty(M_t, \varphi)\right)$, see Lemma 5. What remains to prove is that the three other terms converge to zero in probability.

The fourth term is non-zero with probability (21), and tends to zero as soon as $R \to +\infty$; e.g. $R = \mathcal{O}(P^\beta)$, $\beta \in (0,1)$. Using the inequality $\mathrm{Var}(Y_1 + \ldots + Y_R) \leq$

$R\left(\mathrm{Var}(Y_1) + \ldots + \mathrm{Var}(Y_R)\right)$, we may bound the the L^2 norm of the third term as follows:

$$\mathrm{Var}\left(\frac{1}{\sqrt{P}}\sum_{p=1}^{R}\varphi_M(Z_p^\star)\right) \leq \frac{R^2}{P}\mathrm{Var}_\pi(\bar{\varphi}) \leq 2\frac{R^2}{P}\left\|\varphi\right\|_\infty^2$$

which tends to zero as soon as $R^2 \ll P$, e.g. $R = \mathcal{O}(P^\beta)$, $\beta \in (0, 1/2)$.

The second term equals:

$$R\sqrt{\frac{M}{P}}\left(\frac{1}{M}\sum_{m=1}^{M}\frac{1}{R}\sum_{p=1}^{R}\bar{\varphi}(\tilde{X}_t^{m,p})\right) \tag{23}$$

and, since the M chains $\tilde{X}_t^{m,1:P}$ are independent, for $m = 1, \ldots, M$, conditional on $\mathcal{F}_{t-1} = \sigma(X_{t-1}^{1:N})$, we have:

$$\begin{aligned}
&\mathbb{E}\left[\left(\frac{1}{M}\sum_{m=1}^{M}\frac{1}{R}\sum_{p=1}^{R}\bar{\varphi}(X_t^{m,p})\right)^2\middle|\mathcal{F}_{t-1}\right] \\
&\quad= \left(\mathbb{E}\left[\frac{1}{R}\sum_{p=1}^{R}\bar{\varphi}(\tilde{X}_t^{1,p})\middle|\mathcal{F}_{t-1}\right]\right)^2 + \frac{1}{M}\mathrm{Var}\left(\frac{1}{R}\sum_{p=1}^{R}\varphi(\tilde{X}_t^{m,p})\middle|\mathcal{F}_{t-1}\right) \\
&\quad\leq \left\{\mathbb{Q}_{t-1}^N\left(\frac{1}{R}\sum_{p=1}^{R}M_t^{p-1}\bar{\varphi}\right)\right\}^2 + \frac{2}{M}\left\|\varphi\right\|_\infty^2
\end{aligned}$$

where $\mathbb{Q}_{t-1}^N(\varphi) = \sum_{n=1}^N W_{t-1}^n\varphi(X_{t-1}^n)$.

The expectation of the first term can be bounded by a constant times $\left\|\varphi\right\|_\infty^2/N$ by Proposition 4, thus the $\mathbb{L}^2$ norm of (23) is $\mathcal{O}(R/\sqrt{MP})$, which tends to zero as soon $R^2 \ll MP$. Taking $R = \mathcal{O}(P^\beta)$, $\beta \in (0, 1/2)$ therefore ensures that all the terms in (22), minus the first, goes to zero.

A.4. **Proof of Theorem 2.** Before proving Theorem 2, we need to define some new notations to work comfortably with the convergence of conditional distributions. We start with a simple example.

Most Markov chains used in MCMC algorithms admit a central limit theorem regardless of its starting point, i.e., one has, for a Markov chain (Y_p) with invariant distribution π, and and a fixed point y_1,

$$\sqrt{P}\left(\frac{1}{P}\sum_{p=1}^{P}\varphi(Y_p) - \pi(\varphi)\right)\middle|\, Y_1 = y_1 \Rightarrow \mathcal{N}(0, \sigma^2)$$

for some σ^2, as $P \to \infty$. For uniformly ergodic Markov chains, stronger results hold. For example, for any deterministic sequence $(y_p)_{p=1}^\infty$:

$$\sqrt{P}\left(\frac{1}{P}\sum_{p=1}^{P}\varphi(Y_p) - \pi(\varphi)\right)\middle|\, Y_1 = y_P \Rightarrow \mathcal{N}(0, \sigma^2).$$

If instead of having a single Markov chain, we have $M = M(P)$ chains (Y_p^m), $m = 1, \ldots, M$, running in parallel, then, provided that the number of chains M is negligible compared to their length P, it is possible to average the result of M

chains to get a better one. Specifically, it can be shown that for any deterministic sequence (y_P^m) indexed by m and p,

$$\sqrt{MP}\left(\frac{1}{M}\sum_{m=1}^{M}\frac{1}{P}\sum_{p=1}^{P}\varphi(Y_p^m)-\pi(\varphi)\right)\Bigg|\, Y_1^{1:M}=y_P^{1:M}\Rightarrow\mathcal{N}(0,\sigma^2) \tag{24}$$

as $P\to\infty$. It is natural to reformulate (24) using the following simplified notation:

$$\sqrt{MP}\left(\frac{1}{M}\sum_{m=1}^{M}\frac{1}{P}\sum_{p=1}^{P}\varphi(Y_p^m)-\pi(\varphi)\right)\Bigg|\, Y_1^{1:M}\Rightarrow\mathcal{N}(0,\sigma^2) \tag{25}$$

while keeping in mind that $M=M(P)$ and in particular the σ-algebra generated by $Y_1^{1:M}$ does not stay the same when $P\to\infty$. While the interpretation (24) of the notation of (25) is intuitive, a more rigorous formalization will make manipulations easier. That is the point of the following definition and lemma, which are simple specific cases of more general results in Sweeting (1989). The difference with Sweeting (1989) is that we prefer, if possible, to work with probability conditioned on an event, which is simpler than probability conditioned on a filtration or a variable.

Definition 1 (Convergence of conditional distributions)**.** *Let $(X_n)_{n=1}^{\infty}$ be a sequence of random variables and let $(\mathcal{F}_n)_{n=1}^{\infty}$ be a sequence of σ-algebras (which are not necessarily nested as in a filtration). We say that the sequence $X_n|\mathcal{F}_n$ of conditional distributions converge as $n\to\infty$ to distribution π,*

$$X_n|\mathcal{F}_n\Rightarrow\pi,$$

if for any sequence $(B_n)_{n=1}^{\infty}$ of events such that $B_n\in\mathcal{F}_n$ and $\mathbb{P}(B_n)>0$, we have $X_n|B_n\Rightarrow\pi$.

Lemma 6. *Under the notations of definition 1, we have, for any continuous bounded function $\varphi:\mathcal{X}\to\mathbb{R}$,*

$$\mathbb{E}\left[\varphi(X_n)|\,\mathcal{F}_n\right]\overset{a.s.}{\to}\pi(\varphi).$$

Remark. This result is in fact used in Sweeting (1989) as the *definition* of convergence in conditional probability. As said above, we prefer Definition 1 as we find it more convenient to work with probability conditioned on events than probability conditioned on sigma-algebras.

Proof. For some $\epsilon>0$, define the events B_n as

$$B_n:=\left\{\mathbb{E}\left[\varphi(X_n)|\,\mathcal{F}_n\right]-\pi(\varphi)\geq\epsilon\right\}.$$

If $\mathbb{P}(B_n)>0$, one can write, since $B_n\in\mathcal{F}_n$:

$$\mathbb{E}\left[\varphi(X_n)|\,B_n\right]=\mathbb{E}\left[\mathbb{E}\left[\varphi(X_n)|\,\mathcal{F}_n\right]|\,B_n\right]\geq\pi(\varphi)+\epsilon. \tag{26}$$

If there exists an infinity of n such that $\mathbb{P}(B_n)>0$, we have by Definition 1 that $X_n|B_n\Rightarrow\pi$, which leads to a contradiction if we let $n\to\infty$ in both sides of (26). Thus, there exists some n_1 such that $\mathbb{P}(B_n)=0,\forall n\geq n_1$. Similarly, one may show that there exists n_2 such that $\mathbb{P}(C_n)=0$, $\forall n\geq n_2$, where

$$C_n=\left\{\mathbb{E}[\varphi(X_n)|\mathcal{F}_n]-\pi(\varphi)<-\epsilon\right\}.$$

Now, note that the desired almost-sure convergence is equivalent to the fact that the random variable

$$R:=\limsup_{n\to\infty}\left|\mathbb{E}[\varphi(X_n)|\mathcal{F}_n]-\pi(\varphi)\right|$$

equals 0 almost surely. Indeed, for any $\epsilon > 0$, the event $\{R \geq \epsilon\}$ is contained in $\left(\bigcup_{n=n_1}^{\infty} B_n\right) \cup \left(\bigcup_{n=n_2}^{\infty} C_n\right)$, which has probability zero. □

Lemma 7. *Let $(X_n)_{n=1}^{\infty}$ and $(Y_n)_{n=1}^{\infty}$ be two sequences of random variables such that $X_n \Rightarrow \mathbb{P}_X$ and $Y_n|X_n \Rightarrow \mathbb{P}_Y$ where the latter is understood in terms of Definition 1. Then $(X_n, Y_n) \Rightarrow \mathbb{P}_X \otimes \mathbb{P}_Y$.*

Proof. Let Y be a $\mathbb{P}_Y$-distributed random variable. We have that

$$\left|\mathbb{E}[e^{iuX_n+ivY_n}] - \mathbb{E}[e^{iuX_n}]\mathbb{E}[e^{ivY}]\right| = \left|\mathbb{E}\left[e^{iuX_n}\left(\mathbb{E}[e^{ivY_n}|X_n] - \mathbb{E}[e^{ivY}]\right)\right]\right|$$

tends to 0 by dominated convergence theorem and the fact that $\mathbb{E}[e^{ivY_n}|X_n] - \mathbb{E}[e^{ivY}]$ converges almost surely to 0 (Lemma 6). □

We are now able to prove Theorem 2.

Proof. The idea of the proof is to show something very similar to (24). Indeed, we shall show the following conditional version of (20):

$$\frac{1}{\sqrt{P}} \sum_{p=1}^{P} \varphi_M(Z_p) \Bigg| \mathcal{F}_{t-1} \Rightarrow \mathcal{N}\left(0, v_\infty(M_t, \varphi)\right) \tag{27}$$

which by Definition 1 means

$$\frac{1}{\sqrt{P}} \sum_{p=1}^{P} \varphi_M(Z_p) \Bigg| B_{t-1} \Rightarrow \left(0, v_\infty(M_t, \varphi)\right) \tag{28}$$

for any sequence B_{t-1} (implicitly indexed by P) of events such that $B_{t-1}^P \in \mathcal{F}_{t-1}^P$. The left hand side of (28) can be decomposed into four terms as in (22), where now $(Z_p^\star)$ is a stationary Markov chain constructed via a maximal coupling of $\mathbb{Q}_{t-1}^{\otimes M}$ and the *conditional* (instead of the full) distribution of Z_R. The first, third and the fourth terms of (22) can be treated exactly as before. The second term tends to 0 in probability when $R = N^\epsilon$ for small enough ϵ, because $M = O(N^\alpha)$ for $\alpha < 1/2$. Thus (27) holds. Applying it for $\varphi = G_t$ and using the delta method give the convergence of $\sqrt{N}(\log \hat{\ell}_t - \log \ell_t)|\mathcal{F}_{t-1}$ with asymptotic variance $v_\infty(M_t, \bar{G}_t)$. Furthermore, note that by Definition 1, the convergence of $X_N|\mathcal{F}_N$ implies the convergence of $X_N|\mathcal{F}'_N$ if $\mathcal{F}'_n \subset \mathcal{F}_n$ for all n. Hence

$$\sqrt{N}\left(\log \ell_t^N - \log \ell_t\right) \Big| \sqrt{N}\left(\log L_{t-1}^N - \log L_{t-1}\right) \Rightarrow \mathcal{N}\left(0, v_\infty(M_t, \bar{G}_t)\right). \tag{29}$$

We can now proceed by induction. Suppose that the assertion is verified up to time $t-1$, that is,

$$\sqrt{N}\left(\log L_{t-1}^N - \log L_{t-1}\right) \Rightarrow \mathcal{N}\left(0, \sum_{s=0}^{t-1} v_\infty(M_s, \bar{G}_s)\right). \tag{30}$$

Then, (29), (30) and Lemma 7 prove the assertion at time t. □

A.5. **Proof of Proposition 5.** We first calculate $\mathcal{V}_t^{\mathrm{std},k}(\varphi)$ by using e.g. formula (11.14) in Chopin and Papaspiliopoulos (2020):

$$\mathcal{V}_t^{\mathrm{std},k}(\varphi) = \sum_{s=0}^{t} \mathbb{Q}_{s-1}\left[\left\{\bar{G}_s R_{s+1:t} C_t \varphi\right\}^2\right] \tag{31}$$

where $\bar{G}_t = G_t/r$, $R_t(\varphi) := M_t \bar{G}_t \varphi$, $R_{s+1:t} := R_{s+1} \circ \ldots \circ R_t$, and $C_t(\varphi) := \varphi - \mathbb{Q}_t(\varphi)$. Note that M_t, $\bar{G}_t$ and C_t are all linear functionals. From the definition of M_t, we have

$$M_t(x_{t-1}, B) = (1 - \tilde{p}_k)\mathbb{1}_B(x_{t-1}) + \tilde{p}_k \pi_{t-1}(B),$$

with $\tilde{p}_k = 1 - (1-p)^k$, which leads to

$$\bar{G}_s R_{s+1:t} C_t \varphi = \bar{G}_s \left[\tilde{p}_k \mathbb{Q}_s^\star \bar{G}_{s+1} + (1 - \tilde{p}_k)\bar{G}_{s+1}\right] \ldots \left[\tilde{p}_k \mathbb{Q}_{t-1}^\star \bar{G}_t + (1 - \tilde{p}_k)\bar{G}_t\right] C_t \varphi.$$

It is easy to fully extend the above expression if one remarks that for any $l < t$, $\tilde{p}_k \mathbb{Q}_l^\star \bar{G}_{l+1:t} C_t \varphi = 0$. Therefore only terms without any $\tilde{p}_k \mathbb{Q}_l^\star \bar{G}_{l+1}$ actually contribute to the result. Thus

$$\bar{G}_s R_{s+1:t} C_t \varphi = (1 - \tilde{p}_k)^{t-s} \bar{G}_{s:t} C_t \varphi.$$

We can now plug this into (31) and get

$$\begin{aligned}
\mathcal{V}_t^{\mathrm{std},k}(\varphi) &= \sum_{s=0}^{t} (1 - \tilde{p}_k)^{2(t-s)} \mathbb{Q}_{t-1}\left[\bar{G}_{s:t}^2 (C_t\varphi)^2\right] \\
&= \sum_{s=0}^{t} (1 - \tilde{p}_k)^{2(t-s)} \mathbb{Q}_{s-1}\left[\bar{G}_{s:t} \frac{1}{r^{t-s+1}} (C_t\varphi)^2\right] \\
&= \sum_{s=0}^{t} \frac{1}{r} \left[\frac{(1-\tilde{p}_k)^2}{r}\right]^{t-s} \mathbb{Q}_t\left[(C_t\varphi)^2\right] \\
&= \frac{1}{r} \sum_{s=0}^{t} \left(\frac{(1-p)^{2k}}{r}\right)^s \mathrm{Var}_{\mathbb{Q}_t}(\varphi).
\end{aligned}$$

We thus see that the variance of the standard SMC sampler evolves proportionally to the sum of a geometric series and its stability depends on whether the base of the series is smaller than or greater than 1. This proves the second point of the proposition. For the third point, note that

$$\begin{aligned}
\tilde{\mathcal{V}}_t(\varphi) &= \mathbb{Q}_{t-1}\left[(C_{t-1}\varphi)^2 + 2\sum_{s=1}^{\infty} (C_{t-1}\varphi)(K_t^s C_{t-1}\varphi)\right] \\
&= \mathbb{Q}_{t-1}\left[(C_{t-1}\varphi)^2 + 2\sum_{s=1}^{\infty} (C_{t-1}\varphi)^2 (1-p)^s\right] \\
&= \left(\frac{2}{p} - 1\right) \mathbb{Q}_{t-1}\left[(C_{t-1}\varphi)^2\right],
\end{aligned}$$

from which

$$\begin{aligned}
\mathcal{V}_t^{\mathrm{wf}}(\varphi) &= \tilde{\mathcal{V}}_t(\bar{G}_t C_t \varphi) \\
&= \left(\frac{2}{p} - 1\right) \mathbb{Q}_{t-1}\left[(C_{t-1}\bar{G}_t C_t\varphi)^2\right] \\
&= \left(\frac{2}{p} - 1\right) \mathbb{Q}_{t-1}\left[(\bar{G}_t C_t\varphi)^2\right] \\
&= \frac{1}{r}\left(\frac{2}{p} - 1\right) \mathrm{Var}_{\mathbb{Q}_t}(\varphi).
\end{aligned}$$

Finally, to prove the last point of the proposition, we write

$$\lim_{t\to\infty} \frac{\mathrm{IF}_t^{\mathrm{wf}}}{k\,\mathrm{IF}_t^{\mathrm{std},k}} = \frac{r^{-1}(\frac{2}{p}-1)}{r^{-1}k\left(1-\frac{(1-p)^{2k}}{r}\right)^{-1}} \leq \left(\frac{2}{p}-1\right)\frac{1-(1-p)^{2k}}{k} \tag{32}$$

as the second to last expression is non-decreasing in r. Next, consider the function $f(p) := (1-p)^{2k} + 2kp$ of which the derivative $f'(p) = 2k(1-(1-p)^{2k-1})$ is non-negative thanks to the fact that $k \geq 1$. We have $f(p) \geq f(0) = 1$, which, when plugged into Equation (32), gives

$$\lim_{t\to\infty} \frac{\mathrm{IF}_t^{\mathrm{wf}}}{k\,\mathrm{IF}_t^{\mathrm{std},k}} \leq \left(\frac{2}{p}-1\right)\frac{2kp}{k} \leq 4.$$

References

Amzal, B., Bois, F. Y., Parent, E. and Robert, C. P. (2006) Bayesian-optimal design via interacting particle systems. *J. Amer. Statist. Assoc.*, **101**, 773–785. URL: https://doi.org/10.1198/016214505000001159.

Berzuini, C., Best, N. G., Gilks, W. R. and Larizza, C. (1997) Dynamic conditional independence models and Markov chain Monte Carlo methods. *J. Amer. Statist. Assoc.*, **92**, 1403–1412. URL: https://doi.org/10.2307/2965410.

Beskos, A., Crisan, D. and Jasra, A. (2014) On the stability of sequential Monte Carlo methods in high dimensions. *Ann. Appl. Probab.*, **24**, 1396–1445. URL: https://doi.org/10.1214/13-AAP951.

Beskos, A., Jasra, A., Law, K., Tempone, R. and Zhou, Y. (2017) Multilevel sequential Monte Carlo samplers. *Stochastic Process. Appl.*, **127**, 1417–1440. URL: https://doi.org/10.1016/j.spa.2016.08.004.

Bornn, L., Doucet, A. and Gottardo, R. (2010) An efficient computational approach for prior sensitivity analysis and cross-validation. *Canad. J. Statist.*, **38**, 47–64. URL: http://dx.doi.org/10.1002/cjs.10045.

Buchholz, A., Chopin, N. and Jacob, P. E. (2020) Adaptive tuning of Hamiltonian Monte Carlo within sequential Monte Carlo. *Bayesian Anal.* Advance publication.

Cérou, F., Del Moral, P., Furon, T. and Guyader, A. (2012) Sequential Monte Carlo for rare event estimation. *Stat. Comput.*, **22**, 795–808. URL: https://doi.org/10.1007/s11222-011-9231-6.

Chan, H. P. and Lai, T. L. (2013) A general theory of particle filters in hidden Markov models and some applications. *Ann. Statist.*, **41**, 2877–2904. URL: https://doi.org/10.1214/13-AOS1172.

Chopin, N. (2002) A sequential particle filter method for static models. *Biometrika*, **89**, 539–551.

Chopin, N., Jacob, P. E. and Papaspiliopoulos, O. (2013) SMC^2: an efficient algorithm for sequential analysis of state space models. *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, **75**, 397–426.

Chopin, N. and Papaspiliopoulos, O. (2020) *An Introduction to Sequential Monte Carlo*. Springer Series in Statistics. Springer.

Chopin, N. and Ridgway, J. (2017) Leave Pima Indians alone: binary regression as a benchmark for Bayesian computation. *Statist. Sci.*, **32**, 64–87. URL: https://doi.org/10.1214/16-STS581.

Davies, P. I. and Higham, N. J. (2000) Numerically stable generation of correlation matrices and their factors. *BIT Numerical Mathematics*, **40**, 640–651.

Del Moral, P. (1996) Non-linear filtering: interacting particle resolution. *Markov processes and related fields*, **2**, 555–581.

— (2004) *Feynman-Kac formulae. Genealogical and interacting particle systems with applications*. Probability and its Applications. Springer Verlag, New York.

Del Moral, P., Doucet, A. and Jasra, A. (2006) Sequential Monte Carlo samplers. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, **68**, 411–436.

Douc, R., Moulines, E., Priouret, P. and Soulier, P. (2018) *Markov chains*. Springer Series in Operations Research and Financial Engineering. Springer, Cham. URL: `https://doi.org/10.1007/978-3-319-97704-1`.

Drovandi, C. C. and Pettitt, A. N. (2011) Estimation of parameters for macroparasite population evolution using approximate bayesian computation. *Biometrics*, **67**, 225–233.

Du, Q. and Guyader, A. (2019) Variance estimation in adaptive sequential Monte Carlo. *arXiv e-print 1909.13602*.

Everitt, R. G., Johansen, A. M., Rowing, E. and Evdemon-Hogan, M. (2017) Bayesian model comparison with un-normalised likelihoods. *Stat. Comput.*, **27**, 403–422. URL: `https://doi.org/10.1007/s11222-016-9629-2`.

Finke, A., Doucet, A. and Johansen, A. M. (2020) Limit theorems for sequential MCMC methods. *Adv. in Appl. Probab.*, **52**, 377–403. URL: `https://doi.org/10.1017/apr.2020.9`.

Flegal, J. M. and Jones, G. L. (2010) Batch means and spectral variance estimators in Markov chain Monte Carlo. *Ann. Statist.*, **38**, 1034–1070. URL: `https://doi.org/10.1214/09-AOS735`.

Geyer, C. J. (1992) Practical Markov Chain Monte Carlo. *Statistical science*, **7**, 473–483.

Gibson, G., Glasbey, C. and Elston, D. (1994) Monte carlo evaluation of multivariate normal integrals and sensitivity to variate ordering. *Advances in Numerical Methods and Applications*, 120–126.

Gilks, W. R. and Berzuini, C. (2001) Following a moving target—Monte Carlo inference for dynamic Bayesian models. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, **63**, 127–146.

Heng, J., Bishop, A. N., Deligiannidis, G. and Doucet, A. (2020) Controlled Sequential Monte Carlo. *Annals of Statistics (to appear)*.

Johansen, A. M., Del Moral, P. and Doucet, A. (2006) Sequential Monte Carlo samplers for rare events. In *Proceedings of the 6th International Workshop on Rare Event Simulation*, 256–267.

Kantas, N., Beskos, A. and Jasra, A. (2014) Sequential Monte Carlo methods for high-dimensional inverse problems: a case study for the Navier-Stokes equations. *SIAM/ASA J. Uncertain. Quantif.*, **2**, 464–489. URL: `https://doi.org/10.1137/130930364`.

Lee, A. and Whiteley, N. (2018) Variance estimation in the particle filter. *Biometrika*, **105**, 609–625. URL: `http://dx.doi.org/10.1093/biomet/asy028`.

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. *J. Comput. Graph. Statist.*, **19**, 769–789.

Naesseth, C. A., Lindsten, F. and Schön, T. B. (2014) Sequential monte carlo for graphical models. In *Advances in Neural Information Processing Systems 27* (eds. Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence and K. Q. Weinberger), 1862–1870. Curran Associates, Inc. URL: http://papers.nips.cc/paper/5570-sequential-monte-carlo-for-graphical-models.pdf.

Neal, R. M. (2001) Annealed importance sampling. *Stat. Comput.*, **11**, 125–139.

OEIS Foundation Inc. (2020) The on-line encyclopedia of integer sequences. http://oeis.org.

Olsson, J. and Douc, R. (2019) Numerically stable online estimation of variance in particle filters. *Bernoulli*, **25**, 1504–1535. URL: https://doi.org/10.3150/18-bej1028.

Ridgway, J. (2016) Computation of Gaussian orthant probabilities in high dimension. *Stat. Comput.*, **26**, 899–916.

Ridgway, J., Alquier, P., Chopin, N. and Liang, F. (2014) PAC-bayesian AUC classification and scoring. In *Advances in Neural Information Processing Systems 27* (eds. Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence and K. Q. Weinberger), 658–666. Curran Associates, Inc.

Roberts, G. O. and Rosenthal, J. S. (2004) General state space Markov chains and MCMC algorithms. *Probab. Surv.*, **1**, 20–71.

Salomone, R., South, L. F., Drovandi, C. C. and Kroese, D. P. (2018) Unbiased and consistent nested sampling via sequential Monte Carlo. *arxiv preprint 1805.03924*.

Schäfer, C. and Chopin, N. (2013) Sequential Monte Carlo on large binary sampling spaces. *Stat. Comput.*, **23**, 163–184.

Septier, F., Pang, S. K., Carmi, A. and Godsill, S. (2009) On mcmc-based particle methods for bayesian filtering: Application to multitarget tracking. In *2009 3rd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 360–363. IEEE.

Septier, F. and Peters, G. W. (2016) Langevin and Hamiltonian based Sequential MCMC for Efficient Bayesian Filtering in High-dimensional Spaces. *IEEE Journal of Selected Topics in Signal Processing*.

Skilling, J. (2006) Nested sampling for general Bayesian computation. *Bayesian Anal.*, **1**, 833–859.

South, L. F., Pettitt, A. N. and Drovandi, C. C. (2019) Sequential Monte Carlo samplers with independent Markov chain Monte Carlo proposals. *Bayesian Anal.*, **14**, 773–796. URL: https://doi.org/10.1214/18-BA1129.

Sweeting, T. (1989) On conditional weak convergence. *Journal of Theoretical Probability*, **2**, 461–474.

Tan, Z. (2015) Resampling Markov chain Monte Carlo algorithms: basic analysis and empirical comparisons. *J. Comput. Graph. Statist.*, **24**, 328–356. URL: https://doi.org/10.1080/10618600.2014.897625.

Zhou, Y., Johansen, A. M. and Aston, J. A. D. (2016) Toward automatic model comparison: an adaptive sequential Monte Carlo approach. *J. Comput. Graph. Statist.*, **25**, 701–726. URL: http://dx.doi.org/10.1080/10618600.2015.1060885.

CREST-ENSAE, Institut Polytechnique de Paris, France

On the complexity of backward smoothing algorithms

This work has been made available in the preprint form as:

Dau, H. D., & Chopin, N. (2022). On the complexity of backward smoothing algorithms. arXiv preprint arXiv:2207.00976.

ON THE COMPLEXITY OF BACKWARD SMOOTHING ALGORITHMS

HAI-DANG DAU & NICOLAS CHOPIN

Abstract. In the context of state-space models, backward smoothing algorithms rely on a backward sampling step, which by default has a $\mathcal{O}(N^2)$ complexity (where N is the number of particles). An alternative implementation relying on rejection sampling has been proposed in the literature, with stated $\mathcal{O}(N)$ complexity. We show that the running time of such algorithms may have an infinite expectation. We develop a general framework to establish the convergence and stability of a large class of backward smoothing algorithms that may be used as more reliable alternatives. We propose three novel algorithms within this class. The first one mixes rejection with multinomial sampling; its running time has finite expectation, and close-to-linear complexity (in a certain class of models). The second one relies on MCMC, and has deterministic $\mathcal{O}(N)$ complexity. The third one may be used even when the transition of the model is intractable. We perform numerical experiments to confirm the good properties of these novel algorithms.

1. Introduction

1.1. **Background.** A state-space model is composed of an unobserved Markov process $X_0, \dots, X_T$ and observed data $Y_0, \dots, Y_T$. Given $X_0, \dots, X_T$, the data $Y_0, \dots, Y_T$ are independent and generated through some specified emission distribution $Y_t | X_t \sim \boldsymbol{f}_t(\mathrm{d}y_t | x_t)$. These models have wide-ranging applications (e.g. in biology, economics and engineering). Two important inference tasks related to state-space models are filtering (computing the distribution of X_t given $Y_0, \dots, Y_t$) and smoothing (computing the distribution of the whole trajectory $(X_0, \dots, X_t)$, again given all data until time t). Filtering is usually carried out through a particle filter, that is, a sequential Monte Carlo algorithm that propagates N weighted particles (realisations) through Markov and importance sampling steps; see Chopin and Papaspiliopoulos (2020) for a general introduction to state-space models (Chapter 2) and particle filters (Chapter 10).

This paper is concerned with smoothing algorithms that approximate the smoothing distributions with empirical distributions based on the output of a particle filter (i.e. the locations and weights of the N particles at each time step). A simple example is genealogy tracking (initially introduced by Kitagawa, 1996), which keeps track of the ancestry (past states) of each particles. This smoother suffers from degeneracy: for t large enough, all the particles have the same ancestor at time 0.

The forward filtering backward smoothing (FFBS) algorithm (Godsill et al., 2004) has been proposed as a solution to this problem. Starting from the filtering approximation at time t, the algorithm samples successively particles at times $t-1$, $t-2$, etc. using backward kernels. The naive implementation has an $\mathcal{O}(N^2)$ cost. However, if the Markov transition density is bounded, a rejection sampling–based

scheme can be used (Douc et al., 2011). Its complexity is shown to be $\mathcal{O}(N)$ under restrictive assumptions on the model.

In many applications, one is mainly interested in approximating smoothing expectations of additive functions of the form

$$\mathbb{E}\left[\psi_0(X_0) + \psi_1(X_0, X_1) + \cdots + \psi_t(X_{t-1}, X_t) \middle| Y_0, \ldots, Y_t\right]$$

for some functions $\psi_0, \ldots, \psi_t$. Such expectations can be approximated on-line in $\mathcal{O}(N^2)$ time by a procedure described in Del Moral et al. (2010). Inspired by this, the particle-based, rapid incremental smoother (PaRIS) algorithm of Olsson and Westerborn (2017) replaces some of the calculations with an additional layer of Monte Carlo approximation. Again, it is possible to employ rejection sampling at this level and get an $\mathcal{O}(N)$ cost under strong assumptions.

There are three problems however with the rejection versions of FFBS and PaRIS. First, the $\mathcal{O}(N)$ claim does not hold for realistic models (as we shall demonstrate theoretically and numerically). Second, their running time is random (we elaborate below why this is a drawback). Third (and this problem also apply to the $\mathcal{O}(N^2)$ versions of FFBS and PaRIS), they require the Markov transition of the model to be tractable, which is not the case for certain models of practical interest. These three problems are deeply linked to the way the backward sampling step operates.

1.2. **Motivation and structure.** We will therefore propose new methods to address these issues. Since backward sampling is central to a wide variety of algorithms (e.g. FFBS, forward-additive smoothing, PaRIS), Section 2 presents them in a unified framework. We show how they can all be expressed in terms of discrete backward kernels, which are essentially random $N \times N$ matrices. We specify how these matrices are used differently for off-line and on-line smoothing scenarios. Importantly, we state generic sufficient conditions which ensure that the resulting algorithms are consistent and stable as $T \to \infty$.

Having at hand the necessary theoretical framework, the rest of the article is spent on methodological innovations. We first closely look at the use of rejection sampling and realise that in many models, the resulting execution time may have an infinite expectation; see Section 3. In highly parallel computing architectures, each processor only handles one or a small number of particles. As such, the heavy-tailed nature of the execution time means that a few machine might prevent the whole system from moving forward. In *all* computing architectures, an infinite mean running time makes it difficult to know when a program will stop, even after having performed few pilot runs. We introduce a hybrid rejection sampling procedure which fixes this problem and leads to a nearly $\mathcal{O}(N)$ algorithm (up to some log factor) in an important class of practical models; again see Section 3.

To make the execution time fully linear and deterministic, we propose in Section 4.1 backward kernels based on MCMC (Markov chain Monte Carlo). Convergence and stability of the resulting algorithm are inherited from the general framework developed in Section 2. MCMC methods require evaluation of the likelihood and thus cannot be applied to models with intractable transition densities. In Section 4.2, we show how the use of forward coupling can replace the role of backward MCMC steps in these scenarios. This makes it possible to obtain stable performance in both on-line and off-line scenarios.

Section 5 illustrates the aforementioned algorithms in both on-line and off-line uses. We highlight how hybrid and MCMC samplers lead to a more user-friendly (i.e. smaller, less random and less model-dependent) execution time than the pure rejection sampler. We also apply our smoother for intractable densities to a continuous-time diffusion process with discretisation. Further numerical examples, in particular those concerning Markov jump processes, are planned to be included in a future version of this paper. We observe that our procedure can indeed prevent degeneracy as $T \to \infty$, provided that some care is taken to build couplings with good performance.

Finally, Section 6 concludes the paper with final practical recommendations and further research directions.

1.3. **Related work.** Interestingly, the potential issue with the complexity of FFBS-reject has been suggested by their authors of this method. Indeed, Proposition 1 of Douc et al. (2011) implies that the asymptotic complexity of FFBS-reject might tend to infinity in some situations. However, we show that the infiniteness might already happen at finite values of N. Moreover, we propose concrete solutions to fix this problem and analyse them both theoretically and practically.

Figure 1 of Olsson and Westerborn (2017) and the accompanying discussion provide an excellent intuition on the stability of smoothing algorithms based on the support size of the backward kernels. We formalise these insights and use them to construct new efficient and stable algorithms.

The heavy-tailed distribution of the running time of FFBS-reject have been remarked in Taghavi et al. (2013), who proposed a hybrid algorithm combining multinomial and rejection sampling. However, theoretical analysis of the complexity was not performed and the extension to online smoothing was not considered. (The PaRIS algorithm had not be invented then.) Using MCMC steps (started at the previous ancestor) instead of rejection sampling has been explored in Bunch and Godsill (2013). Again, consistency and stability were not formally proved and the article was limited to the offline scenario. Gloaguen et al. (2019) briefly mention the use of MCMC in PaRIS algorithm, without doing further theoretical or numerical analyses. Moreover, since their MCMC chains do not start at the previous ancestors, a large number of steps are necessary to get a correct algorithm. As we shall see, our procedure requires only one MCMC step.

Another way to reduce the computation time is to perform the expensive backward sampling steps at certain times t only. For other values of t, the naive genealogy tracking smoother is used instead. This idea has been recently proposed by Mastrototaro et al. (2021), who also provided a practical recipe for deciding at which values of t the backward sampling should take place and derived corresponding theoretical results.

Smoothing in models with intractable transition densities is very challenging. If these densities can be estimated accurately, the algorithms proposed by Gloaguen et al. (2019) permit to attack this problem. A case of particular interest is diffusion models, where unbiased transition density estimators are provided in Beskos et al. (2006); Fearnhead et al. (2008). More recently, Yonekura and Beskos (2022) use a special bridge path-space construction to overcome the unavailability of transition densities when the diffusion (possibly with jumps) must be discretised.

Our smoother for intractable models are based on a general coupling principle that is not specific to diffusions. We only require users to be able to simulate their

dynamics (e.g. using discretisation in the case of diffusions) and to manipulate random numbers in their simulations so that dynamics starting from two different points can meet with some probability. Our method does not directly provide an estimator for the gradient of the transition density with respect to model parameters and thus cannot be used in its current form to perform maximum likelihood estimation (MLE) in intractable models; whereas the aforementioned work have been able to do so in the case of diffusions. However, the main advantage of our approach lies in its generality beyond the diffusion case. Furthermore, modifications allowing to perform MLE are possible and might be explored in further work specifically dedicated to the parameter estimation problem.

The idea of coupling has been incorporated in the smoothing problem in a different manner by Jacob et al. (2019). There, the goal is to provide offline unbiased estimates of the expectation under the smoothing distribution. Coupling and more generally ideas based on correlated random numbers are also useful in the context of partially observed diffusions via the multilevel approach (Jasra et al., 2017).

In this work, we consider smoothing algorithms that are based on a unique pass of the particle filter. Offline smoothing can be done using repeated iterations of the conditional particle filter (Andrieu et al., 2010). Another approach to smoothing consists of using an additional information filter (Fearnhead et al., 2010), but it is limited to functions depending on one state only. Each of these algorithmic families has their own advantages and disadvantages, of which a detailed discussion is out of the scope of this article (see however Nordh and Antonsson, 2015).

2. General structure of smoothing algorithms

2.1. **Notations. Measure-kernel-function notations.** Let $\mathcal{X}$ and $\mathcal{Y}$ be two measurable spaces with respective σ-algebras $\mathcal{B}(\mathcal{X})$ and $\mathcal{B}(\mathcal{Y})$. The following definitions involve integrals and only make sense when they are well-defined. For a measure μ on $\mathcal{X}$ and a function $f : \mathcal{X} \to \mathbb{R}$, the notations μf and $\mu(f)$ refer to $\int f(x)\mu(\mathrm{d}x)$. A kernel (resp. Markov kernel) K is a mapping from $\mathcal{X} \times \mathcal{B}(\mathcal{Y})$ to $\mathbb{R}$ (resp. $[0,1]$) such that, for $B \in \mathcal{B}(\mathcal{Y})$ fixed, $x \mapsto K(x,B)$ is a measurable function on $\mathcal{X}$; and for x fixed, $B \mapsto K(x,B)$ is a measure (resp. probability measure) on $\mathcal{Y}$. For a real-valued function g defined on $\mathcal{Y}$, let $Kg : \mathcal{X} \to \mathbb{R}$ be the function $Kg(x) := \int g(y)K(x,\mathrm{d}y)$. We sometimes write $K(x,g)$ for the same expression. The product of the measure μ on $\mathcal{X}$ and the kernel K is a measure on $\mathcal{Y}$, defined by $\mu K(B) := \int K(x,B)\mu(\mathrm{d}x)$.

Other notations. • The notation $X_{0:t}$ is a shorthand for $(X_0, \ldots, X_t)$ • We denote by $\mathcal{M}(W^{1:N})$ the multinomial distribution supported on $\{1, 2, \ldots, N\}$. The respective probabilities are $W_1, \ldots, W_N$. If they do not sum to 1, we implicitly refer to the normalised version obtained by multiplication of the weights with the appropriate constant • The symbol $\xrightarrow{\mathbb{P}}$ means convergence in probability • The geometric distribution with parameter λ is supported on $\mathbb{Z}_{\geq 1}$, has probability mass function $f(n) = \lambda(1-\lambda)^{n-1}$ and is noted by $\mathrm{Geo}(\lambda)$ • Let $\mathcal{X}$ and $\mathcal{Y}$ be two measurable spaces. Let μ and ν be two probability measures on $\mathcal{X}$ and $\mathcal{Y}$ respectively. The o-times product measure $\mu \otimes \nu$ is defined via $(\mu \otimes \nu)(h) := \iint h(x,y)\mu(\mathrm{d}x)\nu(\mathrm{d}y)$ for bounded functions $h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$. If $X \sim \mu$ and $Y \sim \nu$, we sometimes note $\mu \otimes \nu$ by $X \otimes Y$.

2.2. **Feynman-Kac formalism and the bootstrap particle filter.** Let $\mathcal{X}_{0:T}$ be a sequence of measurable spaces and $M_{1:T}$ be a sequence of Markov kernels such that M_t is a kernel from $\mathcal{X}_{t-1}$ to $\mathcal{X}_t$. Let $X_{0:T}$ be an unobserved inhomogeneous Markov chain with starting distribution $X_0 \sim \mathbb{M}_0(\mathrm{d}x_0)$ and Markov kernels $M_{1:T}$; i.e. $X_t|X_{t-1} \sim M_t(X_{t-1}, \mathrm{d}x_t)$ for $t \geq 1$. We aim to study the distribution of $X_{0:T}$ given observed data $Y_{0:T}$. Conditioned on $X_{0:T}$, the data $Y_0, \ldots, Y_T$ are independent and

$$Y_t|X_{0:T} \equiv Y_t|X_t \sim \boldsymbol{f}_t(\cdot|X_t)$$

for a certain emission distribution $\boldsymbol{f}_t(\mathrm{d}y_t|x_t)$. Assume that there exists dominating measures $\tilde{\lambda}_t$ not depending on x_t such that

$$\boldsymbol{f}_t(\mathrm{d}y_t|x_t) = f_t(y_t|x_t)\tilde{\lambda}_t(\mathrm{d}y_t).$$

The distribution of $X_{0:t}|Y_{0:t}$ is then given by

$$\mathbb{Q}_t(\mathrm{d}x_{0:t}) = \frac{1}{L_t}\mathbb{M}_0(\mathrm{d}x_0)\prod_{s=1}^{t} M_s(x_{s-1}, \mathrm{d}x_s)G_s(x_s) \tag{1}$$

where $G_s(x_s) := f(y_s|x_s)$ and $L_t > 0$ is the normalising constant. Moreover, $\mathbb{Q}_{-1} := \mathbb{M}_0$ and $L_{-1} := 1$ by convention. Equation (1) defines a Feynman-Kac model (Del Moral, 2004). It does not require M_t to admit a transition density, although herein we only consider models where this assumption holds. Let λ_t be a dominating measure on $\mathcal{X}_t$ in the sense that there exists a function m_t (not necessarily tractable) such that

$$M_t(x_{t-1}, \mathrm{d}x_t) = m_t(x_{t-1}, x_t)\lambda_t(\mathrm{d}x_t). \tag{2}$$

A special case of the current framework are linear Gaussian state space models. They will serve as a running example for the article, and some of the results will be specifically demonstrated for models of this class. The rationale is that many real-world dynamics are partly, or close to, Gaussian. The notations for linear Gaussian models are given in Appendix A.1 and we will refer to them whenever this model class is discussed.

Particle filters are algorithms that sample from $\mathbb{Q}_t(\mathrm{d}x_t)$ in an on-line manner. In this article, we only consider the bootstrap particle filter (Gordon et al., 1993) and we detail its notations in Algorithm 1. Many results in the following do apply to the auxiliary filter (Pitt and Shephard, 1999) as well, and we shall as a rule indicate explicitly when it is *not* the case.

We end this subsection with the definition of two sigma-algebras that will be referred to throughout the paper. Using the notations of Algorithm 1, let

$$\begin{aligned} \mathcal{F}_t &:= \sigma(X_{0:t}^{1:N}, A_{1:t}^{1:N}), \\ \mathcal{F}_t^- &:= \sigma(X_{0:t}^{1:N}). \end{aligned} \tag{3}$$

2.3. **Backward kernels and off-line smoothing.** In this subsection, we first describe three examples of backward kernels, in which we emphasise both the random measure and the random matrix viewpoints. We then formalise their use by stating a generic off-line smoothing algorithm.

Algorithm 1: Bootstrap particle filter

Input: Feynman-Kac model (1)

Simulate $X_0^{1:N} \overset{\text{i.i.d.}}{\sim} \mathbb{M}_0$

Set $\omega_0^n \leftarrow G_0(X_0^n)$ for $n = 1, \dots, N$

Set $\ell_0^N \leftarrow \sum_{n=1}^N \omega_0^n / N$

Set $W_0^n \leftarrow \omega_0^n / N\ell_0^N$ for $n = 1, \dots, N$

for $t \leftarrow 1$ **to** T **do**

Resample. Simulate $A_t^{1:N} \overset{\text{i.i.d.}}{\sim} \mathcal{M}(W_{t-1}^{1:N})$

Move. Simulate $X_t^n \sim M_t(X_{t-1}^{A_t^n}, \mathrm{d}x_t)$ for $n = 1, \dots, N$

Reweight. Set $\omega_t^n \leftarrow G_t(X_t^n)$ for $n = 1, 2, \dots, N$

Set $\ell_t^N \leftarrow \sum_{n=1}^N \omega_t^n / N$

Set $W_t^n \leftarrow \omega_t^n / N\ell_t^N$ for $n = 1, 2, \dots, N$

Output: For all $t \geq 0$ and function $\varphi : \mathcal{X}_t \to \mathbb{R}$, the quantity $\sum_{n=1}^N W_t^n \varphi(X_t^n)$ approximates $\int \mathbb{Q}_t(\mathrm{d}x_{0:t})\varphi(x_t)$ and the quantity ℓ_t^N approximates L_t / L_{t-1}

Example 1 (FFBS algorithm)**.** *Once Algorithm 1 has been run, the FFBS procedure generates a trajectory approximating the smoothing distribution in a backward manner. More precisely, it starts by simulating index $\mathcal{I}_T \sim \mathcal{M}(W_T^{1:N})$ at time T. Then, recursively for $t = T, \dots, 1$, given indices $\mathcal{I}_{t:T}$, it generates $\mathcal{I}_{t-1} \in \{1, \dots, N\}$ with probability proportional to $W_{t-1}^n m_t(X_{t-1}^n, X_t^{\mathcal{I}_t})$. The smoothing trajectory is returned as $(X_0^{\mathcal{I}_0}, \dots, X_T^{\mathcal{I}_T})$. Formally, given $\mathcal{F}_T$, the indices $\mathcal{I}_{0:T}$ are generated according to the distribution*

$$\mathcal{M}(W_t^{1:N})(\mathrm{d}i_T) \left[B_T^{N,\mathrm{FFBS}}(i_T, \mathrm{d}i_{T-1}) B_{T-1}^{N,\mathrm{FFBS}}(i_{T-1}, \mathrm{d}i_{T-2}) \dots B_1^{N,\mathrm{FFBS}}(i_1, \mathrm{d}i_0) \right]$$

where the (random) backward kernels $B_t^{N,\mathrm{FFBS}}$ are defined by

$$B_t^{N,\mathrm{FFBS}}(i_t, \mathrm{d}i_{t-1}) := \sum_{n=1}^N \frac{W_{t-1}^n m_t(X_{t-1}^n, X_t^{i_t})}{\sum_{k=1}^N W_{t-1}^k m_t(X_{t-1}^k, X_t^{i_t})} \delta_n. \tag{4}$$

More simply, we can also look at these random kernels as random $N \times N$ matrices of which entries are given by

$$\hat{B}_t^{N,\mathrm{FFBS}}[i_t, i_{t-1}] := \frac{W_{t-1}^{i_{t-1}} m_t(X_{t-1}^{i_{t-1}}, X_t^{i_t})}{\sum_{k=1}^N W_{t-1}^k m_t(X_{t-1}^k, X_t^{i_t})}. \tag{5}$$

We will need both the kernel viewpoint (4) and the matrix viewpoint (5) in this paper as the better choice depends on the context.

Example 2 (Genealogy tracking)**.** *It is well known that Algorithm 1 already gives as a by-product an approximation of the smoothing distribution. This information can be extracted from the genealogy, by first simulating index $\mathcal{I}_T \sim \mathcal{M}(W_T^{1:N})$ at time T, then successively appending ancestors until time 0 (i.e. setting sequentially $\mathcal{I}_{t-1} \leftarrow A_t^{\mathcal{I}_t}$). The smoothed trajectory is returned as $(X_0^{\mathcal{I}_0}, \dots, X_T^{\mathcal{I}_T})$. More formally, conditioned on $\mathcal{F}_T$, we simulate the indices $\mathcal{I}_{0:T}$ according to*

$$\mathcal{M}(W_t^{1:N})(\mathrm{d}i_T) \left[B_T^{N,\mathrm{GT}}(i_T, \mathrm{d}i_{T-1}) B_{T-1}^{N,\mathrm{GT}}(i_{T-1}, \mathrm{d}i_{T-2}) \dots B_1^{N,\mathrm{GT}}(i_1, \mathrm{d}i_0) \right]$$

where GT stands for "genealogy tracking" and the kernels $B_t^{N,\mathrm{GT}}$ *are simply*

$$B_t^{N,\mathrm{GT}}(i_t, \mathrm{d}i_{t-1}) := \delta_{A_t^{i_t}}(\mathrm{d}i_{t-1}). \tag{6}$$

Again, it may be more intuitive to view this random kernel as a random $N \times N$ *matrix, the elements of which are given by*

$$\hat{B}_t^{N,\mathrm{GT}}[i_t, i_{t-1}] := \mathbb{1}\left\{i_{t-1} = A_t^{i_t}\right\}.$$

Example 3 (MCMC backward samplers)**.** *In Example 2, the backward variable* $\mathcal{I}_{t-1}$ *is simply set to* $A_t^{\mathcal{I}_t}$*. On the contrary, in Example 1, we need to launch a simulator for the discrete measure* $W_{t-1}^n m_t(X_{t-1}^n, X_t^{\mathcal{I}_t})$*. Interestingly, the current value of* $A_t^{\mathcal{I}_t}$ *is not taken into account in that simulator. Therefore, a natural idea to combine the two previous examples is to apply one (or several) MCMC steps to* $A_t^{\mathcal{I}_t}$ *and assign the result to* $\mathcal{I}_{t-1}$*. The MCMC algorithm operates on the space* $\{1, 2, \ldots, N\}$ *and targets the invariant measure* $W_{t-1}^n m_t(X_{t-1}^n, X_t^{\mathcal{I}_t})$*. If only one independent Metropolis-Hastings (MH) step is used and the proposal is* $\mathcal{M}(W_{t-1}^{1:N})$*, the corresponding random matrix* $\hat{B}_t^{N,\mathrm{IMH}}$ *has values*

$$\hat{B}_t^{N,\mathrm{IMH}}[i_t, i_{t-1}] = W_{t-1}^{i_{t-1}} \min\left(1, m_t(X_{t-1}^{i_{t-1}}, X_t^{i_t}) / m_t(X_{t-1}^{A_t^{i_t}}, X_t^{i_t})\right)$$

if $i_{t-1} \neq A_t^{i_t}$*, and*

$$\hat{B}_t^{N,\mathrm{IMH}}[i_t, A_t^{i_t}] = 1 - \sum_{n \neq A_t^{i_t}} \hat{B}_t^{N,\mathrm{IMH}}[i_t, n].$$

This third example shows that some elements of the matrix $\hat{B}_t^{N,\mathrm{IMH}}$ *might be expensive to calculate. If several MCMC steps are performed,* all *elements of* $\hat{B}_t^{N,\mathrm{IMH}}$ *will have non-trivial expressions. Still, simulating from* $B_t^{N,\mathrm{IMH}}(i_t, \mathrm{d}i_{t-1})$ *is easy as it amounts to running a standard MCMC algorithm. MCMC backward samples are studied in more details in Section 4.1.*

We formalise how off-line smoothing can be done given random matrices $\hat{B}_{1:T}^N$; see Algorithm 2. Note that in the above examples, our matrices $\hat{B}_t^N$ are $\mathcal{F}_t$-measurable (i.e. they depend on particles and indices up to time t), but this is not necessarily the case in general (i.e. they may also depend on additional random variables, see Section 2.5). Furthermore, Algorithm 2 describes how to perform smoothing using the matrices $\hat{B}_{1:T}^N$, but does not say where they come from. At this point, it is useful to keep in mind the above three examples. In Section 2.4, we will give a general recipe for constructing valid matrices $\hat{B}_t^N$ (i.e. those that give a consistent algorithm).

Algorithm 2 simulates, given $\mathcal{F}_T$ *and* $\hat{B}_{1:T}^N$, N i.i.d. index sequences $\mathcal{I}_{0:T}^n$, each distributed according to

$$\mathcal{M}(W_T^{1:N})(\mathrm{d}i_T) \prod_{t=T}^{1} B_t^N(i_t, \mathrm{d}i_{t-1}).$$

Once the indices $\mathcal{I}_{0:T}^{1:N}$ are simulated, the N smoothed trajectories are returned as $(X_0^{\mathcal{I}_0^n}, \ldots, X_T^{\mathcal{I}_T^n})$. Given $\mathcal{F}_T$ and $\hat{B}_{1:T}^N$, they are thus conditionally i.i.d. and their

Algorithm 2: Generic off-line smoother

Input: Filtering results $X_{0:T}^{1:N}$, $W_{0:T}^{1:N}$, and $A_{1:T}^{1:N}$ from Algorithm 1; random matrices $\hat{B}_{1:T}^{N}$ (see Section 2.3 for two examples of such matrices and Section 2.4 for a general recipe to construct them)

for $n \leftarrow 1$ **to** N **do**

  Simulate $\mathcal{I}_T^n \sim \mathcal{M}(W_T^{1:N})$

  for $t \leftarrow T$ **to** *1* **do**

    Simulate $\mathcal{I}_{t-1}^n \sim B_t^N(\mathcal{I}_t^n, \mathrm{d}i_{t-1})$ (the kernel $B_t^N(\mathcal{I}_t, \cdot)$ is defined by the $\mathcal{I}_t$-th row of the input matrix $\hat{B}_t^N$)

Output: The N smoothed trajectories $(X_0^{\mathcal{I}_0^n}, \ldots, X_T^{\mathcal{I}_T^n})$ for $n = 1, \ldots, N$

conditional distribution is described by the $x_{0:T}$ component of the joint distribution

$$\bar{\mathbb{Q}}_T^N(\mathrm{d}x_{0:T}, \mathrm{d}i_{0:T}) := \mathcal{M}(W_T^{1:N})(\mathrm{d}i_T) \left[\prod_{t=T}^{1} B_t^N(i_t, \mathrm{d}i_{t-1})\right] \left[\prod_{t=T}^{0} \delta_{X_t^{i_t}}(\mathrm{d}x_t)\right]. \tag{7}$$

Throughout the paper, the symbol $\bar{\mathbb{Q}}_T^N$ will refer to this joint distribution, while the symbol $\mathbb{Q}_T^N$ will refer to the $x_{0:T}$-marginal of $\bar{\mathbb{Q}}_T^N$ only. This allows the notation $\mathbb{Q}_T^N \varphi$ to make sense, where $\varphi = \varphi(x_0, \ldots, x_T)$ is a real-valued function defined on the hidden states.

2.4. **Validity and convergence.** The kernels $B_t^{N,\mathrm{FFBS}}$ and $B_t^{N,\mathrm{GT}}$ are both valid backward kernels to generate convergent approximation of the smoothing distribution (Del Moral, 2004; Douc et al., 2011). This subsection shows that they are not the only ones and gives a sufficient condition for a backward kernel to be valid. It will prove a necessary tool to build more efficient B_t^N later in the paper.

Recall that Algorithm 1 outputs particles $X_{0:T}^{1:N}$, weights $W_{0:T}^{1:N}$ and ancestor variables $A_{1:T}^{1:N}$. Imagine that the $A_{1:T}^{1:N}$ were discarded after filtering has been done and we wish to simulate them back. We note that, since the $X_{0:T}^{1:N}$ are given, the $T \times N$ variables $A_{1:T}^{1:N}$ are conditionally i.i.d. We can thus simulate them back from

$$p(a_t^n | x_{0:T}^{1:N}) = p(a_t^n | x_{t-1}^{1:N}, x_t^n) \propto w_{t-1}^{a_t^n} m_t(x_{t-1}^{a_t^n}, x_t^n).$$

This is precisely the distribution of $B_t^{N,\mathrm{FFBS}}(n, \cdot)$. It turns out that any other invariant kernel that can be used for simulating back the discarded $A_{1:T}^{1:N}$ will lead to a convergent algorithm as well. For instance, $B_t^{N,\mathrm{GT}}(n, \cdot)$ (Example 2) simply returns back the old A_t^n, unlike $B_t^{N,\mathrm{FFBS}}(n, \cdot)$ which creates a new version. The kernel $B_t^{N,\mathrm{IMH}}(n, \cdot)$ (Example 3) is somewhat an intermediate between the two. We formalise these intuitions in the following theorem. It is stated for the bootstrap particle filter, but as a matter of fact, the proof can be extended straightforwardly to auxiliary particle filters as well.

Assumption 1. *For all* $0 \leq t \leq T$, $G_t(x_t) > 0$ *and* $\|G_t\|_\infty < \infty$.

Theorem 1. *We use the same notations as in Algorithms 1 and 2 (in particular,* $\hat{B}_t^N$ *denotes the transition matrix that corresponds to the considered kernel* B_t^N*). Assume that for any* $1 \leq t \leq T$, *the random matrix* $\hat{B}_t^N$ *satisfies the following conditions:*

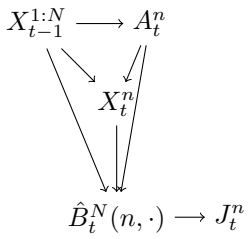

FIGURE 1. Relation between variables described in Theorem 1.

- *given* $\mathcal{F}_{t-1}$ *and* $\hat{B}_{1:t-1}^N$, *the variables* $(X_t^n, A_t^n, \hat{B}_t^N(n,\cdot))$ *for* $n = 1, \dots, N$ *are i.i.d. and their distribution only depends on* $X_{t-1}^{1:N}$, *where* $\hat{B}_t^N(n,\cdot)$ *is the* n*-th row of matrix* $\hat{B}_t^N$;
- *if* J_t^n *is a random variable such that* $J_t^n \mid X_{t-1}^{1:N}, X_t^n, \hat{B}_t^N(n,\cdot) \sim B_t^N(n,\cdot)$, *then* (J_t^n, X_t^n) *has the same distribution as* (A_t^n, X_t^n) *given* $X_{t-1}^{1:N}$.

Then under Assumption 1, there exists constants $C_T > 0$ *and* $S_T < \infty$ *such that, for any* $\delta > 0$ *and function* $\varphi = \varphi(x_0, \dots, x_T)$:

$$\mathbb{P}\left(\left|\mathbb{Q}_T^N \varphi - \mathbb{Q}_T \varphi\right| \geq \frac{\sqrt{-2\log(\delta/2C_T)}S_T \left\|\varphi\right\|_\infty}{\sqrt{N}}\right) \leq \delta \tag{8}$$

where $\mathbb{Q}_N^T$ *is defined by* (7).

A typical relation between variables defined in the statement of the theorem is illustrated by a graphical model in Figure 1. (See Bishop 2006, Chapter 8 for the formal definition of graphical models and how to use them.) By "typical", we mean that Theorem 1 technically allows for more complicated relations, but the aforementioned figure captures the most essential cases.

Theorem 1 is a generalisation of Douc et al. (2011, Theorem 5). Its proof thus follows the same lines (Appendix E.1). However, in our case the measure $\mathbb{Q}_T^N(\mathrm{d}x_{0:T})$ is no longer Markovian. This is because the backward kernel $B_t^N(i_t, \mathrm{d}i_{t-1})$ does not depend on $X_t^{i_t}$ alone, but also possibly on its ancestor and extra random variables.

As we have seen in (7), $\mathbb{Q}_T^N$ is fundamentally a discrete measure of which the support contains N^{T+1} elements. As such, $\mathbb{Q}_T^N \varphi$ cannot be computed exactly in general and must be approximated using N trajectories $(X_0^{\mathcal{I}_0^n}, \dots, X_T^{\mathcal{I}_T^n})$ simulated via Algorithm 2. Theorem 1 is thus completed by the following corollary, which is an immediate consequence of Hoeffding inequality (Appendix E.13).

Corollary 1. *Under the same setting as Theorem 1, we have*

$$\mathbb{P}\left(\left|\frac{1}{N}\sum_n \varphi(X_0^{\mathcal{I}_0^n}, \dots, X_T^{\mathcal{I}_T^n}) - \mathbb{Q}_T\varphi\right| \geq \frac{\sqrt{-2\log\left(\frac{\delta}{2(C_T+1)}\right)}(S_T+1)\left\|\varphi\right\|_\infty}{\sqrt{N}}\right) \leq \delta.$$

2.5. **Generic on-line smoother.** As we have seen in Section 2.3 and Section 2.4, in general, the expectation $\mathbb{Q}_T^N \varphi$, for a real-valued function $\varphi = \varphi(x_0, \dots, x_T)$ of the hidden states, cannot be computed exactly due to the large support (N^{T+1} elements) of $\mathbb{Q}_T^N$. Moreover, in certain settings we are interested in the quantities $\mathbb{Q}_t^N \varphi_t$ for different functions φ_t. They cannot be approximated in an on-line manner

without more assumptions on the connection between φ_{t-1} and φ_t. If the family (φ_t) is additive, i.e. there exists functions ψ_t such that

$$\varphi_t(x_{0:t}) := \psi_0(x_0) + \psi_1(x_0, x_1) + \cdots + \psi_t(x_{t-1}, x_t) \tag{9}$$

then we can calculate $\mathbb{Q}_t^N \varphi_t$ both exactly *and* on-line. The procedure was first described in Del Moral et al. (2010) for the kernel $\mathbb{Q}_t^{N,\mathrm{FFBS}}$ (i.e. the measure defined by (7) and the random kernels $B_t^{N,\mathrm{FFBS}}$), but we will use the idea for other kernels as well. In this subsection, we first explain the principle of the method, then discuss its computational complexity and the link to the PaRIS algorithm (Olsson and Westerborn, 2017).

Principle. For simplicity, we start with the special case $\varphi_t(x_{0:t}) = \psi_0(x_0)$. Equation (7) and the matrix viewpoint of Markov kernels then give

$$\mathbb{Q}_t^N \varphi_t = \left[W_t^1 \dots W_t^N\right] \hat{B}_t^N \hat{B}_{t-1}^N \dots \hat{B}_1^N \begin{bmatrix} \psi_0(X_0^1) \\ \vdots \\ \psi_0(X_0^N) \end{bmatrix}.$$

This naturally suggests the following recursion formula to compute $\mathbb{Q}_t^N \varphi_t$:

$$\mathbb{Q}_t^N \varphi_t = \left[W_t^1 \dots W_t^N\right] \hat{S}_t^N$$

with $\hat{S}_0^N = [\psi_0(X_0^1) \dots \psi_0(X_0^N)]^\top$ and

$$\hat{S}_t^N := \hat{B}_t^N \hat{S}_{t-1}^N. \tag{10}$$

In the general case where functions φ_t are given by (9), simple calculations (Appendix E.2) show that (10) is replaced by

$$\hat{S}_t^N := \hat{B}_t^N \hat{S}_{t-1}^N + \mathrm{diag}(\hat{B}_t^N \hat{\psi}_t^N) \tag{11}$$

where the $N \times N$ matrix $\hat{\psi}_t^N$ is defined by

$$\hat{\psi}_t^N[i_{t-1}, i_t] := \psi_t(X_{t-1}^{i_{t-1}}, X_t^{i_t})$$

and the operator $\mathrm{diag} : \mathbb{R}^{N \times N} \to \mathbb{R}^N$ extracts the diagonal of a matrix. This is exactly what is done in Algorithm 3.

Algorithm 3: Generic on-line smoother for additive functions (one step)

Input: Particles $X_{t-1}^{1:N}$ and weights $W_{t-1}^{1:N}$ at time $t-1$; the $N \times 1$ vector $\hat{S}_{t-1}^N$ (see text); additive function (9)

Generate $X_t^{1:N}$ and $W_t^{1:N}$ according to the particle filter (Algorithm 1)

Calculate the random matrix $\hat{B}_t^N$ (see Section 2.3 and Section 2.4)

Create the $N \times 1$ vector $\hat{S}_t^N$ according to (11). More precisely:

for $i_t \leftarrow 1$ **to** N **do**

$\quad \hat{S}_t^N[i_t] \leftarrow \sum_{i_{t-1}} \hat{B}_t^N[i_t, i_{t-1}] \left(\hat{S}_{t-1}^N[i_{t-1}] + \psi_t(X_{t-1}^{i_{t-1}}, X_t^{i_t})\right)$

Output: Estimate $\sum_n W_t^n \hat{S}_t^N[n]$ of $\mathbb{Q}_t(\varphi_t)$; particles $X_t^{1:N}$, weights $W_t^{1:N}$ and vector S_t^N for the next step

---

Computational complexity and the PaRIS algorithm. Equations (10) and (11) involve a matrix-vector multiplication and thus require, in general, $\mathcal{O}(N^2)$ operations to be evaluated. When $\hat{B}_t^N \equiv \hat{B}_t^{N,\mathrm{FFBS}}$, Algorithm 3 becomes the $\mathcal{O}(N^2)$ on-line smoothing algorithm of Del Moral et al. (2010). The $\mathcal{O}(N^2)$ complexity can however be lowered to $\mathcal{O}(N)$ if the matrices $\hat{B}_t^N$ are *sparse*. This is the idea behind the PaRIS algorithm (Olsson and Westerborn, 2017), where the full matrix $\hat{B}_t^{N,\mathrm{FFBS}}$ is unbiasedly estimated by a sparse matrix $\hat{B}_t^{N,\mathrm{PaRIS}}$. More specifically, for any integer $\tilde{N} > 1$, for any $n \in 1, \ldots, N$, let $J_t^{n,1}, \ldots, J_t^{n,\tilde{N}}$ be conditionally i.i.d. random variables simulated from $B_t^{N,\mathrm{FFBS}}(n, \cdot)$. The random matrix $\hat{B}_t^{N,\mathrm{PaRIS}}$ is then defined as

$$\hat{B}_t^{N,\mathrm{PaRIS}}[n, m] := \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} \mathbb{1}\left\{J_t^{n,\tilde{n}} = m\right\}$$

and the corresponding random kernel is

$$B_t^{N,\mathrm{PaRIS}}(n, \mathrm{d}m) = \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} \delta_{J_t^{n,\tilde{n}}}(\mathrm{d}m). \tag{12}$$

The following straightforward proposition establishes the validity of the $B_t^{N,\mathrm{PaRIS}}$ kernel as well as the corresponding $\mathcal{O}(N)$ complexity of (10) and (11), provided that $\tilde{N}$ is fixed as $N \to \infty$.

Proposition 1. *The matrix $\hat{B}_t^{N,\mathrm{PaRIS}}$ has only $\mathcal{O}(N\tilde{N})$ non-zero elements out of N^2. It is an unbiased estimate of $\hat{B}_t^{N,\mathrm{FFBS}}$ in the sense that*

$$\mathbb{E}\left[\hat{B}_t^{N,\mathrm{PaRIS}} \middle| \mathcal{F}_t\right] = \hat{B}_t^{N,\mathrm{FFBS}}.$$

Moreover, the sequence of matrices $B_{1:T}^{N,\mathrm{PaRIS}}$ satisfies the two conditions of Theorem 1.

It is important to remark that the $\mathcal{O}(N)$ complexity only refers to the cost of computing the recursions (10) and (11). It does not include the cost of generating the matrices $\hat{B}_t^{N,\mathrm{PaRIS}}$ themselves, i.e., the operations required to simulate the indices $J_t^{n,\tilde{n}}$. In Olsson and Westerborn (2017) it is argued that such simulations have an $\mathcal{O}(N)$ cost using the rejection sampling method whenever the transition density is both upper and lower bounded. Section 3 investigates the claim when this hypothesis is violated.

2.6. **Stability.** When $\hat{B}_t^N \equiv \hat{B}_t^{N,\mathrm{GT}}$, Algorithms 2 and 3 reduce to to the genealogy tracking smoother (Kitagawa, 1996). The matrix $\hat{B}_t^{N,\mathrm{GT}}$ is indeed sparse, leading to the well-known $\mathcal{O}(N)$ complexity of this on-line procedure. As per Theorem 1, smoothing via genealogy tracking is convergent at rate $\mathcal{O}(N^{-1/2})$ if T is *fixed*. When $T \to \infty$ however, all particles will eventually share the same ancestor at time 0 (or any fixed time t). Mathematically, this phenomenon is manifested in two ways: (a) for fixed t and function $\phi_t : \mathcal{X}_t \to \mathbb{R}$, the error of estimating $\mathbb{E}[\phi_t(X_t)|Y_{0:T}]$ grows linearly with T; and (b) the error of estimating $\mathbb{E}\left[\sum_{t=0}^{T} \psi_t(x_{t-1}, x_t) \middle| Y_{0:T}\right]$ grows quadratically with T. These correspond respectively to the degeneracy for the fixed marginal smoothing and the additive smoothing problems; see also the introductory section of Olsson and Westerborn (2017) for a discussion. The random matrices $\hat{B}_t^{N,\mathrm{GT}}$ are therefore said to be *unstable* as $T \to \infty$, which is not the case

for $\hat{B}_t^{N,\mathrm{FFBS}}$ or $\hat{B}_t^{N,\mathrm{PaRIS}}$. This subsection gives sufficient conditions to ensure the stability of a general $\hat{B}_t^N$.

The essential point behind smoothing stability is simple: the support of $B_t^{N,\mathrm{FFBS}}(n,\cdot)$ or $B_t^{N,\mathrm{PaRIS}}(n,\cdot)$ for $\tilde{N} \geq 2$ contains more than one element, contrary to that of $B_t^{N,\mathrm{GT}}(n,\cdot)$. This property is formalised by (13). To explain the intuitions, we use the notations of Algorithm 2 and consider the estimate

$$N^{-1}\left(\psi_0(X_0^{\mathcal{I}_0^1}) + \cdots + \psi_0(X_0^{\mathcal{I}_0^N})\right)$$

of $\mathbb{E}\left[\psi_0(X_0)|\, Y_{0:T}\right]$ when $T \to \infty$. The variance of the quantity above is a sum of $\mathrm{Cov}(\psi_0(X_0^{\mathcal{I}_0^i}), \psi_0(X_0^{\mathcal{I}_0^j}))$ terms. It can therefore be understood by looking at a pair of trajectories simulated using Algorithm 2.

At final time $t = T$, $\mathcal{I}_T^1$ and $\mathcal{I}_T^2$ both follow the $\mathcal{M}(W_T^{1:N})$ distribution. Under regularity conditions (e.g. no extreme weights), they are likely to be different, i.e., $\mathbb{P}(\mathcal{I}_T^1 = \mathcal{I}_T^2) = \mathcal{O}(1/N)$. This property can be propagated backward: as long as $\mathcal{I}_t^1 \neq \mathcal{I}_t^2$, the two variables $\mathcal{I}_{t-1}^1$ and $\mathcal{I}_{t-1}^2$ are also likely to be different, with however a small $\mathcal{O}(1/N)$ chance of being equal. Moreover, as long as the two trajectories have not met, they can be simulated independently given $\mathcal{F}_T^-$ (the sigma algebra defined in (3)). In mathematical terms, under the two hypotheses of Theorem 1, given $\mathcal{F}_T^-$ and $\mathcal{I}_{t:T}^{1,2}$, it can be proved that the two variables $\mathcal{I}_{t-1}^1$ and $\mathcal{I}_{t-1}^2$ are independent if $\mathcal{I}_t^1 \neq \mathcal{I}_t^2$ (Lemma 2, Appendix E.3).

Since there is an $\mathcal{O}(1/N)$ chance of meeting at each time step, if $T \gg N$, it is likely that the two paths will meet at some point $t \gg 0$. When $\mathcal{I}_t^1 = \mathcal{I}_t^2$, the two indices $\mathcal{I}_{t-1}$ and $\mathcal{I}_{t-2}$ are both simulated according to $B_t^N(\mathcal{I}_t^1, \cdot)$. In the genealogy tracking algorithm, $B_t^{N,\mathrm{GT}}(i,\cdot)$ is a Dirac measure, leading to $\mathcal{I}_{t-1}^1 = \mathcal{I}_{t-1}^2$ almost surely. This spreads until time 0, so $\mathrm{Corr}(\psi_0(X_0^{\mathcal{I}_0^1}), \psi_0(X_0^{\mathcal{I}_0^2}))$ is almost 1 if $T \gg N$.

Other kernels like $B_t^{N,\mathrm{FFBS}}$ or $B_t^{N,\mathrm{PaRIS}}$ do not suffer from the same problem. For these, the support size of $B_t^N(\mathcal{I}_t^1, \cdot)$ is greater than one and thus there is some real chance that $\mathcal{I}_{t-1}^1 \neq \mathcal{I}_{t-1}^2$. If that does happen, we are again back to the regime where the next states of the two paths can be simulated independently. Note also that the support of $B_t^N(\mathcal{I}_t^1, \cdot)$ does not need to be large and can contain as few as 2 elements. Even if $\mathcal{I}_{t-1}^1$ might still be equal to $\mathcal{I}_{t-1}^2$ with some probability, the two paths will have new chances to diverge at times $t-2$, $t-3$ and so on. Overall, this makes $\mathrm{Corr}(\psi_0(X_0^{\mathcal{I}_0^1}), \psi_0(X_0^{\mathcal{I}_0^2}))$ quite small (Lemma 4, Appendix E.3).

We formalise these arguments in the following theorem, whose proof (Appendix E.3) follows them very closely. The price for proof intuitiveness is that the theorem is specific to the bootstrap filter, although numerical evidence (Section 5) suggests that other filters are stable as well.

Assumption 2. *The transition densities m_t are upper and lower bounded:*

$$\bar{M}_\ell \leq m_t(x_{t-1}, x_t) \leq \bar{M}_h$$

for constants $0 < \bar{M}_\ell < \bar{M}_h < \infty$.

Assumption 3. *The potential functions G_t are upper and lower bounded:*

$$\bar{G}_\ell \leq G_t(x_t) \leq \bar{G}_h$$

for constants $0 < \bar{G}_\ell < \bar{G}_h < \infty$.

Remark. Since Assumption 2 implies that the $\mathcal{X}_t$'s are compact, Assumption 1 automatically implies Assumption 3 as soon as the G_t's' are continuous functions.

Theorem 2. *We use the notations of Algorithms 1 and 2. Suppose that Assumptions 2 and 3 hold and the random kernels $B_{1:T}^N$ satisfy the conditions of Theorem 1. If, in addition, for the pair of random variables $(J_t^{n,1}, J_t^{n,2})$ whose distribution given $X_{t-1}^{1:N}$, X_t^n and $\hat{B}_t^N(n,\cdot)$ is defined by $B_t^N(n,\cdot) \otimes B_t^N(n,\cdot)$, we have*

$$\mathbb{P}\left(J_t^{n,1} \neq J_t^{n,2} \middle| X_{t-1}^{1:N}, X_t^n\right) \geq \varepsilon_{\mathrm{S}} \tag{13}$$

for some $\varepsilon_{\mathrm{S}} > 0$ and all t, n, m; then there exists a constant C not depending on T such that:

- *fixed marginal smoothing is stable, i.e. for $s \in \{0, \ldots, T\}$ and a real-valued function $\phi_s : \mathcal{X}_s \to \mathbb{R}$ of the hidden state X_s, we have*

$$\mathbb{E}\left[\left(\int \mathbb{Q}_T^N(\mathrm{d}x_s)\phi_s(x_s) - \mathbb{E}\left[\phi_s(X_s) \middle| Y_{0:T}\right]\right)^2\right] \leq \frac{C\left\|\phi_s\right\|_\infty^2}{N}; \tag{14}$$

- *additive smoothing is stable, i.e. for $T \geq 2$ and the function φ_T defined in (9), we have*

$$\mathbb{E}\left[\left(\mathbb{Q}_T^N(\varphi_T) - \mathbb{Q}_T(\varphi_T)\right)^2\right] \leq \frac{C\sum_{t=0}^T \left\|\psi_t\right\|_\infty^2}{N}\left(1 + \sqrt{\frac{T}{N}}\right)^2. \tag{15}$$

The $(1+\sqrt{T/N})^2$ term in (15) appeared in Dubarry and Le Corff (2013, Theorem 3.1) (which we used in our proof) and we do not know whether it can be dropped. However, it does not affect the scaling of the algorithm. Indeed, with or without it, the inequality implies that in order to have a constant error in the additive smoothing problem, one only has to take $N = \mathcal{O}(T)$ (instead of $N = \mathcal{O}(T^2)$ without backward sampling). This scaling has also been shown in Del Moral (2013, Chapter 17) in the case of the $B_t^{N,\mathrm{FFBS}}$ kernel. Moreover, from an asymptotic point of view, we always have $\sigma^2(T) = \mathcal{O}(T)$ regardless of the presence of the $(1+\sqrt{T/N})^2$ term, where

$$\sigma^2(T) := \lim_{N\to\infty} N\mathbb{E}\left[\left(\mathbb{Q}_T^N(\varphi_T) - \mathbb{Q}_T(\varphi_T)\right)^2\right].$$

Theorem 2 is stated under strong assumptions (similar to those used in Chopin and Papaspiliopoulos 2020, Chapter 11.4, and slightly stronger than Douc et al. 2011, Assumption 4). On the other hand, it applies to a large class of backward kernels (rather than only FFBS), including the new ones introduced in the forthcoming sections.

3. Sampling from the FFBS Backward Kernels

Sampling from the FFBS backward kernel lies at the heart of both the FFBS algorithm (Example 1) and the PaRIS one (Section 2.5). Indeed, at time t, they require generating random variables distributed according to $B_t^{N,\mathrm{FFBS}}(i_t, \mathrm{d}i_{t-1})$ for i_t running from 1 to N. Since sampling from a discrete measure on N elements requires $\mathcal{O}(N)$ operations (e.g. via CDF inversion), the total computational cost becomes $\mathcal{O}(N^2)$. To reduce this, we start by considering the subclass of models satisfying the following assumption, which is much weaker than Assumption 2.

Assumption 4. *The transition density $m_t(x_{t-1}, x_t)$ is strictly positive and upper bounded, i.e. there exists $\bar{M}_h > 0$ such that $0 < m_t(x_{t-1}, x_t) \leq \bar{M}_h, \forall\ (x_{t-1}, x_t)$.*

The motivation for the first condition $0 < m_t(x_{t-1}, x_t)$ will be clear after Assumption 5 is defined. For now, we see that it is possible to sample from $B_t^{N,\mathrm{FFBS}}(i_t, \mathrm{d}i_{t-1})$ using rejection sampling via the proposal distribution $\mathcal{M}(W_{t-1}^{1:N})$. After an $\mathcal{O}(N)$-cost initialisation, new draws can be simulated from the proposal in amortised $\mathcal{O}(1)$ time; see Chopin and Papaspiliopoulos (2020, Python Corner, Chapter 9), see also Douc et al. (2011, Appendix B.1) for an alternative algorithm with a $\mathcal{O}(\log N)$ cost per draw. The resulting procedure is summarised in Algorithm 4. Compared to traditional FFBS or PaRIS implementations, these rejection–based variants have a random execution time that is more difficult to analyse. Under Assumption 2, Douc et al. (2011) and Olsson and Westerborn (2017) derive an $\mathcal{O}(N\bar{M}_h/\bar{M}_\ell)$ expected complexity. However, the general picture, where the state space is not compact and only Assumption 4 holds, is less clear.

Algorithm 4: Pure rejection sampler for simulating from $B_t^{N,\mathrm{FFBS}}(i_t, \mathrm{d}i_{t-1})$

Input: Particles $X_{t-1}^{1:N}$ and weights $W_{t-1}^{1:N}$ at time $t-1$; particle $X_t^{i_t}$ at time t; constant $\bar{M}_h$; pre-initialised $\mathcal{O}(1)$ sampler for $\mathcal{M}(W_{t-1}^{1:N})$

repeat

  $\mathcal{I}_{t-1} \sim \mathcal{M}(W_{t-1}^{1:N})$ using the pre-initialised $\mathcal{O}(1)$ sampler

  $U \sim \mathrm{Unif}[0,1]$

until $U \leq m_t(X_{t-1}^{\mathcal{I}_{t-1}}, X_t^{i_t})/\bar{M}_h$

Output: $\mathcal{I}_{t-1}$, which is distributed according to $B_t^{N,\mathrm{FFBS}}(i_t, \mathrm{d}i_{t-1})$.

The present subsection intends to fill this gap. Our main focus is the PaRIS algorithm of which the presentation is simpler. Results for the FFBS algorithm can be found in Appendix B. We restrict ourselves to the case where $\mathcal{X}_t = \mathbb{R}^{d_t}$, although extensions to other non compact state spaces are possible. Only the bootstrap particle filter is considered, and results from this section do *not* extend trivially to other filtering algorithms. In Section 5, we shall employ different types of particle filters and see that the performance could change from one type to another, which is an additional weak point of rejection-based algorithms.

Assumption 5. *The hidden state X_t is defined on the space $\mathcal{X}_t = \mathbb{R}^{d_t}$. The measure $\lambda_t(\mathrm{d}x_t)$ with respect to which the transition density $m_t(x_{t-1}, x_t)$ is defined (cf. (2)) is the Lebesgue measure on $\mathbb{R}^{d_t}$.*

This assumption together with the condition $m_t(x_{t-1}, x_t) > 0$ of Assumption 4 ensures that the state space model is "truly non-compact". Indeed, if $m_t(x_{t-1}, x_t)$ is zero whenever $x_{t-1} \notin \mathcal{C}_{t-1}$ or $x_t \notin \mathcal{C}_t$, where $\mathcal{C}_{t-1}$ and $\mathcal{C}_t$ are respectively two compact subsets of $\mathbb{R}^{d_{t-1}}$ and $\mathbb{R}^{d_t}$, then we are basically reduced to a state space model where $\mathcal{X}_{t-1} = \mathcal{C}_{t-1}$ and $\mathcal{X}_t = \mathcal{C}_t$.

3.1. **Complexity of PaRIS algorithm with pure rejection sampling.** We consider the PaRIS algorithm (i.e. Algorithm 3 using the $B_t^{N,\mathrm{PaRIS}}$ kernels). Algorithm 5 provides a concrete description of the resulting procedure, using the

bootstrap particle filter. At each time t, let $\tau_t^{n,\mathrm{PaRIS}}$ be the number of rejection trials required to sample from $B_t^{N,\mathrm{FFBS}}(n, \mathrm{d}m)$. We then have

$$\tau_t^{n,\mathrm{PaRIS}} \mid \mathcal{F}_{t-1}, X_t^n \sim \mathrm{Geo}\left(\frac{\sum_i W_{t-1}^i m_t(X_{t-1}^i, X_t^n)}{\bar{M}_h}\right) \tag{16}$$

with $\bar{M}_h$ defined in Assumption 4.

Algorithm 5: Concrete implementation of PaRIS algorithm (i.e. Algorithm 3 with the $B_t^{N,\mathrm{PaRIS}}$ backward kernel) using the bootstrap particle filter

Input: Particles $X_{t-1}^{1:N}$; weights $W_{t-1}^{1:N}$; vector S_{t-1}^N in $\mathbb{R}^N$; pre-initialised sampler for $\mathcal{M}(W_{t-1}^{1:N})$; function ψ_t (cf. (9)); user-specified parameter $\tilde{N}$

for $n \leftarrow 1$ **to** N **do**

- $A_t^n \sim \mathcal{M}(W_{t-1}^{1:N})$ $(\star)$
- $X_t^n \sim M_t(X_{t-1}^{A_t^n}, \mathrm{d}x_t)$
- Simulate $J_t^{n,1:\tilde{N}} \overset{\text{i.i.d.}}{\sim} B_t^{N,\mathrm{FFBS}}(n, \mathrm{d}n')$ using either the pure rejection sampler (Algorithm 4) or the hybrid rejection sampler (Algorithm 6)
- $S_t^N[n] \leftarrow \tilde{N}^{-1} \sum_{\tilde{n}=1}^{\tilde{N}} \left\{ S_{t-1}^N[J_t^{n,\tilde{n}}] + \psi_t(X_{t-1}^{J_t^{n,\tilde{n}}}, X_t^n) \right\}$

for $n \leftarrow 1$ **to** N **do**

- $W_t^n \leftarrow G_t(X_t^n) / \sum_i G_t(X_t^i)$

$\mu_t^N \leftarrow \sum_{n=1}^N W_t^n S_t^N(n)$

Initialise a sampler for $\mathcal{M}(W_t^{1:N})$

Output: Estimate μ_t^N of $\mathbb{E}\left[\varphi(X_{0:t}) \middle| Y_{0:t}\right]$; particles $X_t^{1:N}$; weights $W_t^{1:N}$; vector S_t^N in $\mathbb{R}^N$ and pre-initialised sampler $\mathcal{M}(W_t^{1:N})$ for the next iteration

By exchangeability of particles, the expected cost of the PaRIS algorithm at step t is proportional to $N\tilde{N}\mathbb{E}[\tau_t^{1,\mathrm{PaRIS}}]$, where $\tilde{N}$ is a fixed user-chosen parameter. Occasionally, X_t^1 falls into an unlikely region of $\mathbb{R}^d$ and the acceptance rate becomes low. In other words, $\tau_t^{1,\mathrm{PaRIS}}$ is a mixture of geometric distribution, some components of which might have a large expectation. Unfortunately, these inefficiencies add up and produce an unbounded execution time in expectation, as shown in the following proposition.

Proposition 2. *Under Assumptions 4 and 5, the version of Algorithm 5 using the pure rejection sampler satisfies* $\mathbb{E}[\tau_t^{1,\mathrm{PaRIS}}] = \infty$, *where* $\tau_t^{1,\mathrm{PaRIS}}$ *is defined in* (16).

Proof. We have

$$\begin{aligned}
\mathbb{E}[\tau_t^{1,\mathrm{PaRIS}}] &= \bar{M}_h \mathbb{E}\left[\frac{1}{\sum_n m_t(X_{t-1}^n, X_t^1) W_{t-1}^n}\right] \quad \text{via (16)} \\
&= \bar{M}_h \mathbb{E}\left[\mathbb{E}\left[\frac{1}{\sum_n m_t(X_{t-1}^n, X_t^1) W_{t-1}^n} \middle| \mathcal{F}_{t-1}\right]\right] \\
&= \bar{M}_h \mathbb{E}\left[\int_{\mathcal{X}_t} \frac{1}{\sum_n m_t(X_{t-1}^n, x) W_{t-1}^n} \left(\sum m_t(X_{t-1}^n, x) W_{t-1}^n\right) \lambda_t(\mathrm{d}x)\right] \\
&= \bar{M}_h \mathbb{E}\left[\int_{\mathcal{X}_t} 1 \times \lambda_t(\mathrm{d}x)\right] = \infty \quad \text{by Assumption 5.}
\end{aligned}$$

□

As we mentioned in Subsection 1.2, a heavy-tailed execution time for each individual particle is not suitable for highly parallel environments, since a small number of processors might block the whole system. In a non-parallel setting, an execution time without expectation is essentially unpredictable. A common practice to estimate execution time is to run a certain algorithm with a small number N of particles, then "extrapolate" to the N_{final} of the definitive run. However, as $\mathbb{E}[\tau_t^{1,\mathrm{PaRIS}}]$ is infinite for any N, it is unclear what kind of information we might get from preliminary runs. In Appendix B, besides studying the execution time of rejection-based implementations of the FFBS algorithm, we will delve deeper into the difference between the non-parallel and parallel computing.

From the proof of Proposition 2, it is clear that the quantity $\sum_n W_{t-1}^n m_t(X_{t-1}^n, x_t)$ will play a key role in the upcoming developments. We thus define it formally.

Definition 1. *The true predictive density function r_t and its approximation r_t^N are defined as*

$$\begin{aligned}
r_t(x_t) &:= \frac{(\mathbb{Q}_{t-1} M_t)(\mathrm{d}x_t)}{\lambda_t(\mathrm{d}x_t)} \\
r_t^N(x_t) &:= \sum W_{t-1}^n m_t(X_{t-1}^n, x_t)
\end{aligned}$$

where the first equation is understood in the sense of the Radon-Nikodym derivative and the density $m_{t-1}(x_{t-1}, x_t)$ is defined with respect to the dominating measure $\lambda_t(\mathrm{d}x_t)$ on $\mathcal{X}_t$ (cf. (2)).

3.2. **Hybrid rejection sampling.** To solve the aforementioned issues of the pure rejection sampling procedure, we propose a hybrid rejection sampling scheme. The basic observation is that, for a single m, direct simulation (e.g. via CDF inversion) of $B_t^{N,\mathrm{FFBS}}(i_t, \mathrm{d}i_{t-1})$ costs $\mathcal{O}(N)$. Thus, once $K = \mathcal{O}(N)$ rejection sampling trials have been attempted, one should instead switch to a direct simulation method. In other words, it does not make sense (at least asymptotically) to switch to direct sampling after K trials if $K \ll \mathcal{O}(N)$ or $K \gg \mathcal{O}(N)$. The validity of this method is established in the following proposition, where we actually allow K to depend on trails drawn so far. The proof, which is *not* an immediate consequence of the validity of ordinary rejection sampling, is given in Appendix E.4.

Proposition 3. *Let $\mu_0(x)$ and $\mu_1(x)$ be two probability densities defined on some measurable space $\mathcal{X}$ with respect to a dominating measure $\lambda(\mathrm{d}x)$. Suppose that there exists $C > 0$ such that $\mu_1(x) \le C\mu_0(x)$. Let $(X_1, U_1), (X_2, U_2), \ldots$ be a sequence of*

i.i.d. random variables distributed according to $\mu_0 \otimes \mathrm{Unif}[0,1]$ *and let* $X^* \sim \mu_1$ *be independent of that sequence. Put*

$$K^* := \inf\left\{n \in \mathbb{Z}_{\geq 1} \text{ such that } U_n \leq \frac{\mu_1(X_n)}{C\mu_0(X_n)}\right\}$$

and let K *be* any *stopping time with respect to the natural filtration associated with the sequence* $\{(X_n, U_n)\}_{n=1}^{\infty}$. *Let* Z *be defined as*

$$Z := \begin{cases} X_{K^*} & \text{if } K^* \leq K \\ X^* & \text{otherwise.} \end{cases}$$

Then Z *is* μ_1*-distributed.*

Proposition 3 thus allows users to pick $K = \alpha N$, where $\alpha > 0$ might be chosen somehow adaptively from earlier trials. In the following, we only consider the simple rule $K = N$, which does not induce any loss of generality in terms of the asymptotic behaviour and is easy to implement. The resulting iteration is described in Algorithm 6.

Algorithm 6: Hybrid rejection sampler for simulating from $B_t^{N,\mathrm{FFBS}}(i_t, \mathrm{d}i_{t-1})$

Input: Particles $X_{t-1}^{1:N}$ and weights $W_{t-1}^{1:N}$ at time $t-1$; particle $X_t^{i_t}$ at time t; constant $\bar{M}_h$; pre-initialised $\mathcal{O}(1)$ sampler for $\mathcal{M}(W_{t-1}^{1:N})$

accepted $\leftarrow$ False

for $i \leftarrow 1$ **to** N **do**

  $\mathcal{I}_{t-1} \sim \mathcal{M}(W_{t-1}^{1:N})$ using the pre-initialised $\mathcal{O}(1)$ sampler

  $U \sim \mathrm{Unif}[0,1]$

  if $U \leq m_t(X_{t-1}^{\mathcal{I}_{t-1}}, X_t^{i_t})/\bar{M}_h$ **then**

    accepted $\leftarrow$ True

    break

if not *accepted* **then**

  $\mathcal{I}_{t-1} \sim \mathcal{M}(W_{t-1}^n m(X_{t-1}^n, X_t^{i_t}))$

Output: $\mathcal{I}_{t-1}$, which is distributed according to $B_t^{N,\mathrm{FFBS}}(i_t, \mathrm{d}i_{t-1})$.

When applied in the context of Algorithm 5, Algorithm 6 gives a smoother of expected complexity proportional to

$$N\tilde{N}\mathbb{E}[\min(\tau_t^{1,\mathrm{PaRIS}})]$$

at time t, where $\tau_t^{1,\mathrm{PaRIS}}$ is defined in (16)). This quantity is no longer infinite, but its growth when $N \to \infty$ might depend on the model. Still, in all cases, it remains strictly larger than $\mathcal{O}(N)$ and strictly smaller than $\mathcal{O}(N^2)$. Perhaps more surprisingly, in linear Gaussian models (see Appendix A.1 for detailed notations), the smoother is of near-linear complexity (up to log factors). The following two theorems formalise these claims.

Assumption 6. *The predictive density* r_t *of* X_t *given* $Y_{0:t-1}$ *and the potential function* G_t *are continuous functions on* $\mathbb{R}^{d_t}$. *The transition density* $m_t(x_{t-1}, x_t)$ *is a continuous function on* $\mathbb{R}^{d_{t-1}} \times \mathbb{R}^{d_t}$.

Theorem 3. *Under Assumptions 1, 4, 5 and 6, the version of Algorithm 5 using the hybrid rejection sampler (Algorithm 6) satisfies* $\lim_{N\to\infty} \mathbb{E}[\min(\tau_t^{1,\mathrm{PaRIS}}, N)] = \infty$ *and* $\lim_{N\to\infty} \mathbb{E}[\min(\tau_t^{1,\mathrm{PaRIS}}, N)]/N = 0$*, where* $\tau_t^{1,\mathrm{PaRIS}}$ *is defined in* (16).

Theorem 4. *We assume the same setting as Theorem 3. In linear Gaussian state space models (Appendix A.1), we have* $\mathbb{E}[\min(\tau_t^{1,\mathrm{PaRIS}}, N)] = \mathcal{O}((\log N)^{d_t/2})$.

4. Alternative backward kernels

4.1. MCMC Backward Kernels.

This subsection analyses and extends the MCMC backward kernel defined in Example 3. As we remarked there, the matrix $\hat{B}_t^{N,\mathrm{IMH}}$ is not sparse and even has some expensive-to-evaluate entries. We thus reserve it for use in the off-line smoother (Algorithm 2) whereas in the on-line scenario (Algorithm 3), we use its PaRIS-like counterpart

$$\hat{B}_t^{N,\mathrm{IMHP}}[i_t, i_{t-1}] := \frac{1}{\tilde{N}} \sum_{\tilde{n}=1}^{\tilde{N}} \mathbb{1}\left\{i_{t-1} = \tilde{J}_t^{i_t,\tilde{n}}\right\} \tag{17}$$

where $\tilde{J}_t^{i_t,1:\tilde{N}}$ is an independent Metropolis-Hastings chain started at $J_t^{i_t,1} := A_t^{i_t}$, targeting the measure $B_t^{N,\mathrm{FFBS}}(i_t, \mathrm{d}i_{t-1})$ and using the proposal distribution $\mathcal{M}(W_{t-1}^{1:N})$. The validity and the stability of $\hat{B}_t^{N,\mathrm{IMH}}$ and $\hat{B}_t^{N,\mathrm{IMHP}}$ are established in the following simple proposition (proved in Appendix E.9). For simplicity, only the case $\tilde{N} = 2$ is examined, but as a matter of fact, the proposition remains true for $\tilde{N} \geq 2$.

Proposition 4. *The kernels* $B_t^{N,\mathrm{IMH}}$ *and* $B_t^{N,\mathrm{IMHP}}$ *with* $\tilde{N} = 2$ *satisfy the hypotheses of Theorem 1 and, under Assumptions 2 and 3, those of Theorem 2. Hence, their respective uses in Algorithms 2 and 3 guarantee a convergent and stable smoother.*

The advantages of independent Metropolis-Hastings MCMC kernels compared to the rejection samplers of Section 3 are the dispensability of specifying an explicit $\bar{M}_h$ and the deterministic nature of the execution time. It is not hard to imagine situations where some proposal smarter than $\mathcal{M}(W_{t-1}^{1:N})$ would be beneficial. However, we only consider that one here, mainly because it already performs satisfactorily in our numerical examples.

4.2. Dealing with intractable transition densities.

4.2.1. *Intuition and formulation.* The purpose of backward sampling is to re-generate, for each particle, a new ancestor that is different from that of the filtering step. However, backward sampling is infeasible if the transition density $m_t(x_{t-1}, x_t)$ cannot be calculated. To get around this, we modify the particle filter so that each particle might, in some sense, have two ancestors right from the *forward* pass.

Consider the standard PF (Algorithm 1). Among the N resampled particles $X_{t-1}^{A_t^{1:N}}$, let us track two of them, say x_{t-1} and x'_{t-1} for simplicity. The move step of Algorithm 1 will push them through M_t using independent noises, resulting in x_t and x'_t (that is, given x_{t-1} and x'_{t-1}, we have $x_t \sim M_t(x_{t-1}, \cdot)$ and $x'_t \sim M_t(x'_{t-1}, \cdot)$ such that x_t and x'_t are independent). Thus, for e.g. linear Gaussian models, we have $\mathbb{P}(x_t = x'_t) = 0$. However, if the two simulations $x_t \sim M_t(x_{t-1}, \cdot)$ and $x'_t \sim M_t(x'_{t-1}, \cdot)$ are done with specifically correlated noises, it can happen that $\mathbb{P}(x_t = x'_t) > 0$. The joint distribution (x_t, x'_t) given (x_{t-1}, x'_{t-1}) is called a *coupling*

of $M_t(x_{t-1}, \cdot)$ and $M_t(x'_{t-1}, \cdot)$; the event $x_t = x'_t$ is called the *meeting* event and we say that the coupling is *successful* when it occurs. In that case, the particle x_t automatically has two ancestors x_{t-1} and x'_{t-1} at time $t-1$ without needing any to perform a backward sampling step.

The precise formulation of the modified forward pass is detailed in Algorithm 7. It consists of building in an on-line manner the backward kernels $B_t^{N,\mathrm{ITR}}$ (where ITR stands for "intractable"). The main interest of this algorithm lies in the fact that while the function m_t may prove impossible to evaluate, it is usually possible to make x_t and x'_t meet by correlating somehow the random numbers used in their simulations. One typical example which this article focuses on is the coupling of continuous-time processes, but it is useful to keep in mind that Algorithm 7 is conceptually more general than that.

Algorithm 7: Modified forward pass for smoothing of intractable models (one time step)

Input: Feynman-Kac model (1), particles $X_{t-1}^{1:N}$ and weights $W_{t-1}^{1:N}$ that approximate the filtering distribution at time $t-1$

for $n \leftarrow 1$ **to** N **do**

Resample. Simulate $(A_t^{n,1}, A_t^{n,2})$ such that marginally each component is distributed according to $\mathcal{M}(W_{t-1}^{1:N})$

Move. Simulate $(X_t^{n,1}, X_t^{n,2})$ such that marginally the two components are distributed respectively according to $M_t(X_{t-1}^{A_t^{n,1}}, \mathrm{d}x_t)$ and $M_t(X_{t-1}^{A_t^{n,2}}, \mathrm{d}x_t)$

Choose $L \sim \mathrm{Uniform}(\{1, 2\})$

Set $X_t^n \leftarrow X_t^{n,L}$

Calculate backward kernel.

if $X_t^{n,1} = X_t^{n,2}$ **then**

$B_t^{N,\mathrm{ITR}}(n, \mathrm{d}i_{t-1}) \leftarrow \left(\delta\left\{A_t^{n,1}\right\} + \delta\left\{A_t^{n,2}\right\}\right)/2$

else

$B_t^{N,\mathrm{ITR}}(n, \mathrm{d}i_{t-1}) \leftarrow \delta\left\{A_t^{n,L}\right\}$

Reweight. Set $\omega_t^n \leftarrow G_t(X_t^n)$ for $n = 1, 2, \ldots, N$

Set $\ell_t^N \leftarrow \sum_{n=1}^N \omega_t^n / N$

Set $W_t^n \leftarrow \omega_t^n / N\ell_t^N$ for $n = 1, 2, \ldots, N$

Output: Particles $X_t^{1:N}$ and weights $W_t^{1:N}$ that approximate the filtering distribution at time t; backward kernel $B_t^{N,\mathrm{ITR}}$ that can be used in either Algorithm 2 or 3

4.2.2. *Validity and stability.* The consistency of Algorithm 7 follows straightforwardly from Theorem 1. To produce a stable routine however, some conditions must be imposed on the couplings $(A_t^{n,1}, A_t^{n,2})$ and $(X_t^{n,1}, X_t^{n,2})$. We want $A_t^{n,1}$ to be different from $A_t^{n,2}$ as frequently as possible. On the contrary, we aim for a coupling of the two distributions $M_t(X_{t-1}^{A_t^{n,1}}, \cdot)$ and $M_t(X_{t-1}^{A_t^{n,2}}, \cdot)$ with high success rate so as to maximise the probability that $X_t^{n,1} = X_t^{n,2}$.

Assumption 7. *There exists an* $\varepsilon_A > 0$ *such that*

$$\mathbb{P}(A_t^{n,1} \neq A_t^{n,2} | X_{t-1}^{1:N}) \geq \varepsilon_A.$$

Assumption 8. *There exists an* $\varepsilon_D > 0$ *such that*

$$\mathbb{P}(X_t^{n,2} = X_t^{n,1} | X_{t-1}^{1:N}, A_t^{n,1}, A_t^{n,2}, X_t^{n,1}) \geq \varepsilon_D \left(1 \wedge \frac{m_t(X_{t-1}^{A_t^{n,2}}, X_t^{n,1})}{m_t(X_{t-1}^{A_t^{n,1}}, X_t^{n,1})}\right).$$

The letters A and D in ε_A and ε_D stand for "ancestors" and "dynamics". Assumption 8 means that the user-chosen coupling of $M_t(X_{t-1}^{A_t^{n,1}}, \cdot)$ and $M_t(X_{t-1}^{A_t^{n,2}}, \cdot)$ must be at least as ε_D times as efficient as their maximal couplings. For details on this interpretation, see Proposition 10 in the Appendix. In Lemma 13, we also show that in spite of its appearance, Assumption 8 is actually symmetric with regards to $X_t^{n,1}$ and $X_t^{n,2}$.

We are now ready to state the main theorem of this subsection (see Appendix E.11 for a proof).

Theorem 5. *The kernels* $B_t^{N,\mathrm{ITR}}$ *generated by Algorithm 7 satisfy the hypotheses of Theorem 1. Thus, under Assumption 1, Algorithm 7 provides a consistent smoothing estimate. If, in addition, the Feynman-Kac model* (1) *satisfies Assumptions 2 and 3 and the user-chosen couplings satisfy Assumptions 7 and 8, the kernels* $B_t^{N,\mathrm{ITR}}$ *also fulfil* (13) *and the smoothing estimates generated by Algorithm 7 are stable.*

4.2.3. *Good ancestor couplings.* It is notable that Assumption 7 only considers the event $A_t^{n,1} \neq A_t^{n,2}$, which is a pure index condition that does not take into account the underlying particles $X_{t-1}^{A_t^{n,1}}$ and $X_{t-1}^{A_t^{n,2}}$. Indeed, if smoothing algorithms prevent degeneracy by creating multiple ancestors for a particle, we would expect that their separation (i.e. that they are far away in the state space $\mathcal{X}_{t-1}$, e.g. $\mathbb{R}^d$) is critical to the performance. Surprisingly, it is unnecessary: two very close particles (in $\mathbb{R}^d$) at time $t-1$ may have ancestors far away at time $t-2$ thanks to the mixing of the model.

We advise choosing an ancestor coupling $(A_t^{n,1}, A_t^{n,2})$ such that the distance between $X_{t-1}^{A_t^{n,1}}$ and $X_{t-1}^{A_t^{n,2}}$ is small. It will then be easier to design a dynamic coupling of $M_t(X_{t-1}^{A_t^{n,1}}, \cdot)$ and $M_t(X_{t-1}^{A_t^{n,2}}, \cdot)$ with a high success rate. Furthermore, simulating the dynamic coupling with two close rather than far away starting points can also take less time when, for instance, the dynamic involves multiple intermediate steps, but the two processes couple early. One way to achieve an ancestor coupling with the aforementioned property is to first simulate $A_t^{n,1} \sim \mathcal{M}(W_{t-1}^{1:N})$, then move $A_t^{n,1}$ through an MCMC algorithm which keeps invariant $\mathcal{M}(W_{t-1}^{1:N})$ and set the result to $A_t^{n,2}$. It suffices to use a proposal looking at indices whose underlying particles are close (in $\mathbb{R}^d$) to $X_{t-1}^{A_t^{n,1}}$. Finding nearby particles are efficient if they are first sorted using the Hilbert curve, hashed using locality-sensitive hashing or put in a KD-tree (see Samet, 2006, for a comprehensive review). In the context of particle filters, such techniques have been studied for different purposes in Gerber and Chopin (2015), Jacob et al. (2019) and Sen et al. (2018).

4.2.4. *Conditionally-correlated version.* In Algorithm 7, the ancestor pairs $(A_t^{n,1}, A_t^{n,2})_{n=1}^N$ are conditionally independent given $\mathcal{F}_t^-$ and the same holds for the particles $(X_t^n)_{n=1}^N$.

These conditional independences allow easier theoretical analysis, in particular, the casting of Algorithm 7 in the framework of Theorems 1 and 2. However, they are not optimal for performance in two important ways: (a) they do not allow keeping both $X_t^{n,1}$ and $X_t^{n,2}$ when the two are not equal, and (b) the set of ancestor variables $(A_t^{n,1})_{n=1}^N$ is multinomially resampled from $\{1, 2, \ldots, N\}$ with weights $W_{t-1}^{1:N}$. We know that multinomial resampling is not the ideal scheme, see Appendix C.1 for discussion.

Consequently, in practice, we shall allow ourselves to break free from conditional independence. The resulting procedure is described in Algorithm 9 (Appendix C). Despite a lack of rigorous theoretical support, this is the algorithm that we will use in Section 5 since it enjoys better performance and it constitutes a fair comparison with practical implementations of the standard particle filter, which are mostly based on alternative resampling schemes.

5. Numerical experiments

5.1. **Linear Gaussian state-space models.** Linear Gaussian models constitute a particular class of state space models. They are characterised by Markov dynamics that are Gaussian and observations that are projection of hidden states plus some Gaussian noises. Appendix A.1 defines, for different components of these models, the notations that we shall use here. In this section, we consider an instance described in Guarniero et al. (2017), where the matrix F_X satisfies $F_X[i, j] = \alpha^{1+|i-j|}$ for some α. We consider the problem with $\dim_X = \dim_Y = 2$ and the observations are noisy versions of the hidden states with C_Y being σ_Y^2 times the identity matrix of size 2. Unless otherwise specified, we take $\alpha = 0.4$ and $\sigma_Y^2 = 0.5$.

In this section, we focus on the performance of different online smoothers based on either genealogy tracking, pure/hybrid rejection sampling or MCMC. Rejection-based online smoothing amounts to the PaRIS algorithm, for which we use $\tilde{N} = 2$ for the $B_t^{N,\mathrm{PaRIS}}$ kernel. We take $T = 3000$ and simulate the data from the model. The benchmark additive function is simply

$$\varphi_t(x_{0:t}) = \sum_{s=0}^{t} x_s(0) \tag{18}$$

where $x_s(0)$ is the first coordinate of the $\mathbb{R}^2$ vector $x_s = [x_s(0), x_s(1)]$. For a study of offline smoothers (including FFBS), see Appendix D.1. In all programs here and there, we choose $N = 1000$ and use systematic resampling for the forward particle filters (see section C.1). Regarding MCMC smoothers, we employ the kernels $B_t^{N,\mathrm{IMH}}$ or $B_t^{N,\mathrm{IMHP}}$ consisting of only one MCMC step. All results are based on 150 independent runs.

Although our theoretical results are only proved for the bootstrap filter, we stress throughout that some of them extend to other filters as well. Therefore, we will also consider guided particle filters in the simulations. An introduction to this topic can be found in Chopin and Papaspiliopoulos (2020, Chapter 10.3.2), where the expression for the optimal proposal is also provided. In linear Gaussian models, this proposal is fully tractable and is the one we use.

To present efficiently the combination of two different filters (bootstrap and guided) and four different algorithms (naive genealogy tracking, pure/hybrid rejection and MCMC) we use the following abbreviations: “B” for bootstrap, “G” for guided, “N” for naive genealogy tracking, “P” for pure rejection, “H” for hybrid

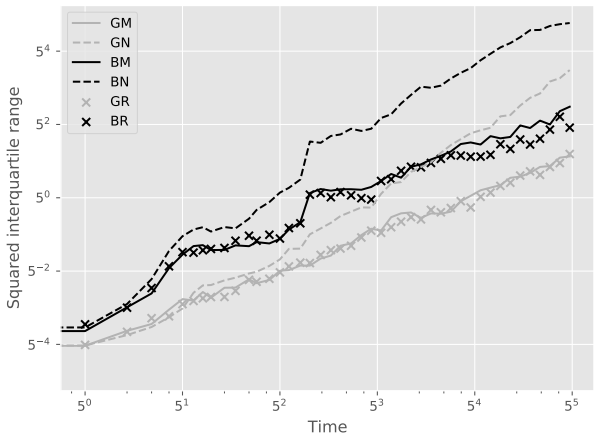

FIGURE 2. Squared interquartile range of the estimators $\mathbb{Q}_t(\varphi_t)$ with respect to t, for different online smoothing algorithms. The model is linear Gaussian with parameters specified in section 5.1. See text for full explanation of the legend. For readability, the curves are down-sampled to 50 points before being drawn.

rejection and "M" for MCMC. For instance, the algorithm referred to as "BM" uses the bootstrap filter for the forward pass and the MCMC backward kernels to perform smoothing. Furthermore, the letter "R" will refer to the rejection kernel whenever the distinction between pure rejection and hybrid rejection is not necessary. (Recall that the two rejection methods produce estimators with the same distribution.)

Figure 2 shows the squared interquartile range for the online smoothing estimates $\mathbb{Q}_t(\varphi_t)$ with respect to t. It verifies the rates of Theorem 2, although linear Gaussian models are not strongly mixing in the sense of Assumptions 2 and 3: the grid lines hint at a variance growth rate of $\mathcal{O}(T)$ for the MCMC and reject-based smoothers and of $\mathcal{O}(T^2)$ for the genealogy tracking ones. Unsurprisingly guided filters have better performance than bootstrap.

Figure 3 (and its zoomed-in, Figure 4) show box-plots of the execution time (divided by NT) for different algorithms over 150 runs. By execution time, we mean the number of Markov kernel transition density evaluations. We see that the bootstrap particle filter coupled with pure rejection sampling has a very heavy-tailed execution time. This behaviour is expected as per Proposition 2. Using the guided particle filter seems to fare better, but Figure 5 (for the same model but with $\sigma_Y^2 = 2$) makes it clear that this cannot be relied on either. Overall, these results highlight two fundamental problems with pure rejection sampling: the computational time has heavy tails and depends on the type of forward particle filter being used.

On the other hand, hybrid rejection sampling, despite having random execution time in principle, displays a very consistent number of transition density evaluations over different independent runs. Thus it is safe to say that the algorithm has a virtually deterministic execution time. The catch is that the average computational load (which is around 16 in Figure 4) cannot be easily calculated beforehand. In any

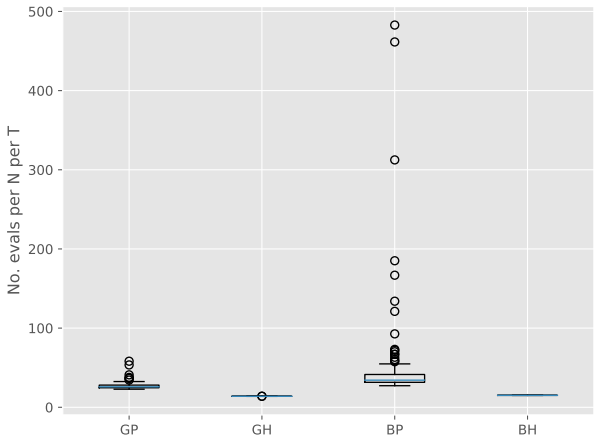

FIGURE 3. Box plots (based on 150 runs) of averaged execution times (numbers of transition density evaluations divided by NT) for different algorithms on the linear Gaussian model of section 5.1.

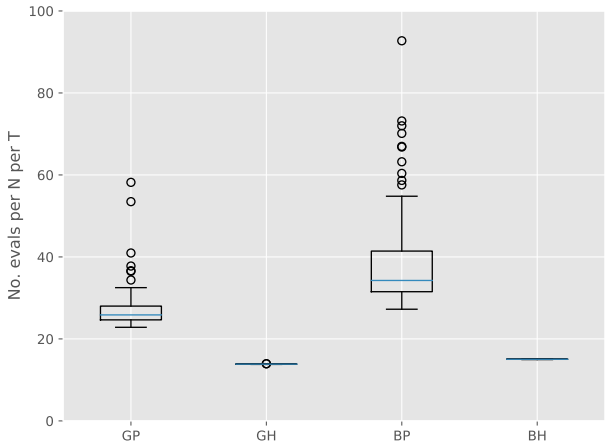

FIGURE 4. Zoomed-in of Figure 3.

case, it is much larger than the value 1 of MCMC smoothers (since only 1 MCMC step is performed in the kernel $B_t^{N,\mathrm{IMHP}}$); whereas the performance (Figure 2) is comparable.

The bottom line is that MCMC smoothers should be the default option, and one MCMC step seems to be enough. If for some reason one would like to use rejection-based methods, using hybrid rejection is a must.

5.2. **Lotka-Volterra SDE.** Lotka-Voleterra models (originated in Lotka, 1925 and Volterra, 1928) describe the population fluctuation of species due to natural birth and death as well as the consumption of one species by others. The emblematic case of two species is also known as the predator-prey model. In this subsection, we study the stochastic differential equation (SDE) version that appears in Hening

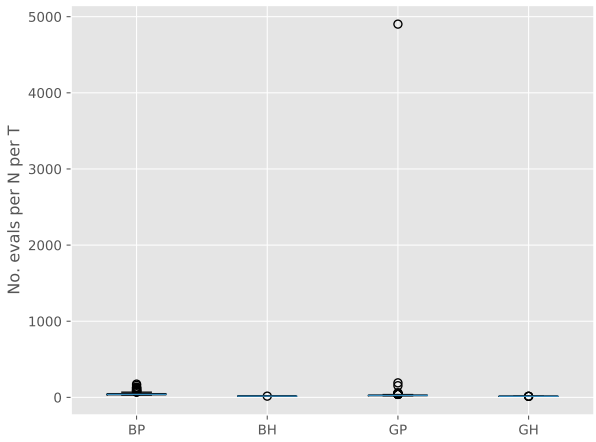

FIGURE 5. Same as Figure 3, but for the modified model where $\sigma_Y^2 = 2$.

and Nguyen (2018). Let $X_t = (X_t(0), X_t(1))$ represent respectively the populations of the prey and the predator at time t and let us consider the dynamics

$$(19) \qquad \begin{cases} \mathrm{d}X_t(0) = \big[\beta_0 X_t(0) - \frac{1}{2}\tau_0[X_t(0)]^2 \quad - \tau_1 X_t(0)X_t(1)\big]\mathrm{d}t + X_t(0)\mathrm{d}E_t(0) \\ \mathrm{d}X_t(1) = \big[\qquad -\beta_1 X_t(1) \qquad\quad + \tau_1 X_t(0)X_t(1)\big]\mathrm{d}t + X_t(1)\mathrm{d}E_t(1) \end{cases}$$

where $E_t = \Gamma W_t$ with W_t being the standard Brownian motion in $\mathbb{R}^2$ and Γ being some 2×2 matrix. The parameters β_0 and β_1 are the natural birth rate of the prey and death rate of the predator. The predator interacts with (eats) the prey at rate τ_1. The quantity τ_0 encodes intra-species competition in the prey population. The $\frac{1}{2}$ in its parametrisation is to line up with the Lotka Volterra jump process in $\mathbb{Z}^2$ (to be included in a future version of this paper), where the population sizes are integers and the interaction term becomes $\tau_0 X_t(0)[X_t(0) - 1]/2$.

The state space model is comprised of the process X_t and its noisy observations Y_t recorded at integer times. The Markov dynamics cannot be simulated exactly, but can be approximated through (Euler) discretisation. Nevertheless, the Euler transition density $m_t^{\mathrm{E}}(x_{t-1}, x_t)$ remains intractable (unless the step size is exactly 1). Thus, the algorithms presented in Subsection 4.2 are useful. The missing bit is a method to efficiently couple $m_t^{\mathrm{E}}(x_{t-1}, \cdot)$ and $m_t^{\mathrm{E}}(x'_{t-1}, \cdot)$, which we carefully describe in Appendix D.2.1.

We consider the model with $\tau_0 = 1/800$, $\tau_1 = 1/400$, $\beta_0 = 0.3125$ and $\beta_1 = 0.25$. The matrix Γ is such that the covariance matrix of E_1 is $\begin{bmatrix} 1/100 & 1/200 \\ 1/200 & 1/100 \end{bmatrix}$. The observations are recorded on the log scale with Gaussian error of covariance matrix $\begin{bmatrix} 0.04 & 0.02 \\ 0.02 & 0.04 \end{bmatrix}$. The distribution of X_0 is two-dimensional normal with mean $[100, 100]$ and covariance matrix $\begin{bmatrix} 100 & 50 \\ 50 & 100 \end{bmatrix}$. This choice is motivated by the fact

that the preceding parameters give the stationary population vector $[100, 100]$. According to Hening and Nguyen (2018), they also guarantee that neither animal goes extinct almost surely as $t \to \infty$.

By discretising (19) with time step $\delta = 1$, one can get some very rough intuition on the dynamics. For instance, per second there are about 31 preys born. Approximately the same number die (to maintain equilibrium), of which 6 die due to internal competition and 25 are eaten by the predator. The duration between two recorded observations corresponds more or less to one-third generation of the prey and one-fourth generation of the predator. The standard deviation of the variation due to environmental noise is about 10 individuals per observation period, for each animal.

Again, these intuitions are highly approximate. For readers wishing to get more familiar with the model, Appendix D.2.2 contains real plots of the states and the observations; as well as data on the performance of different smoothing algorithms for moderate values of T. We now showcase the results obtained in a large scale problem where $T = 3000$ and the data is simulated from the model.

We consider the additive function

$$\varphi_t(x_{0:t}) := \sum_{s=0}^{t} [x_s(0) - 100] \,.$$

Figure 6 represents using box plots the distributions of the estimators for $\mathbb{Q}_T(\varphi_T)$ using either the genealogy tracking smoother (with systematic resampling; see Appendix C.1) or Algorithm 9. Our proposed smoother greatly reduces the variance, at a computational cost which is empirically 1.5 to 2 times greater than the naive method. Since we used Hilbert curve to design good ancestor couplings (see Section 4.2.3), coupling of the dynamics succeeds 80% of the time. As discussed in the aforementioned section, starting two diffusion dynamics from nearby points make them couple earlier, which reduces the computational load afterwards.

Figure 7 plots with respect to t the squared interquartile range of the two methods for the estimation of $\mathbb{Q}_t(\varphi_t)$. Grid lines hint at a quadratic growth for the genealogy tracking smoother (as analysed in Olsson and Westerborn, 2017, Sect. 1) and a linear growth for the kernel $B_t^{N,\mathrm{ITRC}}$ (as described in Theorem 2).

Finally, Figure 21 (Appendix D.2.2) shows properties of the effective sample size (ESS) ratio for this model. In a nutshell, while being globally stable (between 40% and 70%), it has a tendency to drift towards near 0 from time to time due to unusual data points. At these moments, resampling kills most of the particles and aggravates the degeneracy problem for the naive smoother. As we have seen in the above figures, systematic resampling is not enough to mitigate this in the long run.

6. Conclusion

6.1. **Practical recommendations.** Our first recommendation does not concern the smoothing algorithm per se. It is of paramount importance that the particle filter used in in the preliminary filtering step performs reasonably well, since its output defines the support of the approximations generated by the subsequent smoothing algorithm. (Standard recommendations to obtain good performance from a particle filter are to increase N, or to use better proposal distributions, or both.)

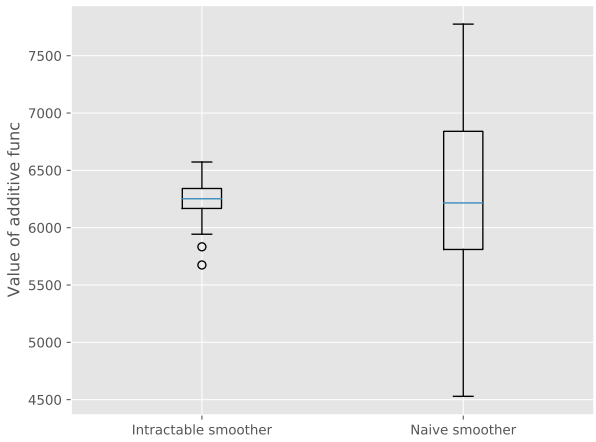

FIGURE 6. Box plot of estimators (over 50 independent runs with $N = 1000$ particles) for $\mathbb{Q}_T(\varphi_T)$ in the Lotka-Volterra SDE model with $T = 3000$. They are calculated using either the naive genealogy tracking smoother or our smoother developed for intractable models (Algorithm 9).

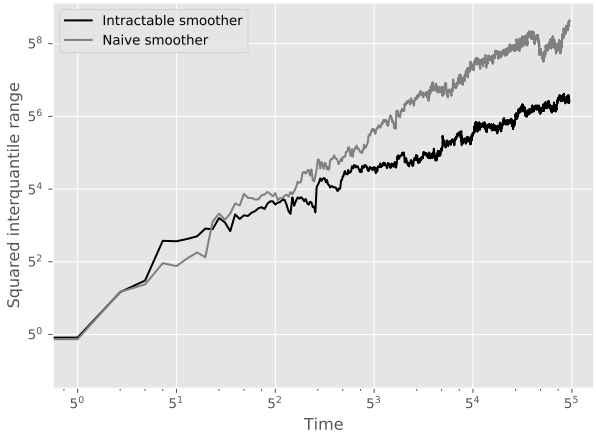

FIGURE 7. Squared interquartile range for the genealogy tracking smoother and our proposed one. Same context as in Figure 6

Then, if the transition density is tractable, we recommend the MCMC smoother by default. One to two MCMC steps seem to be sufficient in most situations. If one still prefers to use rejection sampling (see below for a possible motivation), it is safe to say that there is no reason not to use the hybrid method.

Although the assumptions under which we prove the stability of the smoothing estimates are strong, the general message still holds. The Markov kernel and the potential functions must make the model forget its past in some ways. Otherwise, we get an unstable model for which no smoothing methods can work. The rejection

sampling – based smoothing algorithms can therefore serve as the ultimate test. Since they simulate exactly independent trajectories given the skeleton, there is no hope to perform better, unless one switches to another family of smoothing algorithms.

For intractable models, the key issue is to design couplings with high meeting probability. Fortunately, the inherent chaos of the model makes it possible to choose two very close starting points for the dynamics and thus easy to obtain a reasonable meeting probability.

If further difficulties persist, there is a practical (and very heuristic) recipe to test whether one coupling of $M_t(x, \cdot)$ and $M_t(x', \cdot)$ is close to optimal. It consists in approximating $M_t(x, \cdot)$ and $M_t(x', \cdot)$ by Gaussian distributions and deduce the optimal coupling rate from their total variation distance. There is no closed formula for the total variation distance between two Gaussian distributions in high dimensions. However, it can be reliably estimated using the geometric interpretation of the total variation distance being one minus the area of the intersection created by the corresponding density graphs. In this way, one can get a very rough idea of to what extent a certain coupling realises the meeting potential that the two distributions have. If the coupling seems good and the trajectories still look degenerate, it can very well be that the model is unstable.

6.2. **Further directions.** The major limitation of our work is the exclusive theoretical analysis under the bootstrap particle filter. Moreover, we require that the N new particles generated at step t are conditionally independent given previous particles at time $t-1$. This excludes practical optimisations like systematic resampling and Algorithm 9.

Finally, the backward sampling step is also used in other algorithms (in particular Particle Markov Chain Monte Carlo, see Andrieu et al., 2010) and it would be interesting to see to what extent our techniques can be applied there.

6.3. **Data and code.** The code used to run numerical experiments is available at https://github.com/hai-dang-dau/backward-samplers-code. The algorithms will soon be implemented in the `particles` package at https://github.com/nchopin/particles.

Acknowledgement

The first author acknowledges a CREST PhD scholarship via AMX funding.

References

Andrieu, C., Doucet, A., and Holenstein, R. (2010). Particle Markov chain Monte Carlo methods. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 72(3):269–342.

Beskos, A., Papaspiliopoulos, O., Roberts, G. O., and Fearnhead, P. (2006). Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processes. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 68(3):333–382. With discussions and a reply by the authors.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Information Science and Statistics. Springer, New York.

Bou-Rabee, N., Eberle, A., and Zimmer, R. (2020). Coupling and convergence for Hamiltonian Monte Carlo. *The Annals of Applied Probability*, 30(3):1209 – 1250.

Bunch, P. and Godsill, S. (2013). Improved particle approximations to the joint smoothing distribution using Markov chain Monte Carlo. *IEEE Transactions on Signal Processing*, 61(4):956–963.

Carpenter, J., Clifford, P., and Fearnhead, P. (1999). Improved particle filter for nonlinear problems. *IEE Proc. Radar, Sonar Navigation*, 146(1):2–7.

Chopin, N. (2004). Central limit theorem for sequential Monte Carlo methods and its application to Bayesian inference. *Ann. Statist.*, 32(6):2385–2411.

Chopin, N. and Papaspiliopoulos, O. (2020). *An Introduction to Sequential Monte Carlo*. Springer Series in Statistics. Springer.

Corenflos, A. and Särkkä, S. (2022). The Coupled Rejection Sampler. *arXiv preprint arXiv:2201.09585*.

Del Moral, P. (2004). *Feynman-Kac formulae. Genealogical and interacting particle systems with applications*. Probability and its Applications. Springer Verlag, New York.

Del Moral, P. (2013). *Mean field simulation for Monte Carlo integration*, volume 126 of *Monographs on Statistics and Applied Probability*. CRC Press, Boca Raton, FL.

Del Moral, P., Doucet, A., and Singh, S. S. (2010). A backward particle interpretation of Feynman-Kac formulae. *M2AN Math. Model. Numer. Anal.*, 44(5):947–975.

Douc, R., Cappé, O., and Moulines, E. (2005). Comparison of resampling schemes for particle filtering. In *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis*, pages 64–69. IEEE.

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.

Dubarry, C. and Le Corff, S. (2013). Non-asymptotic deviation inequalities for smoothed additive functionals in nonlinear state-space models. *Bernoulli*, 19(5B):2222–2249.

Fearnhead, P., Papaspiliopoulos, O., and Roberts, G. O. (2008). Particle filters for partially observed diffusions. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 70(4):755–777.

Fearnhead, P., Wyncoll, D., and Tawn, J. (2010). A sequential smoothing algorithm with linear computational cost. *Biometrika*, 97(2):447–464.

Gerber, M. and Chopin, N. (2015). Sequential quasi Monte Carlo. *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, 77(3):509–579.

Gerber, M., Chopin, N., and Whiteley, N. (2019). Negative association, ordering and convergence of resampling methods. *Ann. Statist.*, 47(4):2236–2260.

Gloaguen, P., Corff, S. L., and Olsson, J. (2019). A pseudo-marginal sequential Monte Carlo online smoothing algorithm. *arXiv preprint arXiv:1908.07254*.

Godsill, S. J., Doucet, A., and West, M. (2004). Monte Carlo smoothing for nonlinear times series. *J. Amer. Statist. Assoc.*, 99(465):156–168.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc. F, Comm., Radar, Signal Proc.*, 140(2):107–113.

Guarniero, P., Johansen, A. M., and Lee, A. (2017). The iterated auxiliary particle filter. *J. Amer. Statist. Assoc.*, 112(520):1636–1647.

Hening, A. and Nguyen, D. H. (2018). Stochastic Lotka-Volterra food chains. *J. Math. Biol.*, 77(1):135–163.

Jacob, P. E., Lindsten, F., and Schön, T. B. (2019). Smoothing with couplings of conditional particle filters. *Journal of the American Statistical Association*.

Jacob, P. E., O'Leary, J., and Atchadé, Y. F. (2020). Unbiased Markov chain Monte Carlo methods with couplings. *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, 82(3):543–600.

Janson, S. (2011). Probability asymptotics: notes on notation. *arXiv preprint arXiv:1108.3924*.

Jasra, A., Kamatani, K., Law, K. J., and Zhou, Y. (2017). Multilevel particle filters. *SIAM Journal on Numerical Analysis*, 55(6):3068–3096.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45.

Kalman, R. E. and Bucy, R. S. (1961). New results in linear filtering and prediction theory. *Trans. ASME Ser. D. J. Basic Engrg.*, 83:95–108.

Kitagawa, G. (1996). Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *J. Comput. Graph. Statist.*, 5(1):1–25.

Lévy, P. (1940). Sur certains processus stochastiques homogènes. *Compositio mathematica*, 7:283–339.

Lindvall, T. and Rogers, L. C. G. (1986). Coupling of multidimensional diffusions by reflection. *Ann. Probab.*, 14(3):860–872.

Lotka, A. J. (1925). *Elements of physical biology*. Williams & Wilkins.

Mastrototaro, A., Olsson, J., and Alenlöv, J. (2021). Fast and numerically stable particle-based online additive smoothing: the adasmooth algorithm. *arXiv preprint arXiv:2108.00432*.

Mörters, P. and Peres, Y. (2010). *Brownian motion*, volume 30. Cambridge University Press.

Nordh, J. and Antonsson, J. (2015). A Quantitative Evaluation of Monte Carlo Smoothers. *Technical report*.

Olsson, J. and Westerborn, J. (2017). Efficient particle-based online smoothing in general hidden Markov models: the PaRIS algorithm. *Bernoulli*, 23(3):1951–1996.

Olver, F. W. J., Lozier, D. W., Boisvert, R. F., and Clark, C. W., editors (2010). *NIST handbook of mathematical functions*. U.S. Department of Commerce, National Institute of Standards and Technology, Washington, DC; Cambridge University Press, Cambridge. With 1 CD-ROM (Windows, Macintosh and UNIX).

Pitt, M. K. and Shephard, N. (1999). Filtering via simulation: auxiliary particle filters. *J. Amer. Statist. Assoc.*, 94(446):590–599.

Robert, C. P. and Casella, G. (2004). *Monte Carlo Statistical Methods, 2nd ed.* Springer-Verlag, New York.

Roberts, G. O. and Rosenthal, J. S. (2004). General state space Markov chains and MCMC algorithms. *Probab. Surv.*, 1:20–71.

Samet, H. (2006). *Foundations of multidimensional and metric data structures*. Morgan Kaufmann.

Sen, D., Thiery, A. H., and Jasra, A. (2018). On coupling particle filter trajectories. *Statistics and Computing*, 28(2):461–475.

Taghavi, E., Lindsten, F., Svensson, L., and Schn, T. B. (2013). Adaptive stopping for fast particle smoothing. In *2013 IEEE International Conference on Acoustics,*

Speech and Signal Processing, pages 6293–6297.

Vershynin, R. (2018). *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

Volterra, V. (1928). Variations and fluctuations of the number of individuals in animal species living together. *ICES Journal of Marine Science*, 3(1):3–51.

Yonekura, S. and Beskos, A. (2022). Online smoothing for diffusion processes observed with noise. *Journal of Computational and Graphical Statistics*, 0(0):1–17.

Appendix A. Additional notations

This section defines new notations that do not appear in the main text (except notations for linear Gaussian models) but are used in the Appendix.

A.1. **Linear Gaussian models.** Let $\dim_X$ and $\dim_Y$ be two strictly positive integers and F_X and F_Y be two full-rank matrices of sizes $\dim_X \times \dim_X$ and $\dim_Y \times \dim_X$ respectively. Let C_X and C_Y be two symmetric positive definite matrices of respective sizes $\dim_X \times \dim_X$ and $\dim_Y \times \dim_Y$. A linear Gaussian state space model has the underlying Markov process defined by

$$X_t | X_{0:t-1} \sim \mathcal{N}(F_X X_{t-1}, C_X),$$

where X_0 also follows a Gaussian distribution; and admits the observation process

$$Y_t | X_t \sim \mathcal{N}(F_Y X_t, C_Y).$$

The predictive (X_t given $Y_{0:t-1}$), filtering (X_t given $Y_{0:t}$) and smoothing (X_t given $Y_{0:T}$) distributions are all Gaussian and their parameters can be explicitly calculated via recurrence formulas (Kalman, 1960; Kalman and Bucy, 1961). We shall denote their respective mean vectors and covariance matrices by $(\mu_t^{\mathrm{pred}}, \Sigma_t^{\mathrm{pred}})$, $(\mu_t^{\mathrm{filt}}, \Sigma_t^{\mathrm{filt}})$ and $(\mu_t^{\mathrm{smth}}, \Sigma_t^{\mathrm{smth}})$. In particular, the starting distribution X_0 is $\mathcal{N}(\mu_0^{\mathrm{pred}}, \Sigma_0^{\mathrm{pred}})$.

A.2. **Total variation distance.** Let μ and ν be two probability measures on $\mathcal{X}$. The total variation distance between μ and ν, sometimes also denoted $\mathrm{TV}(\mu, \nu)$, is defined as $\|\mu - \nu\|_{\mathrm{TV}} := \sup_{f:\mathcal{X}\to[0,1]} |\mu(f) - \nu(f)|$. The definition remains valid if f is restricted to the class of indicator functions on measurable subsets of $\mathcal{X}$. It implies in particular that $|\mu(f) - \nu(f)| \leq \|f\|_{\mathrm{osc}} \mathrm{TV}(\mu, \nu)$.

Next, we state a lemma summarising basic properties of the total variation distance and defining coupling-related notions (see, e.g. Proposition 3 and formula (13) of Roberts and Rosenthal (2004)). While the last property (covariance bound) is not in the aforementioned reference and does not seem popular in the literature, its proof is straightforward and therefore omitted.

Lemma 1. *The total variation distance has the following properties:*

- *(Alternative expressions.) If μ and ν admit densities $f(x)$ and $g(x)$ respectively with reference to a dominating measure λ, we have*

$$\mathrm{TV}(\mu, \nu) = \frac{1}{2} \int |f(x) - g(x)| \, \lambda(\mathrm{d}x) = 1 - \int \min(f(x), g(x)) \lambda(\mathrm{d}x).$$

- *(Coupling inequality & maximal coupling.) For any pair of random variables (M, N) such that $M \sim \mu$ and $N \sim \nu$, we have*

$$\mathbb{P}(M \neq N) \geq \mathrm{TV}(\mu, \nu).$$

There exist pairs (M^*, N^*) *for which equality holds. They are called* maximal couplings *of* μ *and* ν.

- *(Contraction property.) Let* (X_n) *be a Markov chain with invariant measure* $\mu^\star$. *Then*

$$\mathrm{TV}(X_n, \mu^*) \geq \mathrm{TV}(X_{n+1}, \mu^*).$$

- *(Covariance bound.) For any pair of random variables* (M, N) *such that* $M \sim \mu$ *and* $N \sim \nu$ *and real-valued functions* h_1 *and* h_2, *we have*

$$|\mathrm{Cov}(h_1(M), h_2(N))| \leq 2 \|h_1\|_\infty \|h_2\|_\infty \mathrm{TV}\left((M, N), \mu \otimes \nu\right).$$

A.3. **Cost-to-go function.** In the context of the Feynman-Kac model (1), define the associated *cost-to-go* function $H_{t:T}$ as (see e.g. Chopin and Papaspiliopoulos (2020, Chapter 5))

$$H_{t:T}(x_t) := \prod_{s=t+1}^{T} M_{s-1}(x_{s-1}, \mathrm{d}x_s) G_s(x_s). \tag{20}$$

This function bridges $\mathbb{Q}_t(\mathrm{d}x_t)$ and $\mathbb{Q}_T(\mathrm{d}x_t)$, since $\mathbb{Q}_T(\mathrm{d}x_t) \propto \mathbb{Q}_t(\mathrm{d}x_t) H_{t:T}(x_t)$.

A.4. **The projection kernel.** Let $\mathcal{X}$ and $\mathcal{Y}$ be two measurable spaces. The projection kernel $\Pi_{\mathcal{X}}^{(\mathcal{X},\mathcal{Y})}$ is defined by

$$\Pi_{\mathcal{X}}^{(\mathcal{X},\mathcal{Y})}((x, y), \mathrm{d}x^*) := \delta_x(\mathrm{d}x^*).$$

In particular, for any function $g : \mathcal{X} \to \mathbb{R}$ and measure $\mu(\mathrm{d}x, \mathrm{d}y)$ defined on $\mathcal{X} \times \mathcal{Y}$, we have

$$\begin{aligned}(\Pi_{\mathcal{X}}^{(\mathcal{X},\mathcal{Y})} g)(x, y) &= g(x) \\ (\mu \Pi_{\mathcal{X}}^{(\mathcal{X},\mathcal{Y})})(g) &= \iint g(x) \mu(\mathrm{d}x, \mathrm{d}y) = \int g(x) \mu(\mathrm{d}x)\end{aligned}$$

where the second identity shows the marginalising action of $\Pi_{\mathcal{X}}^{(\mathcal{X},\mathcal{Y})}$ on μ. In the context of state space models, we define the shorthand

$$\Pi_t^{0:T} := \Pi_{\mathcal{X}_t}^{(\mathcal{X}_0, \ldots, \mathcal{X}_T)}.$$

A.5. **Other notations.** For a real number x, let $\lfloor x \rfloor$ be the largest integer not exceeding x. The mapping $x \mapsto \lfloor x \rfloor$ is called the floor function • The Gamma function $\Gamma(a)$ is defined for $a > 0$ and is given by $\Gamma(a) := \int_{\mathbb{R}_+} e^{-x} x^{a-1} \mathrm{d}x$ • Let $\mathcal{X}$ and $\mathcal{Y}$ be two measurable spaces. Let $K(x, \mathrm{d}y)$ be a (not necessarily probability) kernel from $\mathcal{X}$ to $\mathcal{Y}$. The norm of K is defined by $\|K\|_\infty := \sup_{f:\mathcal{X}\to\mathcal{Y}, f\neq 0} \|Kf\|_\infty / \|f\|_\infty$. In particular, for any function $f : \mathcal{X} \to \mathcal{Y}$, we have $\|Kf\|_\infty \leq \|K\|_\infty \|f\|_\infty$ • Let X_n be a sequence of random variables. We say that $X_n = \mathcal{O}_{\mathbb{P}}(1)$ if for any $\varepsilon > 0$, there exists $M > 0$ and N_0, both depending on ε, such that $\mathbb{P}(|X_n| \geq M) \leq \varepsilon$ for all $n \geq N_0$. For a strictly positive deterministic sequence a_n, we say that $X_n = \mathcal{O}_{\mathbb{P}}(a_n)$ if $X_n / a_n = \mathcal{O}_{\mathbb{P}}(1)$. See Janson (2011) for discussions • We use the notation $\mathcal{N}(x|\mu, \Sigma)$ to refer to the value at x of the density function of the normal distribution $\mathcal{N}(\mu, \Sigma)$ • Let $f : U \to V$ be a function from some space U to another space V. Let S be a subset of U. The restriction of f to S, written $f|_S$, is the function from S to V defined by $f|_S(x) = f(x)$, $\forall x \in S$.

Appendix B. FFBS complexity for different rejection schemes

B.1. **Framework and notations.** The FFBS algorithm is a particular instance of Algorithm 2 where $B_t^{N,\mathrm{FFBS}}$ kernels are used. If backward simulation is done using pure rejection sampling (Algorithm 4), the computational cost to simulate the $t-1$-th index of the n-th trajectory has conditional distribution

$$\tau_t^{n,\mathrm{FFBS}} \mid \mathcal{F}_T, \mathcal{I}_{t:T}^n \sim \mathrm{Geo}\left(\frac{\sum_i W_{t-1}^i m_t(X_{t-1}^i, X_t^{\mathcal{I}_t^n})}{\bar{M}_h}\right). \tag{21}$$

At this point, it would be useful to compare this formula with (16) of the PaRIS algorithm. The difference is subtle but will drive interesting changes to the way rejection-based FFBS behaves.

If hybrid rejection sampling (Algorithm 6) is to be used instead, we are interested in the distribution of $\min(\tau_t^{n,\mathrm{FFBS}}, N)$, for reasons discussed in Subsection 3.2. In a highly parallel setting, it is preferable that the distribution of *individual* execution times, i.e. $\tau_t^{n,\mathrm{FFBS}}$ or $\min(\tau_t^{n,\mathrm{FFBS}}, N)$, are not heavy-tailed. In contrast, for non-parallel hardware, only *cumulative* execution times, i.e. $\sum_{n=1}^N \tau_t^{n,\mathrm{FFBS}}$ or $\sum_{n=1}^N \min(\tau_t^{n,\mathrm{FFBS}}, N)$, matter. Even though the individual times might behave badly, the cumulative times could be much more regular thanks to effect of the central limit theorem, whenever applicable. Nevertheless, studying the finiteness of the k-th order moment of $\tau_t^{1,\mathrm{FFBS}}$ is still a good way to get information about both types of execution times, since it automatically implies k-th order moment (in)finiteness for both of them.

B.2. **Execution time for pure rejection sampling.** We show that under certain circumstances, the execution time of the pure rejection procedure has infinite expectation. Proposition 1 in Douc et al. (2011) hints that the cost per trajectory for FFBS-reject might tend to infinity when $N \to \infty$. In contrast, we show that infinite expectation might very well happen for *finite* sample sizes. We first give the statement for general state space models, then focus on their implications for Gaussian ones. In particular, while infinite expectations occur only under certain configurations, infinite higher moments happen in *all* linear Gaussian models with non-degenerate dynamics.

Theorem 6. *Using the setting and notations of Appendix B.1, under Assumptions 1 and 4 , we have* $\mathbb{E}[\tau_t^{1,\mathrm{FFBS}}] = \infty$ *whenever*

$$\int_{\mathcal{X}_t} G_t(x_t) H_{t:T}(x_t) \lambda_t(\mathrm{d}x_t) = \infty$$

where the cost-to-go function $H_{t:T}$ *is defined in* (20) *and the measure* λ_t *is defined in* (2).

Theorem 7. *Using the setting and notations of Appendix B.1, we consider linear Gaussian models and their notations defined in Appendix A.1. Then we have* $\mathbb{E}[(\tau_t^{1,\mathrm{FFBS}})^k] = \infty$ *whenever* k *is greater than a certain* k_0 *being the smallest eigenvalue of the matrix* $\mathrm{Id} + C_X^{1/2}\left((\Sigma_t^{\mathrm{smth}})^{-1} - (\Sigma_t^{\mathrm{pred}})^{-1}\right) C_X^{1/2}$.

The proofs of the two assertions are given in Appendix E.7. We now look at how they are manifested in concrete examples. The first remark is that for technical reasons, Theorem 7 gives no information on the finiteness of $E[(\tau_t^{1,\mathrm{FFBS}})^k]$ for $k = 1$

(since k_0 is already greater than or equal to 1 by definition). To study the finiteness of $\mathbb{E}[\tau_t^{1,\mathrm{FFBS}}]$, we thus turn to Theorem 6.

Example 4. *In linear Gaussian models, the integral of Theorem 6 is equal to*

$$\int \mathcal{N}(y_t|F_Y x_t, C_Y) \prod_{s=t+1}^{T} \mathcal{N}(x_s|F_X x_{s-1}, C_X)\mathcal{N}(y_s|F_Y x_s, C_Y)\mathrm{d}x_{t:T}$$

where the notation $\mathcal{N}(\mu, \Sigma)$ refers to the density of the normal distribution. The integrand is proportional to $\exp[-0.5(Q(x_{t:T}) - R(x_{t:T}))]$ for some quadratic form $Q(x_{t:T})$ and linear form $R(x_{t:T})$. The integral is finite if and only if Q is positive definite. In our case, this means that there is no non-trivial root for the equation $Q(x_{t:T}) = 0$, which is equivalent to

$$\begin{cases} F_Y x_s & = 0, \forall s = t, \dots T \\ F_X x_{s-1} & = x_s, \forall s = t+1, \dots, T. \end{cases}$$

Put another way, $\mathbb{E}[\tau_t^{1,\mathrm{FFBS}}]$ is infinite whenever the intersection

$$\bigcap_{k=0}^{T-t} \mathrm{Ker}(F_Y F_X^k) = \bigcap_{k=0}^{T-t} F_X^{-k}(\mathrm{Ker}(F_Y))$$

contains other things than the zero vector. A common and particularly troublesome situation is when $F_X = c\,\mathrm{Id}$ for some $c > 0$ (but C_X can be arbitrary) and the dimension of the states ($\dim_X$) is greater than that of the observations ($\dim_Y$). Then the above intersection remains non-trivial no matter how big $T - t$ is. Thus, $\mathbb{E}[\tau_t^{1,\mathrm{FFBS}}]$ has no expectation for any t. In general, the problem is less severe as successive intersections will shrink the space quickly to $\{0\}$. Consequently, Theorem 6 only points out infiniteness of $\mathbb{E}[\tau_t^{1,\mathrm{FFBS}}]$ for t close to T. The bad news however will come from higher moments, as seen in the below example.

We will now focus on a simple but particularly striking example. Our purpose here is to illustrate the concepts as well as to show that their implications are relevant even in small, familiar settings. More advanced scenarios are presented in Section 5 devoted to numerical experiments.

Example 5. *We consider two one-dimensional Gaussian state-space models: they both have $F_X = 0.5$, $C_X = 1$, $X_0 \sim \mathcal{N}(0, C_X^2/(1 - F_X^2))$ and $T = 3$. The only difference between them is that one has $\sigma_y^2 := C_Y = 0.5^2$ and another has $\sigma_y^2 = 3^2$. We are interested in the execution times $\tau_1^{n,\mathrm{FFBS}}$ at time $t = 1$ (i.e. the rejection-based simulation of indices $\mathcal{I}_0^n$ at time $t = 0$). Theorem 7 then gives $k_0 \approx 1.14$ for $\sigma_y = 3$ and $k_0 \approx 5$ for $\sigma_y = 0.5$. The first implication is that in both cases, $\tau_1^{n,\mathrm{FFBS}}$ is a heavy-tailed random variable and therefore FFBS-reject is not a viable option in a highly parallel setting. But an interesting phenomenon happens in the sequential hardware scenario where one is rather interested in the cumulative execution time, i.e. $\sum_{n=1}^{N} \tau_1^{n,\mathrm{FFBS}}$, or equivalently, the* mean *number of trials per* particle. *In the $\sigma_y = 3$ case, non-existence of second moment prevents the cumulative regularisation effect of the central limit theorem. This is not the case for $\sigma_y = 0.5$, in which the cumulative execution time actually behaves nicely (Figures 8 and 9). However, the most valuable message from this example is perhaps that the performance of FFBS-reject depends in a non-trivial (hard to predict) way on the model parameters.*

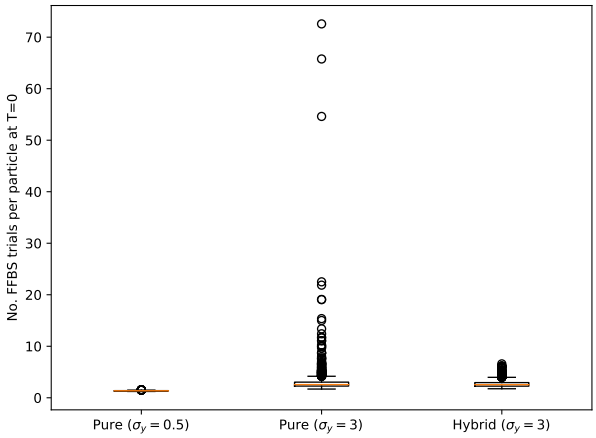

FIGURE 8. Box plots for the mean number of trials per particles to simulate indices at time 0, for models described in Example 5 and for FFBS algorithms based on pure and hybrid rejection sampling. The figure is obtained by running bootstrap particle filters with $N = 500$ over 1500 independent executions.

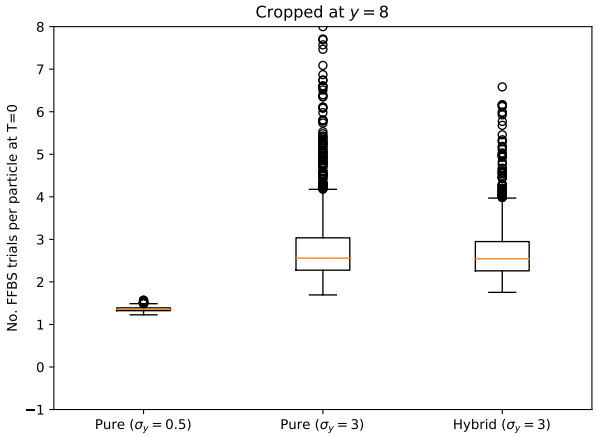

FIGURE 9. Zoom of Figure 8 to $0 \leq y \leq 8$

B.3. **Execution time for hybrid rejection sampling.** Formula (21) suggests defining the limit distribution $\tau_t^{\infty,\mathrm{FFBS}}$ as

$$\tau_t^{\infty,\mathrm{FFBS}} \mid X_t^{\infty,\mathrm{FFBS}} \sim \mathrm{Geo}\left(\frac{r_t(X_t^{\infty,\mathrm{FFBS}})}{\bar{M}_h}\right)$$

where $X_t^{\infty,\mathrm{FFBS}} \sim \mathbb{Q}_T(\mathrm{d}x_t)$ and r_t given in Definition 1. These quantities provide the following characterisation of the cumulative execution time for the hybrid FFBS algorithm (proved in Section E.8).

Theorem 8. *Under Assumptions 1 and 4 and the setting of Section B.1, we have*

$$\frac{\sum_{n=1}^{N} \min(\tau_t^{n,\mathrm{FFBS}}, N)}{N} = \mathcal{O}_{\mathbb{P}}\left(\mathbb{E}[\min(\tau_t^{\infty,\mathrm{FFBS}}, N)]\right)$$

where the notation $\mathcal{O}_{\mathbb{P}}$ is defined in Appendix A.

This theorem admits the following corollary for linear Gaussian models (also proved in Section E.8).

Corollary 2. *For linear Gaussian models (Appendix A.1), if smoothing is performed using the hybrid rejection version of the FFBS algorithm, the mean execution time per particle at time step t is $\mathcal{O}_{\mathbb{P}}(\log^{d_t/2} N)$ where d_t is the dimension of $\mathcal{X}_t$.*

The bound $\mathcal{O}_{\mathbb{P}}(\log^{d_t/2} N)$ is actually quite conservative. For instance, with either $\sigma_y = 0.5$ or $\sigma_y = 3$, the model considered in Example 5 admits $\mathbb{E}[\tau_t^{\infty,\mathrm{FFBS}}] < \infty$. (Gaussian dynamics can be handled using exact analytic calculations and enables to verify the claim straightforwardly.) Theorem 8 then gives an execution time per particle of order $\mathcal{O}_{\mathbb{P}}(1)$ for hybrid FFBS, which is better than the $\mathcal{O}_{\mathbb{P}}(\sqrt{\log N})$ predicted by Corollary 2. Yet another unsatisfactory point of the result is its failure to make sense of the spectacular improvement brought by hybrid rejection sampling over the ordinary procedure in the $\sigma_y = 3$ case (see Figure 8). As explained in Example 5, this is connected to the variance of $\mathbb{E}[\tau_t^{1,\mathrm{FFBS}}]$ and not merely the expectation; so a study of second order properties of $N^{-1}\sum_n \min(\tau_t^{n,\mathrm{FFBS}}, N)$ would be desirable.

Appendix C. Conditionally-correlated versions of particle algorithms

C.1. **Alternative resampling schemes.** In Algorithm 1, the indices $A_t^{1:N}$ are drawn conditionally i.i.d. from the multinomial distribution $\mathcal{M}(W_{t-1}^{1:N})$. They satisfy

$$\mathbb{E}\left[\sum_{j=1}^{N} \mathbb{1}_{A_t^j = i} \middle| \mathcal{F}_{t-1}\right] = N W_{t-1}^i$$

for any $i = 1, \ldots, N$. There are other ways to generate $A_t^{1:N}$ from $W_{t-1}^{1:N}$ that still verify this identity. We call them *unbiased resampling schemes*, and the natural one used in Algorithm 1 *multinomial resampling*.

The main motivation for alternative resampling schemes is performance. We refer to Chopin (2004); Douc et al. (2005); Gerber et al. (2019) for more details, but would like to mention that the theoretical studies of particle algorithms using other resampling schemes are more complicated since $X_t^{1:N}$ are no longer i.i.d. given $\mathcal{F}_{t-1}$. We use systematic resampling (Carpenter et al., 1999) in our experiments. See Algorithm 8 for a succinct description and Chopin and Papaspiliopoulos (2020, Chapter 9) for efficient implementations in $\mathcal{O}(N)$ running time.

C.2. **Conditionally-correlated version of Algorithm 7.** In this part, we present an alternative version of Algorithm 7 that does not create conditionally i.i.d. particles at each time step. The procedure is detailed in Algorithm 9. It creates on the fly backward kernels $B_t^{N,\mathrm{ITRC}}$ (for "intractable, conditionally correlated"). It

Algorithm 8: Systematic resampling

Input: Weights $W_{t-1}^{1:N}$ summing to 1
Generate $U \sim \text{Uniform}[0,1]$
for $n \leftarrow 1$ **to** N **do**
  Set A_t^n to the unique index k satisfying

$$W_1 + \cdots + W_{k-1} \leq nU < W_1 + \cdots + W_k$$

Output: Resampled indices $A_t^{1:N}$

involves a resampling step which can be done in principle using any unbiased resampling scheme. Following the intuitions of Subsection 4.2.3 and the notations of Algorithm 9, we want a scheme such that in most cases, $A_t^{2k-1} \neq A_t^{2k}$ but the Euclidean distance between $X_{t-1}^{A_t^{2k-1}}$ and $X_{t-1}^{A_t^{2k}}$ is small. Algorithm 10 proposes such a method (which we name the Adjacent Resampler). It can run in $\mathcal{O}(N)$ time using a suitably implemented linked list.

Algorithm 9: Conditionally-correlated version of Algorithm 7

Input: Feynman-Kac model (1), particles $X_{t-1}^{1:N}$ and weights $W_{t-1}^{1:N}$ that approximate $\mathbb{Q}_{t-1}(\mathrm{d}x_{t-1})$
Resample $A_t^{1:N}$ from $\{1, 2, \ldots, N\}$ with weights $W_{t-1}^{1:N}$ using any resampling scheme (such as the Adjacent Resampler in Algorithm 10)
for $k \leftarrow 1$ **to** $N/2$ **do**
  Move. Simulate X_t^{2k-1} and X_t^{2k} such that marginally, $X_t^{2k-1} \sim M_t(X_{t-1}^{A_t^{2k-1}}, \cdot)$ and $X_t^{2k} \sim M_t(X_{t-1}^{A_t^{2k}}, \cdot)$
  Calculate backward kernel.
  if $X_t^{2k-1} = X_t^{2k}$ **then**
    Set $B_t^{N,\mathrm{ITRC}}(2k-1, \cdot) \leftarrow \left(\delta\left\{A_t^{2k-1}\right\} + \delta\left\{A_t^{2k}\right\}\right)/2$
    Set $B_t^{N,\mathrm{ITRC}}(2k, \cdot) \leftarrow \left(\delta\left\{A_t^{2k-1}\right\} + \delta\left\{A_t^{2k}\right\}\right)/2$
  else
    Set $B_t^{N,\mathrm{ITRC}}(2k-1, \cdot) \leftarrow \delta\left\{A_t^{2k-1}\right\}$
    Set $B_t^{N,\mathrm{ITRC}}(2k, \cdot) \leftarrow \delta\left\{A_t^{2k}\right\}$
Reweight. Set $\omega_t^n \leftarrow G_t(X_t^n)$ for $n = 1, 2, \ldots, N$
Set $\ell_t^N \leftarrow \sum_{n=1}^N \omega_t^n / N$
Set $W_t^n \leftarrow \omega_t^n / N\ell_t^N$ for $n = 1, 2, \ldots, N$
Output: Particles $X_t^{1:N}$ and weights $W_t^{1:N}$ that approximate $\mathbb{Q}_t(\mathrm{d}x_t)$; backward kernel $B_t^{N,\mathrm{ITRC}}$ for use in Algorithms 2 and 3

Appendix D. Additional information on numerical experiments

D.1. **Offline smoothing in linear Gaussian models.** In this section, we study offline smoothing for the linear Gaussian model specified in Section 5.1. Since offline

Algorithm 10: The Adjacent Resampler

Input: Particles $X_{t-1}^{1:N}$, weights $W_{t-1}^{1:N}$

Sort the particles $X_{t-1}^{1:N}$ using the Hilbert curve. Let $s \leftarrow [s_1 \dots s_N]$ be the corresponding *indices*

Resample from $\{1, \dots, N\}$ with weights $W_{t-1}^{1:N}$ using systematic resampling (Carpenter et al., 1999; Gerber et al., 2019), then let $f : \{1, \dots, N\} \to \mathbb{Z}$ be the function defined by $f(i)$ being the number of times the index s_i was resampled. Obviously $\sum_{i=1}^{N} f(i) = N$

Initialise $i \leftarrow 1$

for $n \leftarrow 1$ **to** N **do**

- Set $A_t^n \leftarrow s_i$
- Update $f(i) \leftarrow f(i) - 1$
- Let Ω_1 be the set $\{\min\{\ell > i \mid f_\ell > 0\}\}$ (which has one element if the minimum is well-defined and zero element otherwise)
- Let Ω_2 be the set $\{\max\{\ell < i \mid f_\ell > 0\}\}$ (which has one element if the maximum is well-defined and zero element otherwise)
- If $\Omega_1 \cup \Omega_2$ is not empty, update $i \leftarrow \operatorname{argmax} f|_{\Omega_1 \cup \Omega_2}$ (see section A.5 for the restriction notation). If there is more than one argmax, pick one randomly

Output: Resampled indices $A_t^{1:N}$

processing requires storing particles at all times t in the memory, we use $T = 500$ here instead of $T = 3000$. Apart from that, the algorithmic and benchmark settings remain the same.

Figure 10 plots the squared interquartile range of the estimators $\mathbb{Q}_T(\varphi_t)$ with respect to t, for different algorithms. For small t, the function φ_t only looks at states close to time 0, whereas for bigger t, recent states less affected by degeneracy are also taken into account. In all cases though, we see that MCMC and rejection-based smoothers have superior performance.

Figure 11 shows box plots of the averaged execution times (per particle N per time t) based on 150 runs. The observations are comparable to those in Section 5.1. We see a performance difference between the rejection-based smoothers using the bootstrap and the guided filters. Both have an execution time that is much more variable than hybrid rejection algorithms. The latter still need around 10 times more CPU load than MCMC smoothers, for essentially the same precision.

We now take a closer look at the reason behind the performance difference between the bootstrap filter and the guided one when pure rejection sampling is used. Figure 12 shows the effective sample size (ESS) of both filters as a function of time. We can see that there is an outlier in the data around time $t = 40$. Figure 13 boxplots the execution times divided by N at $t = 40$ for the pure rejection sampling algorithm, whereas Figures 14 and 15 do the same for $t = 38$ and $t = 42$. The root of the problem is now clear: at most times t there is very few difference between the execution times of the bootstrap and the guided filters. However, if an outlier is present in the data, the guided filter suddenly requires a very high number of transition density evaluation in the rejection sampler. This gives yet another reason to avoid using pure rejection sampling.

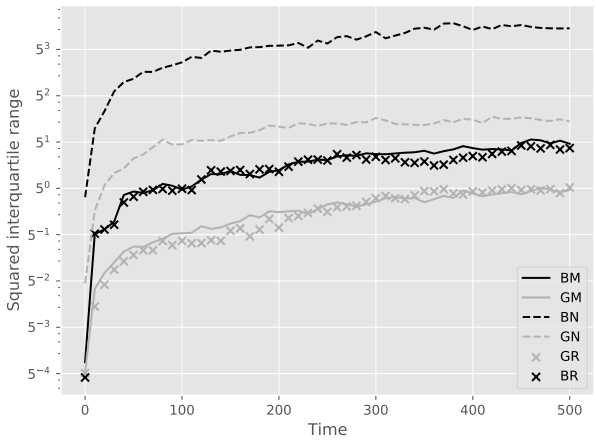

FIGURE 10. Squared interquartile range of the estimators of $\mathbb{Q}_T(\varphi_t)$ with respect to t, for different algorithms applied to the model of Section D.1. See Section 5.1 for the meaning of the acronyms in the legend.

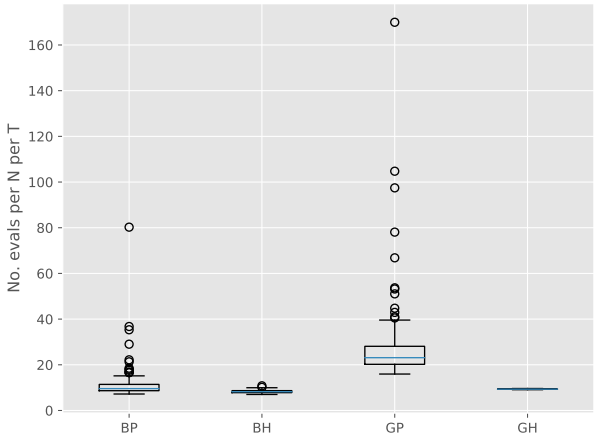

FIGURE 11. Box plots of the number of transition density evaluations divided by NT for different algorithms in the offline linear Gaussian model of Section D.1.

D.2. **Lotka-Volterra SDE.**

D.2.1. *Coupling of Euler discretisations.* Consider the SDE

$$\mathrm{d}X_t = b(X_t)\mathrm{d}t + \sigma(X_t)\mathrm{d}W_t \tag{22}$$

and two starting points X_0^{A} and X_0^{B} in $\mathbb{R}^d$. We wish to simulate X_1^{A} and X_1^{B} such that the transitions from X_0^{A} to X_1^{A} and X_0^{B} to X_1^{B} both follow the Euler-discretised version of the equation, but X_1^{A} and X_1^{B} are correlated in a way that increases, as

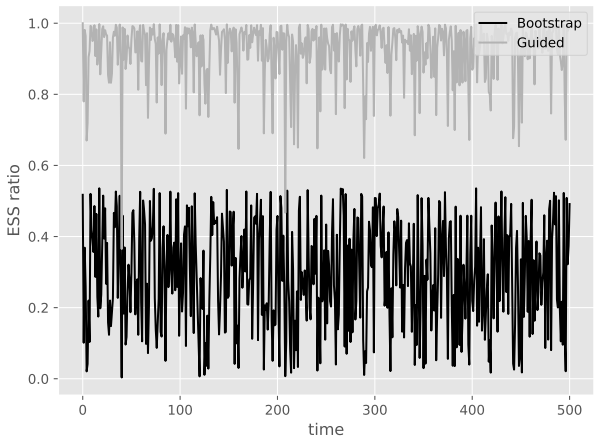

FIGURE 12. Evolution of the ESS for the linear Gaussian model of Section D.1.

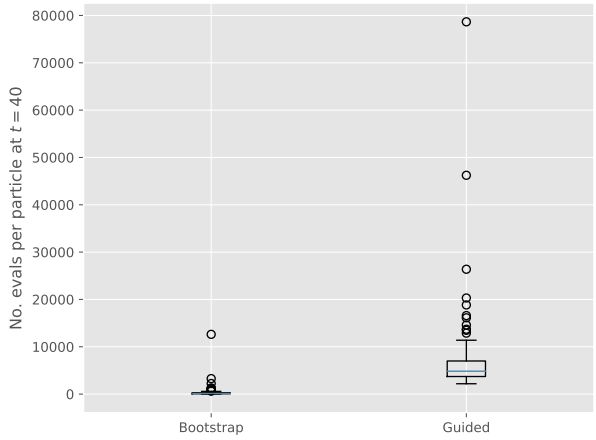

FIGURE 13. Box plots of the average execution time per particle for the pure rejection algorithm at time $t = 40$. Figure produced based on 150 independent runs of the model described in Section D.1.

much as we can, the probability that they are equal. Algorithm 11 makes it clear that it all boils down to the coupling of two Gaussian distributions.

Lindvall and Rogers (1986) propose the following construction: if two diffusions X_t^{A} and X_t^{B} both follow the dynamics of (22), that is,

$$\mathrm{d}X_t^{\mathrm{A}} = b(X_t^{\mathrm{A}})\mathrm{d}t + \sigma(X_t^{\mathrm{A}})\mathrm{d}W_t^{\mathrm{A}}$$
$$\mathrm{d}X_t^{\mathrm{B}} = b(X_t^{\mathrm{B}})\mathrm{d}t + \sigma(X_t^{\mathrm{B}})\mathrm{d}W_t^{\mathrm{B}}$$

and the two Brownian motions are correlated via

$$\mathrm{d}W_t^{\mathrm{B}} = [\mathrm{Id} - 2u(X^{\mathrm{A}}, X^{\mathrm{B}})u(X^{\mathrm{A}}, X^{\mathrm{B}})^\top]\mathrm{d}W_t^{\mathrm{A}} \tag{23}$$

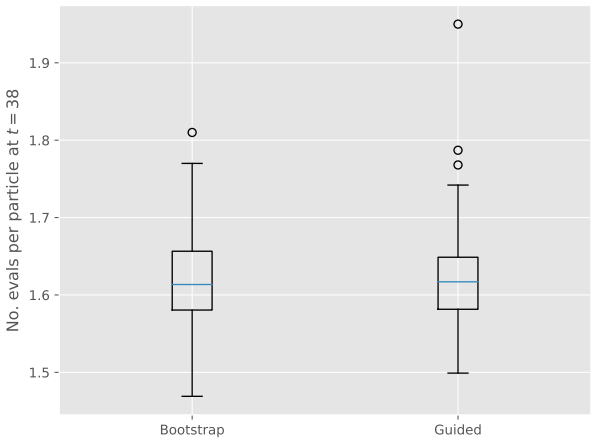

FIGURE 14. Same as Figure 13, but for $t = 38$.

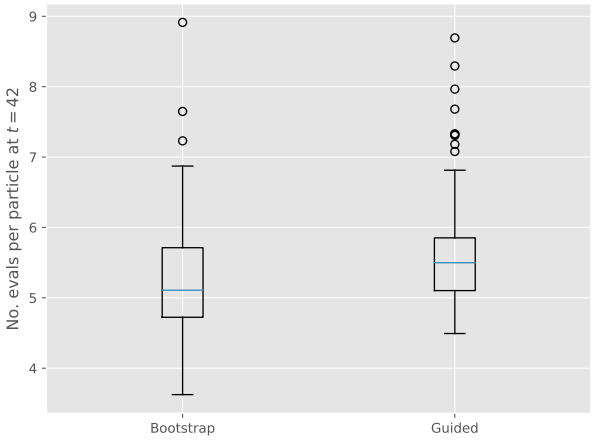

FIGURE 15. Same as Figure 13, but for $t = 42$.

where Id is the identity matrix and the vector u is defined by

$$u(x, x') = \frac{\sigma(x')^{-1}(x - x')}{\|\sigma(x')^{-1}(x - x')\|_2},$$

then under some regularity conditions, the two diffusions meet almost surely. (Note two special features of (23): it is valid because the term in the square bracket is an orthogonal matrix; and it ceases to be well-defined once the two trajectories have met.) Simulating the meeting time τ turns out to be very challenging. The Euler discretisation (Algorithm 11 + Algorithm 12) has a fixed step size δ, and there is zero probability that τ is of the form $k\delta$ for some integer k. Since the coupling transform is deterministic, the two Euler-simulated trajectories will *never* meet. Figure 16 depicts this difficulty in the special case of two Brownian motions in dimension 1 (i.e. $b(x) \equiv 0$ and $\sigma \equiv 1$). Under this setting, (23) means that

Algorithm 11: Coupling of two Euler discretisations

Input: Functions $b : \mathbb{R}^d \to \mathbb{R}^d$ and $\sigma : \mathbb{R}^d \to \mathbb{R}^{d\times d}$, two starting points X_0^{A} and X_0^{B} at time 0, number of discretisation step N_{dist}

Initialise $X^{\mathrm{A}} \leftarrow X_0^{\mathrm{A}}$

Initialise $X^{\mathrm{B}} \leftarrow X_0^{\mathrm{B}}$

Set $\delta \leftarrow 1/N_{\mathrm{dist}}$

for $i \leftarrow 1$ **to** N_{dist} **do**

  Simulate $(\tilde{X}^{\mathrm{A}}, \tilde{X}^{\mathrm{B}})$ from a coupling of

$$\mathcal{N}(X^{\mathrm{A}} + \delta b(X^{\mathrm{A}}), \delta\sigma(X^{\mathrm{A}})\sigma(X^{\mathrm{A}})^\top)$$

  and

$$\mathcal{N}(X^{\mathrm{B}} + \delta b(X^{\mathrm{B}}), \delta\sigma(X^{\mathrm{B}})\sigma(X^{\mathrm{B}})^\top),$$

  such as Algorithm 14

  Update $(X^{\mathrm{A}}, X^{\mathrm{B}}) \leftarrow (\tilde{X}^{\mathrm{A}}, \tilde{X}^{\mathrm{B}})$

Set $(X_1^{\mathrm{A}}, X_1^{\mathrm{B}}) \leftarrow (X^{\mathrm{A}}, X^{\mathrm{B}})$

Output: Two endpoints X_1^{A} and X_1^{B} at time 1, obtained by passing X_0^{A} and X_0^{B} in a correlated manner through a discretised version of (22)

the two Brownian increments are symmetric with respect to the midpoint of the segment connecting their initial states. Note that the two dashed lines do cross at two points, but using them as meeting points is invalid: since they are not part of the discretisation but the result of some heuristic "linear interpolation", it would change the distribution of the trajectories.

Algorithm 12: Lindvall-Rogers coupling of two Gaussian distributions

Input: Two vectors $\mu^{\mathrm{A}}, \mu^{\mathrm{B}}$ in $\mathbb{R}^d$ and two $d \times d$ matrices σ^{A} and σ^{B}

Calculate $u \leftarrow (\sigma^{\mathrm{B}})^{-1}(\mu^{\mathrm{A}} - \mu^{\mathrm{B}})$

Normalise $u \leftarrow u/\left\|u\right\|_2$

Simulate $W^{\mathrm{A}} \sim \mathcal{N}(0, \mathrm{Id})$

Set $W^{\mathrm{B}} \leftarrow (\mathrm{Id} - 2uu^\top)W^{\mathrm{A}}$

Set $X^{\mathrm{A}} \leftarrow \mu^{\mathrm{A}} + \sigma^{\mathrm{A}}W^{\mathrm{A}}$

Set $X^{\mathrm{B}} \leftarrow \mu^{\mathrm{B}} + \sigma^{\mathrm{B}}W^{\mathrm{B}}$

Output: Two correlated points X^{A} and X^{B} marginally distributed according to $\mathcal{N}(\mu^{\mathrm{A}}, \sigma^{\mathrm{A}}(\sigma^{\mathrm{A}})^\top)$ and $\mathcal{N}(\mu^{\mathrm{B}}, \sigma^{\mathrm{B}}(\sigma^{\mathrm{B}})^\top)$ respectively

We therefore need some coupling that has a non-zero meeting probability at each δ-step. This can be achieved by the rejection maximal coupling (Algorithm 13, see also, e.g. Roberts and Rosenthal, 2004) as well as the recently proposed coupled rejection sampler (Corenflos and Särkkä, 2022). However, they all make use of rejection sampling in one way or another, which renders the execution time random. We wish to avoid this if possible. The reflection-maximal coupling (Bou-Rabee et al., 2020; Jacob et al., 2020) has deterministic cost and optimal meeting probability, but is only applicable for two Gaussian distributions of the same covariance matrix, which is not our case.

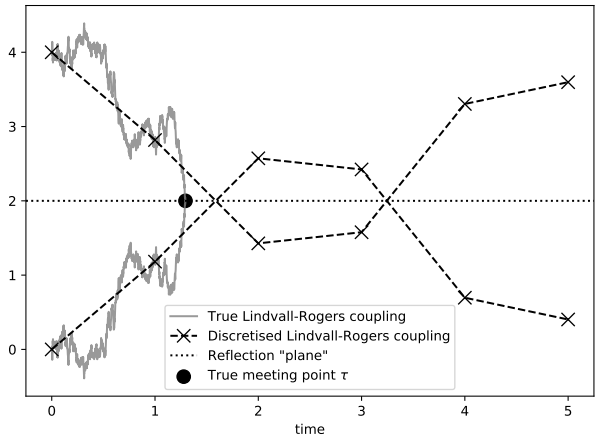

FIGURE 16. Coupling of two Brownian motions in $\mathbb{R}$ starting from 0 and 4 respectively. The true Lindvall-Rogers coupling (23) is represented by the continuous grey lines. The dicretised simulation (Algorithm 11 + Algorithm 12) is shown by the dashed lines. The discretised trajectories not only miss the true meeting point τ but also never meet afterwards (see text).

Algorithm 13: Rejection maximal coupler for two distributions

Input: Two probability distributions f^{A} and f^{B}
Simulate $X^{\mathrm{A}} \sim f^{\mathrm{A}}$
Simulate $U^{\mathrm{A}} \sim \mathrm{Uniform}[0, f^{\mathrm{A}}(X^{\mathrm{A}})]$
if $U^{\mathrm{A}} \leq f^{\mathrm{B}}(X^{\mathrm{A}})$ **then**
  Set $X^{\mathrm{B}} \leftarrow X^{\mathrm{A}}$
else
  repeat
    Simulate $X^{\mathrm{B}} \sim f^{\mathrm{B}}$
    Simulate $U^{\mathrm{B}} \sim \mathrm{Uniform}[0, f^{\mathrm{B}}(X^{\mathrm{B}})]$
  until $U^{\mathrm{B}} > f^{\mathrm{A}}(X^{\mathrm{B}})$
Output: Two maximally-coupled realisations X^{A} and X^{B}, marginally f^{A}-distributed and f^{B}-distributed respectively

As suggested by Figure 16, the discretised Lindvall-Rogers coupling (Algorithm 12) is actually great for bringing together two faraway trajectories. Only when they start getting closer that it misses out. At that moment, the two distributions corresponding to the next δ-step have non-negligible overlap and would preferably be coupled in the style of Algorithm 13. We propose a modified coupling scheme that acts like Algorithm 12 when the two trajectories are at a large distance and behaves as Algorithm 13 otherwise.

The idea is to preliminarily generate a uniform draw in the "overlapping zone" of the two distributions (if they are close enough to make that easy). Next, we perform

Algorithm 12 and then, any of the two simulations belonging to the overlapping zone will be replaced by the aforementioned preliminary draw (if it is available). The precise mathematical formulation is given in Algorithm 14 and the proof in Appendix E.12.

Algorithm 14: Modified Lindvall-Rogers (MLR) coupler of two Gaussian distributions

Input: Two vectors μ^{A} and μ^{B} in $\mathbb{R}^d$, two $d \times d$ matrices σ^{A} and σ^{B}

Let f^{A} and f^{B} be respectively the probability densities of $\mathcal{N}(\mu^{\mathrm{A}}, \sigma^{\mathrm{A}}(\sigma^{\mathrm{A}})^\top)$ and $\mathcal{N}(\mu^{\mathrm{B}}, \sigma^{\mathrm{B}}(\sigma^{\mathrm{B}})^\top)$

Simulate X^{A} and X^{B} from Algorithm 12

Simulate $U \sim \mathrm{Uniform}[0,1]$

Set $U^{\mathrm{A}} \leftarrow U f^{\mathrm{A}}(X^{\mathrm{A}})$ and $U^{\mathrm{B}} \leftarrow U f^{\mathrm{B}}(X^{\mathrm{B}})$

Simulate $Y \sim f^{\mathrm{A}}$ and $V \sim \mathrm{Uniform}[0, f^{\mathrm{A}}(Y)]$

if $V \leq f^{\mathrm{B}}(Y)$ **then**

  if $U^{\mathrm{A}} \leq f^{\mathrm{B}}(X^{\mathrm{A}})$ then update $(X^{\mathrm{A}}, U^{\mathrm{A}}) \leftarrow (Y, V)$

  if $U^{\mathrm{B}} \leq f^{\mathrm{A}}(X^{\mathrm{B}})$ then update $(X^{\mathrm{B}}, U^{\mathrm{B}}) \leftarrow (Y, V)$

Output: Two correlated random vectors X^{A} and X^{B}, distributed marginally according to $\mathcal{N}(\mu^{\mathrm{A}}, \sigma^{\mathrm{A}}(\sigma^{\mathrm{A}})^\top)$ and $\mathcal{N}(\mu^{\mathrm{B}}, \sigma^{\mathrm{B}}(\sigma^{\mathrm{B}})^\top)$

Algorithm 14 has a deterministic execution time, but it does not attain the optimal coupling rate. Yet, as $\delta \to 0$, we see empirically that it still recovers the oracle coupling time defined by (23) (although we did not try to prove this formally). In Figure 17, we couple two standard Brownian motions starting from $a = 0$ and $b = 1.5$ using Algorithm 14 with different values of δ. It is known, by a simple application of the reflection principle (Lévy, 1940; see also Chapter 2.2 of Mörters and Peres, 2010), that the reflection coupling (23) succeeds after a $\mathrm{Levy}(0, (b-a)^2/4)$-distributed time. We therefore have to deal with a heavy-tailed distribution and restrict ourselves to the interval $[0, 5]$. We see that the law of the meeting time is stable and convergent as $\delta \to 0$. Thus, at least empirically, Algorithm 14 does not suffer from the instability problem as $\delta \to 0$, contrary to a naive path space augmentation approach (see Yonekura and Beskos, 2022 for a discussion).

D.2.2. *Supplementary figures.* Figure 18 plots a realisation of the states and data with parameters given in Subsection 5.2, for a relatively small scale dataset ($T = 50$). While the periodic trait seen in classical deterministic Lotka-Volterra equations is still visible (with a period of around 20), it is clear that here random perturbations have added considerable chaos to the system. Figures 19 and 20 show respectively the performances of the naive genealogy tracking smoother and ours (Algorithm 9) on the dataset of Figure 18. Our smoother has successfully prevented the degeneracy phenomenon, particularly for times close to 0. Figure 21 shows, in two different ways, the properties of effective sample sizes (ESS) in the $T = 3000$ scenario (see Section 5.2).

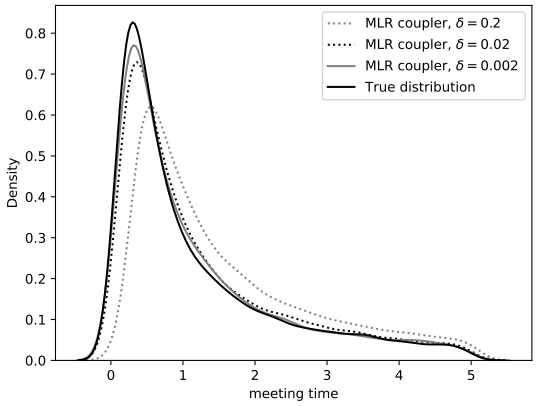

FIGURE 17. Densities of the meeting times restricted to $[0, 5]$ for two Brownian motions started from 0 and 1.5. The curves are drawn using 20 000 simulations from either a Levy distribution (for the "True distribution" curve) or Algorithm 14 (for the MLR ones). The boundary effect of kernel density estimators causes spills beyond 0 and 5.

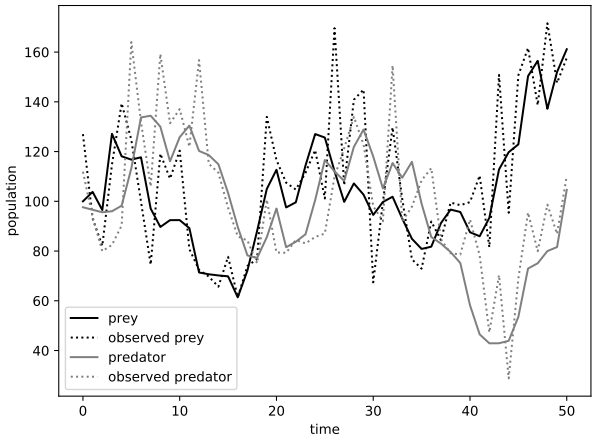

FIGURE 18. A realisation of the Lotka-Volterra SDE with parameters described in Section 5.2. The stationary point of the system is $[100, 100]$.

APPENDIX E. PROOFS

E.1. **Proof of Theorem 1 (general convergence theorem).** In line with (7), we define the distribution $\mathbb{Q}_t^N(\mathrm{d}x_{0:t})$ for $t < T$ as the $x_{0:t}$ marginal of the joint

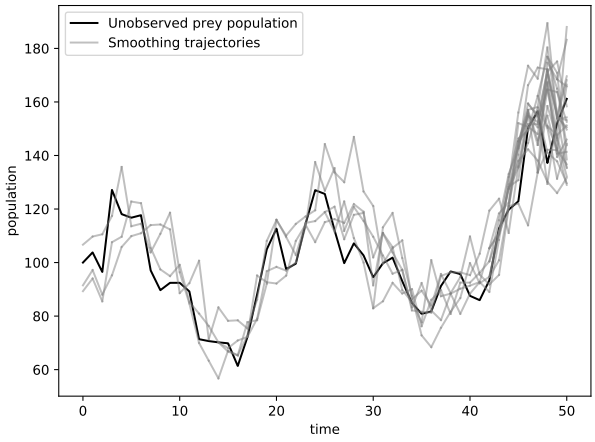

FIGURE 19. Smoothing trajectories for the dataset of Figure 18 using the naive genealogy tracking smoother ($B_t^{N,\mathrm{GT}}$ kernels) with systematic resampling (see Section C.1). We took $N = 100$ and randomly plotted 30 smoothing trajectories.

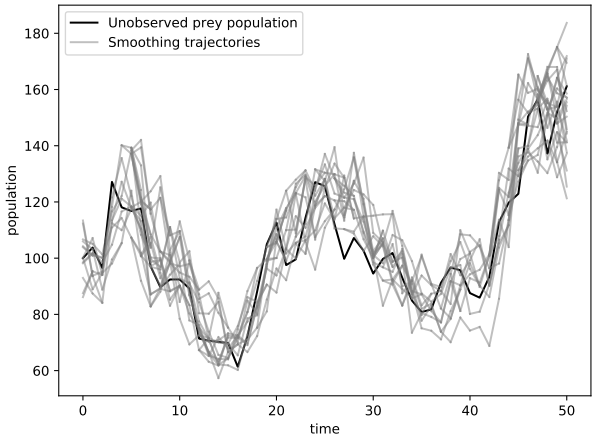

FIGURE 20. Same as Figure 19, but smoothing was done using Algorithm 9 instead.

distribution

$$\bar{\mathbb{Q}}_t^N(\mathrm{d}x_{0:t}, \mathrm{d}i_{0:t}) := \mathcal{M}(W_t^{1:N})(\mathrm{d}i_t) \left[\prod_{s=t}^{1} B_s^N(i_s, \mathrm{d}i_{s-1})\right] \left[\prod_{s=t}^{0} \delta_{X_s^{i_s}}(\mathrm{d}x_s)\right]. \tag{24}$$

The proof builds up on an inductive argument which links $\mathbb{Q}_t^N$ with $\mathbb{Q}_{t-1}^N$ through new innovations at time t. More precisely, we have the following fundamental proposition, where $\mathcal{F}_t^+$ is defined as the smallest σ-algebra containing $\mathcal{F}_t$ and $\hat{B}_{1:t}^N$.

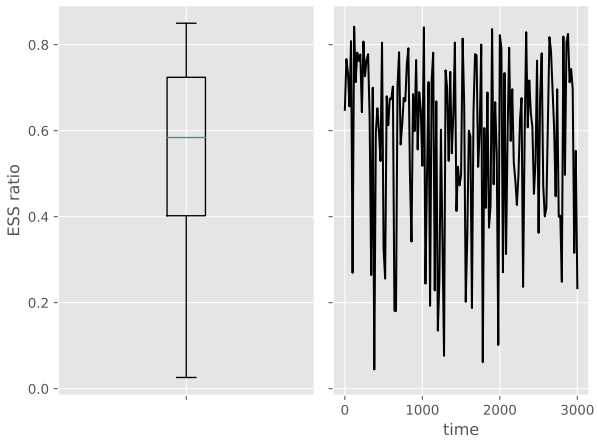

FIGURE 21. Effective sample size (ESS) for the Lotka-Voltterra SDE model with $T = 3000$ (Section 5.2). The left pane draws the box plot of the collection of all estimated ESS for $t = 0, \ldots, 3000$. The right pane plots the evolution of ESS with time. The quantity changes so chaotically that the curve only plots one value every 20 time steps for readability.

Proposition 5. *$\mathbb{Q}_t^N$ is a mixture distribution that admits the representation*

$$\mathbb{Q}_t^N(\mathrm{d}x_{0:t}) = (\ell_t^N)^{-1} N^{-1} \sum_n G_t(x_t) K_t^N(n, \mathrm{d}x_{0:t}) \tag{25}$$

where ℓ_t^N is defined in Algorithm 1 and $K_t^N(n, \mathrm{d}x_{0:t})$ is a certain probability measure satisfying

$$\mathbb{E}\left[K_t^N(n, \mathrm{d}x_{0:t}) \middle| \mathcal{F}_{t-1}^+\right] = \mathbb{Q}_{t-1}^N(\mathrm{d}x_{0:t-1}) M_t(x_{t-1}, \mathrm{d}x_t). \tag{26}$$

In other words, for any (possibly random) function $\varphi_t^N : \mathcal{X}_0 \times \cdots \times \mathcal{X}_t \to \mathbb{R}$ such that $\varphi_t^N(x_{0:t})$ is $\mathcal{F}_{t-1}^+$-measurable, we have

$$\mathbb{E}\left[\int K_t^N(n, \mathrm{d}x_{0:t}) \varphi_t^N(x_{0:t}) \middle| \mathcal{F}_{t-1}^+\right] = \int \mathbb{Q}_{t-1}^N(\mathrm{d}x_{0:t-1}) M_t(x_{t-1}, \mathrm{d}x_t) \varphi_t^N(x_{0:t}).$$

Moreover, $\int K_t^N(n, \mathrm{d}x_{0:t}) \varphi_t^N(x_{0:t})$, for $n = 1, \ldots, N$ are i.i.d. given $\mathcal{F}_{t-1}^+$.

The proof is postponed until the end of this subsection. This proposition gives the expression (25) for $\mathbb{Q}_t^N$, which is easier to manipulate than (24) and which highlights, through (26), its connection to $\mathbb{Q}_{t-1}^N$. To further simplify the notations, let us define, following Douc et al. (2011), the kernel $L_{t_1:t_2}$, for $t_1 \leq t_2$, as

$$L_{t_1:t_2}(x_{0:t_1}^\star, \mathrm{d}x_{0:t_2}) := \delta_{x_{0:t_1}^\star}(\mathrm{d}x_{0:t_1}) \prod_{s=t_1+1}^{t_2} M_s(x_{s-1}, \mathrm{d}x_s) G_s(x_s). \tag{27}$$

In other words, for real-valued functions $\varphi_{t_2} = \varphi_{t_2}(x_0, \ldots, x_{t_2})$, we have

$$L_{t_1:t_2}(x_{0:t_1}^\star, \varphi_{t_2}) = \int \varphi_{t_2}(x_0^\star, \ldots, x_{t_1}^\star, x_{t_1+1}, \ldots, x_{t_2}) \prod_{s=t_1+1}^{t_2} M_s(x_{s-1}, \mathrm{d}x_s) G_s(x_s).$$

The usefulness of these kernels will come from the simple remark $\mathbb{Q}_{t_2} \propto \mathbb{Q}_{t_1} L_{t_1:t_2}$. We also see that

$$\|L_{t_1:t_2}\varphi_{t_2}\|_\infty \leq \|\varphi_{t_2}\|_\infty \prod_{s=t_1+1}^{t_2} \|G_s\|_\infty ,$$

which gives $\|L_{t_1:t_2}\|_\infty < \infty$, where the norm of a kernel is defined in Subsection A. We are now in a position to state an importance sampling-like representation of $\mathbb{Q}_t^N$.

Corollary 3. *Let $\varphi_t^N : \mathcal{X}_0 \times \cdots \times \mathcal{X}_t \to \mathbb{R}$ be a (possibly random) function such that $\varphi_t^N(x_{0:t})$ is $\mathcal{F}_{t-1}^+$-measurable. Suppose that φ_t^N is either uniformly non-negative (i.e. $\varphi_t^N(x_{0:t}) \geq 0$ almost surely) or uniformly bounded (i.e. there exists a deterministic C such that $|\varphi_t^N(x_{0:t})| \leq C$ almost surely). Then*

$$\mathbb{Q}_t^N \varphi_t^N = \frac{N^{-1}\sum_n \tilde{K}_t^N(n, \varphi_t^N)}{N^{-1}\sum_n \tilde{K}_t^N(n, \mathbb{1})},$$

where $\tilde{K}_t^N(n, \cdot)$ is a certain random kernel such that

- $\mathbb{E}\left[\tilde{K}_t^N(n, \varphi_t^N)\middle| \mathcal{F}_{t-1}^+\right] = (\mathbb{Q}_{t-1}^N L_{t-1:t})\varphi_t^N$;
- $N^{-1}\sum_n \tilde{K}_t^N(n, \mathbb{1}) = \ell_t^N$;
- $\left(\tilde{K}_t^N(n, \varphi_t^N)\right)_{n=1,\ldots,N}$ *are i.i.d. given* $\mathcal{F}_{t-1}^+$;
- *almost surely,* $\left|\tilde{K}_t^N(n, \varphi_t^N)\right| \leq \left\|\varphi_t^N\right\|_\infty \|G_t\|_\infty$ *if* φ_t^N *is uniformly bounded and* $\tilde{K}_t^N(n, \varphi_t^N) \geq 0$ *if* φ_t^N *is uniformly non-negative.*

These statements are valid for $t = 0$ under the convention $\mathbb{Q}_{-1}^N L_{-1:0} = \mathbb{Q}_{-1} L_{-1:0} = \mathbb{M}_0$ and $\mathcal{F}_{-1}$ being the trivial σ-algebra.

Proof. Put $\tilde{K}_t^N(n, \varphi_t^N) := \int G_t(x_t) K_t^N(n, \mathrm{d}x_{0:t}) \varphi_t^N(x_{0:t})$ where K_t^N is defined in Proposition 5. Then

$$\begin{aligned}\mathbb{Q}_t^N(\varphi_t^N) &= \frac{N^{-1}\sum_n \int G_t(x_t) K_t^N(n, \mathrm{d}x_{0:t}) \varphi_t^N(x_{0:t})}{\ell_t^N} \\ &= \frac{N^{-1}\sum_n \tilde{K}_t^N(n, \varphi_t^N)}{\ell_t^N}.\end{aligned}$$

Since $\mathbb{Q}_t^N$ is a probability measure, applying this identity twice yields

$$\mathbb{Q}_t^N(\varphi_t^N) = \frac{\mathbb{Q}_t^N(\varphi_t^N)}{\mathbb{Q}_t^N(\mathbb{1})} = \frac{N^{-1}\sum_n \tilde{K}_t^N(n, \varphi_t^N)}{N^{-1}\sum_n \tilde{K}_t^N(n, \mathbb{1})}.$$

The remaining points are simple consequences of the definition of $\tilde{K}_t^N$ and $L_{t-1:t}$. □

The corollary hints at a natural induction proof for Theorem 1.

Proof of Theorem 1. The following calculations are valid for all $T \geq 0$, under the convention defined at the end of Corollary 3. They will prove (8) for $T = 0$ and, at the same time, prove it for any $T \geq 1$ under the hypothesis that it already holds

true for $T-1$. Let $\varphi_T = \varphi_T(x_0, \dots, x_T)$ be a real-valued function on $\mathcal{X}_0 \times \cdots \times \mathcal{X}_T$. Write

$$\sqrt{N}(\mathbb{Q}_T^N \varphi_T - \mathbb{Q}_T \varphi_T) = \sqrt{N}\left(\frac{N^{-1}\sum_n \tilde{K}_T^N(n, \varphi_T)}{N^{-1}\sum_n \tilde{K}_T^N(n, \mathbb{1})} - \frac{\mathbb{Q}_{T-1} L_{T-1:T} \varphi_T}{\mathbb{Q}_{T-1} L_{T-1:T} \mathbb{1}}\right) \tag{28}$$

where the rewriting of $\mathbb{Q}_T \varphi_T$ is a consequence of $Q_T \propto \mathbb{Q}_{T-1} L_{T-1:T}$. We will bound this difference by Hoeffding's inequalities for ratios (see Appendix E.13 for notations, including the definition of sub-Gaussian variables that we shall use below). We have

- that $\sqrt{N}(N^{-1}\sum \tilde{K}_T^N(n, \varphi_T) - \mathbb{Q}_{T-1}^N L_{T-1:T}\varphi_T)$ is $(1, \|\varphi_T\|_\infty \|G_T\|_\infty)$-sub-Gaussian conditioned on $\mathcal{F}_{t-1}^+$ because of Theorem 9 (and thus unconditionally, by the law of total expectation);
- and that $\sqrt{N}(N^{-1}\mathbb{Q}_{T-1}^N L_{T-1:T}\varphi_T - \mathbb{Q}_{T-1} L_{T-1:T}\varphi_T)$ is sub-Gaussian with parameters
$$(C_{T-1}, S_{T-1}\|L_{T-1:T}\|_\infty \|\varphi_T\|_\infty)$$
if $T \geq 1$ by induction hypothesis. The quantity is equal to 0 if $T = 0$.

This permits to apply Lemma 16, which results in the sub-Gaussian properties of

- the quantity $\sqrt{N}(N^{-1}\sum \tilde{K}_T^N(n, \varphi_T) - \mathbb{Q}_{T-1} L_{T-1:T}\varphi_T)$, with parameters $(1 + C_{T-1}, S'_{T-1}\|\varphi_T\|_\infty)$, for a certain constant S'_{T-1};
- and the quantity $\sqrt{N}(N^{-1}\sum \tilde{K}_T^N(n, \mathbb{1}) - \mathbb{Q}_{T-1} L_{T-1:T}\mathbb{1})$, which is a special case of the former one, with parameters $(1 + C_{T-1}, S'_{T-1})$.

Finally, we invoke Proposition 11 and deduce the sub-Gaussian property of (28) with parameters

$$\left(2 + 2C_{T-1}, 2\frac{S'_{T-1}\|\varphi_T\|_\infty}{\mathbb{Q}_{T-1} L_{T-1:T}\mathbb{1}}\right)$$

which finishes the proof. □

Proof of Proposition 5. From (24), we have

$$\begin{aligned}
\mathbb{Q}_t^N(\mathrm{d}x_{0:t}) &= \sum_{i_t} \bar{\mathbb{Q}}_t^N(\mathrm{d}i_t)\bar{\mathbb{Q}}_t^N(\mathrm{d}x_{0:t}|i_t) \\
&= (\ell_t^N)^{-1} N^{-1} \sum_{i_t} G_t(X_t^{i_t})\bar{\mathbb{Q}}_t^N(\mathrm{d}x_{0:t}|i_t) \\
&= (\ell_t^N)^{-1} N^{-1} \sum_{i_t} G_t(x_t)\bar{\mathbb{Q}}_t^N(\mathrm{d}x_{0:t}|i_t)
\end{aligned}$$

since $\bar{\mathbb{Q}}_t^N(dx_{0:t}|i_t)$ has a $\delta_{X_t^{i_t}}(\mathrm{d}x_t)$ term. In fact, the identity

$$\bar{\mathbb{Q}}_t^N(\mathrm{d}x_{0:t}, \mathrm{d}i_{t-1}|i_t) = \delta_{X_t^{i_t}}(\mathrm{d}x_t) B_t^N(i_t, \mathrm{d}i_{t-1})\bar{\mathbb{Q}}_{t-1}^N(\mathrm{d}x_{0:t-1}|i_{t-1})$$

follows directly from the backward recursive nature of Algorithm 2, and thus

$$\bar{\mathbb{Q}}_t^N(\mathrm{d}x_{0:t}|i_t) = \delta_{X_t^{i_t}}(\mathrm{d}x_t) \int_{i_{t-1}} B_t^N(i_t, \mathrm{d}i_{t-1})\bar{\mathbb{Q}}_{t-1}^N(\mathrm{d}x_{0:t-1}|i_{t-1}). \tag{29}$$

The $\mathbb{Q}_{t-1}^N(\mathrm{d}x_{0:t-1}|i_{t-1})$ term is $\mathcal{F}_{t-1}^+$-measurable. We shall calculate the expectation of $\delta_{X_t^{i_t}}(\mathrm{d}x_t)B_t^N(i_t, \mathrm{d}i_{t-1})$ given $\mathcal{F}_{t-1}^+$. The following arguments are necessary for formal verification, but the result (30) is natural in light of the ancestor regeneration intuition explained in Section 2.4.

Let $f_t^N : \{1,\dots,N\} \times \mathcal{X}_t \to \mathbb{R}$ be a (possibly random) function such that $f_t^N(i_{t-1}, x_t)$ is $\mathcal{F}_{t-1}^+$-measurable. Let $J_t^{i_t}$ be a random variable such that given $\mathcal{F}_{t-1}^+$, $X_t^{i_t}$ and $\hat{B}_t^N(i_t, \cdot)$, $J_t^{i_t}$ is $B_t^N(i_t, \mathrm{d}i_{t-1})$-distributed. This automatically makes $J_t^{i_t}$ satisfy the second hypothesis of Theorem 1. Additionally, by virtue of its first hypothesis, the distribution of $(J_t^{i_t}, A_t^{i_t})$ is the same given either $\mathcal{F}_{t-1}^+$ or $X_{t-1}^{1:N}$ (see also Figure 1). We can now write

$$
\begin{aligned}
&\mathbb{E}\left[\int f_t^N(i_{t-1}, x_t)\delta_{X_t^{i_t}}(\mathrm{d}x_t)B_t^N(i_t, \mathrm{d}i_{t-1})\middle|\mathcal{F}_{t-1}^+\right]\\
=&\mathbb{E}\left[\int f_t^N(i_{t-1}, X_t^{i_t})B_t^N(i_t, \mathrm{d}i_{t-1})\middle|\mathcal{F}_{t-1}^+\right]\\
=&\mathbb{E}\left[\mathbb{E}\left[f_t^N(J_t^{i_t}, X_t^{i_t})\middle|\mathcal{F}_{t-1}^+, X_t^{i_t}, \hat{B}_t^N(i_t, \cdot)\right]\middle|\mathcal{F}_{t-1}^+\right]\\
=&\mathbb{E}\left[f_t^N(J_t^{i_t}, X_t^{i_t})\middle|\mathcal{F}_{t-1}^+\right] \text{ by the law of total expectation}\\
=&\mathbb{E}\left[f_t^N(A_t^{i_t}, X_t^{i_t})\middle|\mathcal{F}_{t-1}^+\right] \text{ by the second hypothesis of Theorem 1}\\
=&\int f_t^N(i_{t-1}, x_t)\mathcal{M}(W_{t-1}^{1:N})(\mathrm{d}i_{t-1})M_t(X_{t-1}^{i_{t-1}}, \mathrm{d}x_t).
\end{aligned}
$$

This equality means that

$$
\mathbb{E}\left[\delta_{X_t^{i_t}}(\mathrm{d}x_t)B_t^N(i_t, \mathrm{d}i_{t-1})\middle|\mathcal{F}_{t-1}^+\right] = \mathcal{M}(W_{t-1}^{1:N})(\mathrm{d}i_{t-1})M_t(X_{t-1}^{i_{t-1}}, \mathrm{d}x_t), \tag{30}
$$

Now, put

$$
K^N(i_t, \mathrm{d}x_{0:t}) := \bar{\mathbb{Q}}_t^N(\mathrm{d}x_{0:t}|i_t).
$$

From (29) and (30), we have

$$
\begin{aligned}
\mathbb{E}\left[K^N(i_t, \mathrm{d}x_{0:t})\middle|\mathcal{F}_{t-1}^+\right] &= \int_{i_{t-1}} \mathcal{M}(W_{t-1}^{1:N})(\mathrm{d}i_{t-1})M_t(X_{t-1}^{i_{t-1}}, \mathrm{d}x_t)\bar{\mathbb{Q}}_{t-1}^N(\mathrm{d}x_{0:t-1}|i_{t-1})\\
&= M_t(x_{t-1}, \mathrm{d}x_t)\int_{i_{t-1}} \mathcal{M}(W_{t-1}^{1:N})(\mathrm{d}i_{t-1})\bar{\mathbb{Q}}_{t-1}^N(\mathrm{d}x_{0:t-1}|i_{t-1})\\
&\text{since } \bar{\mathbb{Q}}_{t-1}^N(\mathrm{d}x_{0:t-1}|i_{t-1}) \text{ has a } \delta_{X_{t-1}^{i_{t-1}}(\mathrm{d}x_{t-1})} \text{ term}\\
&= M_t(x_{t-1}, \mathrm{d}x_t)\mathbb{Q}_{t-1}^N(\mathrm{d}x_{0:t-1})
\end{aligned}
$$

which finishes the proof. □

E.2. **Proof of Equation (11) (online smoothing recursion).**

Proof. Using (7) and the matrix notations, the distribution $\bar{\mathbb{Q}}_t^N(\mathrm{d}i_s)$ can be represented by the $1 \times N$ vector

$$
\hat{q}_{s|t}^N := [W_t^1 \dots W_t^N]\hat{B}_t^N \dots \hat{B}_{s+1}^N.
$$

Defining the $N \times N$ matrix $\hat{\psi}_s^N$ as

$$
\hat{\psi}_s^N[i_{s-1}, i_s] := \psi_s(X_{s-1}^{i_{s-1}}, X_s^{i_s}),
$$

we have

$$\begin{aligned}\mathbb{E}_{\mathbb{Q}_t^N}[\psi_s(X_{s-1},X_s)] &= \sum_{i_s,i_{s-1}} \hat{q}^N_{s|t}[1,i_s]\hat{B}^N_s[i_s,i_{s-1}]\hat{\psi}^N_s[i_{s-1},i_s]\\ &= \sum_{i_s} \hat{q}^N_{s|t}[1,i_s](\hat{B}^N_s\hat{\psi}^N_s)[i_s,i_s]\\ &= \hat{q}^N_{s|t}\,\mathrm{diag}(\hat{B}^N_s\hat{\psi}^N_s).\end{aligned}$$

Therefore,

$$\mathbb{Q}^N_t\varphi_t = \sum_{s=0}^{t}[W^1_t\dots W^N_t]\hat{B}^N_t\dots\hat{B}^N_{s+1}\,\mathrm{diag}(\hat{B}^N_s\hat{\psi}^N_s)$$

from which follows the recursion

$$\begin{cases}\mathbb{Q}^N_t\varphi_t &= [W^1_t\dots W^N_t]\hat{S}^N_t,\\ \hat{S}^N_t &:= \hat{B}^N_t\hat{S}^N_{t-1} + \mathrm{diag}(\hat{B}^N_t\hat{\psi}^N_t).\end{cases}$$

This is exactly (11). □

E.3. **Proof of Theorem 2 (general stability theorem).** The following lemma describes the simultaneous backward construction of two trajectories $\mathcal{I}^1_{0:T}$ and $\mathcal{I}^2_{0:T}$ given $\mathcal{F}^-_T$.

Lemma 2. *We use the same notations as in Algorithms 1 and 2. Suppose that the hypotheses of Theorem 1 are satisfied. Then, given $\mathcal{I}^1_{t:T}$, $\mathcal{I}^2_{t:T}$ and $\mathcal{F}^-_T$,*

- *if $\mathcal{I}^1_t \neq \mathcal{I}^2_t$, the two variables $\mathcal{I}^1_{t-1}$ and $\mathcal{I}^2_{t-1}$ are conditionally independent and their marginal distributions are respectively $B^{N,\mathrm{FFBS}}_t(\mathcal{I}^1_t,\cdot)$ and $B^{N,\mathrm{FFBS}}_t(\mathcal{I}^2_t,\cdot)$;*
- *if $\mathcal{I}^1_t = \mathcal{I}^2_t$, under the aforementioned conditioning, the two variables $\mathcal{I}^1_{t-1}$ and $\mathcal{I}^2_{t-1}$ are both marginally distributed according to $B^{N,\mathrm{FFBS}}_t(\mathcal{I}^1_t,\cdot)$. Moreover, if (13) holds, we have*

$$\mathbb{P}\left(\mathcal{I}^1_{t-1}\neq\mathcal{I}^2_{t-1}\middle|\,\mathcal{I}^{1,2}_{t:T},\mathcal{F}^-_T\right)\mathbb{1}_{\mathcal{I}^1_t=\mathcal{I}^2_t} \geq \varepsilon_{\mathrm{S}}\;\mathbb{1}_{\mathcal{I}^1_t=\mathcal{I}^2_t}. \tag{31}$$

In particular, the sequence of variables $(\mathcal{I}^1_{T-s},\mathcal{I}^2_{T-s})_{s=0}^T$ is a Markov chain given $\mathcal{F}^-_T$.

Proof. To simplify the notations, let $\tilde{b}^n_t$ denote the $\mathbb{R}^n$ vector $\hat{B}^N_t(n,\cdot)$. The relation between variables generated by Algorithm 2 is depicted as a graphical model in Figure 22. We consider

$$\begin{aligned}p(\tilde{b}^{1:N}_t, i^{1:2}_{t-1}|\mathcal{F}^-_T, i^{1:2}_{t:T}) &= p(\tilde{b}^{1:N}_t|\mathcal{F}^-_T, i^{1:2}_{t:T})\;p(i^{1:2}_{t-1}|\tilde{b}^{1:N}_t,\mathcal{F}^-_T,i^{1,2}_{t:T})\\ &= p(\tilde{b}^{1:N}_t|x^{1:N}_{t-1},x^{1:N}_t)\;p(i^{1:2}_{t-1}|\tilde{b}^{1:N}_t,i^{1:2}_t)\\ &\qquad\text{(by properties of graphical models, see Figure 22)}\\ &= \left[\prod_n p(\tilde{b}^n_t|x^{1:N}_{t-1},x^n_t)\right]\tilde{b}^{i^1_t}_t(i^1_{t-1})\tilde{b}^{i^2_t}_t(i^2_{t-1}).\end{aligned} \tag{32}$$

The distribution of i^1_{t-1} given $\mathcal{F}^-_T$ and $i^{1:2}_{t:T}$ is thus the i^1_{t-1}-marginal of

$$p(\tilde{b}^{i^1_t}_t|x^{1:N}_{t-1},x^{i^1_t}_t)\tilde{b}^{i^1_t}_t(i^1_{t-1}), \tag{33}$$

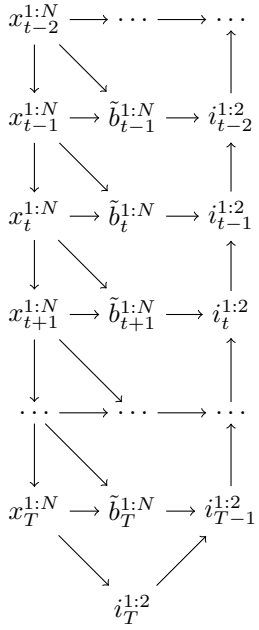

FIGURE 22. Directed graph representing the relations between variables generated in Algorithm 2. Only those necessary for the proof of Lemma 2 are included.

which is exactly the distribution of $p(j_t^{i_t^1}|x_{t-1}^{1:N}, x_t^{i_t^1})$ where the J's are defined in the statement of Theorem 1. By the second hypothesis of that theorem, the aforementioned distribution is equal to $p(a_t^{i_t^1}|x_{t-1}^{1:N}, x_t^{i_t^1})$, which is in turn no other than $B_t^{N,\mathrm{FFBS}}(i_t^1, \cdot)$. Moreover, if $i_t^1 \neq i_t^2$, (32) straightforwardly implies the conditional independence of i_{t-1}^1 and i_{t-1}^2. When $i_t^1 = i_t^2$, the distribution of $i_{t-1}^{1:2}$ given $\mathcal{F}_T^-$ and $i_{t:T}^{1:2}$ is the $i_{t-1}^{1:2}$-marginal of

$$p(\tilde{b}_t^{i_t^1}|x_{t-1}^{1:N}, x_t^{i_t^1})\tilde{b}_t^{i_t^1}(i_{t-1}^1)\tilde{b}_t^{i_t^1}(i_{t-1}^2).$$

Thus, we can apply (13) for $n = i_t^1$, where $i_{t-1}^{1:2}$ here plays the role of $J_t^{1:2}$ there. Equation (31) is now proved. □

As Lemma 2 describes the distribution of two trajectories, it immediately gives the distribution of a single trajectory.

Corollary 4. *Under the same settings as in Lemma 2, given $\mathcal{F}_T^-$, the distribution of $\mathcal{I}_{0:T}^1$ is*

$$\mathcal{M}(W_T^{1:N})(\mathrm{d}i_T)B_T^{N,\mathrm{FFBS}}(i_T, \mathrm{d}i_{T-1})\dots B_1^{N,\mathrm{FFBS}}(i_1, \mathrm{d}i_0).$$

Note that the corollary applies even if the backward kernel used in Algorithm 2 is *not* the FFBS one. This is due to the conditioning on $\mathcal{F}_T^-$ and the second hypothesis of Theorem 1.

Proof of Theorem 2. First of all, we remark that as per Algorithm 2, using index variables $\mathcal{I}_{0:T}^{1:N}$ adds a level of Monte Carlo approximation to $\mathbb{Q}_T^N(\mathrm{d}x_{0:T})$. Therefore

$$\begin{aligned}\mathbb{E}\left[(\mathbb{Q}_T^N(\varphi_T)-\mathbb{Q}_T(\varphi_T))^2\right] &= \mathbb{E}\left[\left(\frac{1}{N}\sum_{n=1}^N \varphi_T(X_0^{\mathcal{I}_0^n},\dots,X_T^{\mathcal{I}_T^n})-\mathbb{Q}_T(\varphi_T)\right)^2\right] \\ &= \mathbb{E}\left[(\mathbb{Q}_T^{N,\mathrm{FFBS}}(\varphi_T)-\mathbb{Q}_T(\varphi_T))^2\right]+ \\ &\quad +\mathbb{E}\left[\mathrm{Var}\left(\frac{1}{N}\sum_{n=1}^N \varphi_T(X_0^{\mathcal{I}_0^n},\dots,X_T^{\mathcal{I}_T^n})\middle|\mathcal{F}_T^-\right)\right] \end{aligned} \tag{34}$$

where the ultimate inequality is justified by the law of total expectation and Corollary 4. (Note that $(\mathcal{I}_{0:T}^n)_{n=1}^N$ are identically distributed but *not* necessarily independent given $\mathcal{F}_T^-$.) Using Lemma 4 (stated and proved below) and putting $\rho := 1-\bar{M}_\ell/\bar{M}_h$, we have

$$\begin{aligned}&\mathrm{Var}\left(\frac{1}{N}\sum_{n=1}^N \varphi_T(X_0^{\mathcal{I}_0^n},\dots,X_T^{\mathcal{I}_T^n})\middle|\mathcal{F}_T^-\right) \\ &= \mathrm{Var}\left(\frac{1}{N}\sum_{n=1}^N\sum_{t=0}^T \psi_t(X_{t-1}^{\mathcal{I}_{t-1}^n},X_t^{\mathcal{I}_t^n})\middle|\mathcal{F}_T^-\right) \\ &= \frac{1}{N^2}\sum_{n,m\le N}\sum_{s,t\le T}\mathrm{Cov}\left(\psi_t(X_{t-1}^{\mathcal{I}_{t-1}^n},X_t^{\mathcal{I}_t^n}),\psi_s(X_{s-1}^{\mathcal{I}_{s-1}^m},X_s^{\mathcal{I}_s^m})\middle|\mathcal{F}_T^-\right) \\ &\le \frac{2}{N^2}\sum_{\substack{n,m\le N\\ n=m}}\sum_{s,t\le T}\|\psi_t\|_\infty\|\psi_s\|_\infty\rho^{|t-s|-1}+ \\ &\quad +\frac{4}{N^2}\sum_{\substack{n,m\le N\\ n\neq m}}\sum_{s,t\le T}\frac{\tilde{C}}{N}\|\psi_t\|_\infty\|\psi_s\|_\infty\rho^{|t-s|-1} \\ &= \left(\sum_{s,t\le T}2\|\psi_t\|_\infty\|\psi_s\|_\infty\rho^{|t-s|-1}\right)\frac{(2\tilde{C}+1)N-2\tilde{C}}{N^2} \\ &\le \left[\sum_{s,t\le T}\left(\|\psi_t\|_\infty^2+\|\psi_s\|_\infty^2\right)\rho^{|t-s|-1}\right]\frac{2\tilde{C}+1}{N} \le \frac{4(2\tilde{C}+1)}{N\rho(1-\rho)}\sum\|\psi_t\|_\infty^2 . \end{aligned} \tag{35}$$

We now look at the first term of (34). In the fixed marginal smoothing case, for any $s\in\mathbb{Z}_+$, $s\le T$ and any function $\phi_s:\mathcal{X}_s\to\mathbb{R}$, Douc et al. (2011) proved that

$$\mathbb{P}\left(\left|\mathbb{Q}_T^{N,\mathrm{FFBS}}(\varphi_T)-\mathbb{Q}_T(\varphi_T)\right|\ge\varepsilon\right)\le B'e^{-C'N\varepsilon^2/\|\phi_s\|_\infty^2}$$

for $\varphi_T(x_{0:T})=\phi_s(x_s)$ and constants B' and C' not depending on T. Using $\mathbb{E}[\Delta^2]=\int_0^\infty\mathbb{P}(\Delta^2\ge t)\mathrm{d}t$, the inequality implies

$$\mathbb{E}\left[\left|\mathbb{Q}_T^{N,\mathrm{FFBS}}(\varphi_T)-\mathbb{Q}_T(\varphi_T)\right|^2\right]\le\frac{B'\|\phi_s\|_\infty^2}{C'N} \tag{36}$$

for $\varphi_T(x_{0:T}) = \phi_s(x_s)$. In the additive smoothing case, Dubarry and Le Corff (2013) proved that, for $T \geq 2$,

$$\mathbb{E}\left[\left|\mathbb{Q}_T^{N,\mathrm{FFBS}}(\varphi_T) - \mathbb{Q}_T(\varphi_T)\right|^2\right] \leq \frac{C'}{N}\left(\sum_{t=0}^{T}\|\psi_t\|_\infty^2\right)\left(1+\sqrt{\frac{T}{N}}\right)^2. \tag{37}$$

Equations (36), (37), (35) and (34) conclude the proof. □

The following lemma quantifies the backward mixing property induced by Assumption 2.

Lemma 3. *Under the same setting as Theorem 2, we have*

$$\mathrm{TV}\left(B_t^{N,\mathrm{FFBS}}(m,\cdot), B_t^{N,\mathrm{FFBS}}(n,\cdot)\right) \leq 1 - \frac{\bar{M}_\ell}{\bar{M}_h}$$

for all $m, n \in \{1,\dots,N\}$ *and* $t \in \{1,\dots,T\}$.

Proof. We have

$$\begin{aligned}
&1 - \mathrm{TV}\left(B_t^{N,\mathrm{FFBS}}(m,\cdot), B_t^{N,\mathrm{FFBS}}(n,\cdot)\right) \\
&\quad= \left[\sum_{i=1}^{N} \min\left(\frac{G_{t-1}(X_{t-1}^i)m_t(X_{t-1}^i, X_t^m)}{\sum_{j=1}^N G_{t-1}(X_{t-1}^j)m_t(X_{t-1}^j, X_t^m)},\right.\right. \\
&\qquad\left.\left.\frac{G_{t-1}(X_{t-1}^i)m_t(X_{t-1}^i, X_t^n)}{\sum_{j=1}^N G_{t-1}(X_{t-1}^j)m_t(X_{t-1}^j, X_t^n)}\right)\right] \text{ by Lemma 1 (Appendix A.2)} \\
&\quad\geq \left[\sum_{i=1}^{N} \frac{G_t(X_{t-1}^i)\bar{M}_\ell}{\sum_{j=1}^N G_t(X_{t-1}^j)\bar{M}_h}\right] \text{ by Assumption 2} \\
&\quad= (\bar{M}_\ell/\bar{M}_h).
\end{aligned}$$

□

Lemma 4. *Under the same settings as in Theorem 2, for any* $m, n \in \{1,\dots,N\}$ *and* $s, s' \in \{0,\dots,T\}$, *we have*

$$\begin{aligned}
&\mathrm{Cov}\left(\psi_s(X_{s-1}^{\mathcal{I}_{s-1}^m}, X_s^{\mathcal{I}_s^m}), \psi_{s'}(X_{s'-1}^{\mathcal{I}_{s'-1}^n}, X_{s'}^{\mathcal{I}_{s'}^n}) \middle| \mathcal{F}_T^-\right) \\
&\qquad\leq 2\left(1-\frac{\bar{M}_\ell}{\bar{M}_h}\right)^{|s-s'|-1}\|\psi_s\|_\infty\|\psi_{s'}\|_\infty \times \begin{cases} 2\tilde{C}/N & \text{if } m \neq n \\ 1 & \text{if } m = n \end{cases}
\end{aligned} \tag{38}$$

where $\tilde{C} = \tilde{C}(\bar{M}_\ell, \bar{M}_h, \bar{G}_\ell, \bar{G}_h, \varepsilon_{\mathrm{S}})$ *is a constant that does not depend on* T *(and which arises in the formulation of Lemma 5). If* s *or* s' *is equal to* 0, *we adopt the natural convention* $\psi_0(x_{-1}, x_0) := \psi_0(x_0)$.

Proof. We first handle the case $m \neq n$. Without loss of generality, assume that $m = 1$, $n = 2$ and $s \geq s'$. The covariance bound of Lemma 1 yields

$$\begin{aligned}
&\mathrm{Cov}\left(\psi_s(X_{s-1}^{\mathcal{I}_{s-1}^1}, X_s^{\mathcal{I}_s^1}), \psi_{s'}(X_{s'-1}^{\mathcal{I}_{s'-1}^2}, X_{s'}^{\mathcal{I}_{s'}^2}) \middle| \mathcal{F}_T^-\right) \\
&\quad\leq 2\|\psi_s\|_\infty\|\psi_{s'}\|_\infty \mathrm{TV}\left((\mathcal{I}_{s-1:s}^1, \mathcal{I}_{s'-1:s'}^2)|\mathcal{F}_T^-, (\mathcal{I}_{s-1:s}^1|\mathcal{F}_T^-) \otimes (\mathcal{I}_{s'-1:s'}^2|\mathcal{F}_T^-)\right).
\end{aligned} \tag{39}$$

We shall bound this total variation distance via the coupling inequality of Lemma 1 (Appendix A.2). The idea is to construct, in addition to $\mathcal{I}^1_{0:T}$ and $\mathcal{I}^2_{0:T}$, two trajectories $\mathcal{I}^{*1}_{0:T}$ and $\mathcal{I}^{*2}_{0:T}$ i.i.d. given $\mathcal{F}^-_T$ such that each of them is conditionally distributed according to $\mathcal{I}^1_{0:T}$ (cf. Corollary 4). To make the coupling inequality efficient, it is desirable to make $\mathcal{I}^1_{0:T}$ and $\mathcal{I}^{*1}_{0:T}$ as similar as possible (same thing for $\mathcal{I}^2_{0:T}$ and $\mathcal{I}^{*2}_{0:T}$).

The detailed construction of the four trajectories $\mathcal{I}^1_{0:T}$, $\mathcal{I}^2_{0:T}$, $\mathcal{I}^{*1}_{0:T}$ and $\mathcal{I}^{*2}_{0:T}$ *given* $\mathcal{F}^-_T$ is described in Algorithm 15. In particular, we ensure that $\forall t \geq s-1$, we have $\mathcal{I}^1_t = \mathcal{I}^{*1}_t$. For $t \leq s-1$, if $\mathcal{I}^2_t = \mathcal{I}^{*2}_t$, it is guaranteed that $\mathcal{I}^2_\ell = \mathcal{I}^{*2}_\ell$ holds $\forall \ell \leq t$. The rationale for different coupling behaviours between the times $t \geq s-1$ and $t \leq s-1$ will become clear in the proof: the former aim to control the correlation between two different trajectories $m = 1$ and $n = 2$ and result in the $\tilde{C}/N$ term of (38); the latter are for bounding the correlation between two times s and s' and result in the $(1 - \bar{M}_\ell/\bar{M}_h)^{|s-s'|-1}$ term of the same equation.

Algorithm 15: Sampler for the variables $\mathcal{I}^1_{0:T}$, $\mathcal{I}^2_{0:T}$, $\mathcal{I}^{*1}_{0:T}$ and $\mathcal{I}^{*2}_{0:T}$ (see proof of Lemma 4)

Input: Feynman-Kac model (1), variables $X^{1:N}_{0:T}$ from the output of Algorithm 1, integer $s \geq 0$ (see statement of Lemma 4)

Sample $\mathcal{I}^1_T, \mathcal{I}^2_T \overset{i.i.d.}{\sim} \mathcal{M}(W^{1:N}_T)$

Set $\mathcal{I}^{*1}_T \leftarrow \mathcal{I}^1_T$ and $\mathcal{I}^{*2}_T \leftarrow \mathcal{I}^2_T$

for $t \leftarrow T$ **to** 1 **do**

  if $\mathcal{I}^1_t \neq \mathcal{I}^2_t$ **then**

    for $k \in \{1, 2\}$ **do**

      Sample $(\mathcal{I}^k_{t-1}, \mathcal{I}^{*k}_{t-1})$ from any maximal coupling of $B^{N,\mathrm{FFBS}}_t(\mathcal{I}^k_t, \cdot)$ and $B^{N,\mathrm{FFBS}}_t(\mathcal{I}^{*k}_t, \cdot)$ (cf. Lemma 1)

  else

    Sample the $\mathbb{R}^N$ vector $\hat{B}^N_t(\mathcal{I}^1_t, \cdot)$ from $p(\hat{b}^N_t(i^1_t, \cdot) | x^{1:N}_{t-1}, x^{i^1_t}_t)$

    Sample $\mathcal{I}^1_{t-1}, \mathcal{I}^2_{t-1} \overset{i.i.d.}{\sim} \hat{B}^N_t(\mathcal{I}^1_t, \cdot)$

    Set $k \leftarrow 1, \ell \leftarrow 2$ if $t \geq s$ and $k \leftarrow 2, \ell \leftarrow 1$ otherwise

    Sample $\mathcal{I}^{*k}_{t-1} \sim B^{N,\mathrm{FFBS}}_t(\mathcal{I}^{*k}_t, \cdot)$ such that $(\mathcal{I}^{*k}_{t-1}, \mathcal{I}^k_{t-1})$ is any maximal coupling of $B^{N,\mathrm{FFBS}}_t(\mathcal{I}^{*k}_t, \cdot)$ and $B^{N,\mathrm{FFBS}}_t(\mathcal{I}^k_t, \cdot)$ given $\mathcal{I}^{1:2}_{t:T}$, $\mathcal{I}^{*1:2}_{t:T}$ and $\mathcal{F}^-_T$ ($(\star)$ - see text for validity of this step)

    Sample $\mathcal{I}^{*\ell}_{t-1} \sim B^{N,\mathrm{FFBS}}_t(\mathcal{I}^{*\ell}_t, \cdot)$

Output: Four trajectories $\mathcal{I}^1_{0:T}$, $\mathcal{I}^2_{0:T}$, $\mathcal{I}^{*1}_{0:T}$, $\mathcal{I}^{*2}_{0:T}$ to be used in the proof of Lemma 4

The correctness of Algorithm 15 is asserted by Lemma 2. Step $(\star)$ is valid because that lemma states that the distribution of $\mathcal{I}^k_{t-1}$ given $\mathcal{F}^-_T$, $\mathcal{I}^{1,2}_{t:T}$ and $\mathcal{I}^{*1,2}_{t:T}$ is $B^{N,\mathrm{FFBS}}_t(\mathcal{I}^k_t, \cdot)$. Furthermore, we note that $(R_{T-t})^T_{t=0}$ where

$$R_t := (\mathcal{I}^1_t, \mathcal{I}^2_t, \mathcal{I}^{*1}_t, \mathcal{I}^{*2}_t),$$

is a Markov chain given $\mathcal{F}^-_T$.

From (39), applying the coupling inequality of Lemma 1 gives

$$\begin{aligned}(40)\quad &\operatorname{Cov}\left(\psi_s(X_{s-1}^{\mathcal{I}^1_{s-1}}, X_s^{\mathcal{I}^1_s}), \psi_{s'}(X_{s'-1}^{\mathcal{I}^2_{s'-1}}, X_{s'}^{\mathcal{I}^2_{s'}}) \middle| \mathcal{F}_T^-\right) \\ &\quad \leq 2\,\|\psi_s\|_\infty \|\psi_{s'}\|_\infty \mathbb{P}\left((\mathcal{I}^1_{s-1:s}, \mathcal{I}^2_{s'-1:s'}) \neq (\mathcal{I}^{*1}_{s-1:s}, \mathcal{I}^{*2}_{s'-1:s'}) \middle| \mathcal{F}_T^-\right) \\ &\quad = 2\,\|\psi_s\|_\infty \|\psi_{s'}\|_\infty \mathbb{P}\left(\mathcal{I}^2_{s'-1:s'} \neq \mathcal{I}^{*2}_{s'-1:s'} \middle| \mathcal{F}_T^-\right)\end{aligned}$$

where the last equality results from the construction of Algorithm 15. The sub-case $s = s'$ following directly from Lemma 5, we now focus on the sub-case $s \geq s' + 1$. For all $t \leq s - 1$,

$$\begin{aligned}(41)\quad &\mathbb{P}\left(\mathcal{I}^2_{t-1} \neq \mathcal{I}^{*2}_{t-1} \middle| \mathcal{F}_T^-\right) \\ &\quad = \mathbb{P}\left(\mathcal{I}^2_{t-1} \neq \mathcal{I}^{*2}_{t-1}, \mathcal{I}^2_t \neq \mathcal{I}^{*2}_t \middle| \mathcal{F}_T^-\right) \\ &\quad \text{by construction of Algorithm 15} \\ &\quad = \mathbb{E}\left[\mathbb{P}\left(\mathcal{I}^2_{t-1} \neq \mathcal{I}^{*2}_{t-1}, \mathcal{I}^2_t \neq \mathcal{I}^{*2}_t \middle| R_t, \mathcal{F}_T^-\right) \middle| \mathcal{F}_T^-\right] \\ &\quad \text{by the law of total expectation} \\ &\quad = \mathbb{E}\left[\operatorname{TV}\left(B_t^{N,\mathrm{FFBS}}(\mathcal{I}^2_t, \cdot), B_t^{N,\mathrm{FFBS}}(\mathcal{I}^{*2}_t, \cdot)\right) \mathbb{1}\left\{\mathcal{I}^2_t \neq \mathcal{I}^{*2}_t\right\} \middle| \mathcal{F}_T^-\right] \\ &\quad \leq \left(1 - \frac{\bar{M}_\ell}{\bar{M}_h}\right) \mathbb{P}\left(\mathcal{I}^2_t \neq \mathcal{I}^{*2}_t \middle| \mathcal{F}_T^-\right) \text{ by Lemma 3.}\end{aligned}$$

Thus

$$\begin{aligned}&\mathbb{P}\left(\mathcal{I}^2_{s'-1:s'} \neq \mathcal{I}^{*2}_{s'-1:s'} \middle| \mathcal{F}_T^-\right) \\ &\quad = \mathbb{P}\left(\mathcal{I}^2_{s'} \neq \mathcal{I}^{*2}_{s'} \middle| \mathcal{F}_T^-\right) \text{ by construction of Algorithm 15} \\ &\quad \leq \left(1 - \frac{\bar{M}_\ell}{\bar{M}_h}\right)^{s-s'-1} \mathbb{P}\left(\mathcal{I}^2_{s-1} \neq \mathcal{I}^{*2}_{s-1} \middle| \mathcal{F}_T^-\right) \text{ by applying (41) recursively} \\ &\quad \leq \left(1 - \frac{\bar{M}_\ell}{\bar{M}_h}\right)^{s-s'-1} \frac{\tilde{C}}{N} \text{ by Lemma 5,}\end{aligned}$$

which, combined with (40) finishes the proof for the current sub-case $s \geq s' + 1$. It remains to show (38) when $m = n$. The proof follows the same lines as in the case $m \neq n$, although we shall briefly outline some arguments to show how the factor $\tilde{C}/N$ disappeared. The case $s = s'$ being trivial, suppose that $s \geq s' + 1$ and without loss of generality that $m = n = 3$. To use the coupling tools of Lemma 1, we construct trajectories $\mathcal{I}^3_{0:T}$, $\mathcal{I}^{*3}_{0:T}$ and $\mathcal{I}^{*4}_{0:T}$ via Algorithm 16 and write, in the spirit of (40):

$$\begin{aligned}(42)\quad &\operatorname{Cov}\left(\psi_s(X_{s-1}^{\mathcal{I}^3_{s-1}}, X_s^{\mathcal{I}^3_s}), \psi_{s'}(X_{s'-1}^{\mathcal{I}^3_{s-1}}, X_{s'}^{\mathcal{I}^3_{s'}}) \middle| \mathcal{F}_T^-\right) \\ &\quad \leq 2\,\|\psi_s\|_\infty \|\psi_{s'}\|_\infty \mathbb{P}\left((\mathcal{I}^3_{s-1:s}, \mathcal{I}^3_{s'-1:s'}) \neq (\mathcal{I}^{*3}_{s-1:s}, \mathcal{I}^{*4}_{s'-1:s'}) \middle| \mathcal{F}_T^-\right) \\ &\quad = 2\,\|\psi_s\|_\infty \|\psi_{s'}\|_\infty \mathbb{P}\left(\mathcal{I}^3_{s'} \neq \mathcal{I}^{*4}_{s'} \middle| \mathcal{F}_T^-\right)\end{aligned}$$

Algorithm 16: Sampler for the variables $\mathcal{I}^3_{0:T}$, $\mathcal{I}^{*3}_{0:T}$ and $\mathcal{I}^{*4}_{0:T}$ (see proof of Lemma 4)

Input: Feynman-Kac model (1), variables $X^{1:N}_{0:T}$ from the output of Algorithm 1, integer $s \geq 0$ (see statement of Lemma 4)

Sample $\mathcal{I}^{*3}_T, \mathcal{I}^{*4}_T \overset{i.i.d.}{\sim} \mathcal{M}(W^{1:N}_T)$
Set $\mathcal{I}^3_T \leftarrow \mathcal{I}^{*3}_T$
for $t \leftarrow T$ **to** *1* **do**
  if $t \geq s$ **then**
    Sample $\mathcal{I}^{*3}_{t-1} \sim B^{N,\mathrm{FFBS}}_t(\mathcal{I}^{*3}_t, \cdot)$ and $\mathcal{I}^{*4}_{t-1} \sim B^{N,\mathrm{FFBS}}_t(\mathcal{I}^{*4}_t, \cdot)$
    Set $\mathcal{I}^3_{t-1} \leftarrow \mathcal{I}^{*3}_{t-1}$
  else
    Sample $(\mathcal{I}^3_{t-1}, \mathcal{I}^{*4}_{t-1})$ from a maximal coupling of $B^{N,\mathrm{FFBS}}_t(\mathcal{I}^3_t, \cdot)$ and $B^{N,\mathrm{FFBS}}_t(\mathcal{I}^{*4}_t, \cdot)$
    Sample $\mathcal{I}^{*3}_{t-1} \sim B^{N,\mathrm{FFBS}}_t(\mathcal{I}^{*3}_t, \cdot)$

Output: Three trajectories $\mathcal{I}^3_{0:T}$, $\mathcal{I}^{*3}_{0:T}$ and $\mathcal{I}^{*4}_{0:T}$ to be used in the proof of Lemma 4

where the last equality follows from the construction of Algorithm 16 and the hypothesis $s \geq s' + 1$. For all $t \leq s - 1$, the inequality

$$\mathbb{P}\left(\mathcal{I}^3_{t-1} \neq \mathcal{I}^{*4}_{t-1} \middle| \mathcal{F}^-_T\right) \leq \left(1 - \frac{\bar{M}_\ell}{\bar{M}_h}\right) \mathbb{P}\left(\mathcal{I}^3_t \neq \mathcal{I}^{*4}_t \middle| \mathcal{F}^-_T\right) \tag{43}$$

can be proved using the same techniques as those used to prove (41): applying Lemma 3 given $(\mathcal{I}^3_t, \mathcal{I}^{*3}_t, \mathcal{I}^{*4}_t)$ then invoking the law of total expectation. Repeatedly instantiating (43) gives

$$\begin{aligned}\mathbb{P}\left(\mathcal{I}^3_{s'} \neq \mathcal{I}^{*4}_{s'} \middle| \mathcal{F}^-_T\right) &\leq \left(1 - \frac{\bar{M}_\ell}{\bar{M}_h}\right)^{s-s'-1} \mathbb{P}\left(\mathcal{I}^3_{s-1} \neq \mathcal{I}^{*4}_{s-1} \middle| \mathcal{F}^-_T\right) \\ &\leq \left(1 - \frac{\bar{M}_\ell}{\bar{M}_h}\right)^{s-s'-1}\end{aligned}$$

which, when plugged into (42), finishes the proof. □

Lemma 5. *For $\mathcal{I}^2_s$ and $\mathcal{I}^{2*}_s$ defined by the output of Algorithm 15, we have*

$$\mathbb{P}\left(\mathcal{I}^2_s \neq \mathcal{I}^{*2}_s | \mathcal{F}^-_T\right) \leq \tilde{C}/N, \text{ and}$$
$$\mathbb{P}\left(\mathcal{I}^2_{s-1} \neq \mathcal{I}^{*2}_{s-1} \middle| \mathcal{F}^-_T\right) \leq \tilde{C}/N, \text{ if } s \geq 1,$$

for some constant $\tilde{C} = \tilde{C}(\bar{M}_\ell, \bar{M}_h, \bar{G}_\ell, \bar{G}_h, \varepsilon_{\mathrm{S}})$.

Proof. Define $A_t := \mathbb{1}\left\{\mathcal{I}^1_t \neq \mathcal{I}^2_t\right\}$, $B_t := \mathbb{1}\left\{\mathcal{I}^2_t = \mathcal{I}^{*2}_t\right\}$ and $\Gamma_t := A_t B_t$ and recall that $R_t := (\mathcal{I}^1_t, \mathcal{I}^2_t, \mathcal{I}^{*1}_t, \mathcal{I}^{*2}_t)$. The sequence $(R_{T-\ell})^T_{\ell=0}$ is a Markov chain given $\mathcal{F}^-_T$, but this is not necessarily the case for the sequence $(\Gamma_{T-\ell})^T_{\ell=0}$ of Bernoulli random variables. Nevertheless, Lemma 6 below shows that one can get bounds on two-step "transition probabilities" for $(\Gamma_{T-\ell})$, i.e. the probabilities under $\mathcal{F}^-_T$ that $\Gamma_{t-2} = 1$ given Γ_t *and* R_t. This motivates our following construction of actual Markov chains

approximating the dynamic of Γ_t. Let Γ^*_T and Γ^*_{T-1} be two independent Bernoulli random variables given $\mathcal{F}^-_T$ such that

$$\begin{aligned}\mathbb{P}\left(\Gamma^*_T = 1 \middle| \mathcal{F}^-_T\right) &= \mathbb{P}\left(\Gamma_T = 1 \middle| \mathcal{F}^-_T\right)\\ \mathbb{P}\left(\Gamma^*_{T-1} = 1 \middle| \mathcal{F}^-_T\right) &= \mathbb{P}\left(\Gamma_{T-1} = 1 \middle| \mathcal{F}^-_T\right).\end{aligned} \tag{44}$$

Let $\Gamma^*_T, \Gamma^*_{T-2}, \Gamma^*_{T-4}, \dots$ and $\Gamma^*_{T-1}, \Gamma^*_{T-3}, \dots$ be two homogeneous Markov chains given $\mathcal{F}^-_T$ with the same transition kernel $\overleftarrow{\mathfrak{C}^2}$ defined by

$$\begin{aligned}\mathbb{P}_{\mathcal{F}^-_T}(\Gamma^*_{t-2} = 1 | \Gamma^*_t = 1) &= 1 - \frac{2}{N}\left(\frac{\bar{G}_h \bar{M}_h}{\bar{G}_\ell \bar{M}_\ell}\right)^2 =: \overleftarrow{\mathfrak{C}^2}_{11}\\ \mathbb{P}_{\mathcal{F}^-_T}(\Gamma^*_{t-2} = 1 | \Gamma^*_t = 0) &= \frac{\bar{M}_\ell \varepsilon_{\mathrm{S}}}{2\bar{M}_h} \qquad\qquad =: \overleftarrow{\mathfrak{C}^2}_{01}\end{aligned} \tag{45}$$

where for two events E_1, E_2, the notation $\mathbb{P}_{\mathcal{F}^-_T}(E_1|E_2)$ is the ratio between $\mathbb{P}\left(E_1, E_2 \middle| \mathcal{F}^-_T\right)$ and $\mathbb{P}\left(E_2 \middle| \mathcal{F}^-_T\right)$. We shall now prove by backward induction the following statement:

$$\mathbb{P}\left(\Gamma_t = 1 \middle| \mathcal{F}^-_T\right) \geq \mathbb{P}\left(\Gamma^*_t = 1 \middle| \mathcal{F}^-_T\right), \forall t \geq s-1. \tag{46}$$

Firstly, (46) holds for $t = T$ and $t = T-1$. Now suppose that it holds for some $t \geq s+1$ and we wish to justify it for $t-2$. By Lemma 6,

$$\begin{aligned}\mathbb{P}\left(\Gamma_{t-2} = 1 \middle| R_t, \mathcal{F}^-_T\right) \mathbb{1}_{\Gamma_t = 1} &\geq \overleftarrow{\mathfrak{C}^2}_{11} \mathbb{1}_{\Gamma_t = 1}\\ \mathbb{P}\left(\Gamma_{t-2} = 1 \middle| R_t, \mathcal{F}^-_T\right) \mathbb{1}_{\Gamma_t = 0} &\geq \overleftarrow{\mathfrak{C}^2}_{01} \mathbb{1}_{\Gamma_t = 0}.\end{aligned}$$

Applying the law of total expectation gives

$$\begin{aligned}\mathbb{P}\left(\Gamma_{t-2} = 1 \middle| \mathcal{F}^-_T\right) &\geq \overleftarrow{\mathfrak{C}^2}_{11} \mathbb{P}\left(\Gamma_t = 1 \middle| \mathcal{F}^-_T\right) + \overleftarrow{\mathfrak{C}^2}_{01} \mathbb{P}\left(\Gamma_t = 0 \middle| \mathcal{F}^-_T\right)\\ &= \left(\overleftarrow{\mathfrak{C}^2}_{11} - \overleftarrow{\mathfrak{C}^2}_{01}\right) \mathbb{P}\left(\Gamma_t = 1 \middle| \mathcal{F}^-_T\right) + \overleftarrow{\mathfrak{C}^2}_{01}\\ &\geq \left(\overleftarrow{\mathfrak{C}^2}_{11} - \overleftarrow{\mathfrak{C}^2}_{01}\right) \mathbb{P}\left(\Gamma^*_t = 1 \middle| \mathcal{F}^-_T\right) + \overleftarrow{\mathfrak{C}^2}_{01}\\ &\quad \text{if } N \text{ is large enough, by induction hypothesis}\\ &= \mathbb{P}\left(\Gamma^*_{t-2} = 1 \middle| \mathcal{F}^-_T\right)\end{aligned}$$

and (46) is now proved. To finish the proof of the lemma, it is necessary to lower bound its right hand side. We start by controlling the distribution Γ^*_t for $t = T$ and $t = T-1$. We have

$$\begin{aligned}\mathbb{P}\left(\Gamma^*_T = 1 \middle| \mathcal{F}^-_T\right) &= \mathbb{P}\left(\Gamma_T = 1 \middle| \mathcal{F}^-_T\right) \text{ by (44)}\\ &= 1 - \mathbb{P}\left(A_T = 0 \middle| \mathcal{F}^-_T\right) \text{ as } B_T = 1 \text{ by Algorithm 15}\\ &= 1 - \sum_{i=1}^N \mathbb{P}\left(\mathcal{I}^1_T = \mathcal{I}^2_T = i \middle| \mathcal{F}^-_T\right)\\ &= 1 - \sum_{i=1}^N \left(\frac{G(X^i_T)}{\sum_{j=1}^N G(X^j_T)}\right)^2\\ &\geq 1 - \frac{1}{N}\left(\frac{\bar{G}_h}{\bar{G}_\ell}\right)^2 \text{ by Assumption 3}\end{aligned} \tag{47}$$

and

$$
\begin{aligned}
\mathbb{P}\left(\Gamma^*_{T-1}=1\middle|\,\mathcal{F}_T^-\right) &\geq \mathbb{P}\left(\Gamma_T=1,\Gamma_{T-1}=1|\,\mathcal{F}_T^-\right)\\
&= \mathbb{E}\left[\mathbb{P}\left(\Gamma_T=1,\Gamma_{T-1}=1|\,R_T,\mathcal{F}_T^-\right)\middle|\,\mathcal{F}_T^-\right]\\
&\quad \text{by the law of total expectation}\\
&= \mathbb{E}\left[\mathbb{P}\left(\Gamma_{T-1}=1|\,R_T,\mathcal{F}_T^-\right)\mathbb{1}_{\Gamma_T=1}\middle|\,\mathcal{F}_T^-\right]\\
&\geq \left[1-\frac{1}{N}\left(\frac{\bar{G}_h\bar{M}_h}{\bar{G}_\ell\bar{M}_\ell}\right)^2\right]\mathbb{P}\left(\Gamma_T=1|\,\mathcal{F}_T^-\right)\ \text{via (53)}\\
&\geq \left[1-\frac{1}{N}\left(\frac{\bar{G}_h\bar{M}_h}{\bar{G}_\ell\bar{M}_\ell}\right)^2\right]\left[1-\frac{1}{N}\left(\frac{\bar{G}_h}{\bar{G}_\ell}\right)^2\right].
\end{aligned}
\tag{48}
$$

The contraction property of Lemma 1 makes it possible to relate the intermediate distributions $\Gamma^*_t|\mathcal{F}_T^-$ to the end point ones $\Gamma^*_{T-1}|\mathcal{F}_T^-$ and $\Gamma^*_T|\mathcal{F}_T^-$. More specifically, (45) and Lemma 1 lead to

$$
\mathrm{TV}(\Gamma^*_t|\mathcal{F}_T^-,\mu^*) \leq \max\left(\mathrm{TV}(\Gamma^*_T|\mathcal{F}_T^-,\mu^*),\mathrm{TV}(\Gamma^*_{T-1}|\mathcal{F}_T^-,\mu^*)\right)
\tag{49}
$$

where μ^* is the invariant distribution of a Markov chain with transition matrix $\overleftarrow{\mathfrak{C}^2}$, namely

$$
\begin{cases}
\mu^*(\{0\}) &= \dfrac{\overleftarrow{\mathfrak{C}^2}_{10}}{\overleftarrow{\mathfrak{C}^2}_{01}+\overleftarrow{\mathfrak{C}^2}_{10}}\\
\mu^*(\{1\}) &= 1-\mu^*(\{0\}).
\end{cases}
\tag{50}
$$

Furthermore, an alternative expression of the total variation distance given in Lemma 1 implies that the total variation distance between two Bernoulli distributions of parameters p and q is $|p-q|$. Combining this with (49), the triangle inequality and the rough estimate $\max(a,b)\leq a+b\ \forall a,b\geq 0$, we get

$$
\mathbb{P}\left(\Gamma^*_t=0|\,\mathcal{F}_T^-\right) \leq 3\mu^*(\{0\})+\mathbb{P}\left(\Gamma^*_T=0|\,\mathcal{F}_T^-\right)+\mathbb{P}\left(\Gamma^*_{T-1}=0\middle|\,\mathcal{F}_T^-\right) \leq \tilde{C}/N
$$

where $\tilde{C}=\tilde{C}(\bar{M}_\ell,\bar{M}_h,\bar{G}_\ell,\bar{G}_h,\varepsilon_{\mathrm{S}})$. The last inequality is straightforwardly derived by plugging respectively (50), (47) and (48) into the three terms of the preceding sum. This combined with (46) finishes the proof. □

Lemma 6. *For s defined in the statement of Lemma 4; A_t, B_t and R_t defined in the proof of Lemma 5 and all $t\geq s+1$, we have*

$$
\begin{aligned}
\mathbb{P}\left(A_{t-2}B_{t-2}=1|\,R_t,\mathcal{F}_T^-\right)\mathbb{1}_{A_tB_t=1} &\geq \left(1-\frac{2}{N}\left(\frac{\bar{G}_h\bar{M}_h}{\bar{G}_\ell\bar{M}_\ell}\right)^2\right)\mathbb{1}_{A_tB_t=1};\\
\mathbb{P}\left(A_{t-2}B_{t-2}=1|\,R_t,\mathcal{F}_T^-\right) &\geq \frac{\bar{M}_\ell\varepsilon_{\mathrm{S}}}{2\bar{M}_h}
\end{aligned}
$$

where the inequalities hold for N large enough, i.e., $N\geq N_0=N_0(\bar{M}_\ell,\bar{M}_h,\bar{G}_\ell,\bar{G}_h,\varepsilon_{\mathrm{S}})$.

Proof. We start by showing the following three inequalities for all $t \geq s$ and N sufficiently large:

$$\mathbb{P}\left(A_{t-1}=1 \middle| R_t, \mathcal{F}_T^-\right) \geq \varepsilon_{\mathrm{S}}; \tag{51}$$

$$\mathbb{P}\left(A_{t-1}B_{t-1}=1 \middle| R_t, \mathcal{F}_T^-\right) \mathbb{1}_{A_t=1} \geq (\bar{M}_\ell/2\bar{M}_h)\mathbb{1}_{A_t=1}; \tag{52}$$

$$\mathbb{P}\left(A_{t-1}B_{t-1}=1 \middle| R_t, \mathcal{F}_T^-\right) \mathbb{1}_{A_tB_t=1} \geq \left[1-\frac{1}{N}\left(\frac{\bar{G}_h\bar{M}_h}{\bar{G}_\ell\bar{M}_\ell}\right)^2\right] \mathbb{1}_{A_tB_t=1}. \tag{53}$$

For (51), we have

$$\mathbb{P}\left(A_{t-1}=1 \middle| R_t, \mathcal{F}_T^-\right) \mathbb{1}_{A_t\neq 1} = \mathbb{P}\left(\mathcal{I}_{t-1}^1 \neq \mathcal{I}_{t-1}^2 \middle| R_t, \mathcal{F}_T^-\right) \mathbb{1}_{\mathcal{I}_t^1=\mathcal{I}_t^2} \geq \varepsilon_{\mathrm{S}}\mathbb{1}_{A_t\neq 1} \tag{54}$$

by Lemma 2. Next,

$$\begin{aligned}
&\mathbb{P}\left(A_{t-1}=1 \middle| R_t, \mathcal{F}_T^-\right) \mathbb{1}_{A_t=1} \\
&\quad= \mathbb{P}\left(\mathcal{I}_{t-1}^1 \neq \mathcal{I}_{t-1}^2 \middle| R_t, \mathcal{F}_T^-\right) \mathbb{1}_{\mathcal{I}_t^1\neq\mathcal{I}_t^2} \\
&\quad= \left[1-\sum_i \mathbb{P}\left(\mathcal{I}_{t-1}^1=\mathcal{I}_{t-1}^2=i \middle| R_t, \mathcal{F}_T^-\right)\right] \mathbb{1}_{\mathcal{I}_t^1\neq\mathcal{I}_t^2} \\
&\quad= \left[1-\sum_{i=1}^N \prod_{k=1}^2 \frac{G_{t-1}(X_{t-1}^i)m_t(X_{t-1}^i, X_t^{\mathcal{I}_t^k})}{\sum_{j=1}^N G_{t-1}(X_{t-1}^j)m_t(X_{t-1}^j, X_t^{\mathcal{I}_t^k})}\right] \mathbb{1}_{\mathcal{I}_t^1\neq\mathcal{I}_t^2} \text{ by Lemma 2} \\
&\quad\geq \left[1-\frac{1}{N}\left(\frac{\bar{G}_h\bar{M}_h}{\bar{G}_\ell\bar{M}_\ell}\right)^2\right] \mathbb{1}_{A_t=1} \text{ by Assumptions 2 and 3.}
\end{aligned} \tag{55}$$

Combining (54) and (55) yields (51) for N large enough. To prove (52), we write

$$\begin{aligned}
&\mathbb{P}\left(A_{t-1}B_{t-1}=1 \middle| R_t, \mathcal{F}_T^-\right) \mathbb{1}_{A_t=1} \\
&\quad= \left[1-\mathbb{P}\left(A_{t-1}B_{t-1}=0 \middle| R_t, \mathcal{F}_T^-\right)\right] \mathbb{1}_{A_t=1} \\
&\quad\geq \left[1-\mathbb{P}\left(A_{t-1}=0 \middle| R_t, \mathcal{F}_T^-\right)-\mathbb{P}\left(B_{t-1}=0 \middle| R_t, \mathcal{F}_T^-\right)\right] \mathbb{1}_{A_t=1} \\
&\quad= \left[\mathbb{P}\left(A_{t-1}=1 \middle| R_t, \mathcal{F}_T^-\right)+\mathbb{P}\left(B_{t-1}=1 \middle| R_t, \mathcal{F}_T^-\right)-1\right] \mathbb{1}_{A_t=1}.
\end{aligned} \tag{56}$$

We analyse the second term in the above expression. We have

$$\begin{aligned}
&\mathbb{P}\left(B_{t-1}=1 \middle| R_t, \mathcal{F}_T^-\right) \mathbb{1}_{A_t=1} \\
&\quad= \mathbb{P}\left(\mathcal{I}_{t-1}^2=\mathcal{I}_{t-1}^{*2} \middle| R_t, \mathcal{F}_T^-\right) \mathbb{1}_{\mathcal{I}_t^1\neq\mathcal{I}_t^2} \\
&\quad= \left[1-\mathrm{TV}\left(B_t^{N,\mathrm{FFBS}}(\mathcal{I}_t^2,\cdot), B_t^{N,\mathrm{FFBS}}(\mathcal{I}_t^{*2},\cdot)\right)\right] \mathbb{1}_{A_t=1} \\
&\quad\quad \text{by construction of Algorithm 15} \\
&\quad\geq (\bar{M}_\ell/\bar{M}_h)\mathbb{1}_{A_t=1} \text{ by Lemma 3.}
\end{aligned} \tag{57}$$

Plugging (55) and (57) into (56) yields

$$\mathbb{P}\left(A_{t-1}B_{t-1}=1 \middle| R_t, \mathcal{F}_T^-\right) \mathbb{1}_{A_t=1} \geq \left(-\frac{1}{N}\left(\frac{\bar{G}_h\bar{M}_h}{\bar{G}_\ell\bar{M}_\ell}\right)^2+\frac{\bar{M}_\ell}{\bar{M}_h}\right) \mathbb{1}_{A_t=1}$$

and thus (52) follows if N is large enough. The inequality (53) is justified by combining (55), the simple decomposition $\mathbb{1}_{A_tB_t=1} = \mathbb{1}_{A_t=1}\mathbb{1}_{B_t=1}$ and the fact that Algorithm 15 guarantees $B_{t-1}=1$ if $A_t=B_t=1$.

We can now deduce the two inequalities in the statement of the Lemma. The first one is a straightforward double application of (53):

$$
\begin{aligned}
&\mathbb{P}\left(A_{t-2}B_{t-2}=1\middle|\, R_t,\mathcal{F}_T^-\right)\mathbb{1}_{A_tB_t=1}\\
&\quad\geq \mathbb{P}\left(A_{t-2}B_{t-2}=1,A_{t-1}B_{t-1}=1\middle|\, R_t,\mathcal{F}_T^-\right)\mathbb{1}_{A_tB_t=1}\\
&\quad= \mathbb{E}\left[\mathbb{P}\left(A_{t-2}B_{t-2}=1,A_{t-1}B_{t-1}=1\middle|\, R_{t-1},R_t,\mathcal{F}_T^-\right)\middle|\, R_t,\mathcal{F}_T^-\right]\mathbb{1}_{A_tB_t=1}\\
&\quad\text{by the law of total expectation}\\
&\quad= \mathbb{E}\left[\mathbb{P}\left(A_{t-2}B_{t-2}=1\middle|\, R_{t-1},\mathcal{F}_T^-\right)\mathbb{1}_{A_{t-1}B_{t-1}=1}\middle|\, R_t,\mathcal{F}_T^-\right]\mathbb{1}_{A_tB_t=1}\\
&\quad\text{since } (R_{T-\ell})_{\ell=0}^T \text{ is Markov given } \mathcal{F}_T^-\\
&\quad\geq \mathbb{E}\left[\left(1-\frac{1}{N}\left(\frac{\bar{G}_h\bar{M}_h}{\bar{G}_\ell\bar{M}_\ell}\right)^2\right)\mathbb{1}_{A_{t-1}B_{t-1}=1}\middle|\, R_t,\mathcal{F}_T^-\right]\mathbb{1}_{A_tB_t=1}\\
&\quad\geq \left[1-\frac{1}{N}\left(\frac{\bar{G}_h\bar{M}_h}{\bar{G}_\ell\bar{M}_\ell}\right)^2\right]^2\mathbb{1}_{A_tB_t=1}\geq\left(1-\frac{2}{N}\left(\frac{\bar{G}_h\bar{M}_h}{\bar{G}_\ell\bar{M}_\ell}\right)^2\right)\mathbb{1}_{A_tB_t=1}.
\end{aligned}
$$

Finally, we have

$$
\begin{aligned}
&\mathbb{P}\left(A_{t-2}B_{t-2}=1\middle|\, R_t,\mathcal{F}_T^-\right)\\
&\quad\geq \mathbb{P}\left(A_{t-2}B_{t-2}=1,A_{t-1}=1\middle|\, R_t,\mathcal{F}_T^-\right)\\
&\quad= \mathbb{E}\left[\mathbb{P}\left(A_{t-2}B_{t-2}=1\middle|\, R_{t-1},\mathcal{F}_T^-\right)\mathbb{1}_{A_{t-1}=1}\middle|\, R_t,\mathcal{F}_T^-\right]\\
&\quad\text{using law of total expectation and the Markov property as above}\\
&\quad\geq \frac{\bar{M}_\ell}{2\bar{M}_h}\mathbb{P}\left(A_{t-1}=1\middle|\, R_t,\mathcal{F}_T^-\right)\text{ by (52)}\\
&\quad\geq \frac{\bar{M}_\ell}{2\bar{M}_h}\varepsilon_{\mathrm{S}}\text{ by (51)}
\end{aligned}
$$

and the second inequality is proved. □

E.4. **Proof of Proposition 3 (hybrid rejection validity).**

Proof. Put $Z_n := (X_n, U_n C\mu_0(X_n))$. Then Z_n is uniformly distributed on

$$\mathcal{G}_0 := \{(x,y)\in\mathcal{X}\times\mathbb{R}_+, y\leq C\mu_0(x)\}.$$

The proof would be done if one could show that, *given* $K^*\leq K$, the variable Z_{K^*} is uniformly distributed on

$$\mathcal{G}_1 := \{(x,y)\in\mathcal{X}\times\mathbb{R}_+, y\leq \mu_1(x)\}.$$

Note that K^* is, by definition, the first time index where the sequence (Z_n) touches $\mathcal{G}_1$. Let B be any subset of $\mathcal{G}_1$. We have

$$\begin{aligned}
\mathbb{P}\left(Z_{K^*} \in B \middle| K^* \leq K\right) &\propto \mathbb{P}(Z_{K^*} \in B, K^* \leq K) \\
&= \sum_{k^*=1}^{\infty} \mathbb{P}\left(Z_{k^*} \in B, K^* = k^*, K \geq k^*\right) \\
&= \sum_{k^*=1}^{\infty} \mathbb{P}\left(Z_{k^*} \in B, Z_{1:k^*-1} \notin \mathcal{G}_1, K > k^* - 1\right) \\
&= \sum_{k^*=1}^{\infty} \mathbb{P}(Z_{k^*} \in B)\mathbb{P}\left(Z_{1:k^*-1} \notin \mathcal{G}_1, K > k^* - 1\right) \text{since } K \text{ stopping time} \\
&= \mathbb{P}(Z_1 \in B) \sum_{k^*=1}^{\infty} \mathbb{P}\left(Z_{1:k^*-1} \notin \mathcal{G}_1, K > k^* - 1\right) \\
&\propto \mathbb{P}(Z_1 \in B) \propto \mathbb{P}\left(Z_1 \in B \middle| Z_1 \in \mathcal{G}_1\right).
\end{aligned} \tag{58}$$

By considering the special case $B = \mathcal{G}_1$, we see that the constant of proportionality between the first and the last terms of (58) must be 1, from which the proof follows. □

E.5. **Proof of Theorem 3 (hybrid algorithm's intermediate complexity).** From (16), one may have the correct intuition that as $N \to \infty$, $\tau_t^{1,\mathrm{PaRIS}}$ tends in distribution to that of the variable $\tau_t^{\infty,\mathrm{PaRIS}}$ defined as

$$\tau_t^{\infty,\mathrm{PaRIS}} \mid X_t^{\infty,\mathrm{PaRIS}} \sim \mathrm{Geo}\left(\frac{r_t(X_t^{\infty,\mathrm{PaRIS}})}{\bar{M}_h}\right) \tag{59}$$

where $X_t^{\infty,\mathrm{PaRIS}} \sim \mathbb{Q}_{t-1}M_t(\mathrm{d}x_t)$ is distributed according to the predictive distribution of X_t given $Y_{0:t-1}$ and r_t is the density of $X_t^{\infty,\mathrm{PaRIS}}$ with respect to the Lebesgue measure (cf. Definition 1). The following proposition formalises the connection between $\tau_t^{1,\mathrm{PaRIS}}$ and $\tau_t^{\infty,\mathrm{PaRIS}}$.

Proposition 6. *We have* $\tau_t^{1,\mathrm{PaRIS}} \Rightarrow \tau_t^{\infty,\mathrm{PaRIS}}$ *as* $N \to \infty$.

Proof. From (16) and Definition 1 one has

$$\tau_t^{1,\mathrm{PaRIS}} \mid X_t^1, \mathcal{F}_{t-1} \sim \mathrm{Geo}\left(\frac{r_t^N(X_t^1)}{\bar{M}_h}\right). \tag{60}$$

In light of (59), it suffices to establish that

$$\frac{r_t^N(X_t^1)}{\bar{M}_h} \Rightarrow \frac{r_t(X_t^{\infty,\mathrm{PaRIS}})}{\bar{M}_h}. \tag{61}$$

Indeed, this would mean that for any continuous bounded function ψ, we have

$$\begin{aligned}
\mathbb{E}[\psi(\tau_t^{1,\mathrm{PaRIS}})] = \mathbb{E}\left[(\mathrm{Geo}^\star\,\psi)\left(\frac{r_t^N(X_t^1)}{\bar{M}_h}\right)\right] &\to \mathbb{E}\left[(\mathrm{Geo}^\star\,\psi)\left(\frac{r_t(X_t^{\infty,\mathrm{PaRIS}})}{\bar{M}_h}\right)\right] \\
&= \mathbb{E}[\psi(\tau_t^{\infty,\mathrm{PaRIS}})]
\end{aligned}$$

where $\mathrm{Geo}^\star$ is the geometric Markov kernel that sends each λ to the geometric distribution of parameter λ, i.e. $\mathrm{Geo}^\star(\lambda, \mathrm{d}x) = \mathrm{Geo}(\lambda)$. To this end, write

$$\begin{aligned} r_t^N(X_t^1) - r_t(X_t^1) &= \frac{\sum_n G_{t-1}(X_{t-1}^n) m_t(X_{t-1}^n, X_t^1)}{\sum_n G_{t-1}(X_{t-1}^n)} - r_t(X_t^1) \\ &= \frac{\sum_n N^{-1} G_{t-1}(X_{t-1}^n) \left[m_t(X_{t-1}^n, X_t^1) - r_t(X_t^1)\right]}{N^{-1} \sum_n G_{t-1}(X_{t-1}^n)}. \end{aligned} \tag{62}$$

We study the mean squared error of the numerator:

$$\begin{aligned} &\mathbb{E}\left\{ \frac{1}{N} \sum_n G_{t-1}(X_{t-1}^n) \left[m_t(X_{t-1}^n, X_t^1) - r_t(X_t^1)\right] \right\}^2 \\ &\quad = \frac{1}{N} \mathbb{E}\left\{ G_{t-1}(X_{t-1}^1)^2 \left[m_t(X_{t-1}^1, X_t^1) - r_t(X_t^1)\right]^2 \right\} \\ &\qquad + \frac{N(N-1)}{N^2} \mathbb{E}\Big\{ G_{t-1}(X_{t-1}^1) G_{t-1}(X_{t-1}^2) \left[m_t(X_{t-1}^1, X_t^1) - r_t(X_t^1)\right] \\ &\qquad\quad \times \left[m_t(X_{t-1}^2, X_t^1) - r_t(X_t^1)\right] \Big\} \end{aligned}$$

where we have again used the exchangeability induced by step $(\star)$ of Algorithm 5. The first term obviously tends to 0 as $N \to \infty$ by Assumptions 4 and 1. The second term also vanishes asymptotically thanks to Lemma 7 below and Assumption 6. Assumption 1 also implies that the denominator of (62) converges in probability to some constant, via the consistency of particle approximations, see e.g. Del Moral (2004) or Chopin and Papaspiliopoulos (2020). Thus, $r_t^N(X_t^1) - r_t(X_t^1) \Rightarrow 0$ by Slutsky's theorem. Moreover, $r_t(X_t^1) \Rightarrow r_t(X_t^{\infty,\mathrm{PaRIS}})$ by the continuity of r_t and the consistency of particle approximations. Using again Slutsky's theorem yields (61). □

The following lemma is needed to complete the proof of Proposition 6 and is related to the propagation of chaos property, see Del Moral (2004, Chapter 8).

Lemma 7. *We have* $(X_{t-1}^1, X_{t-1}^2, X_t^1) \Rightarrow \mathbb{Q}_{t-2} M_{t-1} \otimes \mathbb{Q}_{t-2} M_{t-1} \otimes Q_{t-1} M_t$.

Proof. For vectors u, v, and w, we have, by the symmetry of the distribution of particles:

$$\begin{aligned} &\mathbb{E}\left[\exp\left(iuX_{t-1}^1 + ivX_{t-1}^2 + iwX_t^1\right)\right] \\ &\quad = \mathbb{E}\left[\left(\frac{1}{N}\sum e^{iuX_{t-1}^n}\right)\left(\frac{1}{N}\sum e^{ivX_{t-1}^n}\right)\left(\frac{1}{N}\sum e^{iwX_t^n}\right)\right] \\ &\qquad - \frac{N}{N^2}\mathbb{E}\left[e^{iuX_{t-1}^1} e^{ivX_{t-1}^1}\left(\frac{1}{N}\sum e^{iwX_t^n}\right)\right] \\ &\qquad + \frac{N}{N^2}\mathbb{E}\left[e^{iuX_{t-1}^1} e^{ivX_{t-1}^2}\left(\frac{1}{N}\sum e^{iwX_t^n}\right)\right]. \end{aligned}$$

Note that

$$\frac{1}{N}\sum e^{iuX_{t-1}^n} \xrightarrow{\text{a.s.}} \mathbb{Q}_{t-2} M_{t-1}\left(\exp(iu\bullet)\right)$$

and

$$\frac{1}{N}\sum e^{iwX_t^n} \xrightarrow{\text{a.s.}} \mathbb{Q}_{t-1} M_t\left(\exp(iw\bullet)\right).$$

The dominated convergence theorem, applicable since $\left|e^{iu}\right| \leq 1$ for $u \in \mathbb{R}$, finishes the proof. □

Proof of Theorem 3. First of all,

$$\mathbb{E}[\tau_t^{\infty,\mathrm{PaRIS}}] = \mathbb{E}\left[\frac{\bar{M}_h}{r_t(X_t^{\infty,\mathrm{PaRIS}})}\right] = \int_{\mathcal{X}_t} \frac{\bar{M}_h}{r_t(x_t)} r_t(x_t)\mathrm{d}x_t = \infty \tag{63}$$

by Assumption 5. Next, for any $x \in \mathbb{R} \setminus \mathbb{Z}$ and $N > x$,

$$\mathbb{P}\left(\min(\tau_t^{1,\mathrm{PaRIS}}, N) \leq x\right) = \mathbb{P}\left(\tau_t^{1,\mathrm{PaRIS}} \leq x\right) \to \mathbb{P}\left(\tau_t^{\infty,\mathrm{PaRIS}} \leq x\right)$$

by Proposition 6. Thus, by Portmanteau theorem,

$$\min(\tau_t^{1,\mathrm{PaRIS}}, N) \Rightarrow \tau_t^{\infty,\mathrm{PaRIS}}. \tag{64}$$

Altogether, we have

$$\begin{aligned}
\liminf_{N\to\infty} \mathbb{E}\left[\min(\tau_t^{1,\mathrm{PaRIS}}, N)\right] &= \liminf_{N\to\infty} \sum k\mathbb{P}\left(\min(\tau_t^{1,\mathrm{PaRIS}}, N) = k\right) \\
&\geq \sum \liminf_{N\to\infty} k\mathbb{P}\left(\min(\tau_t^{1,\mathrm{PaRIS}}, N) = k\right) \text{ by Fatou's lemma} \\
&= \sum k\mathbb{P}\left(\tau_t^{\infty,\mathrm{PaRIS}} = k\right) \text{ by (64)} \\
&= \infty \text{ by (63)}
\end{aligned}$$

and

$$\lim_{N\to\infty} \frac{1}{N}\mathbb{E}\left[\min(\tau_t^{1,\mathrm{PaRIS}}, N)\right] = \lim_{N\to\infty} \mathbb{E}\left[\min\left(\frac{\tau_t^{1,\mathrm{PaRIS}}}{N}, 1\right)\right] \to 0$$

since $\tau_t^{1,\mathrm{PaRIS}} \Rightarrow \tau_t^{\infty,\mathrm{PaRIS}}$ implies that the sequence of random variables

$$\min\left(\frac{\tau_t^{1,\mathrm{PaRIS}}}{N}, 1\right)$$

converges to 0 in distribution while being bounded between 0 and 1. □

E.6. **Proof of Theorem 4 (hybrid PaRIS near-linear complexity).** The following proposition shows that the real execution time for the hybrid algorithm is asymptotically at most of the same order as the "oracle" hybrid execution time.

Proposition 7. *We have*

$$\limsup_{N\to\infty} \frac{\mathbb{E}\left[\min(\tau_t^{1,\mathrm{PaRIS}}, N)\right]}{\mathbb{E}\left[\min(\tau_t^{\infty,\mathrm{PaRIS}}, N)\right]} < \infty.$$

Proof. Put

$$z^N(\lambda) := \frac{1-(1-\lambda)^N}{\lambda} = \sum_{n=0}^{N-1} (1-\lambda)^n. \tag{65}$$

One can quickly verify (using the memorylessness of the geometric distribution for example) that $z^N(\lambda) = \mathbb{E}\left[\min(G, N) | \, G \sim \mathrm{Geo}(\lambda)\right]$. It will be useful to keep in

mind the elementary estimate $z^N(\lambda) \leq \min(N, \lambda^{-1})$. We can now write

$$\begin{aligned}
\mathbb{E}\left[\min(\tau_t^{1,\mathrm{PaRIS}}, N)\right] &= \mathbb{E}\left[z^N\left(\frac{r_t^N(X_t^1)}{\bar{M}_h}\right)\right] \text{ (by (60))} = \mathbb{E}\left[\mathbb{E}\left[z^N\left(\frac{r_t^N(X_t^1)}{\bar{M}_h}\right)\middle| \mathcal{F}_{t-1}\right]\right] \\
&= \mathbb{E}\left[\int_{\mathcal{X}_t} z^N\left(\frac{r_t^N(x_t)}{\bar{M}_h}\right) r_t^N(x_t)\lambda_t(\mathrm{d}x_t)\right] \\
&\leq c_t\left(\int_{\mathcal{X}_t} z^N\left(\frac{r_t(x_t)}{\bar{M}_h}\right) r_t(x_t)\lambda_t(\mathrm{d}x_t) + b_t\right) \text{ by Lemma 8} \\
&= c_t\left(\mathbb{E}[\min(\tau_t^{\infty,\mathrm{PaRIS}}, N)] + b_t\right)
\end{aligned}$$

from which the proposition is immediate. □

Lemma 8. *In addition to notations of Algorithm 1, let the function z^N be defined as in (65) and the functions r_t and r_t^N be defined as in Definition 1. Let $\phi_t : \mathcal{X}_t \to \mathbb{R}_{>0}$ be a bounded non-negative deterministic function. Then, under Assumptions 1 and 4, there exist constants b_t and c_t depending only on the model such that*

$$\mathbb{E}\left[\int_{\mathcal{X}_t} z^N\left(\frac{r_t^N(x_t)}{\bar{M}_h}\right) r_t^N\phi_t\right] \leq c_t\left(\int_{\mathcal{X}_t} z^N\left(\frac{r_t(x_t)}{\bar{M}_h}\right) r_t\phi_t + b_t\left\|\phi_t\right\|_\infty\right)$$

where for brevity, we shortened the integration notation (e.g. dropping $\lambda_t(\mathrm{d}x_t)$, dropping x_t from $\phi(x_t)$, etc.) whenever there is no ambiguity.

Proof. We have

$$\mathbb{E}\left[\int_{\mathcal{X}_t} z^N\left(\frac{r_t^N(x_t)}{\bar{M}_h}\right) r_t^N\phi_t\right] \leq \int_{\mathcal{X}_t} z^N\left(\mathbb{E}\left[\frac{r_t^N(x_t)}{\bar{M}_h}\right]\right)\mathbb{E}\left[r_t^N(x_t)\right]\phi_t \tag{66}$$

using Fubini's theorem and the concavity of $\lambda \mapsto \lambda z^N(\lambda)$ on $[0,1]$. By a well-known result on the bias of a particle filter (which is in fact the propagation of chaos in the special case of $q = 1$ particle), we have:

$$\begin{aligned}
\left|\mathbb{E}\left[r_t^N(x_t)\right] - r_t(x_t)\right| &= \left|\mathbb{E}\left[\sum W_{t-1}^n m_t(X_{t-1}^n, x_t)\right] - r_t(x_t)\right| \\
&= \left|\mathbb{E}\left[m_t\left(X_{t-1}^{A_t^1}, x_t\right)\right] - \mathbb{Q}_{t-1}\left(m_t\left(\bullet, x_t\right)\right)\right| \\
&\leq \frac{b_t\bar{M}_h}{N}
\end{aligned}$$

for some constant b_t. We next show that such a bias does not change the asymptotic behavior of z^N. More precisely,

$$\begin{aligned} z^N\left(\mathbb{E}\left[\frac{r_t^N(x_t)}{\bar{M}_h}\right]\right) &\leq z^N\left(\frac{r_t(x_t)}{\bar{M}_h} - \frac{b_t}{N}\right) \\ &= \sum_{n=0}^{N-1}\left(\frac{1 - r_t(x_t)/\bar{M}_h + b_t/N}{1 - r_t(x_t)/\bar{M}_h}\right)^n \left(1 - \frac{r_t(x_t)}{\bar{M}_h}\right)^n \\ &\leq \sum_{n=0}^{N-1}\left(1 + \frac{b_t}{N\left(1 - r_t(x_t)/\bar{M}_h\right)}\right)^N \left(1 - \frac{r_t(x_t)}{\bar{M}_h}\right)^n \\ &\leq \exp\left(\frac{b_t}{1 - r_t(x_t)/\bar{M}_h}\right) z^N\left(\frac{r_t(x_t)}{\bar{M}_h}\right) \\ &\leq e^{2b_t} z^N\left(\frac{r_t(x_t)}{\bar{M}_h}\right) \end{aligned} \tag{67}$$

if x_t is such that $r_t(x_t)/\bar{M}_h \leq 1/2$. In contrast, if $r_t(x_t)/\bar{M}_h \geq 1/2$, then provided that $N \geq 6b_t$, we have

$$z^N\left(\mathbb{E}\left[\frac{r_t^N(x_t)}{\bar{M}_h}\right]\right) \leq z^N\left(\frac{r_t(x_t)}{\bar{M}_h} - \frac{b_t}{N}\right) \leq z^N\left(\frac{1}{3}\right) \leq 3z^N\left(\frac{r_t(x_t)}{\bar{M}_h}\right). \tag{68}$$

Putting together (67) and (68), we have, for $N \geq 6b_t$,

$$z^N\left(\mathbb{E}\left[\frac{r_t^N(x_t)}{\bar{M}_h}\right]\right) \leq \left(e^{2b_t} + 3\right) z^N\left(\frac{r_t(x_t)}{\bar{M}_h}\right)$$

and so, by (66),

$$\begin{aligned} \mathbb{E}\left[\int_{\mathcal{X}_t} z^N\left(\frac{r_t^N(x_t)}{\bar{M}_h}\right) r_t^N \phi_t\right] &\leq \left(e^{2b_t} + 3\right) \int_{\mathcal{X}_t} z^N\left(\frac{r_t(x_t)}{\bar{M}_h}\right) \mathbb{E}\left[r_t^N(x_t)\right] \phi_t \\ &= \left(e^{2b_t} + 3\right) \mathbb{E}\left[z^N\left(\frac{r_t(X_t^1)}{\bar{M}_h}\right) \phi_t(X_t^1)\right]. \end{aligned}$$

Again, using the result on the bias of a particle filter,

$$\left|\mathbb{E}\left[z^N\left(\frac{r_t(X_t^1)}{\bar{M}_h}\right)\phi_t(X_t^1)\right] - \int_{\mathcal{X}_t} z^N\left(\frac{r_t(x_t)}{\bar{M}_h}\right) r_t \phi_t\right| \leq \frac{b_t \left\|z^N\right\|_\infty \|\phi_t\|_\infty}{N} = b_t \|\phi_t\|_\infty$$

which, together with the previous inequality, implies the desired result. □

Proposition 8. *In linear Gaussian state space models, we have*

$$\mathbb{E}\left[\min(\tau_t^{\infty,\mathrm{PaRIS}}, N)\right] = \mathcal{O}\left((\log N)^{d_t/2}\right).$$

Proof. Let μ_t and Σ_t be such that $X_t^{\infty,\mathrm{PaRIS}} \sim \mathcal{N}(\mu_t, \Sigma_t)$. Then

$$\log(r_t(X_t^{\infty,\mathrm{PaRIS}})/\bar{M}_h) = b_t' - W_t$$

where b_t' is some constant and

$$W_t := \frac{(X_t^{\infty,\mathrm{PaRIS}} - \mu_t)^\top \Sigma_t^{-1} (X_t^{\infty,\mathrm{PaRIS}} - \mu_t)}{2} \sim \mathrm{Gamma}\left(\frac{d_t}{2}, 1\right).$$

We have

$$\begin{aligned}\mathbb{E}\left[\min(\tau_t^{\infty,\mathrm{PaRIS}}, N)\right] &= \mathbb{E}\left[z^N\left(\frac{r_t(X_t^{\infty,\mathrm{PaRIS}})}{\bar{M}_h}\right)\right] = \mathbb{E}\left[z^N(e^{b'_t - W_t})\right] \\ &= \int_0^\infty z^N(e^{b'_t - w})\frac{w^{d_t/2-1}e^{-w}}{\Gamma(d_t/2)}\mathrm{d}w \\ &\leq \int_0^{\log N} e^{w-b'_t}\frac{w^{d_t/2-1}e^{-w}}{\Gamma(d_t/2)}\mathrm{d}w + \int_{\log N}^\infty N\frac{w^{d_t/2-1}e^{-w}}{\Gamma(d_t/2)}\mathrm{d}w\end{aligned}$$

using the bound $z^N(\lambda) \leq \min(N, 1/\lambda)$. The first term is of order $\mathcal{O}(\log^{d_t/2} N)$ by elementary calculus, while the second term is of order $\mathcal{O}(\log^{d_t/2-1} N)$ using asymptotic properties of the incomplete Gamma function, see Olver et al. (2010, Section 8.11). □

Proof of Theorem 4. The theorem is a straightforward consequence of Proposition 7 and Proposition 8. □

E.7. **Proof of Theorems 6 and 7 (pure rejection FFBS complexity).** We start with a useful remark linking the projection kernels Π and the cost-to-go functions defined in Appendix A with the L-kernels formulated in (27). The proof is simple and therefore omitted.

Lemma 9. *We have $L_{t:T}(x_{0:t}, \mathbb{1}) = H_{t:T}(x_t)$ for all $x_{0:t}$. Moreover, for any function $\phi_t : \mathcal{X}_t \to \mathbb{R}$, we have*

$$L_{t:T}\Pi_t^{0:T}\phi_t = \Pi_t^{0:t}(\phi_t \times H_{t:T}).$$

Theorems 6 and 7 both rely on an induction argument wrapped up in the following proposition.

Proposition 9. *We use the notations of Algorithm 2. Let $\mathbb{Q}_t^N$ be defined as in (24), where the B_s^N kernels can be $B_s^{N,\mathrm{FFBS}}$ or any other kernels satisfying the hypotheses of Theorem 1. Suppose that Assumption 1 holds. Let $f_t^N : \mathcal{X}_t \to \mathbb{R}_{\geq 0}$ be a (possibly random) function such that $f_t^N(x_t)$ is $\mathcal{F}_{t-1}$-measurable. Then the following assertions are true:*

(a) *Suppose that $\mathbb{E}\left[\int_{\mathcal{X}_t}\left\{r_t^N \times f_t^N \times G_t \times H_{t:T}\right\}(x_t)\lambda_t(\mathrm{d}x_t)\right] = \infty$, where r_t^N and λ_t are defined in Definition 1. Then*

$$\mathbb{E}\left[\int \mathbb{Q}_T^N(\mathrm{d}x_t)f_t^N(x_t)\right] = \infty.$$

(b) *Suppose that $\int_{\mathcal{X}_t}\left\{r_t^N \times f_t^N \times G_t \times H_{t:T}\right\}(x_t)\lambda_t(\mathrm{d}x_t) \xrightarrow{\mathbb{P}} 0$. Then*

$$\int \mathbb{Q}_T^N(\mathrm{d}x_t)f_t^N(x_t) \xrightarrow{\mathbb{P}} 0.$$

Proof. **Part (a).** We shall prove by induction the statement

$$\mathbb{E}\left[\mathbb{Q}_s^N L_{s:T}\Pi_t^{0:T} f_t^N\right] = \infty, \forall\ t-1 \leq s \leq T.$$

For $s = t-1$, it follows from part (a)'s hypothesis and Lemma 9. Indeed,

$$\begin{aligned}
&\mathbb{Q}_{t-1}^N L_{t-1:T} \Pi_t^{0:T} f_t^N \\
&\quad = \mathbb{Q}_{t-1}^N L_{t-1:t} L_{t:T} \Pi_t^{0:T} f_t^N = \mathbb{Q}_{t-1}^N L_{t-1:t} \Pi_t^{0:t} (f_t^N \times H_{t:T}) \\
&\quad = \iint_{\mathcal{X}_{t-1} \times \mathcal{X}_t} \mathbb{Q}_{t-1}^N(\mathrm{d}x_{t-1}) m_t(x_{t-1}, x_t) \lambda_t(\mathrm{d}x_t) G_t(x_t) (f_t^N \times H_{t:T})(x_t) \\
&\quad = \int_{\mathcal{X}_t} \left\{ r_t^N \times f_t^N \times G_t \times H_{t:T} \right\}(x_t) \lambda_t(\mathrm{d}x_t).
\end{aligned}$$

For $s \geq t$, we have

$$\begin{aligned}
\mathbb{E}\left[\mathbb{Q}_s^N L_{s:T} \Pi_t^{0:T} f_t^N\right] &= \mathbb{E}\left[\frac{N^{-1} \sum \tilde{K}_s^N(n, L_{s:T} \Pi_t^{0:T} f_t^N)}{\ell_s^N}\right] \text{ by Corollary 3} \\
&\geq \frac{1}{\|G_s\|_\infty} \mathbb{E}\left[N^{-1} \sum \tilde{K}_s^N(n, L_{s:T} \Pi_t^{0:T} f_t^N)\right] \\
&\quad \text{by Assumption 1 and definition of } \ell_s^N \text{ (see Algorithm 1)} \\
&\geq \frac{1}{\|G_s\|_\infty} \mathbb{E}\left[\mathbb{Q}_{s-1}^N L_{s-1:s} L_{s:T} \Pi_t^{0:T} f_t^N\right] \\
&\quad \text{by Corollary 3 and law of total expectation} \\
&= \mathbb{E}\left[\mathbb{Q}_{s-1}^N L_{s-1:T} \Pi_t^{0:T} f_t^N\right] = \infty \text{ (induction hypothesis).}
\end{aligned}$$

Part (b). Similar to part (a), we shall prove by induction the statement

$$\mathbb{Q}_s^N L_{s:T} \Pi_t^{0:T} f_t^N \xrightarrow{\mathbb{P}} 0, \forall\ t-1 \leq s \leq T.$$

Again, by Corollary 3, this quantity is equal to

$$\frac{N^{-1} \sum \tilde{K}_s^N(n, L_{s:T} \Pi_t^{0:T} f_t^N)}{\ell_s^N},$$

and the expectation of the numerator given $\mathcal{F}_{s-1}$ is $\mathbb{Q}_{s-1}^N L_{s-1:T} \Pi_t^{0:T} f_t^N$, which tends to 0 in probability by induction hypothesis. Lemma 12 (see below at the end of the section), the classical result $\ell_s^N \xrightarrow{\mathbb{P}} \ell_s := \mathbb{Q}_{s-1} M_s(G_s)$ and Stutsky's theorem concludes the proof. □

Proof of Theorem 6. By (21), we have $\mathbb{E}[\tau_t^{1,\mathrm{FFBS}}] = \mathbb{E}[\int \mathbb{Q}_T^{N,\mathrm{FFBS}}(\mathrm{d}x_t) f_t^N(x_t)]$ where

$$f_t^N(x_t) = \frac{\bar{M}_h}{\sum W_{t-1}^n m_t(X_{t-1}^n, x_t)} = \frac{\bar{M}_h}{r_t^N}$$

with r_t^N given in Definition 1. Proposition 9(a) gives a sufficient condition for $\mathbb{E}[\tau_t^{1,\mathrm{FFBS}}] = \infty$ to hold, namely

$$\int_{\mathcal{X}_t} (r_t^N \times f_t^N \times G_t \times H_{t:T})(x_t) \lambda_t(\mathrm{d}x_t) = \infty,$$

which is equivalent to the hypothesis of the theorem. □

Proof of Theorem 7. We use notations from Definition 1 and Appendix A.1. We note $\mathcal{N}(x|\mu, \Sigma)$ the density of the specified normal distribution at point x. Using

Lemma 10, Proposition 9 and (21), we have

$$\begin{aligned}
\mathbb{E}[(\tau_t^{1,\mathrm{FFBS}})] = \infty &\Leftrightarrow \mathbb{E}\left[\frac{1}{r_t^N(X_t^{\mathcal{I}_t^1})^k}\right] = \infty \\
&\Leftrightarrow \mathbb{E}\left[\int \mathbb{Q}_T^N(\mathrm{d}x_t)\frac{1}{r_t^N(x_t)^k}\right] = \infty \\
&\Leftarrow \mathbb{E}\left[\int_{\mathcal{X}_t} \frac{r_t^N G_t H_{t:T}}{(r_t^N)^k}(x_t)\lambda_t(\mathrm{d}x_t)\right] = \infty \\
&\Leftarrow \int_{\mathcal{X}_t} \frac{r_t G_t H_{t:T}}{(r_t^N)^{k-1} r_t}(x_t)\lambda_t(\mathrm{d}x_t) = \infty \text{ almost surely} \\
&\Leftrightarrow \int_{\mathcal{X}_t} \frac{\mathcal{N}(x_t|\mu_t^{\mathrm{smth}}, \Sigma_t^{\mathrm{smth}})}{r_t^N(x_t)^{k-1}\mathcal{N}(x_t|\mu_t^{\mathrm{pred}}, \Sigma_t^{\mathrm{pred}})}\lambda_t(\mathrm{d}x_t) = \infty \text{ a.s.}
\end{aligned}$$

The theorem then follows from elementary arguments, by noting that r_t^N is a mixture of N Gaussian distributions with covariance matrix C_X. □

Lemma 10. *Let L be a $]0,1]$-valued random variable. Suppose X is another random variable such that $X|L \sim \mathrm{Geo}(L)$. Then for any real number $k > 0$,*

$$\mathbb{E}[X^k] = \infty \Leftrightarrow \mathbb{E}[L^{-k}] = \infty.$$

Proof. By the definition of X, we have

$$\mathbb{E}[X^k] = \mathbb{E}\left[\sum_{x=1}^{\infty} x^k (1-L)^{x-1} L\right].$$

A natural idea is then to approximate the sum by the integral $\int_0^\infty x^k(1-L)^{x-1}L\mathrm{d}x$, from which one easily extracts the L^{-k} factor. This is however technically laborious, since the function $x \mapsto x^k(1-L)^{x-1}L$ is not monotone on the whole real line. It is only so starting from a certain x_0 which itself depends on L. We would therefore rather write

$$\begin{aligned}
\mathbb{E}[X^k] &= \int_0^\infty \mathbb{P}(X^k \geq x)\mathrm{d}x = \int_0^\infty \mathbb{P}(X \geq x^{1/k})\mathrm{d}x \\
&= \int_0^\infty \mathbb{E}\left[(1-L)^{\lfloor x^{1/k} \rfloor}\right]\mathrm{d}x \\
&\quad \text{where the two integrands are equal Lebesgue–almost-everywhere} \\
&= \mathbb{E}\left[\int_0^\infty \exp\left(-\left|\log(1-L)\right| \lfloor x^{1/k} \rfloor\right)\mathrm{d}x\right]
\end{aligned}$$

with the natural interpretation of expressions when $L = 1$. Using $u \sim v$ as a shorthand for "u and v are either both finite or both infinite", we have

$$\begin{aligned}
\mathbb{E}[X^k] &\sim \mathbb{E}\left[\int_0^\infty \exp\left(-\left|\log(1-L)\right| x^{1/k}\right)\mathrm{d}x\right] \text{ by Lemma 11} \\
&= k\,\Gamma(k)\,\mathbb{E}\left[\frac{1}{\left|\log(1-L)\right|^k}\right] \sim \mathbb{E}\left[\frac{1}{L^k}\right] \text{ by Lemma 11 again.}
\end{aligned}$$

□

The following lemma is elementary. Its proof is therefore omitted.

Lemma 11. *Let L be a $]0,1]$-valued random variable and let f_1 and f_2 be two continuous functions from $]0,1]$ to $\mathbb{R}$. Suppose that $\limsup_{\ell\to 0^+} f_1(\ell)/f_2(\ell)$ and $\limsup_{\ell\to 0^+} f_2(\ell)/f_1(\ell)$ are both finite. Then $\mathbb{E}[f_1(L)]$ is finite if and only if $\mathbb{E}[f_2(L)]$ is so.*

Lemma 12. *Let $Z_1, Z_2, \ldots$ be non-negative random variables. Suppose that there exist σ-algebras $\mathcal{F}_1, \mathcal{F}_2, \ldots$ such that $\mathbb{E}[Z_n|\mathcal{F}_n] \xrightarrow{\mathbb{P}} 0$. Then $Z_n \xrightarrow{\mathbb{P}} 0$.*

Proof. Fix $\varepsilon > 0$. By Markov's inequality, $\mathbb{P}(Z_n \geq \varepsilon|\mathcal{F}_n) \leq \varepsilon^{-1}\mathbb{E}[Z_n|\mathcal{F}_n]$. Therefore, the $[0,1]-$bounded random variable $\mathbb{P}(Z_n \geq \varepsilon|\mathcal{F}_n)$ tends to 0 in probability, hence also in expectation. The law of total expectation then gives $\mathbb{P}(Z_n \geq \varepsilon) \to 0$, which, by varying ε, establishes the convergence of Z_n to 0 in probability. □

E.8. Proof of Theorem 8 and Corollary 2 (hybrid FFBS complexity).

Proof of Theorem 8. According to Janson (2011, Lemma 3), it is sufficient to show that

$$\frac{\sum_n \min(\tau_t^{n,\mathrm{FFBS}}, N)}{N\alpha_N} \xrightarrow{\mathbb{P}} 0$$

for any deterministic sequence α_N such that $\alpha_N/\mathbb{E}[\min(\tau_t^{\infty,\mathrm{FFBS}}, N)] \to \infty$. By Lemma 12, we can take expectation with respect to $\mathcal{F}_T$ to derive a sufficient condition, namely

$$\begin{aligned}
&\int_{\mathcal{X}_t} {\alpha_N}^{-1} z^N \left(\frac{r_t^N(x_t)}{\bar{M}_h}\right) \mathbb{Q}_T^N(\mathrm{d}x_t) \xrightarrow{\mathbb{P}} 0 \text{ with } z^N \text{ defined in (65)} \\
\Leftarrow &\int_{\mathcal{X}_t} {\alpha_N}^{-1} z^N \left(\frac{r_t^N(x_t)}{\bar{M}_h}\right) r_t^N \times G_t \times H_{t:T}\, \mathrm{d}\lambda_t \xrightarrow{\mathbb{P}} 0 \text{ by Proposition 9(b)} \\
\Leftarrow &\mathbb{E}\left[\int_{\mathcal{X}_t} {\alpha_N}^{-1} z^N \left(\frac{r_t^N(x_t)}{\bar{M}_h}\right) r_t^N \times G_t \times H_{t:T}\, \mathrm{d}\lambda_t\right] \to 0 \\
\Leftarrow &\int_{\mathcal{X}_t} {\alpha_N}^{-1} z^N \left(\frac{r_t(x_t)}{\bar{M}_h}\right) r_t \times G_t \times H_{t:T}\, \mathrm{d}\lambda_t \to 0 \text{ by Lemma 8} \\
\Leftrightarrow &\frac{\mathbb{E}[\min(\tau_t^{\infty,\mathrm{FFBS}}, N)]}{\alpha_N} \to 0.
\end{aligned}$$

The proof is now complete. □

Proof of Corollary 2. We have, using the cost-to-go, the z^N functions and the $\tau_t^{\infty,\mathrm{PaRIS}}$ distribution defined respectively in (20), (65) and (59):

$$\begin{aligned}
&\mathbb{E}[\min(\tau_t^{\infty,\mathrm{FFBS}}, N)] = \int_{\mathcal{X}_t} z^N \left(\frac{r_t(x_t)}{\bar{M}_h}\right) \mathbb{Q}_T(\mathrm{d}x_t) \\
&= [(\mathbb{Q}_{t-1}M_t)(G_t H_{t:T})]^{-1} \int_{\mathcal{X}_t} z^N \left(\frac{r_t(x_t)}{\bar{M}_h}\right) (G_t H_{t:T})(x_t)(\mathbb{Q}_{t-1}M_t)(\mathrm{d}x_t) \\
&\leq \|G_t H_{t:T}\|_\infty [(\mathbb{Q}_{t-1}M_t)(G_t H_{t:T})]^{-1} \int_{\mathcal{X}_t} z^N \left(\frac{r_t(x_t)}{\bar{M}_h}\right) (\mathbb{Q}_{t-1}M_t)(\mathrm{d}x_t) \\
&= \|G_t H_{t:T}\|_\infty [(\mathbb{Q}_{t-1}M_t)(G_t H_{t:T})]^{-1} \mathbb{E}[\min(\tau_t^{\infty,\mathrm{PaRIS}}, N)].
\end{aligned}$$

Proposition 8 then finishes the proof. □

E.9. **Proof of Proposition 4 (MCMC kernel properties).**

Proof. To show that a certain kernel B_t^N satisfies (13), we look at two conditionally i.i.d. simulations $J_t^{n,1}$ and $J_t^{n,2}$ from $B_t^N(n,\cdot)$ and lower bound the probability that they are different. For the kernel $B_t^{N,\mathrm{IMH}}$, the variables $J_t^{n,1}$ and $J_t^{n,2}$ both result from one step of MH applied to A_t^n. Let $J_t^{n,1*}$ and $J_t^{n,2*}$ be the corresponding MH proposals. A sufficient condition for $J_t^{n,1} \neq J_t^{n,2}$ is that $J_t^{n,1*} \neq J_t^{n,2*}$ and the two proposals are both accepted. The acceptance rate is at least $\bar{M}_\ell/\bar{M}_h$ by Assumption 2 and the probability that $J_t^{n,1*} \neq J_t^{n,2*}$ is

$$1 - \sum_{n=1}^{N} (W_{t-1}^n)^2 \geq 1 - \frac{1}{N}\left(\frac{\bar{G}_h}{\bar{G}_\ell}\right)^2$$

by Assumption 3. Thus (13) is satisfied for $\varepsilon_{\mathrm{S}} = \bar{M}_\ell/2\bar{M}_h$ for N large enough. Similarly, the probability that $J_t^{n,1} \neq J_t^{n,2}$ for the $B_t^{N,\mathrm{IMHP}}$ kernel with $\tilde{N} = 2$ can be lower-bounded via the probability that $\tilde{J}_t^{n,1} \neq \tilde{J}_t^{n,2}$ (where $\tilde{J}_t^{n,1}$ and $\tilde{J}_t^{n,2}$ are defined in (17)). Thus using the same arguments, (13) is satisfied here for $\varepsilon_{\mathrm{S}} = \bar{M}_\ell/4\bar{M}_h$. □

E.10. **Conditional probability of maximal couplings.** In general, there exist multiple maximal couplings of two random distributions (i.e. couplings that maximise the probability of equality of the two variables). However, they all satisfy a certain conditional probability property stated in the following lemma. Its proof, which we were unable to find in the literature, is obvious in the discrete case but requires lengthier arguments in the continuous one.

Proposition 10. *Let X_1 and X_2 be two random variables with densities f_1 and f_2 with respect to some dominating measure defined on a space $\mathcal{X}$. Then, the following inequality holds almost surely:*

$$\mathbb{P}(X_2 = X_1 | X_1) \leq 1 \wedge \frac{f_2(X_1)}{f_1(X_1)}. \tag{69}$$

Moreover, the equality occurs almost surely if and only if X_1 and X_2 form a maximal coupling.

Proof. Let h be any non-negative test function from $\mathcal{X}$ to $\mathbb{R}$. Putting

$$A_1 := \{x \in \mathcal{X} \mid f_1(x) \geq f_2(x)\}$$
$$A_2 := \{x \in \mathcal{X} \mid f_2(x) \geq f_1(x)\},$$

we have

$$\begin{aligned}
\mathbb{E}[\mathbb{P}(X_2 = X_1|X_1)h(X_1)] &= \mathbb{E}[\mathbb{1}_{X_2=X_1} h(X_1)] \\
&= \mathbb{E}[\mathbb{1}_{X_2=X_1}\mathbb{1}_{X_1\in A_1} h(X_1)] + \mathbb{E}[\mathbb{1}_{X_2=X_1}\mathbb{1}_{X_1\in A_2} h(X_1)] \\
&= \mathbb{E}[\mathbb{1}_{X_2=X_1}\mathbb{1}_{X_2\in A_1} h(X_2)] + \mathbb{E}[\mathbb{1}_{X_2=X_1}\mathbb{1}_{X_1\in A_2} h(X_1)] \\
&\leq \mathbb{E}[\mathbb{1}_{X_2\in A_1} h(X_2)] + \mathbb{E}[\mathbb{1}_{X_1\in A_2} h(X_1)] \\
&= \int h(x) f_2 \wedge f_1(x)\mathrm{d}x = \mathbb{E}\left[\left(1 \wedge \frac{f_2(X_1)}{f_1(X_1)}\right) h(X_1)\right].
\end{aligned}$$

The inequality (69) is now proved almost-surely. As a result, almost-sure equality occurs if and only if the expectation of the two sides of (69) are equal, which means, via Lemma 1, that the two variables are maximally coupled. □

The following lemma establishes the symmetry of Assumption 8. Again, its statement is obvious in the discrete case, though some work is needed to rigorously justify the continuous one.

Lemma 13. *Let X_1 and X_2 be two random variables of densities f_1 and f_2 w.r.t. some dominating measure defined on some space $\mathcal{X}$. Suppose that almost-surely*

$$\mathbb{P}(X_2 = X_1 | X_1) \geq \varepsilon \left(1 \wedge \frac{f_2(X_1)}{f_1(X_1)}\right)$$

for some $\varepsilon > 0$. Then almost-surely,

$$\mathbb{P}(X_1 = X_2 | X_2) \geq \varepsilon \left(1 \wedge \frac{f_1(X_2)}{f_2(X_2)}\right).$$

Proof. We introduce a non-negative test function $h_2 : \mathcal{X} \to \mathbb{R}$ and write

$$\begin{aligned}
\mathbb{E}[\mathbb{P}(X_1 = X_2 | X_2) h(X_2)] &= \mathbb{E}[\mathbb{1}_{X_1 = X_2} h(X_2)] = \mathbb{E}[\mathbb{1}_{X_2 = X_1} h(X_1)] \\
&= \mathbb{E}[\mathbb{P}(X_2 = X_1 | X_1) h(X_1)] \\
&\geq \mathbb{E}\left[\varepsilon \left(1 \wedge \frac{f_2(X_1)}{f_1(X_1)}\right) h(X_1)\right] \\
&= \int \varepsilon f_1 \wedge f_2(x) h(x) \mathrm{d}x \\
&= \mathbb{E}\left[\varepsilon \left(1 \wedge \frac{f_1(X_2)}{f_1(X_2)}\right) h(X_2)\right]
\end{aligned}$$

which implies the desired result. □

E.11. **Proof of Theorem 5 (intractable kernel properties).**

Proof. Let J_t^n be a random variable such that

$$J_t^n | X_{t-1}^{1:N}, X_t^n, \hat{B}_t^{N,\mathrm{ITR}}(n, \cdot) \sim B_t^{N,\mathrm{ITR}}(n, \cdot).$$

By construction of Algorithm 7, given $X_{t-1}^{1:N}$, the couple (J_t^n, X_t^n) has the same distribution as $(A_t^{n,L}, X_t^{n,L})$. Thus, $B_t^{N,\mathrm{ITR}}$ satisfies the hypotheses of Theorem 1.

To verify (13), we define the variables $J_t^{n,1:2}$ accordingly and write:

$$
\begin{aligned}
&\mathbb{P}\left(J_t^{n,1} \neq J_t^{n,2} \middle| X_t^n = x_t, X_{t-1}^{1:N} = x_{t-1}^{1:N}\right) \\
&= \frac{1}{2}\mathbb{P}\left(X_t^{n,1} = X_t^{n,2}, A_t^{n,1} \neq A_t^{n,2} \middle| X_t^n = x_t, X_{t-1}^{1:N} = x_{t-1}^{1:N}\right) \\
&= \mathbb{P}\left(X_t^{n,1} = X_t^{n,2}, A_t^{n,1} \neq A_t^{n,2}, L = 1 \middle| X_t^n = x_t, X_{t-1}^{1:N} = x_{t-1}^{1:N}\right) \\
&= \frac{1}{2}\mathbb{P}\left(X_t^{n,1} = X_t^{n,2}, A_t^{n,1} \neq A_t^{n,2} \middle| L = 1, X_t^n = x_t, X_{t-1}^{1:N} = x_{t-1}^{1:N}\right) \text{ (by symmetry)} \\
&= \frac{1}{2}\mathbb{P}\left(X_t^{n,1} = X_t^{n,2}, A_t^{n,1} \neq A_t^{n,2} \middle| X_t^{n,1} = x_t, X_{t-1}^{1:N} = x_{t-1}^{1:N}\right) \\
&= \frac{1}{2} \sum_{a_t^{n,1} \neq a_t^{n,2}} \mathbb{P}\left(X_t^{n,2} = x_t \middle| A_t^{n,1} = a_t^{n,1}, A_t^{n,2} = a_t^{n,2}, X_t^{n,1} = x_t, X_{t-1}^{1:N} = x_{t-1}^{1:N}\right) \times \\
&\qquad \times \mathbb{P}\left(A_t^{n,1} = a_t^{n,1}, A_t^{n,2} = a_t^{n,2} \middle| X_t^{n,1} = x_t, X_{t-1}^{1:N} = x_{t-1}^{1:N}\right) \\
&\geq \frac{1}{2}\,\varepsilon_D \frac{\bar{M}_\ell}{\bar{M}_h} \sum_{a_t^{n,1} \neq a_t^{n,2}} \mathbb{P}\left(A_t^{n,1} = a_t^{n,1}, A_t^{n,2} = a_t^{n,2} \middle| X_t^{n,1} = x_t, X_{t-1}^{1:N} = x_{t-1}^{1:N}\right) \\
&\qquad\qquad \text{(by Assumptions 8 and 2)} \\
&\geq \frac{1}{2}\varepsilon_D \left(\frac{\bar{M}_\ell}{\bar{M}_h}\right)^2 \varepsilon_A \text{ by Lemma 14.}
\end{aligned}
$$

The proof is complete. □

Lemma 14. *We use the notations of Algorithm 7. Under Assumptions 2 and 7, we have*

$$\mathbb{P}\left(A_t^{n,1} \neq A_t^{n,2} \middle| X_t^{n,1}, X_{t-1}^{1:N}\right) \geq \frac{\bar{M}_\ell}{\bar{M}_h}\varepsilon_A.$$

Proof. We write (and define new notations along the way):

$$
\begin{aligned}
\pi(a_t^{n,1}, a_t^{n,2}) &:= p(a_t^{n,1}, a_t^{n,2} | X_t^{n,1}, X_{t-1}^{1:N}) \\
&\propto p(a_t^{n,1}, a_t^{n,2} | X_{t-1}^{1:N}) m_t(X_{t-1}^{a_t^{n,1}}, X_t^{n,1}) \\
&=: p(a_t^{n,1}, a_t^{n,2} | X_{t-1}^{1:N}) \phi(a_t^{n,1}) \\
&=: \pi_0(a_t^{n,1}, a_t^{n,2}) \phi(a_t^{n,1}).
\end{aligned}
$$

Thus

$$
\begin{aligned}
\mathbb{P}\left(A_t^{n,1} \neq A_t^{n,2} \middle| X_t^{n,1}, X_{t-1}^{1:N}\right) &= \int \mathbb{1}\left\{a_t^{n,1} \neq a_t^{n,2}\right\} \pi(a_t^{n,1}, a_t^{n,2}) \\
&= \frac{\int \mathbb{1}\left\{a_t^{n,1} \neq a_t^{n,2}\right\} \pi_0(a_t^{n,1}, a_t^{n,2}) \phi(a_t^{n,1})}{\int \pi_0(a_t^{n,1}, a_t^{n,2}) \phi(a_t^{n,1})} \\
&\geq \int \mathbb{1}\left\{a_t^{n,1} \neq a_t^{n,2}\right\} \pi_0(a_t^{n,1}, a_t^{n,2}) \frac{\bar{M}_\ell}{\bar{M}_h}
\end{aligned}
$$

by the boundedness of the function ϕ between $\bar{M}_\ell$ and $\bar{M}_h$. From this, we get the desired result by virtue of Assumption 7. □

E.12. **Validity of Algorithm 14 (modified Lindvall-Rogers coupler).** Recall that generating a random variable is equivalent to uniformly simulating under the graph of its density (see e.g. Robert and Casella, 2004, The Fundamental Theorem of Simulation, chapter 2.3.1). Algorithm 14's correctness is thus a direct corollary of the following intuitive lemma.

Lemma 15. *Let S_A and S_B be two subsets of $\mathbb{R}^d$ with finite Lebesgue measures. Let A and B be two not necessarily independent random variables distributed according to* Uniform(S_A) *and* Uniform(S_B) *respectively. Denote by S_0 the intersection of S_A and S_B; and by C a certain* Uniform(S_A)*-distributed random variable that is independent from (A, B). Define $A^\star$ and $B^\star$ as*

$$A^\star = \begin{cases} C & \text{if } (A, C) \in S_0 \times S_0 \\ A & \text{otherwise} \end{cases}$$

and

$$B^\star = \begin{cases} C & \text{if } (B, C) \in S_0 \times S_0 \\ B & \text{otherwise} \end{cases}$$

Then $A^\star \sim$ Uniform(S_A) and $B^\star \sim$ Uniform(S_B).

Proof. Given $(A, C) \in S_0 \times S_0$, the two variables A and C have the same distribution (which is Uniform(S_0)). Thus, the definition of $A^\star$ implies that A and $A^\star$ have the same (unconditional) distribution. The same argument applies to B and $B^\star$ notwithstanding the asymmetry in the definition of C. □

E.13. **Hoeffding inequalities.** This appendix proves a Hoeffding inequality for ratios, which helps us to bound (28). It is essentially a reformulation of Douc et al. (2011, Lemma 4) in a slightly more general manner.

Definition 2. *A real-valued random variable X is called (C, S)-sub-Gaussian if*

$$\mathbb{P}\left(\frac{|X|}{S} > t\right) \leq 2Ce^{-t^2/2}, \forall\ t \geq 0.$$

This definition is close to other sub-Gaussian definitions in the literature, see e.g. Vershynin (2018, Chapter 2.5). It basically means that the tails of X decreases at least as fast as the tails of the $\mathcal{N}(0, S^2)$ distribution, which is itself $(1, S)$-sub-Gaussian. The following result is classic.

Theorem 9 (Hoeffding's inequality)**.** *Let $X_1, \ldots, X_N$ be N i.i.d. random variables with mean μ and almost surely contained between a and b. Then $N^{1/2}(\sum X_i/N - \mu)$ is $(1, (b-a)/2)$-sub-Gaussian.*

The following lemma is elementary from Definition 2. The proof is omitted.

Lemma 16. *Let X and Y be two (not necessarily independent) random variables. If X is (C_1, S_1)-sub-Gaussian and Y is (C_2, S_2)-sub-Gaussian, then $X+Y$ is $(C_1 + C_2, S_1 + S_2)$-sub-Gaussian.*

We are ready to state the main result of this section.

Proposition 11 (Hoeffding's inequality for ratios)**.** *Let a_N, b_N, a^*, b^* be random variables such that $\sqrt{N}(a_N - a^*)$ is (C_a, S_a)-sub-Gaussian and $\sqrt{N}(b_N - b^*)$ is*

(C_b, S_b)-sub-Gaussian. Then $\sqrt{N}\,(a_N/b_N - a^*/b^*)$ is sub-Gaussian with parameters (C^*, S^*) where

$$\begin{cases} C^* & = C_a + C_b \\ S^* & = \left\|\frac{1}{b^*}\right\|_\infty (S_a + S_b \left\|\frac{a_N}{b_N}\right\|_\infty). \end{cases}$$

The terms with inf-norm can be infinite if the corresponding random variables are unbounded.

Proof. We have

$$\begin{aligned} \left|\sqrt{N}\left(\frac{a_N}{b_N} - \frac{a^*}{b^*}\right)\right| &\leq \left|\sqrt{N}\left(\frac{a_N}{b_N} - \frac{a_N}{b^*}\right)\right| + \left|\sqrt{N}\left(\frac{a_N}{b^*} - \frac{a^*}{b^*}\right)\right| \\ &= \left|\frac{a_N}{b_N}\right| \left|\frac{1}{b^*}\right| \left|\sqrt{N}(b_N - b^*)\right| + \left|\frac{1}{b^*}\right| \left|\sqrt{N}(a_N - a^*)\right| \end{aligned}$$

by which the proposition follows from Lemma 16. □

CREST-ENSAE, Institut Polytechnique de Paris, France

Robust importance sampling via Median of Means

Ongoing work in collaboration with Nicolas Chopin, Hugues Gallier and Mathieu Gerber.

ROBUST IMPORTANCE SAMPLING VIA MEDIAN OF MEANS

Abstract. Although importance sampling estimates are asymptotically normal under standard conditions, they may have a heavy-tailed distribution in practical scenarios. We apply the idea of median of means from the robust statistics literature to produce more robust estimates. In addition, we propose a new way to build confidence intervals that prove more reliable than those derived from asymptotic normality.

1. Introduction

Importance sampling is a Monte Carlo method that approximates expectations with respect to a target distribution using simulations from another distribution (the so-called proposal distribution). The main requirement is that one is able to calculate the ratio between the densities of the two distributions, either exactly or up to some unknown normalising constant. We focus on the latter (more general) case from now on. Importance sampling is a classical method which is covered in all textbooks on Monte Carlo, e.g. Robert and Casella (2004); Gobet (2016).

Importance sampling estimates are consistent and asymptotically normal as $N \to +\infty$ (N being the Monte Carlo sample size) under second moment assumptions. One may construct asymptotic confidence intervals for such estimates by estimating their asymptotic variance. Unfortunately, the distribution of importance sampling estimates is often far from Gaussian for a finite N, and may have heavy tails. When this happens, an estimate of its variance (or equivalently of its second moment) will be even more unreliable. Another common recipe is to measure the performance of importance sampling via the effective sample size (ESS, Kong et al., 1994), but this quantity is also based on the second moment of the weights.

The solutions proposed so far in the literature include truncated importance sampling (Ionides, 2008) and Pareto-smoothed importance sampling (Vehtari et al., 2015). However, these methods either converge under conditions that are difficult to check or do not even converge. In fact, there is an inherent danger in modifying the weights without knowing more about the underlying distributions. Furthermore, building a confidence interval for such estimates would still require second-moment estimation.

In the robust statistics literature, the median of means (MoM) estimator has gained considerable traction in recent years; see the excellent survey of Lugosi and Mendelson (2019). The objective of this paper is to apply this idea to importance sampling, and to derive more reliable confidence intervals for the resulting estimates. Our construction of these intervals is reasonably straightforward, but we are not aware of similar efforts elsewhere. Recently, Orenstein (2019) suggested a Bayesian version of median of means for use in importance sampling. However, auto-normalised importance sampling (the version of importance sampling where the ratio of densities is known up to a constant) is not studied, and the proposed confidence interval simply uses the bootstrap method. Derumigny et al. (2019) derived confidence intervals for ratios of expectations, of which auto-normalised

importance sampling is a special case. However, their construction still aims at accompanying the traditional importance sampling estimator and requires the estimation of second moments.

After introducing importance sampling and the median of means method in Section 2, we combine the two to produce better importance sampling estimators. Their construction and properties are studied in Section 3. In Section 4, we consider a new method to build confidence intervals. We show that it guarantees asymptotically the required cover rate. We also prove that it works better than confidence intervals built via the central limit theorem, especially when high confidence levels are required. Simple numerical examples accompany the discussion. More extensive experiments are expected in a future version of the paper.

2. Framework and notations

2.1. **Importance sampling.** Let $\mathcal{X}$ be a measurable space and let $\mathbb{Q}$, $\mathbb{M}$ be respectively the target and proposal probability distributions on $\mathcal{X}$. (We assume it is easy to sample independent random variables from $\mathbb{M}$.) Let $\bar{\omega} := \mathrm{d}\mathbb{Q}/\mathrm{d}\mathbb{M}$ be the Radon-Nikodym derivative of $\mathbb{Q}$ with respect to $\mathbb{M}$ and suppose that we can calculate the function ω defined by $\omega(x) := Z\bar{\omega}(x)$ where Z is some unknown normalising constant. Let $X^1, \dots, X^N$ be i.i.d. random variables distributed according to $\mathbb{M}$. Then, given a function $\varphi : \mathcal{X} \to \mathbb{R}$, the auto-normalised importance sampling estimator of $\mu := \mathbb{Q}(\varphi) := \int \varphi(x)\mathbb{Q}(\mathrm{d}x)$ is

$$\mu_{\mathrm{AIS}}^N := \frac{\sum_{n=1}^N \omega(X^n)\varphi(X^n)}{\sum_{n=1}^N \omega(X^n)} \tag{1}$$

and an unbiased estimator of the normalising constant is

$$Z^N := \frac{1}{N}\sum_{n=1}^N \omega(X^n). \tag{2}$$

The estimate μ_{AIS}^N can also be rewritten in the form $\mu_{\mathrm{AIS}}^N = \sum_n W^n \varphi(X^n)$, where $W^n := \omega(X^n)/NZ^N$. Then $\sum_n W^n = 1$ and the quantity W^n can be interpreted as the weight of the draw X^n. More formally, one can define a weighted approximation $\mathbb{Q}^N$ of $\mathbb{Q}$ as

$$\mathbb{Q}^N(\mathrm{d}x) := \sum_{n=1}^N W^n \delta_{X^n}(\mathrm{d}x)$$

and μ_{AIS}^N is then equal to $\mathbb{Q}^N(\varphi)$.

It is elementary to show that

$$\sqrt{N}\left(\mu_{\mathrm{AIS}}^N - \mu\right) \Rightarrow \mathcal{N}(0, \mathbb{M}(\bar{\omega}^2\bar{\varphi}^2)) \tag{3}$$

where $\bar{\varphi}(x) := \varphi(x) - \mathbb{Q}(\varphi)$ and $\Rightarrow$ denotes convergence in distribution. A natural question is then how much the asymptotic variance $\mathbb{M}(\bar{\omega}^2\bar{\varphi}^2)$ is inflated with respect to $\mathbb{M}(\bar{\omega}\bar{\varphi}^2)$, the ideal asymptotic variance if we were able to simulate from $\mathbb{Q}$ directly. The following proposition (proved in Appendix A.1) answers this.

Proposition 1. *Using the notations defined in this subsection, suppose that $\mathcal{X} = \mathbb{R}^d$ and $\mathbb{Q}$ is dominated by the Lebesgue measure. Then*

$$\sup_{\varphi:\mathcal{X}\to\mathbb{R}} \frac{\mathbb{M}(\bar{\omega}^2\bar{\varphi}^2)}{\mathbb{M}(\bar{\omega}\bar{\varphi}^2)} = \mathrm{esssup}_{\mathbb{Q}}(\bar{\omega})$$

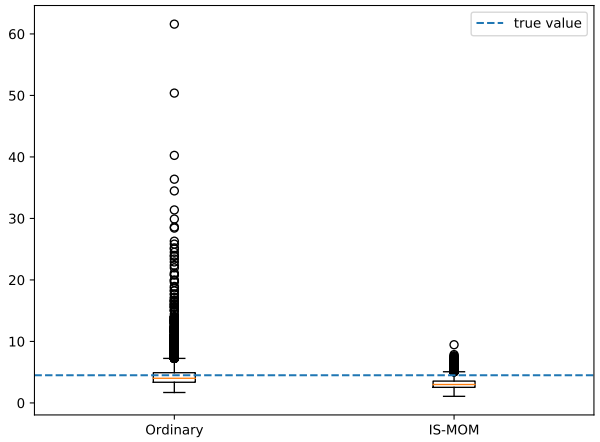

FIGURE 1. Box plots of 10^4 independent replicates of two estimates of $\mathbb{Q}(\varphi)$ when $N = 250$ in Example 1: the ordinary importance sampling estimate (1), and our MoM estimate (10), with $K = 6$ subgroups.

where $\operatorname{esssup}_{\mathbb{Q}}(\bar{\omega})$ *is the essential supremum of* $\bar{\omega}$ *with respect to the measure* $\mathbb{Q}$.

In words, the worst-case variance inflation factor coincides with the peak value of the normalised weight function. This justifies the use of the largest empirical weights among $W^1, \ldots, W^N$ to measure the quality of importance sampling (Chatterjee and Diaconis, 2018; Huggins and Roy, 2019). In particular, scenarios in which the function ω is unbounded are troublesome. For instance, if $\mathcal{X} = \mathbb{R}^d$, we could have $\lim_{x\to\infty} \omega(x) = \infty$; i.e. the target distribution $\mathbb{Q}$ has heavier tails than the proposal distribution $\mathbb{M}$. If in our sample $X^1, \ldots, X^N$ there is a certain draw X^n coming from a rare region of $\mathbb{M}$, it is likely to be assigned a high weight. It will then dominate the rest of the sample and thus "spoils" the resulting estimate. This is made clear by the following running example.

Example 1 (Running example)**.** *Let* $\mathbb{M}$ *and* $\mathbb{Q}$ *be exponential distributions with respective mean* $\lambda_0^{-1} := 1$ *and* $\lambda_1^{-1} := 1.5$*, and* φ *be the function* $\varphi(x) = x^2$*. Take* $N = 250$*.*

It is easy to check that the normalised weight function $\bar{\omega}(x) \propto e^{-(\lambda_1-\lambda_0)x}\mathbb{1}_{x\geq 0}$ is unbounded, while the asymptotic variance $\mathbb{M}(\bar{\omega}^2\bar{\varphi}^2)$ is finite. At a finite sample size ($N = 250$ here), the distribution of the estimate looks very skewed and non-Gaussian (left side of Figure 1).

A popular diagnostic for the quality of importance sampling is the effective sample size (Kong et al., 1994), defined as a function of the weights $W^1, \ldots, W^N$:

$$\mathrm{ESS}(W^1, \ldots, W^N) = \frac{1}{\sum_n W_n^2}.$$

This quantity lies in $[1, N]$. In Figure 2, we show the box plot of the effective sample size divided by N over the runs. Two lessons may be learnt here: (a) the estimation of the effective sample size itself is unstable, and (b) in most cases the

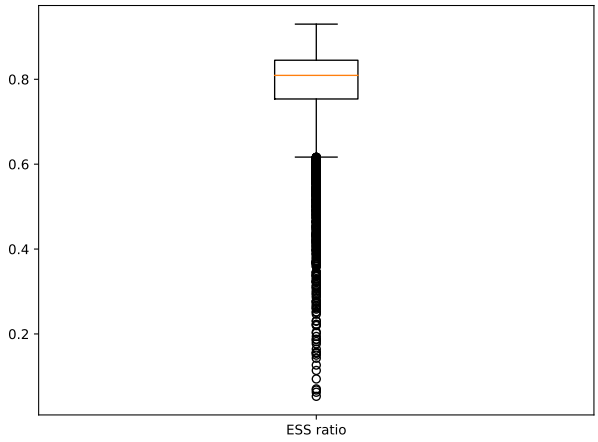

FIGURE 2. Box plot of the effective sample size divided by N in Example 1, again based on 10^4 independent replications.

diagnostic does not detect any problem, as a high ESS ratio (more than 60%) is reported.

Truly or practically unbounded weight functions happen in Bayesian cross validation as well as discrete problems, such as the simulation of random walks or permutations. The references (Vehtari et al., 2015; Orenstein, 2019; Chatterjee and Diaconis, 2018) discuss these examples, among others. We provide yet another scenario in Example 2.

Example 2 (Approximate Bayesian Computation)**.** *Consider a Bayesian inference problem where θ is the parameter variable with prior $p(\theta)$ and y is the data variable. Suppose that it is possible to simulate y given θ, but the probability density $p(y|\theta)$ is intractable. Approximate Bayesian Computation provides a workaround by sampling from $p(\theta, y \mid \|y - y_0\| \leq \varepsilon)$, where y_0 is the observed data and the threshold ε and the norm $\|\cdot\|$ are user-chosen. Importance sampling can be employed using the proposal distribution $r(\theta)p(y|\theta)$, where r can be any distribution on θ. The weight function*

$$\omega(\theta, y) = \frac{p(\theta)}{r(\theta)} \mathbb{1}_{\|y-y_0\| \leq \varepsilon}$$

is fully computable. When the data is informative and ε is small, a concentrated r is needed to prevent ω from being 0 most of the time. Unfortunately, this will also make ω unbounded.

2.2. **Median of means.** We now deviate from importance sampling and investigate a simpler problem: mean estimation. We remark that Z^N (Equation (2)) is an empirical mean estimator, whereas μ^N_{AIS} is more complicated, being the ratio of two estimators of that type. The normalised importance sampling estimator (computable in the case Z is known) is also simply an empirical mean. Studying mean estimation is therefore central to understanding importance sampling, and is done at depth in this paper. To distinguish between the two problems, we make the following convention: importance sampling aims at estimating $\mathbb{Q}(\varphi) = \int \mathbb{Q}(\mathrm{d}x)\varphi(x)$

and mean estimation aims at estimating $\mathbb{M}(\psi) = \int \mathbb{M}(\mathrm{d}x)\psi(x)$, both using i.i.d. $\mathbb{M}$-distributed random variables $X^1, \ldots, X^N$.

The most natural estimator for $\mathbb{M}(\psi)$ is the empirical mean estimator

$$\theta^N_{\mathrm{EMP}} := \frac{\psi(X^1) + \cdots + \psi(X^N)}{N}.$$

Unfortunately, if $\mathbb{M}$ is heavy-tailed and ψ is unbounded, the distribution of θ^N_{EMP} may have heavy tails, and may tend to generate outliers, like the importance sampling estimator in Figure 1. The median of means (MoM) method has been proposed as a solution (see Lugosi and Mendelson, 2019 and references therein). It consists in dividing the N data points into K groups of size M (so that $N = MK$). Next, it calculates the classical mean estimate for each group. This results in K estimates $\hat{\theta}_1, \ldots, \hat{\theta}_K$ defined by

$$\hat{\theta}_k := \frac{1}{M} \sum_{n=M(k-1)+1}^{Mk} \psi(X^n). \tag{4}$$

The final mean estimate is returned as the empirical median of $\{\hat{\theta}_1, \ldots, \hat{\theta}_K\}$:

$$\theta^{N,K}_{\mathrm{MoM}} := \theta^{N,K}_{\mathrm{MoM}}(\psi(X^1), \ldots, \psi(X^N)) := \mathrm{empmed}(\hat{\theta}_1, \ldots, \hat{\theta}_K). \tag{5}$$

To understand in which sense the MoM estimator is better than the empirical mean, it is necessary to change the benchmark for performance. Monte Carlo users usually focus on the mean squared error (MSE), which might not fully capture the behaviour of an estimator. Consider the shifted log-normal distribution with the density

$$f^*(x) = \frac{1}{s_0} f\left(\frac{x - \ell}{s_0}, s\right) \tag{6}$$

where $s = 1.5$, $s_0 = \left[(e^{s^2} - 1)e^{s^2}\right]^{-1/2}$, $\ell = -e^{s^2/2}s$ and f is the non-shifted density

$$f(x, s) = \frac{1}{sx\sqrt{2\pi}} \exp\left(-\frac{\log^2 x}{2s^2}\right).$$

The shifted distribution has mean 0 and variance 1. The first two box plots of Figure 3 correspond to 10^4 replicates of the empirical mean estimator for the $\mathcal{N}(0, 1)$ distribution and the aforementioned shifted log-normal distribution respectively. According to the MSE benchmark, the two cases should lead to similar performance. However, it is clear that the estimates in the normal distribution scenario have a much more desirable behaviour.

Therefore, we switch our attention from the MSE to the tail behaviour of the error. Given some threshold $\delta > 0$, we can benchmark a certain estimator θ^N based on the $(1 - \delta)$-quantile of its error $\left|\theta^N - \mathbb{M}(\psi)\right|$; and use this criterion to compare different estimators. From Figure 3, we see that it is desirable to have $\theta^N - \mathbb{M}(\psi)$ look as Gaussian-like as possible.

An estimator is said to be sub-Gaussian if the $(1 - \delta)$-quantile of its error $\left|\theta^N - \mathbb{M}(\psi)\right|$ is comparable to the $(1-\delta)$-quantile of the absolute value of a suitably scaled Gaussian distribution. The following definition formalises this.

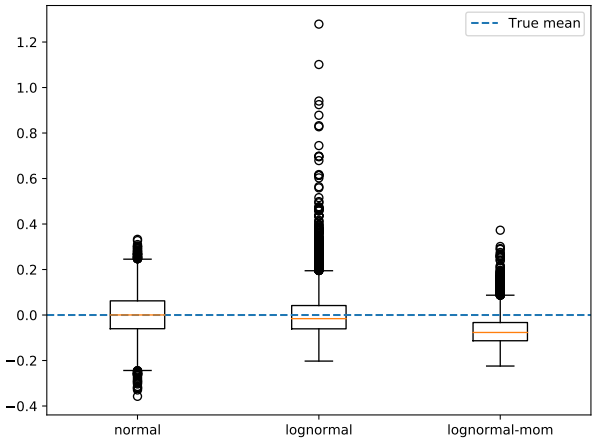

FIGURE 3. The first two box plots represent the empirical distribution (over 10^4 independent runs) of the empirical mean estimator with $N = 120$ when $\mathbb{M}$ is the $N(0,1)$ distribution (left), and the shifted log-normal distribution (6) described in the text (middle). The third box plot shows the distribution of the MoM estimator ($N = 120$ and $K = 6$) of the expectation of (6), using the same number of independent runs.

Definition 1. *An estimator* $\theta^N = \theta^N(\psi(X^1), \dots, \psi(X^N))$ *of* $\mathbb{M}(\psi)$ *is said to be* (C, D)*-sub-Gaussian at level* $\delta \in]0,1[$ *if*

$$\mathbb{P}\left(\left|\theta^N - \mathbb{M}(\psi)\right| \geq \frac{C\sigma\sqrt{\log(D/\delta)}}{\sqrt{N}}\right) \leq \delta$$

where $\sigma = \sqrt{\mathrm{Var}_{\mathbb{M}}(\psi)} := \sqrt{\mathbb{M}(\psi^2) - [\mathbb{M}(\psi)]^2}$.

If the distribution of θ^N is absolutely continuous with respect to the Lebesgue measure, this can be rewritten more intuitively as

$$q_{1-\delta}(\left|\theta^N - \mathbb{M}(\psi)\right|) \leq \frac{C\sigma\sqrt{\log(D/\delta)}}{\sqrt{N}} \tag{7}$$

where $q_{1-\delta}(X)$ denotes the $(1-\delta)$-quantile of a random variable X. Devroye et al. (2016) show that in general, there does not exist an estimator that is (C, D)-sub-Gaussian for simultaneously all δ. However, there exists absolute constants C and D such that, for each δ and for N large enough with respect to δ, one can construct a (C, D)-sub-Gaussian estimator at level δ. The following theorem has been proved in Lerasle and Oliveira (2011); Lugosi and Mendelson (2019).

Theorem 1. *For all* $\delta \in]0,1[$, *for* $N > 8\log(1/\delta)$ *and for* $K = \lceil 8\log(1/\delta)\rceil$, *the estimator* $\theta^{N,K}_{\mathrm{MoM}}$ *is* $(32,1)$*-sub-Gaussian at level* δ.

In contrast, the empirical mean estimator is not sub-Gaussian. The following result has been shown in Catoni (2012, Prop. 6.2).

Theorem 2. *For all $\delta < 1/e$ and $N \geq 1$, there exists a distribution $\mathbb{M}$ and a function ψ such that $\mathrm{Var}_{\mathbb{M}}(\psi) = \sigma^2$, $\mathbb{M}(|\psi|^3) < \infty$ and*

$$\mathbb{P}\left(\left|\theta^N_{\mathrm{EMP}} - \mathbb{M}(\psi)\right| \geq \sqrt{\frac{\sigma^2}{N\delta}}\left(1 - \frac{\delta e}{N}\right)^{\frac{N-1}{2}}\right) \geq \delta.$$

Similar to (7), one can express this in term of the quantile function as

$$\sup_{\substack{\mathbb{M},\psi \\ \mathrm{Var}_{\mathbb{M}}(\psi)=\sigma^2}} q_{1-\delta}\left(\left|\theta^N_{\mathrm{EMP}} - \mathbb{M}(\psi)\right|\right) \geq \sqrt{\frac{\sigma^2}{N\delta}}\left(1 - \frac{\delta e}{N}\right)^{\frac{N-1}{2}}. \tag{8}$$

Comparing (7) and (8) shows that the smaller δ is, the more the empirical mean estimator is dominated by MoM (at least in the worst-case scenario) in the sense of the $(1-\delta)$-quantile of the error. In practice, Figure 3 shows that using MoM prevents outliers in the resulting estimates.

Theorem 1 makes it clear that the number of groups K depends on the error threshold δ. As such, the user chooses K based on the risk of extreme errors in the particular application at hand. The log dependence of K on δ means that the behaviour of the estimate is expected to be robust to K. We will return to the choice of this parameter when discussing confidence intervals.

3. Combining importance sampling and median of means

There are at least two ways to combine importance sampling and median of means. Recall that μ^N_{AIS} (defined in (1)) is a ratio of two empirical means, and μ can be expressed as $\mathbb{M}(\omega\varphi)/\mathbb{M}(\omega)$. One natural idea is then to use MoM twice to estimate $M(\omega\varphi)$ and $\mathbb{M}(\omega)$ individually, then return the ratio of the two estimations. However, if φ is replaced by $\varphi + 1$, the estimator using this idea is not guaranteed to increase by exactly 1. Therefore, we shall prefer the second way described below.

We divide the N data points into K groups of size M each. For each group, we compute an importance sampling estimator using only its data points. This results in K different estimates, $\hat{\mu}_1, \ldots, \hat{\mu}_K$, defined as

$$\hat{\mu}_k := \frac{\sum_{n=M(k-1)+1}^{Mk} \varphi(X^n)\omega(X^n)}{\sum_{n=M(k-1)+1}^{Mk} \omega(X^n)}. \tag{9}$$

The final estimate is returned as their empirical median

$$\mu^{N,K}_{\mathrm{MoM}} := \mathrm{empmed}(\hat{\mu}_1, \ldots, \hat{\mu}_K). \tag{10}$$

The first result establishes the sub-Gaussian property of μ_{MoM}. For that, it is useful to define the following two quantities:

$$\begin{aligned} \sigma^2_{\mathrm{CS}} &:= \mathbb{M}(\bar{\omega}^2) - 1 \\ \sigma^2_{\mathrm{IS}} &:= \mathbb{M}(\bar{\omega}^2\bar{\varphi}^2) \end{aligned} \tag{11}$$

where $\bar{\omega} = \omega/Z$ is the normalised weight function; σ^2_{IS} is simply the asymptotic variance in (3), whereas σ^2_{CS} is the chi-squared pseudo-distance of $\mathbb{Q}$ with respect to $\mathbb{M}$.

Proposition 2. *Let $\delta \in]0,1[$ and suppose that $N \geq 8(32\sigma^2_{\mathrm{CS}} \vee 1)\log(1/\delta)$. Then the $\mu^{N,K}_{\mathrm{MoM}}$ estimator with $K = \lceil 8\log(1/\delta)\rceil$ satisfies*

$$\mathbb{P}\left(\left|\mu^{N,K}_{\mathrm{MoM}} - \mathbb{Q}(\varphi)\right| \geq 16\sqrt{\frac{\sigma^2_{\mathrm{IS}}\log(1/\delta)}{N}}\right) \leq \delta.$$

Proposition 2 is proved in Appendix A.2. The main difference with Theorem 1 is that the latter only requires $N > 8\log(1/\delta)$, instead of $N > 8(32\sigma^2_{\mathrm{CS}} \vee 1)(\log(1/\delta))$. For importance sampling, the minimum sample size does not depend only on δ but also on σ^2_{CS}, which is strongly linked to the ESS.

A natural question is whether a value of N depending only on δ, like Theorem 1, is possible in the case of importance sampling. The negative answer is given in the following proposition (proved in Appendix A.3).

Proposition 3. *For any function f, let $f(X^{1:N})$ be a shorthand for $(f(X^1),\dots,f(X^N))$. Suppose that $0 < \delta < 1/2$. Then, there do not exist a function $N_0 = N_0(\delta)$, an estimator $\mu^N = \mu^N(\varphi(X^{1:N}),\omega(X^{1:N}),\delta)$ and absolute constants C, D such that; for all $N \geq N_0(\delta)$ and all importance sampling problems defined by the triple $(\mathbb{M},\omega,\varphi)$ with $\sigma^2_{\mathrm{IS}} < \infty$ and $\sigma^2_{\mathrm{CS}} < \infty$, we have*

$$\mathbb{P}\left(\left|\mu^N - \mathbb{Q}(\varphi)\right| \geq C\sqrt{\frac{\sigma^2_{\mathrm{IS}}\log(D/\delta)}{N}}\right) \leq \delta. \tag{12}$$

We end this section by returning to the running example (Example 1). Figure 1 shows that the MoM estimator (with $K = 6$ subgroups here) again displays a much more regular behaviour than the ordinary importance sampling one.

4. Constructing confidence intervals

4.1. **CI for mean estimation.** As before, we start the discussion with the simpler problem of mean estimation, targeting $\mathbb{M}(\psi)$. The usual way to construct a confidence interval is to use the central limit theorem. However, this requires estimating its asymptotic variance. If the mean estimation is already difficult, the variance estimate will be even more so. Even if the variance is known, the asymptotic normality might not yet manifest at small sample sizes. As far as we know, in the literature there has been little effort to get around the variance estimation issue. One reason is that the variance is ubiquitous in theoretical bounds, including that of the MoM estimates (e.g. see (7) or Proposition 2).

We propose a new confidence interval for the mean estimation problem that is based on three observations. The first one is that the MoM estimator (5) does not directly target $\mathbb{E}_{\mathbb{M}}[\psi(X)]$, but rather

$$\mathrm{med}\left[\frac{\psi(X^1)+\cdots+\psi(X^M)}{M}\right],$$

where $\mathrm{med}[Z]$ denotes the median of the random variable Z.

Secondly, let $Z^1,\dots,Z^N$ be an i.i.d. sample from some real distribution Z. While it is impossible to get exact non-asymptotic confidence interval for $\mathbb{E}[Z]$ without further information, the task is possible for $\mathrm{med}[Z]$. Denote by $Z^{(1)},\dots,Z^{(N)}$ the sorted observations. One can construct a confidence interval with the desired level of the form $[Z^{(u)}, Z^{(v)}]$ using concentration inequalities for binomial random variables.

Thirdly, under finite third moment assumptions, med $\left[\frac{\psi(X^1)+\cdots+\psi(X^M)}{M}\right]$ converges to $\mathbb{M}(\psi)$ at rate $\mathcal{O}(1/M)$. This result is well-known, see e.g. Orenstein (2019); Minsker (2019). It can be proved by a simple application of the Berry-Esseen theorem, which we detail in a more general setting in Appendix A.6. If K is fixed while $N \to \infty$, the $\mathcal{O}(1/M)$ gap is negligible compared to the $\mathcal{O}(1/\sqrt{N})$ length of any confidence interval.

The following proposition describes our confidence interval. Its proof (Appendix A.4) simply puts together the three points above.

Proposition 4. *Let $K = \lceil \log_2(1/\delta) \rceil + 1$ and let $\hat{\theta}_{(1)} < \ldots < \hat{\theta}_{(K)}$ be the sorted group means, where the unsorted values are defined in (4). Then, if $\mathbb{M}(|\psi|^3) < \infty$, we have*

$$\lim_{N\to\infty} \mathbb{P}\left([\hat{\theta}_{(1)}, \hat{\theta}_{(K)}] \ni \mathbb{M}(\psi)\right) = 1 - \delta.$$

A disadvantage of this proposition is that the length of the resulting confidence intervals may vary a lot (since it relies on a minimum and a maximum). One workaround is to increase K and choose other order statistics. For instance, it can be proved, using similar techniques, that if $K = \lceil 8 \log(2/\delta) \rceil$ then the confidence interval $[\hat{\theta}_{(K/4)}, \hat{\theta}_{(3K/4)}]$ also has level δ asymptotically. However, we still recommend to stick with Proposition 4, in order to take M as large as possible. Indeed, the non-asymptotic cover rate of this kind of confidence intervals depends strongly on the symmetry of $\psi(X^1) + \cdots + \psi(X^M)$, which is unknown to the user. Hence we should not take any chance to take M smaller than possible.

We will now quantify the superiority of this confidence interval relative to confidence intervals relying o the central limit theorem (CLT). In this comparison, we will give an advantage to the CLT confidence interval by assuming that it has access to the true variance. The CLT interval is then given by

$$\left[\theta^N_{\mathrm{EMP}} - z_{1-0.5\delta}\sqrt{\frac{\sigma^2}{N}}, \theta^N_{\mathrm{EMP}} + z_{1-0.5\delta}\sqrt{\frac{\sigma^2}{N}}\right] \tag{13}$$

where $\sigma^2 := \mathrm{Var}_{\mathbb{M}}(\psi)$ and z_γ is the γ quantile of the standard normal distribution. This interval, as well as the one of Proposition 4, are only valid asymptotically. To study their adequateness for finite sample sizes, we introduce the notion of length-to-go, which quantifies, for a given sample size N, the "missing length" needed to really achieve the asymptotic level.

Definition 2 (Length-to-go)**.** *The length-to-go of a random interval $[\theta^N_\ell, \theta^N_h]$ with respect to a target quantity θ and a level δ is defined as*

$$R := \inf\{r \geq 0 \mid \mathbb{P}\left([\theta^N_\ell - \frac{r}{2}, \theta^N_h + \frac{r}{2}] \ni \theta\right) \geq 1 - \delta\}.$$

Proposition 5. *For all distributions $\mathbb{M}$ and functions ψ such that $\mathrm{Var}_{\mathbb{M}}(\psi) = \sigma^2 < \infty$, for all $\delta > 0$ and $N > 2 + \log_2(1/\delta)$, the length-to-go of the confidence interval constructed in Proposition 4 is less than*

$$2\sqrt{\frac{\sigma^2}{N}(2 + \log_2(1/\delta))}.$$

In contrast, given any $N \geq 1$ and $0 < \delta < 1/e$, there exists a distribution $\mathbb{M}$ and a function ψ such that $\mathrm{Var}_{\mathbb{M}}(\psi) = \sigma^2$, $\mathbb{M}(|\psi|^3) < \infty$ and the length-to-go of the

CLT-based confidence interval (13) *is at least*

$$2\sqrt{\frac{\sigma^2}{N}}\left(\frac{1}{\sqrt{\delta}}\left(1-\frac{\delta e}{N}\right)^{\frac{N-1}{2}} - z_{1-0.5\delta}\right)^+.$$

The main difference between the two length-to-go expressions is their dependences on δ (which are $\sqrt{\log(\delta)}$ for one and $\sqrt{1/\delta}$ for the other). Thus, this proposition (proved in Appendix A.5) shows that the smaller δ is, the more our confidence interval dominates the CLT-based one. A similar result also holds for the $[\hat{\theta}_{(K/4)}, \hat{\theta}_{(3K/4)}]$ interval described above.

4.2. **CI for importance sampling.** We show that it is possible to extend the confidence interval of Proposition 4 to importance sampling.

Proposition 6. *Let* $K = \lceil \log_2(1/\delta) \rceil + 1$ *and let* $\hat{\mu}_{(1)}, \dots, \hat{\mu}_{(K)}$ *be the sorted group means, where the unsorted values are defined in* (9)*. Suppose that the distribution of* μ_{AIS}^M *has a unique median, where* $M = N/K$ *and* μ_{AIS}^M *defined in* (1)*. Then, if* $\mathbb{M}(|\omega|^3) < \infty$ *and* $\mathbb{M}(|\omega^3\varphi^3|) < \infty$*, we have*

$$\lim_{N\to\infty} \mathbb{P}\left([\hat{\mu}_{(1)}, \hat{\mu}_{(K)}] \ni \mathbb{Q}(\varphi)\right) = 1 - \delta.$$

Proposition 6 is proved in Appendix A.6. We now come back to our running example (Example 1). We compare two constructions of confidence intervals: either using (3) or Proposition 6. We aim at confidence level 96.875% (which is $1 - 2^{-5}$ and results in $K = 6$). Using 10^5 runs, we see that the CLT-based interval only covers the true value 78% of the time, whereas our interval reaches 87%. While both values are still behind the required level, an increase of nearly 10% is significant.

5. Conclusion

We recommend to use our MoM estimator (10) and the confidence interval defined in Proposition 6 in lieu of the classical importance sampling estimate (1) and the CLT-based confidence interval. Proposition 2 suggests that the required sample size for good performance of the IS-MOM estimator is related to the chi-squared distance, whereas a related result for the classical importance sampling estimator (Chatterjee and Diaconis, 2018) relies on the Kullback-Leiber distance instead. It would be interesting to understand more thoroughly the connection between the two bounds. Finally, more numerical experiments are necessary to understand the behaviour of the newly proposed estimator and confidence interval in real-world settings.

References

Catoni, O. (2012). Challenging the empirical mean and empirical variance: A deviation study. *Annales de l'Institut Henri Poincar, Probabilits et Statistiques*, 48(4):1148 – 1185.

Chatterjee, S. and Diaconis, P. (2018). The sample size required in importance sampling. *Ann. Appl. Probab.*, 28(2):1099–1135.

Derumigny, A., Girard, L., and Guyonvarch, Y. (2019). On the construction of confidence intervals for ratios of expectations. *arXiv preprint arXiv:1904.07111*.

Devroye, L., Lerasle, M., Lugosi, G., and Oliveira, R. I. (2016). Sub-gaussian mean estimators. *The Annals of Statistics*, 44(6):2695–2725.

Gobet, E. (2016). *Monte Carlo methods and stochastic processes: from linear to non-linear*. Chapman and Hall/CRC.

Huggins, J. H. and Roy, D. M. (2019). Sequential Monte Carlo as approximate sampling: bounds, adaptive resampling via ∞-ESS, and an application to particle Gibbs. *Bernoulli*, 25(1):584–622.

Ionides, E. L. (2008). Truncated importance sampling. *Journal of Computational and Graphical Statistics*, 17(2):295–311.

Kong, A., Liu, J. S., and Wong, W. H. (1994). Sequential imputation and Bayesian missing data problems. *J. Am. Statist. Assoc.*, 89:278–288.

Lerasle, M. and Oliveira, R. I. (2011). Robust empirical mean estimators. *arXiv preprint arXiv:1112.3914*.

Lugosi, G. and Mendelson, S. (2019). Mean estimation and regression under heavy-tailed distributions: A survey. *Foundations of Computational Mathematics*, 19(5):1145–1190.

Minsker, S. (2019). Distributed statistical estimation and rates of convergence in normal approximation. *Electronic Journal of Statistics*, 13(2):5213–5252.

Orenstein, P. (2019). Robust mean estimation with the bayesian median of means. *arXiv preprint arXiv:1906.01204*.

Robert, C. P. and Casella, G. (2004). *Monte Carlo Statistical Methods, 2nd ed.* Springer-Verlag, New York.

Vehtari, A., Simpson, D., Gelman, A., Yao, Y., and Gabry, J. (2015). Pareto smoothed importance sampling. *arXiv preprint arXiv:1507.02646*.

Appendix A. Proofs

A.1. **Proof of Proposition 1.**

Proof. Let M_n be a strictly increasing sequence of real numbers such that $M_n \to \operatorname{esssup}_{\mathbb{Q}}(\bar{\omega})$. Define $S_n := \{x \in \mathcal{X} \mid \bar{\omega}(x) \geq M_n\}$. By definition of the essential supremum, we have $\mathbb{Q}(S_n) > 0$. Moreover

$$0 < \mathbb{Q}(S_n) = \int q(x) \mathbb{1}_{S_n}(x) \, \mathrm{d}x$$

where q is the density of $\mathbb{Q}$ with respect to the Lebesgue measure. The function

$$R \mapsto \int q(x) \mathbb{1}_{S_n}(x) \mathbb{1}_{\|x\| \leq R} \mathrm{d}x$$

equals to 0 at $R = 0$, tends to $\mathbb{Q}(S_n)$ as $R \to \infty$ and is continuous in between, thanks to the monotone convergence theorem. As such, it maps a certain R^* to $1/2$, which means that one can divide the set S_n into S_n^1 and S_n^2 such that $\mathbb{Q}(S_n^1) = \mathbb{Q}(S_n^2) = \mathbb{Q}(S_n)/2$. Define the function φ_n as

$$\varphi_n(x) := \begin{cases} 1 & \text{if } x \in S_n^1 \\ -1 & \text{if } x \in S_n^2 \\ 0 & \text{otherwise.} \end{cases}$$

Then $\bar{\varphi}_n = \varphi_n$ and $\varphi_n^2 = \mathbb{1}_{S_n}$. Consequently,

$$\frac{\mathbb{M}(\bar{\omega}^2 \bar{\varphi}_n^2)}{\mathbb{M}(\bar{\omega} \bar{\varphi}_n^2)} = \frac{\mathbb{Q}(\bar{\omega} \bar{\varphi}_n^2)}{\mathbb{Q}(\bar{\varphi}_n^2)} = \frac{\mathbb{Q}(\bar{\omega} \mathbb{1}_{S_n})}{\mathbb{Q}(\mathbb{1}_{S_n})} \geq M_n$$

by the definition of S_n. The proof follows by letting $n \to \infty$. □

A.2. **Proof of Proposition 2.**

Proof. Define the local importance sampling estimates $\hat{\mu}_1, \dots, \hat{\mu}_K$ as in (9). We have

$$|\hat{\mu}_1 - \mathbb{Q}(\varphi)| = \left| \frac{M^{-1} \sum_{m=1}^{M} \bar{\varphi}(X_m)\bar{\omega}(X_m)}{M^{-1} \sum_{m=1}^{M} \bar{\omega}(X_m)} \right|.$$

Using Chebychev's inequality, we have

$$\begin{cases} \left| M^{-1} \sum_{m=1}^{M} \bar{\varphi}(X_m)\bar{\omega}(X_m) \right| \leq \sqrt{8\sigma_{\mathrm{IS}}^2/M} \text{ with prob. at least } 7/8 \\ \left| M^{-1} \sum_{m=1}^{M} \bar{\omega}(X_m) - 1 \right| \leq \sqrt{8\sigma_{\mathrm{CS}}^2/M} \text{ with prob. at least } 7/8. \end{cases}$$

Therefore, using the fact that $M \geq 32\sigma_{\mathrm{CS}}^2$, the following holds with prob. at least $3/4$:

$$|\hat{\mu}_1 - \mathbb{Q}(\varphi)| \leq \frac{\sqrt{8\sigma_{\mathrm{IS}}^2/M}}{1 - \sqrt{8\sigma_{\mathrm{CS}}^2/M}} \leq \frac{\sqrt{8\sigma_{\mathrm{IS}}^2/M}}{1 - 1/2} = 16\sqrt{\frac{\sigma_{\mathrm{IS}}^2 \log(1/\delta)}{N}} =: \varepsilon.$$

Thus

$$\begin{aligned} \mathbb{P}\left(|\mathrm{empmed}(\hat{\mu}_1, \dots, \hat{\mu}_K) - \mathbb{Q}(\varphi)| \geq \varepsilon\right) &= \mathbb{P}\left(\sum_{k=1}^{K} \mathbb{1}\{|\hat{\mu}_k - \mathbb{Q}(\varphi)| \geq \varepsilon\} \geq K/2 \right) \\ &\leq \mathbb{P}\left(\mathrm{Bin}(K, 1/4) \geq K/2\right) \\ &\leq e^{-K/8} \text{ by Hoeffding's inequality} \\ &= \delta \end{aligned}$$

where $\mathrm{Bin}(n, p)$ refers to the binomial distribution of parameters n and p. □

A.3. **Proof of Proposition 3.**

Proof. We use similar ideas to the proof of Theorem 3.1 in Devroye et al. (2016). Suppose that there exist N_0 and μ^N satisfying (12). By considering two importance sampling problems $(\mathbb{M}_-, \omega_-, \varphi_-)$ and $(\mathbb{M}_+, \omega_+, \varphi_+)$ such that $\mathbb{Q}_-(\varphi_-) \neq \mathbb{Q}_+(\varphi_+)$ but the law of μ_-^N and μ_+^N are very similar, we conclude that at least one of the two estimators must have a bad performance. More specifically, let $0 < \alpha < 1$, $\beta := 1 - \alpha$ and define

$$\begin{aligned} \mathbb{M}_+ &:= \beta\delta_0 + \alpha\delta_1 & \mathbb{M}_- &:= \beta\delta_0 + \alpha\delta_{-1} \\ \mathbb{Q}_+ &:= \alpha\delta_0 + \beta\delta_1 & \mathbb{Q}_- &:= \alpha\delta_0 + \beta\delta_{-1} \\ \varphi_+(x) &:= \mathbb{1}\{x = 1\} & \varphi_-(x) &:= -\mathbb{1}\{x = -1\} \end{aligned}$$

Let $X_+^{1:N} \overset{\mathrm{iid}}{\sim} \mathbb{M}_+$, $X_-^{1:N} \overset{\mathrm{iid}}{\sim} \mathbb{M}_-$ and define μ_+^N and μ_-^N as deterministic functions of $X_+^{1:N}$ and $X_-^{1:N}$ respectively

$$\begin{aligned} \mu_+^N &= \mu^N(\varphi_+(X_+^{1:N}), \omega_+(X_+^{1:N}), \delta) \\ \mu_-^N &= \mu^N(\varphi_-(X_-^{1:N}), \omega_-(X_-^{1:N}), \delta) \end{aligned}$$

where $\omega_+ = \frac{\mathrm{d}\mathbb{Q}_+}{\mathrm{d}\mathbb{M}_+}$ and $\omega_- = \frac{\mathrm{d}\mathbb{Q}_-}{\mathrm{d}\mathbb{M}_-}$.

Since $\mathbb{P}(X_+^{1:N} = 0) = \mathbb{P}(X_-^{1:N} = 0) = \beta^N$, the two estimators μ_+^N and μ_-^N are equal with probability at least β^{2N}. However, their target quantities $\mathbb{Q}_+(\varphi_+)$ and $\mathbb{Q}_-(\varphi_-)$ are 2β apart. Therefore

$$\mathbb{P}\left(\left|\mu_+^N - \mathbb{Q}_+(\varphi_+)\right| \geq \beta \text{ or } \left|\mu_-^N - \mathbb{Q}_-(\varphi_-)\right| \geq \beta\right) \geq \beta^{2N}.$$

Hence one of the two events under consideration must have probability greater than $\beta^{2N}/2$. Without loss of generality, we may suppose that

$$\mathbb{P}\left(\left|\mu_+^N - \mathbb{Q}_+(\varphi_+)\right| \geq \beta\right) \geq \beta^{2N}/2. \tag{14}$$

Comparing (12) and (14) would lead to contradiction if one could choose a β such that

$$\begin{cases} \beta^{2N}/2 & \geq \delta \\ \beta & \geq C\sqrt{\frac{\sigma_{\mathrm{IS}}^2 \log(D/\delta)}{N}}. \end{cases}$$

Satisfying these conditions is easy by letting $\beta \to 1$, since $\sigma_{\mathrm{IS}}^2 = \alpha\beta$ and $\delta \in]0, 1/2[$. □

A.4. **Proof of Proposition 4.**

Proof. Put $\tilde{\theta} := \mathrm{med}\left[\frac{\psi(X^1)+\cdots+\psi(X^M)}{M}\right]$ where $\mathrm{med}[L]$ denotes the median of a random variable L. For any $v \in \{1, 2, \ldots, K\}$, we have

$$\begin{aligned} \mathbb{P}\left(\tilde{\theta} > \hat{\theta}_{(v)}\right) &= \mathbb{P}\left(\hat{\theta}_{(1)} < \tilde{\theta}, \ldots, \hat{\theta}_{(v)} < \tilde{\theta}\right) \\ &= \mathbb{P}\left(\sum_{k=1}^{K} \mathbb{1}\{\theta_k < \tilde{\theta}\} \geq v\right) \\ &= \mathbb{P}\left(\mathrm{Bin}(K, 1/2) \geq v\right) \end{aligned}$$

where $\mathrm{Bin}(\cdot, \cdot)$ refers to the Binomial distribution with specified parameters. In particular, for $v = K$, we have $\mathbb{P}(\tilde{\theta} > \hat{\theta}_{(K)}) = 2^{-K} = \delta/2$. Similarly $\mathbb{P}(\tilde{\theta} < \hat{\theta}_{(1)}) = \delta/2$. Thus

$$\mathbb{P}\left([\hat{\theta}_{(1)}, \hat{\theta}_{(K)}] \ni \tilde{\theta}\right) = 1 - \delta. \tag{15}$$

It remains to bound

$$\begin{aligned} &\left|\mathbb{P}\left([\hat{\theta}_{(1)}, \hat{\theta}_{(K)}] \ni \tilde{\theta}\right) - \mathbb{P}\left([\hat{\theta}_{(1)}, \hat{\theta}_{(K)}] \ni \mathbb{M}(\psi)\right)\right| \\ &\quad \leq \mathbb{P}\left([\hat{\theta}_{(1)}, \hat{\theta}_{(K)}] \text{ contains exactly one of } \tilde{\theta} \text{ and } \mathbb{M}(\psi)\right) \\ &\quad \leq \mathbb{P}\left(\text{either } \hat{\theta}_{(1)} \text{ or } \hat{\theta}_{(K)} \text{ lies between } \tilde{\theta} \text{ and } \mathbb{M}(\psi)\right) \\ &\quad \leq K\mathbb{P}\left(\hat{\theta}_1 \text{ lies between } \tilde{\theta} \text{ and } \mathbb{M}(\psi)\right) \\ &\quad = K\mathbb{P}\left(\sqrt{M}\left(\hat{\theta}_1 - \mathbb{M}(\psi)\right) \text{ lies between 0 and } \sqrt{M}\left(\tilde{\theta} - \mathbb{M}(\psi)\right)\right). \end{aligned} \tag{16}$$

Note that K is fixed, so M tends to infinity at the same rate as N. Moreover, $\sqrt{M}(\hat{\theta}_1 - \mathbb{M}(\psi))$ tends to a normal distribution and $\sqrt{M}(\tilde{\theta} - \mathbb{M}(\psi))$ tends to 0 by a simple application of the Berry-Esseen theorem (see proof of Proposition 6, Appendix A.6 for a proof of this result in a more general setting). Therefore (16) converges to 0 as $N \to \infty$. □

A.5. **Proof of Proposition 5.**

Proof. Recall that the confidence interval of Proposition 4 satisfies

$$\mathbb{P}\left([\hat{\theta}_{(1)}, \hat{\theta}_{(K)}] \ni \tilde{\theta}\right) = 1 - \delta$$

where $\tilde{\theta} = \text{med}\left[\frac{\psi(X^1)+\ldots+\psi(X^M)}{M}\right]$ (see (15)). Using the classical inequality $|\mathbb{E}[Z] - \text{med}[Z]| \leq \sqrt{\text{Var}(Z)}$ for any r.v. Z (see, e.g. Orenstein, 2019, Prop. 16), we have that $\left|\tilde{\theta} - \mathbb{M}(\psi)\right| \leq \sqrt{\sigma^2/M}$, from which the first part of the proposition follows.

For the second part, let R be the length-to-go for the CLT-based confidence interval. Then

$$\mathbb{P}\left(\mathbb{M}(\psi) \geq \theta^N_{\text{EMP}} - z_{1-0.5\delta}\sqrt{\frac{\sigma^2}{N}} - \frac{R}{2}\right) \geq 1 - \delta$$

which is equivalent to

$$\mathbb{P}\left(\theta^N_{\text{EMP}} - \mathbb{M}(\psi) > z_{1-0.5\delta}\sqrt{\frac{\sigma^2}{N}} + \frac{R}{2}\right) \leq \delta.$$

According to Theorem 2, there exists $\mathbb{M}$ and ψ such that

$$\mathbb{P}\left(\theta^N_{\text{EMP}} - \mathbb{M}(\psi) > \sqrt{\frac{\sigma^2}{N\delta}}\left(1 - \frac{\delta e}{N}\right)^{\frac{N-1}{2}}\right) \geq \delta.$$

Thus

$$z_{1-0.5\delta}\sqrt{\frac{\sigma^2}{N}} + \frac{R}{2} \geq \sqrt{\frac{\sigma^2}{N\delta}}\left(1 - \frac{\delta e}{N}\right)^{\frac{N-1}{2}}$$

from which the second part follows. □

A.6. **Proof of Proposition 6.**

Proof. We need to prove that

$$\text{med}\left[\frac{\sum_{m=1}^M \varphi(X^m)\omega(X^m)}{\sum_{m=1}^M \omega(X^m)}\right] = \mathbb{Q}(\varphi) + \mathcal{O}(\frac{1}{M}) \tag{17}$$

as $M \to \infty$. The rest follows the same lines as the proof of Proposition 4 (Appendix A.4) and is therefore omitted. Put

$$\gamma^M := \frac{\sqrt{M}}{\sigma_{\text{IS}}}\left[\frac{\sum_{m=1}^M \varphi(X^m)\omega(X^m)}{\sum_{m=1}^M \omega(X^m)} - \mathbb{Q}(\varphi)\right]$$

where σ_{IS} is defined in (11). Let F^M be the cumulative distribution function of γ^M. Recall that $\bar{\varphi} = \varphi - \mathbb{Q}(\varphi)$ and $\bar{\omega}(x) = \omega(x)/\mathbb{M}(\omega)$. For any $t \geq 0$, we have

$$\begin{aligned}
1 - F^M(t) = \mathbb{P}(\gamma^M > t) &= \mathbb{P}\left(\frac{\sum_m \bar{\varphi}(X^m)\bar{\omega}(X^m)}{\sqrt{M}\sigma_{\text{IS}}} > t\frac{\sum_m \bar{\omega}(X^m)}{M}\right) \\
&\leq \mathbb{P}\left(\frac{\sum_m \bar{\varphi}(X^m)\bar{\omega}(X^m)}{\sqrt{M}\sigma_{\text{IS}}} > \frac{t}{2}\right) + \mathbb{P}\left(\frac{\sum_m \bar{\omega}(X^m)}{M} \leq \frac{1}{2}\right) \\
&\leq 1 - \Phi\left(\frac{t}{2}\right) + \mathcal{O}\left(\frac{1}{\sqrt{M}}\right) + \frac{4\sigma^2_{\text{CS}}}{M}
\end{aligned}$$

where Φ is the cumulative distribution function of the standard normal variable. In the last inequality, the first three terms follow from the Berry-Esseen theorem and the hypothesis $\mathbb{M}(\left|\bar{\varphi}^3\bar{\omega}^3\right|) < \infty$, whereas the last term results from Chebychev inequality. Thus

$$\Phi\left(\frac{t}{2}\right) \leq F^M(t) + \mathcal{O}\left(\frac{1}{\sqrt{M}}\right), \forall t \geq 0.$$

A similar argument leads to

$$\Phi\left(\frac{3}{2}t\right) \geq F^M(t) + \mathcal{O}\left(\frac{1}{\sqrt{M}}\right), \forall t \leq 0.$$

Now put

$$z^M := \text{med}[\gamma^M]. \tag{18}$$

By the hypothesis on the uniqueness of z^M, we have $F^M(z^M) = 1/2$. Thus, if $z^M \geq 0$,

$$\Phi\left(\frac{1}{2}z^M\right) \leq \frac{1}{2} + \mathcal{O}(1/\sqrt{M})$$

which entails

$$\frac{1}{2}z^M \leq \Phi^{-1}\left(\frac{1}{2} + \mathcal{O}(1/\sqrt{M})\right) = \mathcal{O}(1/\sqrt{M})$$

since the inverse of Φ at $1/2$ is 0 and Φ^{-1} is differentiable everywhere. If $z^M \leq 0$, similar arguments lead to

$$\frac{3}{2}z^M \geq \mathcal{O}(1/\sqrt{M}).$$

In all cases, we have $|z^M| = \mathcal{O}(1/\sqrt{M})$, which, via (18), justifies (17). □

List of talks

- 2021, UCL Centre for AI, Online. Waste-free Sequential Monte Carlo (talk with Nicolas Chopin)
- 2021, End-to-End Bayesian Learning workshop, Marseille. Waste-free Sequential Monte Carlo (30 minutes)
- 2022, 5^{th} workshop on Sequential Monte Carlo methods, Madrid. On the backward sampling step in the smoothing of state-space models (poster)
- 2022, CREST research day, Palaiseau. Importance sampling using Median of means.
- 2022 (forthcoming), Monte Carlo Methods and Approximate Dynamic Programming workshop, ESSEC Business School, Cergy. On the complexity of backward sampling algorithms.

ECOLE
DOCTORALE
DE MATHEMATIQUES
HADAMARD

Titre : Calcul bayésien séquentiel

Mots clés : échantilonneur, séquentiel, lissage, Monte-Carlo, particules

Résumé : Cette thèse est composée de deux parties. La première concerne les échantillonneurs dits de Monte-Carlo séquentiel (les échantillonneurs SMC). Il s'agit d'une famille d'algorithmes pour produire des échantillons venant d'une suite de distributions, grâce à une combinaison de l'échantillonnage pondéré et la méthode de Monte-Carlo par chaîne de Markov (MCMC). Nous proposons une version améliorée qui exploite les particules intermédiaires engendrées par l'application de plusieurs pas de MCMC. Elle a une meilleure performance, est plus robuste et permet la construction d'estimateurs de la variance. La deuxième partie analyse des algorithmes de lissage existants et en propose des nouveaux pour les modèles espace-état. Le lissage étant coûteux en temps de calcul, l'échantillonnage par rejet a été proposé dans la littérature comme une solution. Cependant, nous démontrons que son temps d'exécution est très variable. Nous développons des algorithmes ayant des coûts de calcul plus stables et ainsi plus adaptés aux architectures parallèles. Notre cadre peut aussi traiter des modèles dont la densité de transition n'est pas calculable.

Title : Sequential Bayesian Computation

Keywords : sampler, sequential, smoothing, Monte Carlo, particles

Abstract : This thesis is composed of two parts. The first part focuses on Sequential Monte Carlo samplers, a family of algorithms to sample from a sequence of distributions using a combination of importance sampling and Markov chain Monte Carlo (MCMC). We propose an improved version of these samplers which exploits intermediate particles created by the application of multiple MCMC steps. The resulting algorithm has a better performance, is more robust and comes with variance estimators. The second part analyses existing and develops new smoothing algorithms in the context of state space models. Smoothing is a computationally intensive task. While rejection sampling has been proposed as a solution, we prove that it has a highly variable execution time. We develop algorithms which have a more stable computational cost and thus are more suitable for parallel environments. We also extend our framework to handle models with intractable transition densities.

Institut Polytechnique de Paris
91120 Palaiseau, France