Contributions to multilevel splitting for rare events, and applications to air traffic

Damien Jacquemart

ANNÉE 2014

THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Européenne de Bretagne

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : mathématiques et applications

École doctorale MATISSE

présentée par

Damien Jacquemart

préparée à
ONERA centre de Palaiseau
INRIA Rennes Bretagne–Atlantique

Contributions aux méthodes de branchement multi-niveaux pour les évènements rares, et applications au trafic aérien.

Thèse soutenue à Palaiseau
le 08/12/2014

devant le jury composé de :

Emmanuel Gobet
Professeur, École Polytechnique / Rapporteur

Éric Moulines
Professeur, Télécom-ParisTech / Rapporteur

Josselin Garnier
Professeur, Université Paris Diderot / Examinateur

Véronique Serfaty
Responsable Scientifique, DGA / Examinatrice

François Le Gland
Directeur de recherche, INRIA / Directeur de thèse

Jérôme Morio
Ingénieur chercheur, ONERA / Encadrant de thèse

“Je m’avancerai bientôt dans ce désert immense, parfaitement plat et incommensurable, où le cœur vraiment pieux succombe, bienheureux. Je m’abîmerai dans la ténèbre divine, en un silence muet et en une union ineffable, et m’abîmant seront perdues toute égalité et toute inégalité, et en cet abîme mon esprit se perdra lui-même, et il ne connaîtra ni l’égal ni l’inégal ni rien d’autre : et seront oubliées toutes les différences, je serai dans le fondement simple, dans le désert silencieux où jamais l’on ne vit de diversité, dans l’intime où personne ne se retrouve dans son propre lieu. Je tomberai dans la divinité silencieuse et inhabitée où il n’est ni œuvre ni image.

Il fait froid dans le scriptorium, j’ai mal au pouce. Je laisse cet écrit, je ne sais pour qui, je ne sais plus à propos de quoi.”

Umberto Eco, Le Nom de la rose

Contents

Remerciements

Publications and communications

- D. Jacquemart and J. Morio, Conflict probability estimation between aircraft with dynamic importance Splitting, *Safety Science*, 2013, Vol. 51, n°1, pp. 94-100.
- J. Morio, D. Jacquemart, M. Balesdent and J. Marzat, Optimisation of interacting particle system for rare event estimation, *Computational Statistics and Data Analysis*, 2013, Vol. 66, pp. 117-128.
- J. Morio, M. Balesdent, D. Jacquemart and C. Vergé, A survey of rare event estimation methods for static input-output models, conditionally accepted in *Simulation Modelling Practice and Theory* (minor revision).
- D. Jacquemart, F. Le Gland, and J. Morio (2013). A combined important splitting important sampling for rare event estimation, in *Winter Simulation Conference*, Washington.
- D. Jacquemart and F. Le Gland (2014). Convergence of a two step adaptive multilevel splitting algorithm, in Proceedings of *6th International Workshop on Rare Event Simulation, RESIM*, Tinbergen Institute, Amsterdam.
- D. Jacquemart (2014). Adaptive particle methods for rare event simulation in a Markovian framework, in *MascotNum Annual Conference*, ETH Zurich.
- L. Santinelli, J. Morio, G. Dufour and D. Jacquemart (2014). On the Sustainability of the Extreme Value Theory for WCET Estimation, in *WCET 2014*: 21-30, Madrid, Spain.

Résumé étendu en Français

Contexte

Ces dernières décennies, les normes de sécurité et les standards de qualité sont devenus de plus en plus exigeants dans de nombreux domaines industriels et scientifiques. Les communautés scientifiques ont alors porté un grand intérêt à l'estimation et la simulation d'événements de probabilité très petites, typiquement entre 10^{-4} et 10^{-10}. C'est notamment le cas en finance [Carmona and Crepey, 2010, Giesecke et al., 2010], climatologie [Blanchet et al., 2009], fiabilité des systèmes en énergie renouvellable [Wadman et al., 2013], réseaux de communications [Garvels, 2000], transmission par fibre optique [Garnier and Del Moral, 2006] ou régulation du trafic aérien [Prandini et al., 2011]. Afin d'étudier de tels systèmes des modèles mathématiques ont été mis au point. Il est alors possible d'utiliser des méthodes statistiques et numériques pour estimer et simuler ces évènements de très faible probabilité.

Dans le contexte de la simulation d'évènements rares, les méthodes de Monte Carlo traditionnelles ne sont plus efficaces. En effet, pour estimer la probabilité d'un évènement qui a une chance sur 10^9 de se produire il faudrait répéter en moyenne 10^9 fois l'expérience pour voir ne serait-ce qu'une seule réalisation de cet évènement. Les temps de calcul seraient alors déraisonnables.

Dans cette thèse, nous nous intéressons à deux différents problèmes d'estimation. Tout d'abord, nous nous sommes concentrés sur l'estimation de la probabilité, supposée très petite, qu'un processus aléatoire Markovien $\{X(t),\ t \geq 0\}$ atteigne un ensemble B avant un temps final T

$$\mathbb{P}(X(t) \in B,\ \text{pour un certain } t \leq T). \tag{1}$$

Le temps final T peut être déterministe ou bien aléatoire. Dans cette dernière éventualité, on s'intéressera par exemple à la probabilité que le processus $\{X(t),\ t \geq 0\}$ atteigne un ensemble B avant d'atteindre un autre ensemble Δ.

Le deuxième problème auquel nous nous intéressons concerne l'estimation de la probabilité, supposée très faible aussi, qu'une chaîne de Markov $\{\Xi_k,\ k = 0,,\ldots,,m\}$ atteigne un ensemble critique au dernier instant déterministe de son évolution

$$\mathbb{P}(\Xi_m \in C). \tag{2}$$

Ces deux problèmes sont liés de plusieurs manières, comme on le décrira plus bas.

Étude bibliographique

Pour estimer les probabilités d'évènements rares (1) et (2), on distingue trois différentes approches: statistiques, analytiques et numériques. Les méthodes statistiques peuvent être utilisées pour les deux problèmes d'estimation mentionnés ci-dessus. Les deux autres contiennent chacune des méthodes propres aux deux problèmes d'intérêt.

Méthodes statistiques et théorie des valeurs extrêmes

Sous certaines conditions, la théorie des valeurs extrêmes affirme que la loi du maximum d'une séquence de variables aléatoires indépendantes de identiquement distribuées converge vers une loi d'extremum généralisé [Embrechts et al., 2011, Neves and Fraga A., 2004, Pickands, 1975]. Cette méthode est d'un grand intérêt lorsque l'on travaille avec un échantillon de taille fixée et qu'il n'est pas possible de ré-échantillonner. Cette méthode requiert un paramétrage qui influence grandement la qualité de l'estimation. De plus les hypothèses assurant cette convergence ne peuvent pas être vérifié en pratique. Enfin, étudier un processus dans son régime non critique pour en déduire des informations sur son comportement dans son régime critique (*i.e.* conditionnellement au fait que l'ensemble critique soit atteint) peut conduire à de mauvaises estimées, en particulier si la dynamique du processus change au delà du régime non extrême.

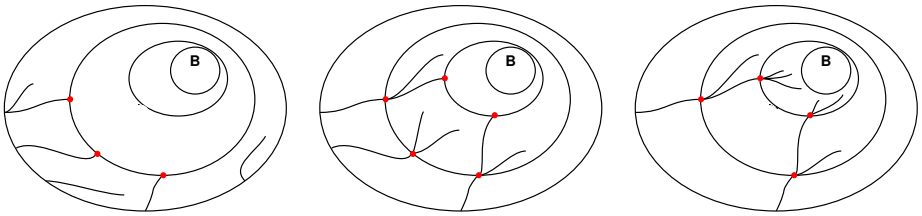
L'ensemble critique est atteint avant le temps final T

Résultats théoriques Des résultats théoriques pour l'estimation de la probabilité de l'équation (1) sont proposés dans [Albin, 1998, Albin, 1990, Albin and Sunden, 2008]. On y trouve en particulier des résultats pour les processus stationnaires, Gaussien, de Rayleigh et de Lévy. Le livre [Aldous, 1989] propose aussi une heuristique sans démonstration. De telles approches paraissent utiles mais leur puissance est limitée par la complexité du problème. Seuls certains cas simples peuvent être traités.

Méthodes de simulations numériques Dans ce cas, les méthodes disponibles dans la littérature scientifiques sont l'algorithme de branchement multi-niveaux et les méthodes d'échantillonnage préférentiel.

Algorithme de branchement multi-niveaux Pour augmenter le nombre de visites du processus à l'ensemble critique, l'algorithme de branchement multi-niveaux (splitting ou encore RESTART en anglais) repose sur l'existence de régions intermédiaires, bien identifiables, avant l'ensemble critique. L'idée est de travailler avec un ensemble de trajectoires partant de chaque région intermédiare. Celles qui ont atteint une région intermédiaire sont multipliées, les autres sont éliminées. Les trajectoires sont alors d'une certaine manière orientées vers l'ensemble critique. Cet algorithme a été mis en place sous la forme de nombreuses variantes dont on peut trouver une présentation dans [L'Ecuyer and Tuffin, 2007, L'Ecuyer et al., 2006] et des comparaisons d'efficacité dans [Garvels and Kroese, 1998]. Les avancées les plus récentes sont exposées dans [Cérou et al., 2006b]. Dans tous les cas, la probabilité d'atteindre la région B est décomposée en un produit de probabilités conditionnelles d'atteindre une région intermédiaire partant de la précédente. Une illustration de l'algorithme se trouve en figure 0.1.

L'enjeu principal dans la mise en place de cet algorithme est le choix des régions intermédiaires. Une des idées proposée dans la littérature est de considérer des quantiles de la loi



Figures 0.1: L'algorithme de branchement multi-niveaux.

du supremum du processus en question [Garvels, 2000, Cérou and Guyader, 2007]. Une autre manière [Wadman et al., 2013] est de mettre un certain nombre de régions intermédiaires et de sectionner, grâce aux les travaux de [Amrein and Kunsch, 2011], celles qui réduisent le plus la variance. Enfin dans le contexte du trafic aérien [Prandini et al., 2011], les régions sont en quelque sorte déjà données par la réglementation internationale et correspondent à des zones de conflit de plus en plus critiques.
Dans [Garvels, 2000, Cérou and Guyader, 2007], les trajectoires sont systématiquement recommencées à $t = 0$ avec les mêmes noyaux de transitions ce qui restreint le cadre de travail aux processus homogènes en temps et à un temps final aléatoire. Quant aux travaux de [Wadman et al., 2013], ils ne s'appliquent que si les régions intermédiaires ne sont pas trop irrégulièrement placées. Enfin les zones de conflit proposées dans[Prandini et al., 2011] ne prennent pas en compte le contrôle de la variance en fonction du scénario considéré.

Une méthode plus générale pour l'estimation des régions intermédiaire est à élaborer. En particulier, il n'existe pas de méthode générale pour tout temps final T et tout processus Markovien.

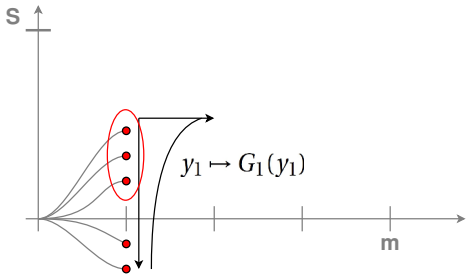
Échantillonnage préférentiel Les méthodes d'échantillonnage préférentiel [Sandmann, 2005, Glynn and Iglehart, 1989, Juneja and Shahabuddin, 2006] ont pour but de simuler le processus sous une autre loi qui le forcerait à aller vers l'ensemble critique. Ces méthodes sont efficaces dans le cas de processus évoluant dans un espace discret [Heidelberger, 1995, Goyal et al., 1992, Alexopoulos and Shultes, 2001, Goyal et al., 1987, Nakayama, 1996, Ahamed et al., 2006]. Pour un processus en à espace d'état continu, il n'existe pas de méthode générale pour déterminer une loi auxiliaire efficace. Ainsi, nous n'étudierons pas cette méthodes, ni n'y comparerons nos résultats.

L'ensemble critique est atteint au temps final T Dans ce cas, on trouve essentiellement des techniques d'échantillonnage préférentiel et l'algorithme des redistributions pondérées.

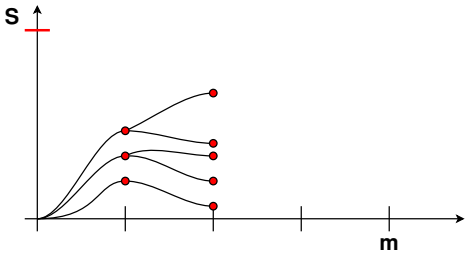
Échantillonnage préférentiel pour un processus de diffusion Pour réduire la variance dans l'estimation Monte Carlo dans le cas d'un processus de diffusion, les auteurs de [Newton, 1994, Fouque and Tullie, 2002, Carmona and Crepey, 2010] proposent une mesure auxiliaire définie sur l'espace des trajectoire du processus. Néanmoins, un paramétrage est nécessaire et est impossible à mettre en pratique pour des cas complexes.

Algorithme des redistributions pondérées Cet algorithme est présenté originellement dans [Del Moral and Garnier, 2005]. L'idée de travailler avec un ensemble de trajectoires et de sélectionner et multiplier, à chaque instant d'une sequence déterministe fixée à l'avance de temps de l'évolution du processus, les trajectoires qui ont le plus de chance d'atteindre

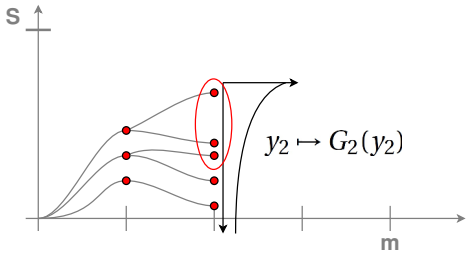
l'ensemble critique. Cet algorithme a l'avantage sur l'échantillonnage préférentiel de ne pas requérir de distribution auxiliaire. C'est souvent l'alternative la plus confortable, si ce n'est la seule. Les figures 0.2 illustrent cet algorithme.



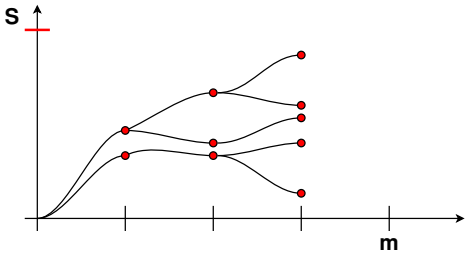
1. Sélection.



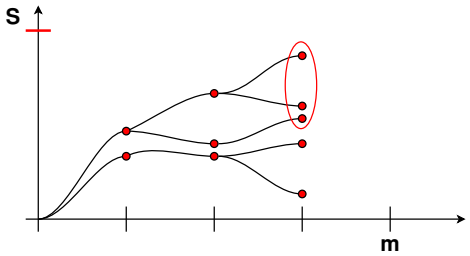
2. Mutation.



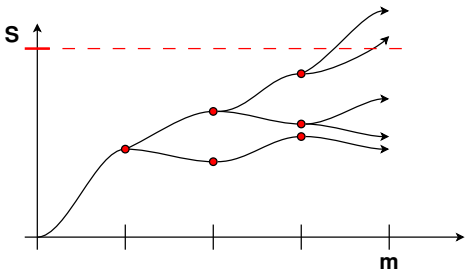
3. Sélection.



4. Mutation.



5. Sélection.



6. Mutation et estimation.

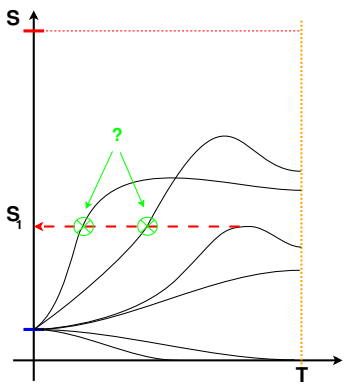
Figures 0.2: L'algorithme des redistributions pondérées utilisant des fonctions de sélections $G_k(y_k) = G_k(x_1, \ldots, x_k)$.

Cet algorithme requiert un paramétrage et, à notre connaissance, il n'existe ni règle ni méthode. En pratique, il est nécessaire d'effectuer plusieurs tests de paramètre pour trouver une valeur efficace de celui-ci. Ce genre de méthode peut être coûteuse en budget de simulation.

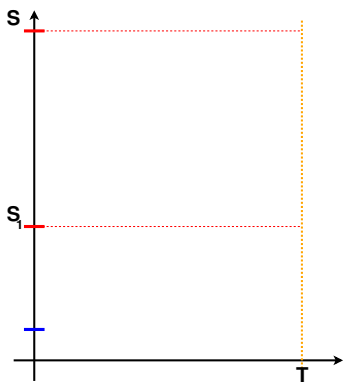
Contributions

Dans cette thèse, nous proposons des paramétrages automatiques pour des algorithmes existants ainsi que de nouveaux algorithmes pour simuler des événements critiques pouvant survenir lors de l'évolution d'un processus aléatoire $\{X(t),\ t \geq 0\}$.

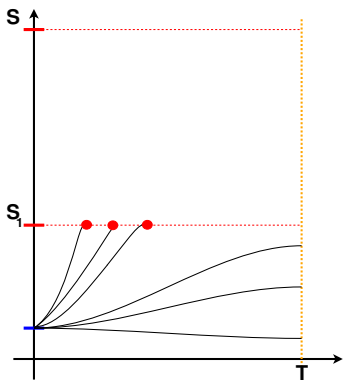
Tout d'abord, nous avons développé une méthode qui peut être utilisée avec tout processus stochastique et pour tout temps final T pour l'estimation de la probabilité de l'equation 1. Nous proposons une modification de l'algorithme de branchement multi-niveaux. Celui-ci se compose de deux étapes. La première consiste en l'estimation d'une région intermédiaire et la seconde donne l'estimation de la probabilité conditionnelle d'atteindre cette nouvelle région partant de la précédente. L'algorithme est résumé dans les figures 0.3. Des résultats de convergences presque sûre sur l'estimation de la probabilité rare et de l'approximation de la distribution d'entrée dans les niveaux intermédiaires ont été obtenus. De plus, notre méthode est étendue à l'estimation de quantiles extrêmes du processus. Ces résultats ont fait l'objet d'une communication [Jacquemart and Le Gland, 2014].



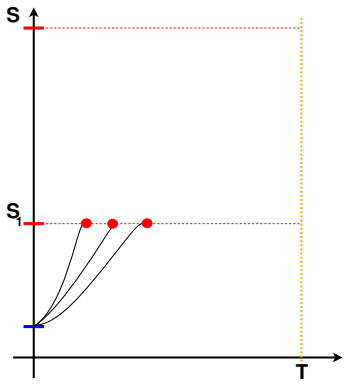
a) Estimation du quantile.



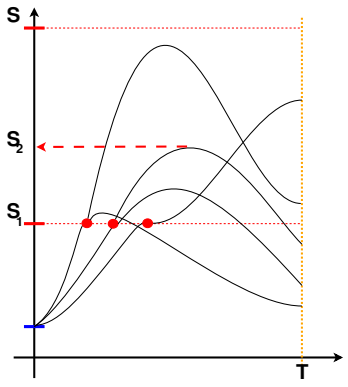
b) Création de la première région.



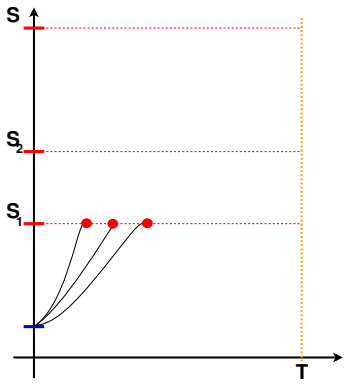
c) Approximation distribution d'entrée.



d) Trajectoires conservées.



e) Estimation du quantile.



f) Création de la deuxième région.

Figures 0.3: Les premières étapes de l'algorithme de branchement multi-niveaux adaptatif.

Nous nous sommes ensuite tournés vers la réduction du bais et de la variance dans l'algorithme de branchement multi-niveaux. L'algorithme de branchement multi-niveaux est théoriquement non biaisé mais la discrétisation des trajectoires dans l'implémentation peut con-

duire à une sous-estimation de la probabilité si une trajectoire discrétisée *manque* une région intermédiaire. Nous proposons un algorithme avec des niveaux intermédiaires déformés (voir figure 0.1), en nous basant sur les travaux de [Gobet and Menozzi, 2010]. Nous montrons numériquement que notre méthode corrige le biais sans affecter la variance ni les temps de calcul. En ce qui concerne la variance, nous savons qu'il en existe une borne inférieure, qu'il est théoriquement possible d'atteindre si nous choisissons de manière très particulière les régions intermédiaires, appelés alors régions optimales. Nous avons proposé pour des cas simples deux méthodes d'estimation de ces niveaux et nous avons numériquement montré que nous atteignons la borne inférieure pour la variance.

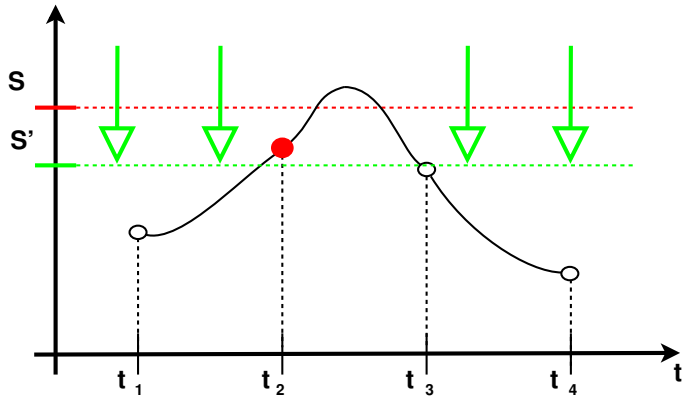


Figure 0.1: Le processus discrétisé rate le seuil mais S mais pas le seuil S'.

La suite des sous ensembles emboités réduisant la variance à sa borne inférieure peut s'avérer très délicate à déterminer, voire impossible pour des cas complexes. Nous avons alors proposé une variante à l'algorithme de branchement multi-niveaux visant à privilégier les trajectoires les plus précoces à l'atteinte d'une région intermédiaire. En effet, intuitivement, les trajectoires qui atteignent une région le plus tôt ont plus de temps (et dans la plupart des cas cela équivaut à plus de chance) d'atteindre l'ensemble critique cible. Nous avons appelé cette méthode l'algorithme de branchement multi-niveau pondéré. Nous avons numériquement montré que les performances de notre algorithme, en terme de variance, sont à mi-chemin entre l'algorithme de branchement multi-niveaux avec des régions standards et avec les régions optimales. Nous avons développé un cadre théorique à cet algorithme. Il est basé sur des approximations particulaires de distributions de Feynmna-Kac. Nous avons également étendu ce cadre de travail afin de mettre dans un même cadre théorique l'algorithme des redistributions pondérées et l'algorithme de branchement multi-niveaux. Ainsi, les possibilités de l'algorithme des redistributions pondérées sont augmentées, en particulier nous pouvons alors considérer des fonctions de sélection pouvant s'annuler. Nous avons enfin démontré des théorèmes central-limite pour l'estimation de la probabilité avec l'algorithme de branchement multi-niveaux pondéré et l'algorithme des redistributions pondérées avec des fonctions de sélection positives ou nulles. Ces résultats ont fait l'objet d'une communication [Jacquemart et al., 2013].

Nous nous sommes par la suite demandé si l'algorithme de branchement multi-niveaux était le seul disponible, facile à paramétrer pour l'estimation de la probabilité (1). Nous nous sommes orientés vers l'algorithme des redistributions pondérées. En effet, si $B = \{x, \phi(x) \geq S\}$

pour une certaine fonction à valeurs réelles ϕ et un certain réel S, la probabilité

$$\mathbb{P}(X(t) \in B, \text{ for some } t \leq T)$$

peut se réécrire

$$\mathbb{P}(M(T) \geq S),$$

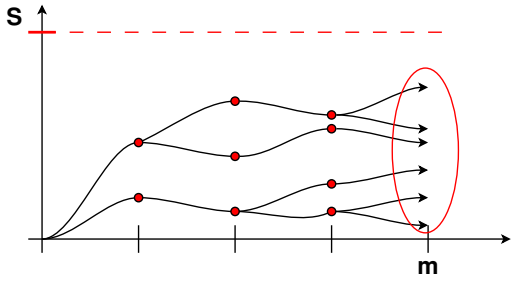
où le processus $\{M(t),\ t \geq 0\}$ est défini par $M(t) = \max_{0 \leq s \leq t} \phi(X(s))$. Nous pouvons alors utiliser l'algorithme des redistributions pondérées pour l'estimation de la probabilité définie en equation (1) en considérant la suite $\{M(t_k),\ k = 1,,\ldots,,m\}$ où $t_1 < \cdots < t_m$ sont des instants de discrétisation du processus $\{M(t),\ t \geq 0\}$. Nous avons alors proposé une méthode itérative automatique pour le choix d'un bon paramétrage. L'idée principale est que si la sélection est trop exigeante, toutes le trajectoires vont atteindre l'ensemble critique à l'instant final et la diversité sera appauvrie, donc la variance augmentée. D'un autre côté, si aucune trajectoire n'atteint l'ensemble critique à l'instant final, il faut augmenter la sélection. Ceci est illustré dans la figure 0.2. Il suffit alors de connaître deux paramètres dont on est sûr que pour l'un toutes les trajectoires passent le seuil final et l'autre dont on est sûr qu'aucune ne va l'atteindre. On procède ensuite à une simple dichotomie jusqu'à avoir un paramètre dont un certain pourcentage raisonnable de trajectoires passent le seuil.

Les résultats sur l'utilisation de l'algorithme des redistributions pondérées à la place de l'algorithme de branchement multi-niveaux sont très prometteurs. Pour des performances équivalentes, on réduit jusqu'à 40 fois les temps de calcul dans certains cas. Nous avons aussi testé cette procédure de paramétrage automatique de l'algorithme des redistributions pondérées dans les cas étudiés dans la littérature, à savoir en finance et en fibre optique [Carmona and Crepey, 2010, Carmona et al., 2009, Garnier and Del Moral, 2006]. Nous atteignons les paramètres optimaux en un très petit nombre d'appels à l'algorithme des redistributions pondérées (moins de 8 sont nécessaires pour tous les cas étudiés). Ces résultats ont été présentés dans [Jacquemart, 2014].

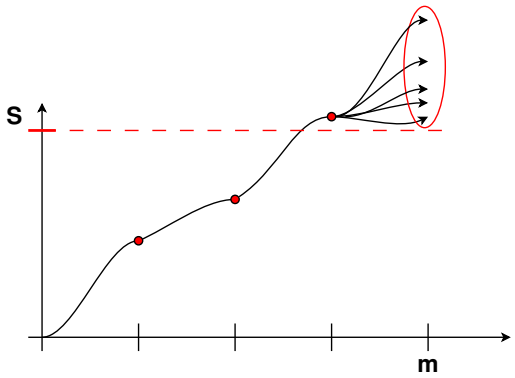
Enfin, nous avons appliqué les algorithmes proposés dans la thèse à l'estimation de la probabilité de conflit dans le trafic aérien. Nous distinguons deux types de conflits. Le premier correspond au fait que des avions sont trop près l'un de l'autre. Le second prend en compte les vitesses horizontales et verticales relatives entre les appareils. Comme tous les algorithmes proposés sont écrits en terme d'une fonction de score, nous avons caractérisé les zones de conflit en terme de dépassement de seuils par une fonction à valeurs réelles. Pour le premier type de conflit, cette caractérisation est évidente. Cela est plus complexe dans le second cas . Avec l'algorithme de branchement multi-niveaux adaptatif pour l'estimation de quantiles extrêmes, nous avons pu donner des estimations de zones de sécurité à un niveau de probabilité donné prenant en compte les vitesses relatives horizontales et verticales de tous les appareils pris en compte dans le scénario. Cette étude a fait l'objet d'une publication [Jacquemart and Morio, 2013].

Conclusions et perspectives

Nous nous sommes intéressés à la probabilité, supposée très petite, qu'un processus aléatoire Markovien $\{X(t),\ t \geq 0\}$ à valeurs dans un espace non discret atteigne une certaine région critique B avant un temps final T. Nous avons proposé des implémentations automatiques pour deux algorithmes de la littérature pouvant traiter ce sujet : l'algorithme de branchement multi-niveaux et l'algorithme des redistributions pondérées. Pour l'algorithme des redistributions pondérées, nous avons montré qu'une certaine modification du processus considéré



(a) Sélection trop faible: aucune trajectoire n'atteint l'ensemble critique.



(b) Sélection trop forte: toutes les trajectoires ont atteint l'ensemble critique mais la diversité est appauvrie.

Figure 0.2: Illustration de l'idée pour un bon paramétrage.

étendait le cadre de travail de cet algorithme à l'estimation de probabilité de temps d'atteinte. Aussi, nous avons amélioré à l'algorithme de branchement multi-niveaux en terme de réduction du biais et de la variance. Enfin, nous avons proposé une modification de l'algorithme de branchement multi-niveaux afin d'estimer des quantiles extrêmes de la loi du supremum de $\{\phi(X(t)),\ t \leq T\}$ pour une certaine fonction réelle ϕ. Enfin, notons que la plupart des algorithmes proposés dans cette thèse requiert que l'ensemble critique B soit de la forme $B = \{x,\ \phi(x) \geq S\}$. Tous les problèmes de la littérature sont exprimés en ces termes, hormis le problème d'estimation de la probabilité de conflit entre avion. Nous avons enfin montré que c'était encore le cas pour ce dernier.

Bien que nous ayons démontré des résultats de convergence presque sûre pour l'algorithme de branchement multi-niveaux adaptatif, un théorème central limite reste à établir. En ce qui concerne l'algorithme des redistributions pondérées adaptatif, l'implémentation sur certains cas d'estimation de probabilité de conflit n'est pas toujours satisfaisante. Il serait possible d'améliorer les performances en se basant sur les travaux de [Jourdain and Lelong, 2009]. Par ailleurs, l'extension des algorithmes proposés à l'étude de processus stochastiques hybrides [Krystul, 2006] est tout à fait immédiate. Enfin, il serait utile et intéressant de combiner les ré-

sultats de cette thèse avec des techniques de filtrage [Ichard and Baehr, 2013] ou encore avec des modèles d'ilots de particules [Vergé et al., 2013].

Part I

Context and overview of rare event simulation

Context of the thesis

General problem In recent decades, standards of quality and safety requirements is increasingly demanding in numerous industrial and scientific areas. Estimation of probability of extreme and rare events is therefore become of great interest. It is for example the case in financial engineering [Carmona and Crepey, 2010, Giesecke et al., 2010], climatology [Blanchet et al., 2009], electricity network reliability [Wadman et al., 2013], communication network [Garvels, 2000], optical fiber [Garnier and Del Moral, 2006] and air traffic management [Prandini et al., 2011]. To study these systems, some stochastic models have been worked out. It is thus possible to use some statistical and computational methods to estimate the very small probability that a critical event occurs.

In the context of rare event estimation, the standard Monte Carlo method is no longer efficient. Indeed, if the event under consideration is very rare, it is highly possible that even with a huge sample size, none of the sampled events will see the rare event probability set. From a more computational point of view, rewriting the quantity of interest as the expectation of a given Bernoulli random variable (*r.v.*) Y of mean $\mathbb{P}(\mathscr{R})$, the law of large numbers ensures that

$$\frac{1}{N}\sum_{i=1}^{N} Y_i \underset{N\to\infty}{\longrightarrow} \mathbb{P}(\mathscr{R}),$$

where Y_i are N independent samples of Y. Knowing the variance of a Bernoulli *r.v.* of mean $\mathbb{P}(\mathscr{R})$, we can thus determine the variance of $\widehat{p}_{CMC}$

$$Var(\widehat{p}_{CMC}) = \frac{1}{N}\mathbb{P}(\mathscr{R})(1-\mathbb{P}(\mathscr{R})).$$

If the event $\mathscr{R}$ is rare, the standard deviation of $\widehat{p}_{CMC}$ can be well approximated by:

$$\sigma(\widehat{p}_{CMC}) \simeq \sqrt{\frac{\mathbb{P}(\mathscr{R})}{N}}.$$

The relative standard deviation $\mu(\widehat{p}_{CMC})$ is then defined with

$$\begin{aligned}\mu(\widehat{p}_{CMC}) &= \frac{\sigma(\widehat{p}_{CMC})}{\widehat{p}_{CMC}} \\ &\simeq \frac{1}{\sqrt{N\widehat{p}_{CMC}}}.\end{aligned}$$

10^{11} CMC samples are required to estimate a probability of order 10^{-9} with a 10^{-1} relative standard deviation. This necessary number of samples is often intractable in practice.

The field of interest Our main field of interest will take place in the context or air traffic management. Here, to measure some extreme risk has the following major issues

- The number of aircraft is continuously increasing, and the risks of conflict and collision have to be reliably estimated for every flight plan scenario.
- The more the safety zone are precisely determined, the greater the number of aircraft can flight at the same time.
- The measure of the risk and of the safety requirement may facilitate the introduction of unmanned aircraft.
- The avoiding procedure has to be precisely certified.

As an insight, the International Air Transport Association reports the following results for 2010

- 2.4 billion people flew safely on 36.8 million flights
- 94 accidents (all aircraft types)
- 23 fatal accidents (all aircraft types)

The probability $p_{\text{Acc.}}$ of accident is then

$$p_{\text{Acc.}} = \frac{94}{2.4.10^9} \approx 3.9.10^{-8}.$$

This a small number compared to the huge number of aircraft flying at the same time. To illustrate this, figure 0.3 presents a shot screen of Flightradar24 website. This website gives in particular the real-time position of flying airplanes in the world.

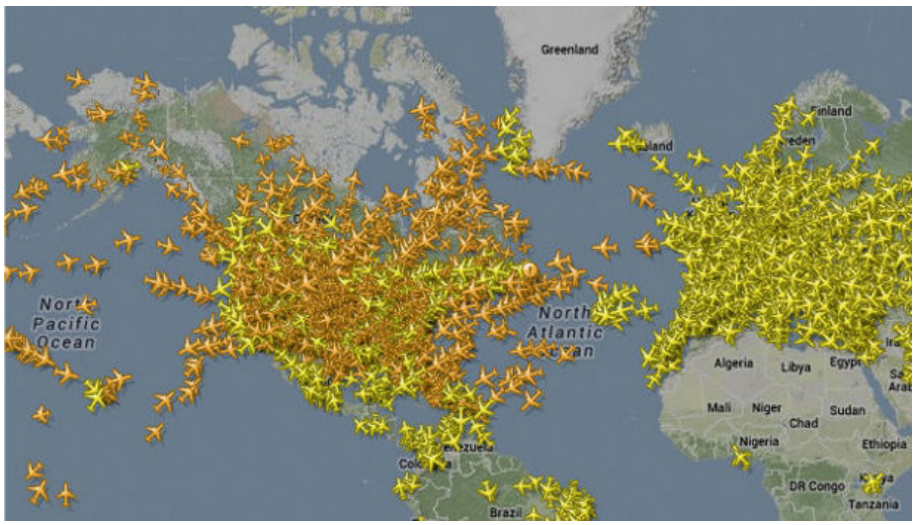


Figure 0.3: A Flightradar24 website screen shot.

CHAPTER 1

Bibliography study

In this thesis, we are interested in two different estimation problems. The first one aims at estimating the probability that a Markov process $\{X(t),\ t \geq 0\}$ enters a set B before a final time T

$$\mathbb{P}(X(t) \in B, \text{ for some } t \leq T). \tag{1.1}$$

In that case, the Markov process under consideration is assumed to have continuous state space.

The second one concerns the estimation of the probability that a Markov chain (or process) enters a rare event probability set at some final deterministic horizon time. Both estimation problems are linked as we will see in chapter 6 and section 7.3.

To estimate a rare event probability, we distinguish three different approaches: statistical, numerical and simulation techniques. The statistical approach may be used for both estimation problems above mentioned. The other methods are related to each problem and are thus studied in separated sections.

1.1 Statistical approach with the extreme value theory

Extreme value theory (EVT) focuses on the behaviour of the distribution tail of a real random variable, based on a reasonable number of observations [Embrechts et al., 2011, Kotz and Nadarajah, 2000]. Because of its general applicative conditions, this theory have been widely used to model extreme risks in meteorological phenomena [Blanchet et al., 2009], finance and insurance [Embrechts et al., 2011, Reiss and Thomas, 1997] and engineering [Castillo et al., 2005]. The way to proceed is of great interest when one has to work with only a fixed set of data.

Under some conditions, the founder theorem of the extreme value theory [Gnedenko, 1943, De Haan and Ferreira, 2006, Embrechts et al., 2011] claims that the maxima of an independent and identically distributed (i.i.d.) sequence converges to a generalized extreme value distribution (GEV). There are two equivalent ways of characterizing extremes with such an approach. A first one uses a Poisson point process [De Haan and Ferreira, 2006] which counts the number

of exceedances of the threshold. The other one, named Peak over Threshold (POT), works with the distribution of the observations above a certain threshold. This distribution is given by the generalized Pareto (GP) distribution whose parameters are estimated with the initial sample [Pickands, 1975, Embrechts et al., 2011]. Both approaches require to determine a threshold. Graphical and analytical threshold selection methods are presented in [Davison and Smith, 1990] and [Neves and Fraga A., 2004], but no recommendation have been clearly established to select a method and a careful check of several techniques seems to be needed to provide relevant results. Besides, the hypothesis using fundamental theorem of EVT cannot be verified in practice. Finally, studying the dynamic of the process in the non extreme regime (rare event probability set not reached) to deduce information on the rare event probability may lead to bad estimates, especially if the dynamic changes beyond the non extreme regime.

1.2 Reaching a rare set before final time

1.2.1 Analytical results

Theoretical results for the approximation of (1.1) can be found in the literature and several cases have been successfully studied. Amongst them we can found stationary process [Albin, 1998, Albin, 1990] with application of small set hitting probabilities for Gaussian and Rayleigh processes. Lévy processes, which frequently arise in financial engineering, are studied in [Albin and Sunden, 2008]. An other example of rare events in continuous time can be found in [Aldous, 1989], where the author presents a Poisson clumping heuristic.

Such analytical approaches appear to be useful but require many simplifying assumptions on the underlying models. Moreover, due to complex analytical expressions involved by such techniques, only simple cases are workable. Besides, the Poisson clumping heuristic does not rely on any theoretical justification.

1.2.2 Splitting algorithm

To increase the number of visits in the rare event probability set, the splitting relies on the fact that there exists some well identifiable intermediate regions before the rare event probability set. We work with a set of trajectories and the trajectories that have reached an intermediate region before the final time T are multiplied, the others are killed. With such a method, the trajectories are somehow forced to reach the rare set before the final time T.

The splitting algorithm has been first proposed in a physical context [Kahn and Harris, 1951], and a few variants have been then worked out such as in [Villén-Altamirano and Villén-Altamirano, 1994],[Glasserman et al., 1999],[Garvels, 2000] and [Cérou et al., 2006a]. Presentations of the summary of the methods can be found in [L'Écuyer et al., 2007, L'Écuyer et al., 2009] and some comparisons in [Garvels and Kroese, 1998]. The most recent advances and most rigorous proofs of convergence are given in [Cérou et al., 2006b] where the authors understood that the splitting actually fits the framework of the approximation of some Feynman Kac distributions. Central limit theorems and non-asymptotic L^p bounds for the estimation of the rare event probability with the splitting algorithm were thus worked out (see [Cérou et al., 2005] for a review). This work is based based on the previous works in the field of Feynman Kac distribution approximations [Del Moral and Miclo, 2001].

Selecting the intermediate regions The main issue in the implementation of the splitting algorithm is to select the intermediate region that are required. The choice of these regions is delicate because it strongly influences the variance in the estimation of the rare event probability. We found in the scientific literature three different approaches. One of these ideas is to use quantile considerations [Garvels, 2000, Cérou and Guyader, 2007]. A region is estimated in a way that a given percentage of the trajectories reaches a region. An other way [Wadman et al., 2013] is to consider an arbitrary set of region, and to retain the ones that minimize the variance using the results derived in [Amrein and Kunsch, 2011]. In that case, the regions are characterized by a real threshold. Finally, in the context of air traffic management the intermediate regions are somewhat already given by some regulation rules [Blom et al., 2007b, Prandini et al., 2011]. They correspond to some conflict zones that are more and more critical.

In [Garvels, 2000, Cérou and Guyader, 2007], the trajectories are systematically restarted a time $t = 0$ using same transition kernels. This implicitly implies that the underlying process is time-homogeneous. Moreover, the framework is limited to random final time T. Namely, the event of interest is that a trajectory enters a critical set before it enters next a recurrent set. In [Wadman et al., 2013], the method can only works if the sequence of optimal thresholds, with regards to a minimization of the variance, that characterizes the regions are not irregularly spaced. As far as the third idea [Blom et al., 2007b, Prandini et al., 2011] is concerned, the use of some given zones that do not depend on the considered flight plan scenario cannot enable to control the variance in the estimation.

The conditional probability values We denote p_k the probability of reaching the region $k+1$ starting from the region k. In a simplified setting, [Lagnoux, 2006] shows that $p_k \approx e^{-2}$ is optimal. If for any entrance point x in the region k the probability of reaching the region $k+1$ starting at x, denoted $p_k(x)$ does not depend on x, [Garvels et al., 2002] proves that the value $p_k \approx e^{-2}$ is also optimal, but this result fails if $p_k(x)$ depends on x. The authors of the splitting variant named RESTART [Villén-Altamirano and Villén-Altamirano, 1994] mention also that p_k should be equals to $p_k \approx e^{-2}$ but in a later paper [Villén-Altamirano and Villén-Altamirano, 2002] they recommend to use $p_k = 1/2$.

The p_k value is chosen to minimise the variance and the total computation time. No definitive results was carried out.

Optimal regions The optimal regions, in term of a minimization of the variance of the rare event estimation with the splitting algorithm, are described in [Kahn and Harris, 1951] without any justification, in [Garvels et al., 2002] with a heuristic and [Cérou et al., 2006b] gives a rigorous proof. A method in the case of a discrete state space for the process $\{X(t),\ t \geq 0\}$, with random final time T is given in [Garvels et al., 2002]. Another characterization of the optimal regions is given in [Dean and Dupuis, 2009], where the problem of finding optimal subsets is replaced by the one of solving a variational equation. Besides, it is shown in [Glasserman et al., 1998, Glasserman et al., 1999] that a bad choice of importance function increases the variance.

The first method is not tractable in the case of continuous state space and deterministic final time T. As far as the second method is concerned, the lack of clarity in the proposed algorithms makes them difficult to understand. Moreover, the lack of comparison of the variance obtained between the use of optimal and non optimal subsets makes impossible makes impossible the performance evaluation of the proposed approach.

The bias The splitting is theoretically unbiased. However, the adaptive algorithm that considers quantile estimation is slightly biased [Cérou and Guyader, 2007]. Moreover, if the discretization step during the implementation of the trajectory is too large, an additional bias can appear [Cérou and Guyader, 2007].

Despite of these remarks, no correction procedure has been given in the literature.

Truncation The aim of the splitting is to give fast simulation of some rare event. To decrease the total computation time, authors of [Kahn and Harris, 1951] and [Hammersley and Handscomb, 1964] propose to stop a trajectory if it falls lower than one or several threshold below the starting threshold. Many others truncation methods are proposed and compared in [L'Ecuyer et al., 2006].

1.2.3 Importance sampling

The idea of importance sampling is to sample the trajectory with an other distribution for which the trajectories of the process are more likely to reach the rare event probability set before time T. An introduction is presented [Sandmann, 2005] and numerous examples are given in [Glynn and Iglehart, 1989, Juneja and Shahabuddin, 2006]. The works related to importance sampling in a dynamic framework mainly focuses on finite or countable state space models. Examples in that case are found in [Heidelberger, 1995, Goyal et al., 1992] and some techniques are presented in [Alexopoulos and Shultes, 2001, Goyal et al., 1987, Nakayama, 1996, Ahamed et al., 2006]. In the special case of finite time T a review of existing methods and asymptotic analysis is presented in [De Boer et al., 2007]. Finally, the case of continuous state space is studied in [Johansen et al., 2006].

For a general Markov process with continuous state space, there is, to our knowledge, no general method to determine efficient importance sampling measure. This is due to the very broad variety of instances that can be involved. In particular, the method of [Johansen et al., 2006] is difficult to tune. The author uses only uses a specific tunes for each studied example. It is thus practically impossible to use this method in an other case. However, the choice of a efficient auxiliary distribution in the case of continuous state space requires the user to have a significant experience of the studied system, and may faces delicate calculation problem.

1.2.4 Is assuming the probability of interest to be define by threshold exceedance of real valued function restrictive ?

Characterizing the threshold exceedance probability with a real valued function (1.1) is of great interest. In particular, the characterization of intermediate regions in the splitting algorithm can be reduced to the choice of a parameter. The following examples use such a characterization for the probability (1.1).

- In [Kahn and Harris, 1951], a beam of particles evolves through a slab. The computation of the fraction of the original particles that are transmitted is addressed. Although the geometry of a slab can easily given by a score function, the authors also discuss optimal and non-optimal important functions that can be used.
- The telecommunication network study raises the issue of buffer overflow in queuing systems, and an obvious importance function is immediately given by the problem [Garvels and Kroese, 1998],[Glasserman et al., 1999]

- In the field of renewal energy [Wadman et al., 2013], the reliability of an electrical grid is modelled by threshold exceedance of a real function of the nodal voltage and the connection current.

- One of the safety requirements in air traffic management is to respect some given separation distance between aircraft, which obviously defines an important function [Jacquemart and Morio, 2013].

To our knowledge, only one estimation problem of the probability (1.1) involving a Markov process $\{X(t),\ t \geq 0\}$ with continuous time and continuous state space does not make use of an important function. It consists of estimating the conflict probability between aircraft, where the conflict zones depend on the relative position and the relative speed of aircraft (see [Prandini et al., 2011]). In chapter 8, a score function relative to this problem will be derived. Hence, we will be able to conclude that considering the problem to be defined as threshold exceedance of a real valued function is not restrictive.

1.2.5 Conclusion in the case of a continuous time Markov process

In this thesis, we are interested in some Markov process $\{X(t),\ t \geq 0\}$ with continuous time and continuous state space. The conclusions of this study lead to the following fact. For complex models, if the intermediate regions required by the splitting algorithm implementation are well chosen, it appears to be the best algorithm to be used for the estimation of the very small probability that the process $\{X(t),\ t \geq 0\}$ reaches a set before final time.

1.3 Reaching the rare set at final time

1.3.1 Important sampling for diffusion

To reduce the variance in Monte Carlo estimation in the case of a diffusion process, authors of [Newton, 1994, Fouque and Tullie, 2002] propose to use an important measure defined on the state space of trajectories. In [Carmona and Crepey, 2010] such a probability change of measure is used to compute some rare event probability of critical event that can occur a final time of a diffusion process and basis methodological of the use of Girsanov's theorem is reminded.

Nevertheless, a tuning is required and does not seem be completely handled in the literature. See [Carmona and Crepey, 2010] for some discussion.

1.3.2 Weighted redistribution algorithm

This algorithm is originally presented in [Del Moral and Garnier, 2005]. The idea of this algorithm is to work with a set of trajectories, and to select and multiply at some deterministic iteration time the paths that are more likely to reach the rare event probability set. This algorithm has the advantage on important sampling not to require an other sampling distribution. It is often the more comfortable alternative for the user, or even the exclusive one. Some applications are presented in [Garnier and Del Moral, 2006, Carmona and Crepey, 2010, Giesecke et al., 2010].

Tuning of the algorithm The weighted redistribution algorithm precision relies on a good choice of some selection functions. This choice is always replaced by the tuning of a real parameter, say α. The original paper [Del Moral and Garnier, 2005] proposes exponential selection functions. This idea is used in [Carmona and Crepey, 2010, Carmona et al., 2009, Garnier and Del Moral, 2006, Giesecke et al., 2010]. Authors of the paper [Carmona and Crepey, 2010] proposes other selection functions that still depends on a real parameter.

To choose a good α parameter, the authors of the original article [Del Moral and Garnier, 2005] consider different estimates of the relative standard deviation as a function of α and detect, for each rare event to estimate, which α is the minimizer of the variance. On the other hand, the paper [Carmona and Crepey, 2010] pointed out that a good α should maximize the occurrence of the rare event they consider, and select such an α with brute force.

Both methods can be very costly and have to be improved.

Weighted redistribution algorithm generalization A more general approach that underlies this algorithm can be found in [Le Gland, 2007]. It highlights the weighted importance resampling procedure in a sense that a part of the weights is used for re-sampling, and an other part is used for weighting trajectories.

Recently, [Chan and Lai, 2011] combines this algorithm to importance sampling to get some additional variance reduction. Asymptotic normality results are provided and computations of optimal weights for some special cases are deduced. An example is provided in [Deng et al., 2012].

Despite the theoretical point of view, the practical side of these algorithms has not been deeply tackled. A sensitive tuning is still necessary. There is to our knowledge no formula neither rules.

1.4 Conflict probability estimation in air traffic management

Maintaining some specific separation zones between aircraft to avoid conflict and collision is mandatory in air traffic management (ATM). These zones have to be tune in a way that the risk probability is very small.

Conflict probability estimation The estimation of the (very small) probability of conflict or collision was studied in several papers. For some simple cases, authors of [Paielli and Erzberger, 1999, Paielli, 1998] estimate the probability that a pair of aircraft being closer than a certain distance with analytical or geometrical methods.

Splitting-like algorithms such as HIPS and HHIPS [Krystul, 2006, Krystul and Blom, 2006, Blom et al., 2007a] enable to deal with stochastic hybrid system that takes into account regime changes in the dynamic of the aircraft. An efficient splitting-like algorithm with importance sampling for rare switching regime diffusion can also deal with models using hybrid stochastic processes [Krystul and Blom, 2005]. This method was applied to estimate conflict probability in [Blom et al., 2006]. Then, the algorithm proposed in [Blom et al., 2007a] is applied to a free flight modelling described by advanced stochastic Petri nets [Blom et al., 2007b]. Recently, [Prandini et al., 2011] proposes to discard, in the splitting algorithm, trajectories that are below

some threshold given by air traffic complexity measures, which enables to reduce the simulation time.

For complex realistic models, all the proposed algorithms are based on a decomposition of the state space of the underlying process. This decomposition consists of a sequence of nested supersets of the conflict or collision zone. The probability of conflict or collision is then decomposed into the product of conditional probabilities of reaching a set starting from the previous one. That is to say, all the proposed algorithms are based on the splitting algorithm.

However, the choice of the considered sequence of nested supersets suffers from the following points. In [Krystul and Blom, 2005], this choice is empirical and there is no proposed automatic procedure. As far as the idea of [Prandini et al., 2011, Blom et al., 2007b] is concerned, the use of some given zones that do not depend on the considered flight plan scenario cannot enable to control the variance in the estimation. To this purposes, authors of [Rubino and Tuffin, 2009] writes

> "The [nested supersets] were identified through an iterative process of learning from conducting MC simulations. This quite easily led to the identification of a [sequence of nested supersets] that appeared to work well. Although it is likely that further optimization of the [sequence of nested supersets] may lead to a reduction in the variance and confidence interval of the estimates [...], we have not yet tried to do so."

About the same idea of adding intermediate conflict zones, the authors of [Blom et al., 2007b, Prandini et al., 2011] wrote

> "These values have been determined through two steps. The first was to let an operational expert make a best guess of proper parameter values. Next, during initial simulations with the [splitting algorithm] some fine tuning of the number of levels and of parameter values per level has been done."

Moreover, we cannot make sure that the variance in the estimation is controlled for any flight plan scenario.

As a conclusion, even though the splitting algorithm is comfortable to use, the problem of selecting intermediate regions is not handled completely in the ATM context. Avoiding such an "on the thumb" tuning and give automatic procedure that control the variance in the estimation is one of the main goals of this thesis.

Estimation of conflict zone that ensure a risk lower to a given value On the other hand, the zones given by ATM regulation does not depend on the considered flight plan. For some given risk (namely for some given small probability), there is no proposed method in the scientific literature for the construction of a related safety zone.

1.5 Splitting, IPS, RESTART, genealogical models or random sampling. Terminology choices throughout the manuscript.

This thesis essentially focuses on two algorithms for which terminology may vary upon authors and upon fields of application. We explain here the two named that will be used in the follow-

ing.

In the scientific literature that we study in this manuscript, several different names have been given for the same algorithm. Moreover, the same name has also been given to different algorithms. Thus, algorithms of [L'Ecuyer et al., 2006] and [Cérou et al., 2006b] are called splitting or RESTART in the field of telecommunication network [Garvels and Kroese, 1998],[Glasserman et al., 1999], random sampling in nuclear engineering [Kahn and Harris, 1951] and splitting in renewal energy [Wadman et al., 2013]. Moreover, the same algorithms are called IPS, an acronym for interacting particle system, in the context of Air Traffic Management [Prandini et al., 2011], and the variant names are based upon the IPS terminology, such as HIPS and HHIPS [Krystul, 2006, Krystul and Blom, 2006, Blom et al., 2007a].

Things become even more confusing with the IPS terminology. In the domain of financial engineering [Carmona and Crepey, 2010],[Fouque and Tullie, 2002] and optical fiber reliability [Garnier and Del Moral, 2006], the algorithm of [Del Moral and Garnier, 2005], which is very different of the one of [Cérou et al., 2006b], is also named IPS.

The confusions might be done for two reasons. First, authors of [Cérou et al., 2006b] and [Del Moral and Garnier, 2005] did not clearly named the proposed algorithms. The first paper [Cérou et al., 2006b] title is *Genetic Genealogical Models in Rare Event Analysis* and the second paper [Del Moral and Garnier, 2005] title is *Genetic Genealogical Particle Analysis of Rare Events*. Furthermore, all the studied algorithms aim at estimating some Feynman-Kac distribution measures with a set of "particles". They thus all match the framework of interacting particle system algorithm type given for example in [Del Moral, 2004].

Terminology choices

- The algorithm that are based upon a sequence of nested supersets of the rare event probability set [Cérou et al., 2006b, L'Ecuyer et al., 2006, Garvels and Kroese, 1998, Amrein and Kunsch, 2011, Cérou et al., 2012, Cérou and Guyader, 2007] will be called splitting.
- The algorithm of [Del Moral and Garnier, 2005] and used in [Carmona and Crepey, 2010, Carmona et al., 2009, Garnier and Del Moral, 2006, Giesecke et al., 2010]. will be called the weighted redistribution algorithm.

CHAPTER 2

Outline of the thesis

We will focus on the case of stochastic processes with continuous state-space and continuous time. With such a framework, we are interested in the probability that the process enters a rare set before some final stopping time. As already mentioned in chapter 1, there are several methods proposed in the scientific literature. Each of these methods requires a potentially hard tuning that may decrease their usefulness. For our case of interest the splitting algorithm is most of the time used, as see in section 1.2.5.

In this thesis we propose automatic tunings for some existing algorithms and new algorithms to simulate and estimate the probability of critical events that can occur during the evolution of a random process.

First of all, the study will focus on the elaboration of an automatic tuning for the splitting algorithm. The principle of the splitting algorithm relies on a decomposition of the state space of the process into a sequence of nested supersets of the rare event probability set. For that purpose, we give a characterization of the rare event of interest with an exceedance over a certain threshold of a real valued function. Thus, the nested supersets are simply characterized by a real quantile of a certain random variable. This idea is not new [Garvels, 2000],[Cérou and Guyader, 2007] [Cérou et al., 2012] but have to be improved and generalized to all study cases. Our goal will be to derive a method that can be used for all stochastic processes with random or deterministic final time. We will propose a two step algorithm. The first step estimates a nested superset and the second one gives estimates of the conditional probability as the original splitting algorithm [L'Ecuyer and Tuffin, 2007] does. We will extend this approach to compute some extreme quantile and the whole cumulative distribution of the supremum of a random process over final time. This first part of the study led to a communication [Jacquemart and Le Gland, 2014] and will be detailed in the chapter 4.

Then, the objective will be to improve the performances of the splitting algorithm in terms of bias and variance. First, we will assume that the sequence of nested supersets required for the implementation of the splitting algorithm is given, for example given in chapter 4. The goal will be to highlight a bias that can appear with the discretization of the trajectories of the process. More precisely, the discrete point process can miss a nested superset whereas the real

non-discrete process actually reaches it. This leads to underestimation of the probability. The next step will consist in giving some correction procedure for this bias. Next, to decrease the variance in estimation of the rare event probability with the splitting algorithm, it is shown that some optimal sequence of nested superset should be used [Cérou et al., 2006b, prop 3.1]. The aim will be to derive some methods to estimate and use these optimal sequence. This will be the subject of the chapter 5.

The sequence of optimal nested supersets studied in chapter 5 can be very hard to estimate. The goal will be next to consider a non optimal sequence of nested superset, for example given in chapter 4, and to improve the variance in the estimation. To this purpose, we will give more importance to some well chosen trajectories. Typically, those which manage to reach an intermediate superset the earlier are favoured. The theoretical study of this new method will rely on the elaboration of an interacting particle system algorithm for rare event with non-negative potential functions. This was partially published in [Jacquemart et al., 2013] and a more detailed version will be presented in the chapter 6.

From then on, we will have described a fully implementable adaptive splitting algorithm, and will have given procedures to reduce the bias and the variance. The next objective will be to give an other algorithm that can be used instead of the splitting algorithm. We will work on the algorithm presented in [Del Moral and Garnier, 2005]. We will faces two problem. First, there is no proposed tuning in the scientific literature for the choice of good parameter. Second, the goal of [Del Moral and Garnier, 2005] is to estimate the probability that a Markov chain enters a set at final deterministic iteration time. These two problems will be addressed and the final objective will be to use the algorithm of [Del Moral and Garnier, 2005] instead of the splitting algorithm. This work has been presented in [Jacquemart, 2014] and will be detailed in the chapter 7.

Finally, we will implement our algorithm for the estimation of conflict probability between aircraft during a fixed flight duration. Two cases will be studied. The first one will consider the conflict as a minimum separation between aircraft. The second case will be the one of conflict zone that take into account the relative speed between aircraft. For both cases, there will be two objectives. The first will be to estimate the rare event probability of interest with algorithm of chapter 4 and 7. The second will be to estimate some conflict zones for a given fixed probability with algorithms of chapter 4. This was partially published in [Jacquemart and Morio, 2013] and will be detailed in the chapter 8.

CHAPTER 3

Materials and methods

3.1 Basic background

Relative standard deviation For any real H-sample $(x_1,\dots,x_H)$, the accuracy of an estimator is measured by its relative standard deviation (rSTD), defined as follow

$$\mathrm{rSTD}(x_1,\dots,x_H)=\frac{\sqrt{\frac{1}{H}\sum_{j=1}^{H}(x_j-\overline{x})^2}}{\overline{x}},\quad \text{where}\quad \overline{x}=\frac{1}{H}\sum_{l=1}^{H}x_l.$$

It corresponds to the average spreading percentage between the mean and the estimates.

Time Relative Variance product The main challenge in rare event simulation is to give sharp estimates of some extreme quantities with the smallest run time. That is why the relative standard deviation is not always the best way to discuss the validity of a method since it does not take into account the run time. The best method would be the one that minimize the rSTD and the run time. To this end, we define the time relative variance product [Garvels et al., 2002, L'Ecuyer et al., 2006], for any H-sample $(x_1,\dots,x_H)$ of an estimation of a quantity by

$$rTV=\mathrm{rSTD}(x_1,\dots,x_H)\times\overline{T},$$

where $\overline{T}$ is the average run time of the method under consideration.

Markov kernels and related operations

Definition 3.1. *Let $(X,\mathscr{A})$ and $(Y,\mathscr{B})$ be measurable spaces. A Markov kernel is a map $K: X\times\mathscr{B}$ such as*

- *The map $x\in X\mapsto K(x,B)$ is measurable for every $B\in\mathscr{B}$.*
- *The map $B\in\mathscr{B}\mapsto K(x,B)$ is a probability measure on $(Y,\mathscr{B})$ for every $x\in X$.*

Let K be a Markov kernel K as above. Given a measure μ and a measurable function f defined on suitable spaces, we set the following notations

- $\mu(f)$ or $\langle \mu, f \rangle$ is the real number defined with

$$\mu(f) = \langle \mu, f \rangle = \int f(x) \mathrm{d}\mu(x).$$

- μK is a measure defined with

$$\mu K(\mathrm{d}y) = \int K(x, \mathrm{d}y) \mathrm{d}\mu(x).$$

- Kf is the function defined with

$$Kf(x) = \mathbb{E}[f(Y)|X = x],$$

where the law of the random variable Y is given with $Y \sim K(x, \cdot)$.

In particular, we have that

$$(\mu K)(f) = \int \int f(x) K(x, \mathrm{d}y) \mathrm{d}\mu(x)$$

and we can write that

$$(\mu K)(f) = \mu(Kf) = \mu K f,$$

or, equivalently

$$\langle \mu K, f \rangle = \langle \mu, Kf \rangle.$$

Remark 3.1. *The notations $\mu(f)$ and $\langle \mu, f \rangle$ are strictly equivalent. The notation $\langle \mu, f \rangle$ has the advantage of separating the measure and the function. It prevents ambiguities. However, the two notations will be used in this manuscript, depending on the chapter under consideration.*

3.2 Splitting algorithm

3.2.1 Description of the algorithm

Splitting methods are of great interest when one has to work with stochastic processes that evolve in a continuous time. The framework is the following. Given a probability space $(\Omega, \mathscr{F}, \mathbb{P})$ and a measurable metric space E, let

$$X : (\omega, t) \in \Omega \times [0, +\infty) \to X(t)(\omega) \in E$$

be a Markov process with continuous, or at least right continuous with left limited trajectories, $\mathscr{B}$ a measurable subset of the state space E and T_B the first hit time of B, given by

$$T_B = \inf\{t \geq 0, X(t) \in B\}.$$

Given T a deterministic time, or a random stopping time with finite expectation, the event

$$\{T_B \leq T\} = \{X(T) \in B, \text{ for some } t \leq T\}$$

is supposed to be very rare, typically its probability is smaller than 10^{-4}. In this context, the splitting methods give efficient numerical approximations to

- the rare event probability $\mathbb{P}(T_B \leq T)$
- the law of the process in the rare regime $\mathbb{E}(f(X(t), 0 \leq t \leq T_B) | T \leq T_B)$.

The second point is equivalent to the knowledge of an approximation of the distribution of the critical trajectories $(X(t), 0 \le t \le T_B)$.

Denoting by A the subset of $E \times [0, +\infty)$

$$A = B \times [0, +\infty),$$

the principle of the splitting is to consider some region defined as a sequence of decreasing supersets of A

$$A_0 = E \times [0 + \infty) \supset A_1 \supset \cdots \supset A_m = B \times [0, +\infty). \tag{3.1}$$

and to estimate each probability that the process $\{(X(t), t),\ t \ge 0\}$ reaches A_k starting from A_{k-1}. Thus, we are led to define the process $\{Z(t),\ t \ge 0\}$ with

$$Z(t) = (X(t), t).$$

Let us define

$$T_k = \inf\{t \ge 0, Z(t) \in A_k\}, \quad \text{for } k = 0, \ldots, m.$$

It is worth noting that the existence of left-hand limits ensures that $Z(t) \in \overline{A}_k$ where $\overline{A}_k$ is the closure of A_k, hence $Z(T_k) \in A_k$ provided A_k is a closed subset. From now on, it is assumed that the regions introduced in 3.1 are all closed subsets.

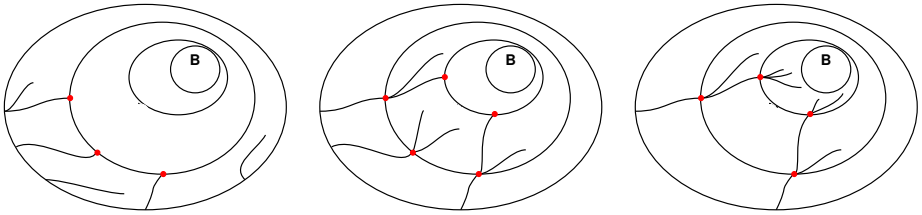
A Bayes formula gives the following decomposition product:

$$\mathbb{P}(T_B \le T) = \prod_{k=1}^{m} \mathbb{P}(T_k \le T \mid T_{k-1} \le T),$$

and we denote also the conditional probability value $\mathbb{P}(T_k \le T \mid T_{k-1} \le T)$ with

$$p_1 = \mathbb{P}(T_1 \le T) \quad \text{and} \quad p_k = \mathbb{P}(T_k \le T \mid T_{k-1} \le T).$$

Splitting principle consists in estimating separately each probability $\mathbb{P}(T_k \le T \mid T_{k-1} \le T)$ with accuracy for a small sample size. The key point is to choose them large enough to be estimated with the standard Monte Carlo method. In a first stage, N samples of Markov process $\{Z(t),\ t \ge 0\}$ are generated until time $T_1 \wedge T = \min(T_1, T)$. If I_1 is the set of entrance point of the process $\{Z(t),\ t \ge 0\}$ into the set A_1, then $\mathbb{P}(T_1 \le T)$ is estimated by $|I_1|/N$. For stage $k \ge 1$, N starting points are randomly and uniformly chosen amongst the $|I_k|$ crossing points between the region A_k and the previously sampled trajectories. N paths of the process $\{Z(t),\ t \ge 0\}$ are sampled from these crossing points according to the Markov dynamic of the process until time $T_{k+1} \wedge T$. We denote I_{k+1} the set of entrance point into A_{k+1} and $\mathbb{P}(T_{k+1} \le T \mid T_k \le T)$ is estimated by $|I_{k+1}|/N$. See figures 3.1 for an illustration.



Figures 3.1: The first steps of the splitting algorithm.

3.2.2 Feynman-Kac interpretation of the algorithm

3.2.2.1 Description of the objects used in the splitting in terms of Feynman-Kac distributions

We will now give a Feynman-Kac interpretation of the algorithm, slightly different of the one given in [Cérou et al., 2006b]. The present representation only takes into account the entrance point into the regions $A_k, k = 1,\dots,m$. To model the algorithm, we consider the $E \times [0,+\infty)$ valued random variable

$$Z_k = \left(X_{T_k \wedge T}, T_k \wedge T\right).$$

The sequence $\{Z_k,\ k = 0,\dots,m\}$ is a Markov chain with initial distribution $\eta_0 \times \delta_0$, where η_0 is the distribution of X_0 and δ_0 is the Dirac function in 0. Its transition kernels are denoted by

$$Q_k(x,t,\mathrm{d}x',\mathrm{d}t') = \mathbb{P}(X_{T_k \wedge T} \in \mathrm{d}x', T_k \wedge T \in \mathrm{d}t' | X_{T_{k-1} \wedge T} = x, T_{k-1} \wedge T = t).$$

Considering next the selection function g_k

$$g_k(x,t) = \mathbb{1}_{(x,t) \in A_k},$$

it yields

$$g_k(Z_k) = \mathbb{1}_{T_k \leq T}.$$

After that, we define the following Feynman-Kac un-normalised distribution

$$\langle \gamma_k, f \rangle = \mathbb{E}(f(Z_k) \prod_{p=0}^{k} g_k(Z_k)) = \mathbb{E}(f(X(T_k), T_k) \mathbb{1}_{T_k \leq T}). \tag{3.2}$$

and the associated normalised distribution

$$\langle \mu_k, f \rangle = \frac{\langle \gamma_k, f \rangle}{\langle \gamma_k, 1 \rangle}. \tag{3.3}$$

The latter one is understood as the entrance distribution into the set A_k for the process $\{Z(t),\ t \geq 0\}$ given the event $\{T_k \leq T\}$

$$\mu_k(\mathrm{d}x,\mathrm{d}t) = \mathbb{P}(X(T_k) \in \mathrm{d}x, T_k \in \mathrm{d}t | T_k \leq T),$$

and the rare event probability is rewritten with

$$\langle \gamma_m, 1 \rangle = \mathbb{E}(\mathbb{1}_{T_B \leq T}) = \mathbb{P}(T_B \leq T).$$

To connect the conditional probabilities p_k with the distribution γ_k and μ_k we introduce the distribution η_k defined as follows

$$\langle \eta_k, f \rangle = \mathbb{E}[f(Z_k) | T_{k-1} \leq T],$$

for which the following the interpretation holds

$$\eta_k(\mathrm{d}t,\mathrm{d}x) = \mathbb{P}(X_{T_k \wedge T} \in \mathrm{d}x, T_k \wedge T \in \mathrm{d}x | T_{k-1} \leq T). \tag{3.4}$$

The action of η_k on the potential function g_k gives that

$$\langle \eta_k, g_k \rangle = \mathbb{E}(\mathbb{1}_{T_k \leq T} | T_{k-1} \leq T) = p_k,$$

and

$$\mathbb{P}(T_B \leq T) = \langle \gamma_m, 1 \rangle = \prod_{k=1}^{m} \langle \eta_k, g_k \rangle.$$

Dividing from the both sides of this equality we get

$$\begin{aligned}\langle \mu_k, f\rangle &= \frac{\langle \gamma_k, f\rangle}{\langle \gamma_k, 1\rangle} \\ &= \frac{\langle \eta_k, g_k f\rangle}{\langle \eta_k, g_k\rangle},\end{aligned}$$

that is to say

$$\mu_k = \frac{g_k \eta_k}{\langle \eta_k, g_k\rangle}. \tag{3.5}$$

Equation (3.5) together with the following proposition enable to give a recursive relation for sequence of normalised distributions $\mu_k,\ k = 0,\dots,m$.

Proposition 3.1. *We have the following relation*

$$\eta_k = \mu_{k-1} Q_k.$$

Proof. By definition we have

$$\begin{aligned}\langle \eta_k, f\rangle &= \mathbb{E}(f(Z_k) \mid T_{k-1} \le T) \\ &= \frac{\mathbb{E}(f(Z_k)\mathbb{1}_{T_{k-1}\le T})}{\mathbb{P}(T_{k-1} \le T)}\end{aligned}.$$

Besides, using the elementary properties of the conditional expectation we have that

$$\begin{aligned}\mathbb{E}(f(Z_k) g_{k-1}(Z_{k-1})) &= \mathbb{E}(\mathbb{E}(f(Z_k) \mid Z_{k-1}) g_{k-1}(Z_{k-1})) \\ &= \mathbb{E}(Q_k f(Z_{k-1}) g_{k-1}(Z_{k-1}).\end{aligned}$$

Hence,

$$\begin{aligned}\langle \eta_k, f\rangle &= \frac{\mathbb{E}(Q_k f(Z_{k-1}) g_{k-1}(Z_{k-1}))}{\mathbb{P}(T_{k-1} \le T)} \\ &= \frac{\mathbb{E}(Q_k f(Z_{k-1})\mathbb{1}_{T_{k-1}\le T})}{\mathbb{P}(T_{k-1} \le T)} \\ &= \mathbb{E}(Q_k f(Z_{k-1}) \mid T_{k-1} \le T) \\ &= \langle \mu_{k-1}, Q_k f\rangle \\ &= \langle \mu_{k-1} Q_k, f\rangle.\end{aligned}$$

□

Now, recall the mechanism of the splitting algorithm relies on the generation of trajectories with initial distribution μ_k. It turns out that proposition 3.1 and the recursive equation lead to the following scheme

$$\mu_{k-1} \xrightarrow{\text{mutation}} \eta_k = \mu_{k-1} Q_k \xrightarrow{\text{updating}} \mu_k.$$

3.2.2.2 Associated interacting particle system

We are interested in some approximation with the form

$$\eta_k \approx \eta_k^N = \frac{1}{N}\sum_{k=1}^{N} \delta_{\xi_k^i} \quad \text{and} \quad \mu_k \approx \mu_k^N = \sum_{k=1}^{N} w_k^i \delta_{\xi_k^i} \quad \text{with} \quad \sum_{i=1}^{N} w_k^i = 1,$$

where the *particle* ξ_k^i equal to

$$\xi_k^i = (Z_k^i(T_k^i \wedge T)) = (X_k^i(T_k^i \wedge T), T_k^i \wedge T).$$

Starting with

$$\eta_k \approx \eta_k^N = \frac{1}{N}\sum_{k=1}^{N} \delta_{\xi_k^i},$$

we obtain with equation (3.5) that

$$\mu_k \approx \mu_k^N = \frac{g_k \eta_k^N}{\langle \eta_k^N, g_k \rangle} = \frac{\sum_{i=1}^N g_k(\xi_k^i)\delta_{\xi_k^i}}{\sum_{j=1}^N g_k(\xi_k^j)}$$

and

$$\begin{aligned} p_k \approx \langle \eta_k^N, g_k \rangle &= \frac{1}{N}\sum_{i=1}^{N} g_k(\xi_k^i) \\ &= \frac{|I_k|}{N}, \end{aligned}$$

where

$$\begin{aligned} I_k &= \{i = 1, \dots, N,\ T_k^i \leq T\} \\ &= \{i = 1, \dots, N,\ g_k(\xi_k^i) = 1\}. \end{aligned}$$

Finally, the approximation of the rare event probability stands as follows

$$\mathbb{P}(T_B \leq T) = \prod_{k=1}^{N} p_k \approx \prod_{k=1}^{N} \frac{|I_k|}{N}.$$

3.2.3 The algorithm

The above described procedure is summarized into algorithm 3.1. Notice that if the final horizon time T is random it may depend on the trajectory $t \mapsto Z^i(t)$ under consideration. However, it is still denoted T not to overload the writing.

3.2.4 Central limit theorem

We first define the life time of the set of particles $\{\xi_k^i,\ i = 1, \dots, N\}$ with

$$\tau^N = \inf\{k \geq 0,\ |I_k| = 0\},$$

for which the exponential lower bound holds [Del Moral, 2004, theo. 7.4.1]

$$\mathbb{P}(\tau^N \leq m) \leq a_m \exp(-b_m N).$$

That is to say, the probability that the algorithm stops, or equivalently the probability that the estimates of the rare event probability equals to zero, decreases exponentially with the number of particles N. The approximation of the rare event probability defined on the *good* set $\{\tau^N > m\}$ converges in law, as detailed in the following theorem.

Theorem 3.1. *The central limit theorem for the estimation of the rare event probability with the splitting algorithm, defined on $\{\tau^N > m\}$, stands as follows*

$$\sqrt{N}\left(\frac{\widehat{P}_B - P_B}{P_B}\right) \mathbb{1}_{\tau^N > m} \underset{N \to +\infty}{\overset{Law}{\Longrightarrow}} \mathcal{N}(0, \sigma_m^2),$$

where the asymptotic relative variance equals to

$$\sigma_m^2 = \sum_{k=1}^{m}\left(\frac{1}{p_k} - 1\right) + \sum_{k=1}^{m} \frac{1}{p_k} \frac{\mathrm{var}(u_B, \mu_k)}{\langle \mu_k, u_B \rangle^2},$$

Algorithm 3.1: The splitting algorithm

Data: Rare event probability set B, number N of trajectories to estimate conditional probability, sequence of nested subsets $\{A_k,\ k = 1,\dots,m\}$, final time T

Result: $\widehat{P}_B \approx \mathbb{P}(T_B \le T)$

Initialization

  for $i = 1,\dots,N$ **do**

    Set $T_0^i = 0$ and sample Z_0^i independently from law μ_0

    Set $e_0^i = Z_0^i$

  Set $I_0 = \{1,2,\dots,N\}$

for $k = 0,\dots,m-1$ **do**

  */*Conditional entrance distribution*/*

  for $i = 1,\dots,N$ **do**

    Choose randomly and uniformly a subscript $j_i \in I_k$

    Sample a path $Z^i(t)$ starting from $e_k^{j_i}$ at time $T_k^{j_i}$ and until time $T_{k+1}^i \wedge T$

    where $T_{k+1}^i = \inf\{T_k^{j_i} \le t \le T,\ Z^i(t) \in A_{k+1}\}$

    Set $e_{k+1}^i = Z^i(T_{k+1}^i \wedge T)$

  */*Conditional probability value estimation*/*

  Set $I_{k+1} = \{i,\ (T_{k+1}^i \wedge T) < T\}$ and

$$\widehat{p}_{k+1} = \frac{|I_{k+1}|}{N}$$

Estimation

$$\mathbb{P}(T_B \le T) \approx \widehat{P}_B = \prod_{k=1}^{m} \widehat{p}_k$$

where

$$u_B(x,t) = \mathbb{P}(T_B \le T | X(t) = x),$$

and

$$\mathrm{var}(u_B,\mu_k) = \int_0^\infty \int_E u_B(x,t)^2 \mu_k(\mathrm{d}x,\mathrm{d}t) - |\int_0^\infty \int_E u_B(x,t)\mu_k(\mathrm{d}x,\mathrm{d}t)|^2.$$

It is worth remarking that if the boundary of the intermediate regions A_k are define as the contour lines of the function

$$(x,t) \mapsto \mathbb{P}(T_B \le T | X(t) = x) = c_k, \tag{3.6}$$

for an increasing sequence of real number $c_k \in (0,1)$, we readily check that the asymptotic variance equals to

$$\sigma_m^2 = \sum_{k=1}^{m} \left(\frac{1}{p_k} - 1\right).$$

That is why these subsets are often referred as the *optimal regions* with regard to the minimization of the asymptotic variance.

Moreover, optimizing $\sigma_m^2 = \sum_{k=1}^m \left(1/p_k - 1\right)$ under the constraint $p_1 \cdots p_m = P_B$ leads to choose

$$p_1 = \cdots p_m = P_B^{1/m},$$

for which

$$\frac{1}{P_B} - 1 \geq \sigma_m^2 = m\left(\frac{1}{P_B^{1/m}} - 1\right) \geq -\log P_B.$$

The asymptotic variance σ_m^2 decreases towards the limit $-\log P_B$ as the number of subsets m increases.

Remark 3.2. *The choice of optimal subsets such as those in section (3.2.4) is unrealistic in practice since it depends on quantities we want to estimate. However, it can give some ideas for the choice of an importance function. Notice that an optimal importance function should depend on time, to be consistent with the optimal subsets definition.*

3.3 Weighted redistribution algorithm

The weighted redistribution algorithm enables to sample trajectories of a Markov chain conditionally to the rare event under consideration is reached. It is completely different from the splitting algorithm. See section 1.5 for further terminology explanations.

We suppose here that the studied system is modelled by a Markov chain $\{X_k,\, k = 0,\ldots,m\}$. For a fixed positive integer m, we are interested in the estimation of the probability of rare events such as

$$\mathbb{P}(V(X_m) \in A), \tag{3.7}$$

for V a real valued function and A a subset of $\mathbb{R}$.

The background of the weighted redistribution algorithm essentially relies on some Feynman-Kac interpretation of the rare event. To begin, we denote, $K_k(x_{k-1}, \mathrm{d}x_k)$ the transition kernels of the Markov chain $\{X_k,\, k = 0,\ldots,m\}$ and we assume that X_k belongs to some measurable space $(E_k, \mathscr{E}_k)$. Then, let $\{Y_k,\, k = 0,\ldots,m\}$ denotes the historical process of the Markov chain $\{X_k,\, k = 0,\ldots,m\}$:

$$Y_k = X_{0:k} = (X_0,\ldots,X_k).$$

The sequence $\{Y_k,\, k = 0,\ldots,m\}$ is a Markov chain, where for all $k = 0,\ldots,n$ $Y_k \in F_k = E_0 \times E_1 \times \cdots \times E_k$. Probability transitions of $\{Y_k,\, k = 0,\ldots,m\}$ are given by

$$\begin{aligned}\mathscr{K}_k(y_{k-1}, \mathrm{d}y'_k) &= \mathscr{K}_k(x_0,\ldots,x_{k-1}, \mathrm{d}x'_0,\ldots,\mathrm{d}x'_k)\\ &= \delta_{(x_0,\ldots,x_{k-1})}(\mathrm{d}x'_0,\ldots,\mathrm{d}x'_{k-1})K_k(x_{k-1}, \mathrm{d}x'_k).\end{aligned}$$

The key idea lies in the following decomposition

$$\mathbb{E}(f(X_m)) = \mathbb{E}\left[f(X_m)\prod_{k=0}^m G_k(Y_k)\prod_{k=0}^m G_k^-(Y_k)\right],$$

for some given positive function $G_k : F_k \to (0,+\infty)$, and where $G_k G_k^- = 1$.

Feynman-Kac interpretation of rare events

This paragraph recalls the main methods for the use of Feynman-Kac twisted expectation in rare event analysis as notably described in [Del Moral and Garnier, 2005].

Given some potential functions G_k, the following un-normalized and normalised Feynman-Kac distributions on the path space are defined by

$$\gamma_k(f_k) = \mathbb{E}(f_k(Y_k)\prod_{p=1}^{k-1} G_p(Y_p)) \quad \text{and} \quad \eta_k(f_k) = \frac{\gamma_k(f_k)}{\gamma_k(1)}.$$

To connect the probability $\mathbb{P}(V(X_m) \in A)$ with the above formulae, we use the $T_m^{(A)}(\phi_m)$ function defined on the path space by

$$T_m^{(A)}(\phi_m)(y_m) = \phi_m(y_m)\mathbb{1}_{V(x_m)\in A}\prod_{p=1}^{m-1} G_p^-(y_p).$$

It is now possible to give Feynman-Kac interpretations of the rare event probability $\{V(X_m) \in A\}$

$$\mathbb{P}(V(X_m) \in A) = \gamma_m(T_m^{(A)}(1)). \tag{3.8}$$

In addition, the law of the paths $(X_0,\dots,X_m)$ given the rare event $\{V(X_m) \in A\}$ is characterised as follows

$$\mathbb{E}(\phi_m(X_0,\dots,X_m) \mid V(X_m) \in A) = \frac{\gamma_m(T_m^{(A)}(\phi_m))}{\gamma_m(T_m^{(A)}(1))}, \tag{3.9}$$

where ϕ_m is any bounded measurable function from F_m into $\mathbb{R}$. To connect the rare event probability and the critical trajectories with the normalized measures η_k, the decomposition product formula [Del Moral and Garnier, 2005]

$$\gamma_k(f_k) = \eta_k(f_k)\prod_{p=1}^{k-1}\eta_p(G_p). \tag{3.10}$$

gives the following interpretations

$$\mathbb{P}(V_m(X_m) \in A) = \eta_m(T_m^{(A)}(1))\prod_{k=1}^{m-1}\eta_k(G_k), \quad \text{and} \quad \mathbb{E}(\phi_m(X_0,\dots,X_m) \mid V(X_m) \in A) = \frac{\eta_m(T_m^{(A)}(\phi_m))}{\eta_m(T_m^{(A)}(1))}. \tag{3.11}$$

The benefits of using the normalised measures η_k is that they obey to the recursive equation

$$\eta_k = \Psi_{G_{k-1}}(\eta_{k-1})\mathcal{K}_k, \tag{3.12}$$

where $\mathcal{K}_k\eta_{k-1}$ is the measure obtained by the product of $\mathcal{K}_k$ and η_{k-1}, and $\Psi_G(\mu)$ denotes the Boltzmann-Gibbs transformation of a measure μ, associated to a potential function G:

$$\Psi_G(\mu)(\mathrm{d}y) := \frac{1}{\mu(G)}G(y)\mu(\mathrm{d}y).$$

Associated interacting path particle system

The weighted redistribution algorithm consists of a set of N paths of the Markov chain $\{X_k,\ k = 0,\dots,m\}$. The trajectories are updated from k to $k+1$ to advantage the ones that can potentially reach the rare event A. It is performed in two steps. First, the selection stage consists in choosing with replacement the trajectories according to an empirical weighted measure built with a

function G_k that can depend on the whole past of the process $X_{0:k} = (X_0, X_1, ..., X_k)$. Trajectories which are more likely to reach the rare set are multiplied and the others are killed. Secondly, the mutation stage consists in applying the Markov transition kernel K_k to the trajectory evolution. This section recalls the basic methodology and the main results for the interacting particle system algorithm, as explained in [Del Moral and Garnier, 2005].
For simplification reasons, it is assumed that the initial value $X_0 = x_0$ is deterministic. One is interested in approximations of the normalized distribution η_k with the following form

$$\eta_k^N = \sum_{i=1}^{N} w_k^i \delta_{\xi_k^i} \quad \text{with} \quad \sum_{j=1}^{N} w_k^j = 1,$$

where, for all $i = 1, \dots, N$ the particle $\xi_k^i \in F_k$ is a random trajectory $\xi_k^i = (\xi_{0,k}^i, \xi_{1,k}^i, \dots, \xi_{k,k}^i)$.
The initial generation of the Markov chain ξ, that is to say $\xi_1 = (\xi_1^i)_{1 \le i \le N}$, consists of a N-sample from the distribution

$$\eta_1(\mathrm{d}(x_0', x_1')) = \mathscr{K}_1(x_0, \mathrm{d}(x_0', x_1')) = \delta_{x_0}(\mathrm{d}x_0') K_1(x_0', \mathrm{d}x_1').$$

In other words, $\xi_1^i = (\xi_{0,1}^i, \xi_{1,1}^i) = (x_0, \xi_{1,1}^i) \in F_1 = E_0 \times E_1$ are independent, and the N random variables $\xi_{1,1}^i$ are sampled from $K(x_0, \cdot)$. The recursive equation 3.12 gives the following interpretation of the selection/mutation scheme of the particles $\xi_k = (\xi_k^i,\ i = 1, \dots, N)$:

$$F_{k-1}^N \ni \xi_{k-1} \xrightarrow{\text{selection}} \hat{\xi}_{k-1} \in F_{k-1}^N \xrightarrow{\text{mutation}} \xi_k \in F_k^N.$$

In practice, independently for all $i = 1, \dots, N$:

1. a path $\hat{\xi}_{k-1} = (\hat{\xi}_{0,k-1}^i, \dots, \hat{\xi}_{k-1,k-1}^i)$ is selected among the current population $\{\xi_{k-1}^i,\ i = 1, \dots, N\}$ according to the Boltzmann-Gibbs particle measure

$$\sum_{i=1}^{N} \frac{G_{k-1}(\xi_{0,k-1}^i, \dots, \xi_{k-1,k-1}^i)}{\sum_{j=1}^{N} G_{k-1}(\xi_{0,k-1}^j, \dots, \xi_{k-1,k-1}^j)} \delta_{(\xi_{0,k-1}^i, \dots, \xi_{k-1,k-1}^i)}.$$

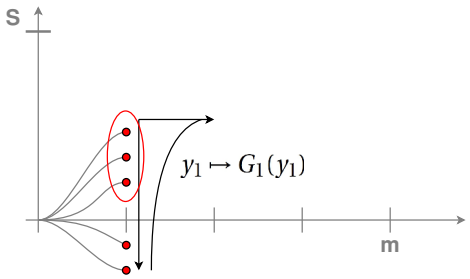
2. a random path $\xi_k^i = (\xi_{0,k}^i, \dots, \xi_{k,k}^i)$ is sampled from the distribution $\mathscr{K}_k(\hat{\xi}_{k-1}^i, \mathrm{d}y_k')$. In other words, we set $\xi_{p,k}^i = \hat{\xi}_{p,k-1}^i$ for all $p = 0, \dots, k-1$, and the random variable $\xi_{k,k}^i$ is sampled from the distribution $K_k(\hat{\xi}_{k-1,k-1}^i, \mathrm{d}x_k')$.

An illustration of the algorithm is provided in figure 3.2. The particle approximation of the normalised measure η_k stands as follows

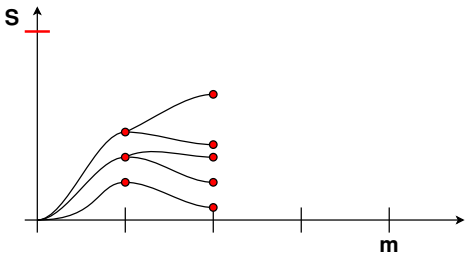
$$\eta_k \approx \eta_k^N = \frac{1}{N} \sum_{i=1}^{N} \delta_{(\xi_{0,k}^i, \dots, \xi_{k,k}^i)}. \tag{3.13}$$

From equation 3.10, we can also give some unbiased particle approximation measures γ_k^N of the un-normalised measure γ_k

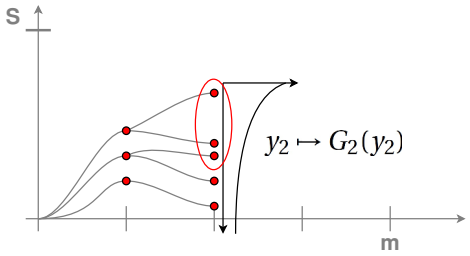
$$\gamma_k(f_k) \approx \gamma_k^N(f_k) \prod_{p=1}^{k-1} \eta_p^N(G_p).$$



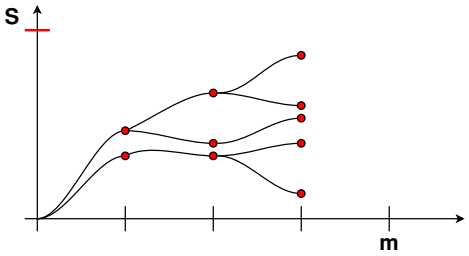
1. Selection.



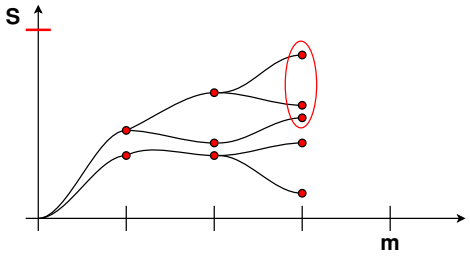
2. Mutation.



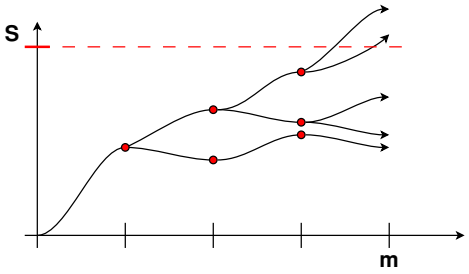
3. Selection.



4. Mutation.



5. Selection.



6. Mutation and estimation.

Figures 3.2: The weighted redistribution algorithm with the selection function $G_k(y_k) = G_k(x_1, \ldots, x_k)$.

With equations 3.8 and 3.9, the rare event probability and the law of the trajectories in the rare event regime are approximated by

$$P_A \approx \gamma_m^N(T_m^{(A)}(1)) = \eta_m^N(T_m^{(A)}(1)) \prod_{k=1}^{m-1} \eta_k^N(G_k) \quad \text{and} \quad \mathbb{E}(\phi_m(X_0, \ldots, X_m) \mid V(X_m) \in A) \approx \frac{\eta_m^N(T_m^{(A)}(\phi_m))}{\eta_m^N(T_m^{(A)}(1))}. \tag{3.14}$$

Results concerning weak convergence of the sequence of measures $(\eta_k^N)_{N\in\mathbb{N}^*}$ and $(\gamma_k^N)_{N\in\mathbb{N}^*}$, almost sure convergence and central limit theorem for approximations given at equations 3.14 and some L^p non-asymptotic estimates can be found in [Del Moral, 2004, Del Moral and Garnier, 2005]. Concentration inequalities are presented in [Del Moral et al., 2011].

Based on large deviation theory, and assuming that $E_0 = \ldots = E_n$, the original article [Del Moral and Garnier, 2005] proposes to use the following family of potential functions

$$G_k^\alpha(y_k) = G_k^\alpha(x_1, \ldots, x_k) = \exp[\alpha\,(V(x_k) - V(x_{k-1}))]. \tag{3.15}$$

Most of authors using the weighted redistribution algorithm includes the same idea [Carmona and Crepey, 2010, Carmona et al., 2009, Garnier and Del Moral, 2006, Giesecke et al., 2010]. Paper [Carmona and Crepey, 2010] also proposes other potential functions.

For the sake of understanding, the interacting path particle algorithm described in section 3.3 is rewritten in algorithm 3.2 in terms of the potential functions defined in equation (3.15).

3.4 Conflict zones in air traffic management

According to the international flight regulation, there are five types of conflicts in air traffic management which are mid-term conflict, short term conflict, conflict, near-collision and collision. Each of them is defined in the following way [Blom et al., 2007b, Prandini et al., 2011]. Assume given a triplet $(\Delta^*, d^*, h^*) \in ([0, +\infty))^3$. Two aircraft are said to be in conflict at time t if it exists a time $t' \in [t, t + \Delta^*]$ such that both of the following statements are verified

1. the expected horizontal relative position of the two aircraft at time t' is smaller that d^*,
2. the expected relative vertical position of the two aircraft at time t' is smaller that h^*.

The triplets five triplets (Δ^*, d^*, h^*) that characterizes the five types of conflicts above mentioned are written (Δ_k, d_k, h_k) for $k = 1, \ldots, 5$ and presented in table 3.3. Moreover, in the following we will make use of the term "conflict of level k" if the corresponding conflict in table 3.3 occurs.

We assume the air traffic scenario to involve n_a aircraft. The positions of these aircraft at time t are given by equation (8.6) with the vector $X_t = (X_t^1, \ldots, X_t^{n_a})$. The X_t vector belongs to $\mathbb{R}^{N_A}$ with $N_A = 3n_A$ if the aircraft positions are supposed to be in $\mathbb{R}^3$.

We can define some conflict zones as subsets of the state space of X_t in the following way. If a pair (i, j) of aircraft is in conflict of level k at time t, it is straightforward to check that the vector X_t actually belongs to the set

$$\begin{aligned} D_k^{i,j} = \{x \in \mathbb{R}^{N_A}, |y^{i,j}(x) + \Delta v^{i,j}(x)| \le d_k, \text{ and} \\ |z^{i,j}(x) + \Delta r^{i,j}(x)| \le h_k, \text{ for some } \Delta \in [0, \Delta_k]\}, \end{aligned}$$

where, for a pair (i, j) of aircraft, $i, j \in \{1, \ldots, n_a\}$, $v^{i,j}(X_t)$ is their relative speed and $r^{i,j}(X_t)$ is their relative rate of climbing.

Conversely, if X_t belongs to $D_k^{i,j}$ then the aircraft i ad j are in conflict of level k at time t.

We set next two additional definitions. First, an aircraft i is said to be in conflict of level k (with any other aircraft) if the vector X_t belongs to

$$D_k^i = \bigcup_{j \neq i} D_k^{i,j},$$

Algorithm 3.2: The weighted redistribution algorithm with multiplicative exponential selection functions

Data: Rare event probability set A, sample size N, potential functions G_k defined in equation (3.15), parameter $\alpha > 0$.

Result: $\hat{P}(A,\alpha) \approx \mathbb{P}(V(X_m) \in A)$

Initialization

  for $i = 1,\dots,N$ **do**

    Sample independently X_0^i from the initial distribution of the Markov chain $\{X_k,\, k = 0,\dots,m\}$.

    Set $W_0^i = x_0$, where x_0 is an arbitrary point, and denote $V_0 = V(x_0)$.

    This forms a set of N particles, denoted by $\{(X_0^i, W_0^i),\, i = 1,\dots,N\}$.

for $k = 0,\dots,m-1$ **do**

  Compute the normalising constant

$$\hat{\eta}_k^N = \frac{1}{N}\sum_{i=1}^{N} \exp[\alpha(V(X_k^i) - V(W_k^i))]$$

  */*Selection*/*

  Choose independently N particles amongst the set $\{(X_k^i, W_k^i),\, i = 1,\dots,N\}$ according to the empirical measure

$$\mu_k(\mathrm{d}\tilde{X}_k, \mathrm{d}\tilde{W}_k) = \frac{1}{N\hat{\eta}_k^N}\sum_{i=1}^{N} \exp[\alpha(V(X_k^i) - V(W_k^i))]\delta_{(X_k^i, W_k^i)}(\mathrm{d}\tilde{X}_k, \mathrm{d}\tilde{W}_k)$$

  The selected particles are denoted by $(\hat{X}_k^i, \hat{W}_k^i),\, i = 1,\dots,N$.

  */*Mutation*/*

  for $i = 1,\dots,N$ **do**

    Set $W_{k+1}^i = \hat{W}_k^i$.

    The chain evolves under the transition kernels K_{k+1}, $\hat{X}_k^i \xrightarrow{K_{k+1}} X_{k+1}^i$.

Estimation

$$\mathbb{P}(V(X_m) \in A) \approx \hat{P}(A,\alpha) = \prod_{k=1}^{m-1} \hat{\eta}_k^N \times \left(\frac{1}{N}\sum_{i=1}^{N} \mathbb{I}_A(V(X_n^i)) \exp[-\alpha(V(W_m^i) - V_0)]\right)$$

Table 3.3: Parameters for conflict zones.

Conflict type	mid-term conflict	short term conflict	conflict	near-collision	collision
k	1	2	3	4	5
d_k	4.5	4.5	4.5	1.25	0.054
h_k	900	900	900	500	131
Δ_k	8	2.5	0	0	0

second, the whole scenario is said to be in conflict of level k if the vector X_t belongs to

$$D_k = \bigcup_{i=1}^{n_a} D_k^i.$$

As an insight, for $i \neq j$ in $\{1,\dots,n_A\}$ and $k \geq 1$, we will be interested in the events that aircraft being in conflict during a fight duration of time T

$$\left\{X_t \in D_k^{i,j},\ \text{for some } 0 \leq t \leq T\right\},$$

$$\left\{X_t \in D_k^{i},\ \text{for some } 0 \leq t \leq T\right\}$$

and

$$\{X_t \in D_k,\ \text{for some } 0 \leq t \leq T\}.$$

Part II

Scientific Contributions

CHAPTER 4

Elaboration and validation of a robust adaptive splitting algorithm for rare event and extreme quantile estimation

In this chapter

- We elaborate a method to make fully adaptive the implementation of the splitting algorithm for rare event estimation for any Markov process and any time horizon.
- We extend the proposed method for the computation of extreme quantiles of the maximum of a real functional of a Markov process over its trajectory, and to compute in one run of the proposed adaptive splitting algorithm the whole cumulative distribution of the maximum of a Markov process up to a final time horizon.
- We provide almost sure convergence results for the estimated rare event probability, the approximate entrance distribution into the rare event probability set and the estimated extreme quantile.

4.1 Introduction

The problem under consideration is to estimate the very small probability that a random process $\{X(t),\ t \geq 0\}$ enters a set B before some final time T. This chapter is written in terms of a deterministic time T. All the methods and the algorithms elaborated here can readily be rewritten with a random stopping time T.

As discussed in section 1.2, several methods are available. They are statistical methods with the extreme value theory [Embrechts et al., 2011, Kotz and Nadarajah, 2000], numerical with some analytical results [Albin, 1998, Albin, 1990, Albin and Sunden, 2008, Aldous, 1989] and simulation techniques with essentially importance sampling [Sandmann, 2005, Glynn and Iglehart, 1989, Juneja and Shahabuddin, 2006, Heidelberger, 1995, Goyal et al., 1992, Alexopoulos and Shultes, 2001, Goyal et al., 1987, Nakayama, 1996, Ahamed et al., 2006] and splitting algorithm [L'Écuyer et al., 2007, L'Écuyer et al., 2009, Garvels and Kroese, 1998, Cérou et al.,

2006b]. Advantages and drawbacks of each methods are discussed in section 1.1 for the extreme value theory and in section section 1.2 for the other ones.

As far as the splitting is concerned, if the intermediate regions required for its implementation (see equation (3.1)) are well chosen, it appears to be a very good method to be used for the estimation of the considered probability if for example the state space is continuous. See section 1.2.5. However, as discussed in section 1.2.2, the problem of selecting the intermediate regions is not completely handled and has to be improved.

In this chapter, we propose an adaptive algorithm to estimate the intermediate regions to be used in the splitting algorithm. The rare event probability set B is assumed to be defined by some threshold exceedance of a real valued function (see section 1.2.4 for a discussion of such a characterization). In that case, each intermediate region can also be characterised as exceedance over a lower threshold of the same real valued function. The good point is that this latter threshold can easily be estimated using quantile consideration with a set of trajectories. However, it is practically impossible to remember where (at which place) and when (at which time) the trajectories used to estimate this new threshold did reach it. Indeed, this would imply to record all the whole trajectories. We propose here a two step algorithm. In a first step, the algorithm estimates the threshold that characterizes an intermediate region using quantiles of the maxima of a set of trajectories. Then, an other set of trajectories is used to obtain an approximation of the hitting place and hitting time of the newly intermediate region, and approximation of the conditional probability.

4.2 Adaptive algorithm for rare event estimation

Given a probability space $(\Omega, \mathscr{F}, \mathbb{P})$ and a complete measurable separable metric space E, let

$$X : (\omega, t) \in \Omega \times \mathbb{R}^+ \mapsto X(t)(\omega) \in E$$

be a Markov process with càdlàg trajectories. For B be a measurable subset of the state space E, we are interested here in giving some numerical approximations of

$$\mathbb{P}(T_B \leq T) = \mathbb{P}(X_t \in B, \text{ for some } 0 \leq t \leq T)$$

with the importance splitting algorithm.

The time T is the final time for the evolution of the process $\{X(t),\ t \geq 0\}$. It can be random with finite expectation *or* deterministic. Besides, we assume the set B to be characterized by a score function ϕ_0 as follows

$$B = \phi_0^{-1}\{[S, +\infty)\} = \left\{x \in E, \phi_0(x) \geq S\right\}.$$

Hence, we have that

$$\mathbb{P}(T_B \leq T) = \mathbb{P}\left(\max_{0 \leq t \leq T} \phi_0(X(t)) \geq S\right).$$

We recall that the splitting method requires a sequence of intermediate supersets of B, denoted by

$$B_1 \supsetneq \cdots \supsetneq B_m = B.$$

The key point of this chapter is to give a way to construct a sequence of thresholds

$$S_1 < \ldots < S_m = S \tag{4.1}$$

that characterizes the B_k regions required by the splitting algorithm in the following way

$$B_k = \phi_0^{-1}\{[S_k, +\infty)\} = \{x \in E, \phi_0(x) \geq S_k\}.$$

Thereby, we suppose ϕ_0 to be defined on E.

Remark 4.1. *As explained in section 3.2.4, the B_k regions can be time-dependent and the ϕ_0 function accordingly. But in section 1.2.4, we have seen that ϕ_0 is most of the time given by the problem for the definition of B and such a definition does not include time dependencies. To emphasize the practical side of the algorithms given in this chapter, we assume that ϕ_0 does not depend on the time. The whole chapter can readily be rewritten in terms of a time-dependent $\phi_0(x,t)$ function and time-dependant intermediate regions of $B \times [0, +\infty)$ if necessary.*

It is worth noting that the existence of left-hand limits for trajectories of the process $\{X(t),\ t \geq 0\}$ ensures that $X(T_k) \in \overline{B}_k$, where

$$T_k = \inf\{t \geq 0,\ X(t) \in B_k\}$$

and where $\overline{B}_k$ is the closure of B_k. Thus, $X(T_k) \in B_k$ provided B_k is a closed subset. The closure of the B_k regions is achieved for example if ϕ_0 is upper semi-continuous. From now on, we suppose to be in that case.

4.2.1 Threshold estimation and entrance distribution approximation

We first remark that the choice of the sequence (4.1) faces delicate problems. If two consecutive thresholds S_k, S_{k+1} are misplaced, the conditional probability $\mathbb{P}(T_{k+1} \leq T | T_k \leq T)$ could be very small and we lost the benefit of using the splitting algorithm, which is to consider some conditional probability sufficiently large to be reliably estimated with Monte Carlo. The expression of the asymptotic relative variance V_m^* in the case of optimal time-dependent region of $B_k \times \mathbb{R}^+$ detailed in section 3.2.4 highlights this remark. In that case, we have that [Cérou et al., 2012]

$$V_m^* = \sum_{k=1}^{m} \frac{1-p_k}{p_k}, \quad \text{where } p_k = \mathbb{P}(T_{k+1} \leq T | T_k \leq T),$$

which shows that the precision, in term of relative variance, decreases if a conditional probability value p_k becomes small. It is even highly possible that none of the trajectories starting from S_k reaches S_{k+1} and the algorithm stops. On the other hand, let us remark that the total computation time of the splitting algorithm increases with the number of re-sample steps, that is to say with the number of thresholds. Consequently, if the thresholds are too close from each others - in other words if the conditional probabilities value p_k are too close to 1 - the number m of thresholds increases, and so does the total computation time. Consequently, we would rather consider more trajectories with more space between thresholds. See also section 1.2.2 for a state of art for the choice of the p_k value.

From these observations, we conclude that a trade-off must be done between accuracy and computation time. The conditional probability must not be too small, or too large. Hence, we expect that a certain percentage, not too close to 0 or to 1, of the trajectories sampled from a certain threshold will reach the next threshold.

Threshold estimation via quantile considerations

Let us assume a given threshold S_k. To create the next threshold S_{k+1}, we consider first a given conditional probability value

$$p_k = \mathbb{P}(T_{k+1} \leq T | T_k \leq T) \in (0,1).$$

Obviously, we expect the threshold S_{k+1} to verify

$$\mathbb{P}(\exists\, t \leq T,\ \phi_0(X(t)) \geq S_{k+1} | T_k \leq T) = p_k.$$

Hence, if we denote by M_k the real random variable

$$M_k = \max_{T_k \leq t \leq T} \phi_0(X(t)),$$

we have that

$$\mathbb{P}(\exists\, t \leq T,\ \phi_0(X(t)) \geq S_{k+1} | T_k \leq T) = \mathbb{P}(M_k \geq S_{k+1} | T_k \leq T) = p_k. \tag{4.2}$$

In other words,

$$S_{k+1} \text{ is the } (1-p_k)\text{-quantile of } M_k. \tag{4.3}$$

To achieve an approximation of the threshold S_{k+1}, we assume that we know the threshold S_k and a sample $\{(X^i(T_k^i), T_k^i),\ i \in I_k\}$ of the space-time joined distribution of the entrance into the set B_k of the process $\{X(t),\ t \geq 0\}$. The theoretical entrance space-time joined distribution into B_k is defined by

$$\mu_k(\mathrm{d}x, \mathrm{d}t) = \mathbb{P}((X(T_k), T_k) \in \mathrm{d}x \times \mathrm{d}t | T_k \leq T) \tag{4.4}$$

and is approximated here by

$$\frac{1}{|I_k|} \sum_{i \in I_k} \delta_{(X^i(T_k^i), T_k^i)}(\mathrm{d}x \times \mathrm{d}t). \tag{4.5}$$

After that, we generate an N-sample of the random variable M_k denoted by

$$M_k^1, \ldots, M_k^N \tag{4.6}$$

of the process $\{\phi_0(X(t)),\ t \geq 0\}$, where the initial distribution of $\{(X(t), t),\ t \geq 0\}$ is the empirical distribution defined in equation 4.5. According to equation (4.3), we set $\tilde{S}_{k+1}$, an approximation of the threshold S_{k+1}, to be equal to the empirical $(1-p_k)$-quantile as follows

$$\tilde{S}_{k+1} = M_k^{([N \times (1-p_k)]+1)},$$

where $a \mapsto [a]$ is the floor function and $M_k^{(1)}, \ldots, M_k^{(N)}$ is the ordered sampled

$$M_k^{(1)} \leq \ldots \leq M_k^{(N)}. \tag{4.7}$$

For the sake of clarity, the index $[N \times (1-p_k)] + 1$ is now denoted by $q_{p_k}^N$ where

$$q_p^N = [N \times (1-p)] + 1, \text{ for } p \in (0,1). \tag{4.8}$$

To emphasize the fact that we have to know at what place and at what time the process $\{X(t),\ t \geq 0\}$ enters the set B_k, we define the process $\{Z(t),\ t \geq 0\}$ with

$$Z(t) = (X(t), t).$$

Accordingly, we define also the ϕ function by

$$\phi(z) = \phi_0(x), \quad \text{for } z = (x,t) \in E \times \mathbb{R}^+.$$

Entrance distribution approximation

The creation of a threshold relies on the approximation of $\mathbb{P}((X(T_k), T_k) \in \mathrm{d}x \times \mathrm{d}t | T_k \leq T)$ given in equation (4.5). To finish the recursive construction of the thresholds S_k, we have thus to give an approximation of the entrance distribution in the $(k+1)$-th region B_{k+1}.

We assume that approximations of the thresholds S_k and S_{k+1} have been obtained. They are denoted by $\tilde{S}_k$ and $\tilde{S}_{k+1}$. We assume also having a set $\{(\tilde{X}^i(\tilde{T}^i_k), \tilde{T}^i_k),\ i \in I_k\}$, which gives an approximation of μ_k, denoted $\tilde{\mu}_k$ defined as follows

$$\tilde{\mu}_k(\mathrm{d}x, \mathrm{d}t) = \frac{1}{|I_k|} \sum_{i \in I_k} \delta_{(\tilde{X}^i(\tilde{T}^i_k), \tilde{T}^i_k)}(\mathrm{d}x \times \mathrm{d}t). \tag{4.9}$$

We want to give an approximation of the distribution $\mathbb{P}((X(T_{k+1}), T_{k+1}) \in \mathrm{d}x \times \mathrm{d}t | T_{k+1} \leq T)$ in the same way as the standard splitting algorithm does. To achieve this goal, N trajectories of $\{Z(t),\ t \geq 0\}$ are sampled with initial distribution $\tilde{\mu}_k$ until the time

$$\tilde{T}_{k+1} = \inf\{t \geq 0, \phi(Z(t)) \geq \tilde{S}_{k+1}\}.$$

More precisely, for each $i = 1, \ldots, N$, we randomly and uniformly pick an element into the set $\{(\tilde{X}^i(\tilde{T}^i_k), \tilde{T}^i_k),\ i \in I_k\}$, which is denoted by

$$\widehat{Z}^i_k = (\widehat{X}^i_k, \widehat{T}^i_k). \tag{4.10}$$

Obviously, the law of $\widehat{Z}^i_k$ is characterized with equation (4.9) as follows

$$\widehat{Z}^i_k \sim \tilde{\mu}_k. \tag{4.11}$$

We denote by $t \mapsto \tilde{Z}^i(t)$ the trajectory starting from $\widehat{Z}^i_k$ at time $\widehat{T}^i_k$ and the stopping time $\tilde{T}^i_{k+1}$ can be rewritten as follows

$$\tilde{T}^i_{k+1} = \inf\left\{t \geq 0,\ \tilde{Z}^i(t) \text{ enters the set } \{\phi(z) \geq \tilde{S}_{k+1}\}\right\}.$$

Then, the approximation of the joint entrance place, entrance time distribution into the set B_{k+1} by $\{X(t),\ t \geq 0\}$ follows

$$\tilde{\mu}_{k+1}(\mathrm{d}x, \mathrm{d}t) \approx \frac{1}{|I_{k+1}|} \sum_{i \in I_{k+1}} \delta_{(\tilde{X}^i(\tilde{T}^i_{k+1}), \tilde{T}^i_{k+1})}(\mathrm{d}x \times \mathrm{d}t),$$

where $I_{k+1} = \left\{i = 1, \ldots, N,\ \tilde{T}^i_{k+1} < T\right\}$.

Conditional probability estimation

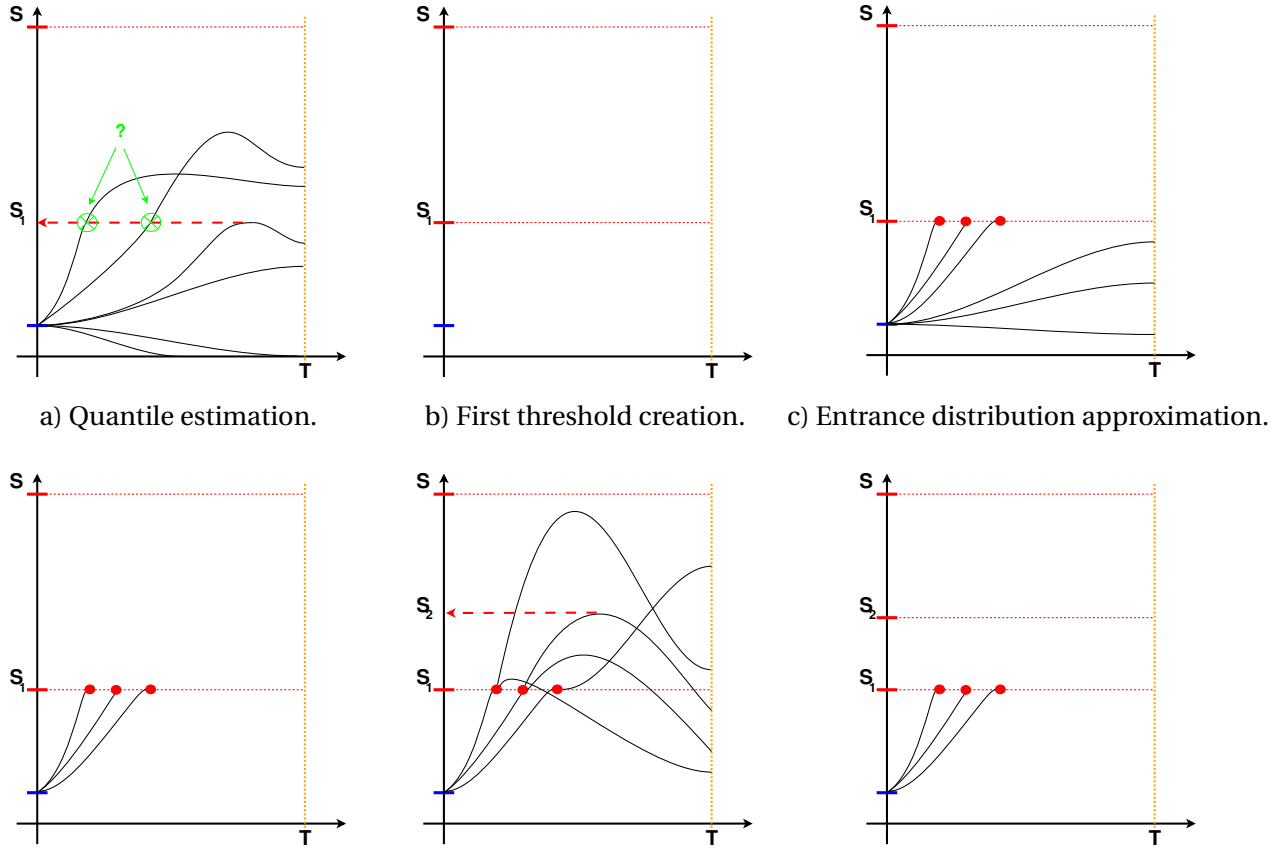
With the above detailed construction of the regions B_k required by the splitting algorithm we have two choices for the estimation of the conditional probability $\mathbb{P}(T_{k+1} \leq T | T_k \leq T)$. We could either use p_{k+1}, since this value is fixed and known, or we could estimate this probability by Monte Carlo simulation. This latter idea is achieved in the same way the splitting algorithm 3.1 (non adaptive) does. Namely, the p_{k+1} value could be estimated by

$$p_{k+1} \approx \frac{|I_{k+1}|}{N}. \tag{4.12}$$

Authors of [Cérou et al., 2012] mentioned that a bias can appear when the probability is estimated by the expected fixed value p_{k+1}. To prevent from this bias, the rest of this chapter focuses of the Monte Carlo estimation of p_{k+1} given in (4.12).

4.2.2 Summary for the implementation of the algorithm

The proposed method is thus performed in two steps between each threshold : a first one to estimate the threshold, and a second one to estimate the conditional probability value *and* the entrance distribution μ_k into the newly estimated threshold. Figures 4.1 presents the first steps of the algorithm. It clearly appears in the figure 4.1.a) that if we want the quantile estimation *and* the entrance distribution in the newly created region $\{(x,t),\ \phi(x) \geq S_1\}$ we have to keep in memory all the whole trajectories. Such a requirement may lead to memory storage problems.



a) Quantile estimation. b) First threshold creation. c) Entrance distribution approximation.

d) Keeping successful trajectories. e) Quantile estimation. f) Second threshold creation.

Figures 4.1: The first steps of the adaptive splitting algorithm 4.1 in the case of a deterministic final time T.

With this manner to proceed, we have fully determined adaptively and for every time horizon T deterministic or random :

- the intermediate regions B_k, implicitly defined with the importance function ϕ_0,
- approximations of the entrance distributions μ_k,
- approximations of the un-normalised distribution γ_k and the distribution η_k given in equations (3.2) and (3.4) with

$$\tilde{\gamma}_k(\mathrm{d}x,\mathrm{d}t) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{\phi(\tilde{X}^i(\tilde{T}_k^i)\geq \tilde{S}_k}\delta_{(\tilde{X}^i(\tilde{T}_k^i),\tilde{T}_k^i)}(\mathrm{d}x\times\mathrm{d}t), \tag{4.13}$$

and

$$\tilde{\eta}_k(\mathrm{d}x,\mathrm{d}t) = \frac{1}{N}\sum_{i=1}^{N} \delta_{(\tilde{X}^i(\tilde{T}_k^i),\tilde{T}_k^i)}(\mathrm{d}x\times\mathrm{d}t),$$

- an estimation of the rare event probability.

Remark 4.2. *With such a construction of the thresholds, we control the probability of extinction and we have approximations of entrance measures as the standard splitting does. That is why the term "robust splitting" is employed.*

4.2.3 The algorithm

The adaptive splitting algorithm is presented in algorithm 4.1. For the sake of simplicity, the sequence p_k is supposed to be constant and equals to $p \in (0,1)$. The algorithm is written in term of a deterministic stopping time T. If T is random it depends on the trajectory under consideration during the run of the algorithm 4.1 and have to be denoted by T^i.

4.3 Convergence results

4.3.1 Preliminaries

4.3.1.1 Notations

Due to the number of different objects involved in the algorithm and in the demonstrations below, we list here the notations.

- $F = E\times\mathbb{R}^+$ the state space of the process $\{Z(t),\ t\geq 0\}$,
- N the number of trajectories sampled at each threshold,
- $(S_1,\ldots,S_k,\ldots)$ the theoretical thresholds,
- $(\tilde{S}_1,\ldots,\tilde{S}_k,\ldots)$ the empirical thresholds created by algorithm 4.1,
- $\mu_0 = \mu_0^X\otimes\delta_0$, where μ_0^X is the initial distribution of $\{X(t),\ t\geq 0\}$ and δ_0 is the Dirac mass distribution in $0\in\mathbb{R}$. μ_0 is the initial distribution of the process $\{Z(t),\ t\geq 0\}$,
- $\{Z^0(t),\ t\geq 0\} = \{\tilde{Z}^0(t),\ t\geq 0\} = \{Z(t),\ t\geq 0\}$.
- $\{Z^k(t),\ t\geq 0\}$ the process with initial distribution μ_k defined at equation (4.4),
- $\{\tilde{Z}^k(t),\ t\geq 0\}$ the process with initial distribution $\tilde{\mu}_k$ defined at equation (4.9),
- $T_0 = \tilde{T}_0 = 0$ and by induction we define

$$T_k = \inf\{t\geq T_{k-1},\phi(Z^{k-1}(t))\geq S_k\}$$

and

$$\tilde{T}_k = \inf\{t\geq \tilde{T}_{k-1},\phi(\tilde{Z}^{k-1}(t))\geq \tilde{S}_k\}.$$

We assume that all the processes have the following transition kernels

$$K_{s,t}(z,\mathrm{d}z') = \mathbb{P}(Z^{z,s}(t)\in\mathrm{d}z').$$

We also need the cumulative distribution of the supremum of the image by ϕ of a trajectory of $\{Z(t),\ t\geq 0\}$ over final time T. Hence, we denote

Algorithm 4.1: The adaptive splitting algorithm

Data: Rare event probability set B, number N of trajectories to estimate a threshold and to estimate conditional probability, expected proportion p of trajectories that reach a threshold starting from the previous one, final time T, importance function ϕ.

Result: $\widehat{p}_B \approx \mathbb{P}(T_B \leq T)$

Initialization

  for $i = 1, \ldots, N$ **do**

    Set $\tilde{T}_0^i = 0$ and sample $Z_0'^i$ independently from law μ_0

    Set $e_0^i = Z_0'^i$

  Set $\widehat{p}_0 = 1$, $\tilde{S}_0 = 0$ and $I_0 = \{1, 2, \ldots, N\}$

  */*Iteration*/*

  $k = 0$

while $\tilde{S}_k < S$ **do**

  */*Estimation of the threshold*/*

  for $i = 1, \ldots, N$ **do**

    Choose randomly and uniformly a subscript $j_i \in I_k$

    Sample a path $Z'^i(t)$ starting from $e_k^{j_i}$ at time $\tilde{T}_k^{j_i}$ and until final time T

    Set $M_k^i = \max_{\tilde{T}_k^{j_i} \leq t \leq T} \phi(Z'^i(t))$

  Compute the threshold

$$\tilde{S}_{k+1} = M_k^{([(1-p)\tilde{N}]+1)}$$

  if $\tilde{S}_{k+1} \geq S$ **then**

    Set $\tilde{S}_{k+1} = S$

  */*Conditional entrance distribution and conditional probability estimations*/*

  for $i = 1, \ldots, N$ **do**

    Choose randomly and uniformly a subscript $j_i \in I_k$

    Sample a path $Z^i(t)$ starting from $e_k^{j_i}$ at time $\tilde{T}_k^{j_i}$ and until time $\tilde{T}_{k+1}^i \wedge T$

    where $\tilde{T}_{k+1}^i = \inf\{T_k^{j_i} \leq t \leq T, \phi(Z^i(t)) \geq \tilde{S}_{k+1}\}$

    Set $e_{k+1}^i = Z^i(\tilde{T}_{k+1}^i \wedge T)$

  Set $I_{k+1} = \{i, (\tilde{T}_{k+1}^i \wedge T) < T\}$ and

$$\widehat{p}_{k+1} = \frac{|I_{k+1}|}{N}$$

  */*Iteration*/*

  $k \leftarrow k + 1$

Estimation

$$\mathbb{P}(T_B \leq T) \approx \prod_{l=1}^{k} \widehat{p}_l$$

- F^k the cumulative distribution function of $\sup_{t\leq T}\phi(Z^k(t))$,
- $\tilde{\mathbb{F}}^k$ the empirical cumulative distribution function of $\sup_{t\leq T}\phi(\tilde{Z}^k(t))$, given by

$$\tilde{\mathbb{F}}^k(x) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{M_k^i \leq x},$$

where M_k^i, for $i = 1,\dots,N$, is the sequence of maxima given in equation (4.6).

Remark 4.3. *The adaptive algorithm only gives $\tilde{S}_k$, the distributions $\tilde{\mu}_k$ and the empirical cumulative distribution functions $\tilde{\mathbb{F}}^k$.*

Note that the number N of trajectories does not appear in several object definitions, and is included into the notation $\tilde{}$ for the sake of clarity.

4.3.1.2 The Skorohod topology

Since the process $\{Z(t),\ t \geq 0\}$ is a càdlàg process, we embed the space of trajectories with a topology that takes into account this specificity in order to have good measurability properties. That is why we make use of the Skorohod topology. We denote by D_t the set of all the càdlàg functions from $[0,t]$ into F. The set D_t is embedded with the Skorohod distance for which it is complete and separable, and the associated Borelian σ-field is denoted by $\mathscr{D}_t$. This distance on D_t is topologically equivalent to the following one [Billingsley, 1999, theorem 12.1]:

$$\tilde{d}(u,u') = \inf_{\lambda\in\Lambda}\left\{\|\lambda - I\| \vee \sup_{0\leq s\leq t} V\left(u(s), u'\circ\lambda(s)\right)\right\}, \tag{4.14}$$

where I is the identity function, $\|\cdot\|$ is the uniform norm on $[0,t]$, Λ is the set of all increasing bijections from $[0,t]$ into itself and V is the product metric on F defined with the metric of the space E. Consequently, for a sequence u_n and an element u, both in D_t, we have that u_n converges to u into $(D_t,\mathscr{D}_t)$ if, and only if, it exists a sequence of bijections $\lambda_n \in \Lambda$ such as

$$\lambda_n \xrightarrow[n\to+\infty]{} I \text{ and } u_n\circ\lambda_n \xrightarrow[n\to+\infty]{} u, \text{ both uniformly on } [0,t] \text{ for the absolute value and for } V. \tag{4.15}$$

We denote also D_∞ the set of all càdlàg functions from $[0,+\infty)$ into F embedded with the Skorohod distance for which the space is complete and separable, and $\mathscr{D}_\infty$ the associated σ-field. In [Billingsley, 1999, thm 16.2], a sequence of càdlàg functions u_n converges to u into $(D_\infty,\mathscr{D}_\infty)$ if the convergence holds for any restriction to $(D_t,\mathscr{D}_t)$ for all t where u is continuous. In the following, we assume that $\{Z(t),\ t\geq 0\}$ is defined for $t\in\mathbb{R}^+$ and that T is a random stopping time. Thus, we use $(D_\infty,\mathscr{D}_\infty)$ for the space of the trajectories of $\{Z(t),\ t\geq 0\}$. The rest of the chapter can readily be rewritten in terms of a deterministic final time T if necessary.

4.3.2 Main theorem and consequences

For the sake of clarity, we first show the following proposition for the uniform convergence of the empirical cumulative distribution function of the maxima of the process $\{\phi(Z(t)),\ t\geq 0\}$ over time horizon T.

Proposition 4.1. *Assume F^k to be continuous. If $\tilde{\mu}_k \xrightarrow[N\to+\infty]{a.s.} \mu_k$, then we have that*

$$\|F^k - \tilde{\mathbb{F}}^k\| \xrightarrow[N\to+\infty]{a.s.} 0$$

Proof. We first show, along the lines of [Ethier and Kurtz, 2009, chap 4,th 2.5], that $\{\tilde{Z}^k(t),\, t \geq 0\}$ converges in distribution to $\{Z^k(t),\, t \geq 0\}$ for the topology $\mathscr{D}_\infty$ of D_∞ at section 4.3.1.2. Equivalently, for any continuous function f from D_∞ into $\mathbb{R}$, we show that

$$\mathbb{E}(f(\tilde{Z}^k(t),\, t \geq 0)) \xrightarrow[N\to+\infty]{} \mathbb{E}(f(Z^k(t),\, t \geq 0)). \tag{4.16}$$

We first remark that

$$\mathbb{E}(f(\tilde{Z}^k(t),\, t \geq 0)) = \int \mathbb{E}(f(Z^{z,s}(t),\, t \geq 0))\mathrm{d}\mathbb{P}_{\widehat{Z}^k}(z,s), \tag{4.17}$$

where, for $z = (x,s)$, $Z^{z,s}$ is the process starting at z at time s and $\widehat{Z}^k$ is defined in (4.10). Equation (4.11) and the hypothesis $\tilde{\mu}_k \xrightarrow[N\to+\infty]{a.s.} \mu_k$ of the proposition imply that

$$\widehat{Z}^k \underset{N\to+\infty}{\overset{\text{Law}}{\Longrightarrow}} \mu_k. \tag{4.18}$$

Besides, the fact that the mapping

$$z \mapsto \mathbb{E}(f(Z^{z,s}(t),\, t \geq 0))$$

is continuous is a direct consequence of the definition of a Feller process. Hence, equations (4.17) and (4.18) lead to the following convergence

$$\{\tilde{Z}^k(t),\, t \geq 0\} \underset{N\to+\infty}{\overset{\text{Law}}{\Longrightarrow}} \{Z^k(t),\, t \geq 0\}. \tag{4.19}$$

Now we consider f in (4.16) to be the mapping Ψ:

$$\Psi(\{w(t),\, t \geq 0\}) = \sup_{t\geq 0} \phi(w(t)),$$

where w is a càdlàg function with $w(t) = (v(t), t) \in F = E \times \mathbb{R}^+$. The mapping Ψ is continuous as shown in the lemma 4.1.

Lemma 4.1. *The mapping Ψ is continuous from $(D_\infty, \mathscr{D}_\infty)$ into $\mathbb{R}$.*

Proof of the lemma 4.1. Since ϕ is continuous, it is sufficient to show the continuity of the mapping

$$\begin{aligned}(D_\infty(\mathbb{R}), \mathscr{D}_\infty(\mathbb{R})) &\to \mathbb{R}\\ u &\mapsto \sup\{u(t), t \geq 0\},\end{aligned}$$

where $(D_\infty(\mathbb{R}), \mathscr{D}_\infty(\mathbb{R}))$ is the set of càdlàg functions from $[0 + \infty)$ into $\mathbb{R}$ embedded with the Skorohod topology. To this purpose, let u_n be a sequence of elements of D_t that tends to u into D_t, for a given t for which u is continuous. With the reminder given in section 4.3.1.2, we show the continuity of Ψ restricted to D_t. Let λ_n a bijection of $[0,t]$ such as in (4.15). We have that

$$\sup_{0\leq s\leq t} u_n \circ \lambda_n(s) = \sup_{0\leq s\leq t} u_n(s)$$

since λ_n is a bijection. Then, the uniform convergence $u_n \circ \lambda_n \to u$ on $[0,t]$ gives that

$$\sup_{0\leq s\leq t} u_n(s) = \sup_{0\leq s\leq t} u_n \circ \lambda_n(s) \xrightarrow[n\to+\infty]{} \sup_{0\leq s\leq t} u(s), \text{ uniformly on } [0,t].$$

□

The continuity of Ψ together with the convergence (4.19) give that

$$\Psi(\{\tilde{Z}^k(t),\ t\geq 0\}) \underset{N\to+\infty}{\overset{\text{Law}}{\Longrightarrow}} \Psi(\{Z^k(t),\ t\geq 0\}). \tag{4.20}$$

The next step consists in showing that $\|F^k-\tilde{\mathbb{F}}^k\| \xrightarrow[N\to+\infty]{a.s.} 0$. To this end, we consider $\tilde{F}^k$ to be the theoretical cumulative distribution function of $\sup_{0\leq t\leq T}\phi(\tilde{Z}^k(t))$. The triangle inequality gives that

$$\|F^k-\tilde{\mathbb{F}}^k\| \leq \|F^k-\tilde{F}^k\| + \|\tilde{F}^k-\tilde{\mathbb{F}}^k\|. \tag{4.21}$$

Equation (4.20) and the continuity of F^k gives that $\tilde{F}^k$ converges uniformly to F^k [Van der Vaart, 1988, lemma 2.11]. On the other hand, the almost sure convergence

$$\|\tilde{F}^k-\tilde{\mathbb{F}}^k\| \xrightarrow[N\to+\infty]{a.s.} 0 \tag{4.22}$$

is a direct application of [Cérou and Guyader, 2007, lemma 2], using the DKW inequality [Massart, 1990] that stands as follows : if $\widehat{G}_N$ denotes the empirical distribution of an N-sample of a given random variable with distribution function G, then for all $\lambda>0$

$$\mathbb{P}\left(\sqrt{N}\|\widehat{G}_N-G\|\geq\lambda\right)\leq 2\exp(-2\lambda^2).$$

Hence equations (4.21) and (4.22) enable to conclude that

$$\|F^k-\tilde{\mathbb{F}}^k\| \xrightarrow[N\to+\infty]{a.s.} 0.$$

□

Now we prove the almost sure convergence of the thresholds created by the algorithm. We suppose the p_k sequence of conditional probability values for which we want to estimate the quantiles $\tilde{S}_k$ to be equal. We still denote $p=p_k$ for all k.

Besides, we say that a quadruplet $((\mu_N)_{N\in\mathbb{N}},\mu,(T_N)_{N\in\mathbb{N}},T_\infty)$, where μ_N is a sequence of distribution that converges to μ and T_N a sequence of stopping times that converges to T_∞ fulfils the H hypothesis if the couple $\left(T_N,\overline{Z}^N(T_N)\right)$ converges in distribution

$$\text{(H)}:\left(T_N,\overline{Z}^N(T_N)\right)\underset{N\to+\infty}{\overset{\text{Law}}{\Longrightarrow}}\left(T_\infty,\overline{Z}(T_\infty)\right),$$

where $\left\{\overline{Z}^N(t),\ t\geq 0\right\}$ and $\left\{\overline{Z}(t),\ t\geq 0\right\}$ are the processes with μ_N and μ for initial distribution respectively and $K_{s,t}(z,\mathrm{d}z')$ for transition kernels. The main theorem of this section stands at follows.

Theorem 4.1. *For each $k\geq 0$, assume that*

- *the cumulative distribution functions of the random variable*

$$\sup\{\phi(Z^k(t)),t\leq a\}$$

  is continuous functions for all $a\geq 0$,

- *$(F^k)^{-1}$ is continuous at $1-p$,*

- *the quadruplet $((\tilde{\mu}_k)_{N\in\mathbb{N}},\mu_k,(\tilde{T}_k)_{N\in\mathbb{N}},T_k)$ fulfils the H hypothesis,*

- $\mathbb{P}(T_k = T) = 0$.

Then we have the almost sure convergence of the empirical thresholds

$$\tilde{S}_k \xrightarrow[N\to+\infty]{a.s.} S_k.$$

Moreover, for each continuous and bounded function f, we have that

$$\tilde{\mu}_k(f) \xrightarrow[N\to+\infty]{a.s.} \mu_k(f) \quad \textit{and} \quad \tilde{\gamma}_k(f) \xrightarrow[N\to+\infty]{a.s.} \gamma_k(f).$$

Proof. The proof is made by induction. The induction hypothesis is the following

$(\mathrm{HR_k})$: $\tilde{S}_k \xrightarrow[N\to+\infty]{a.s.} S_k$, and for each continuous and bounded function f,

$$\tilde{\mu}_k(f) \xrightarrow[N\to+\infty]{a.s.} \mu_k(f) \quad \text{and} \quad \tilde{\gamma}_k(f) \xrightarrow[N\to+\infty]{a.s.} \gamma_k(f).$$

We assume $(\mathrm{HR_k})$.

♠ *Step 1.* Convergence of the sequence $\tilde{S}_{k+1}$ to S_{k+1}

The empirical $(1-p)$-quantile of $(M_k^i)_{1\le i\le N}$ is $M_k^{(q_p^N)}$, where $(M_k^{(i)})_{1\le i\le N}$ is the ordered sample and q_p^N is defined at equation (4.8). We first remark that

$$\tilde{\mathbb{F}}^k(M_k^{(q_p^N)}) = \frac{1}{N} q_p^N = \frac{1}{N}([(1-p)N]+1) \xrightarrow[N\to+\infty]{} 1-p. \tag{4.23}$$

Then, by the hypothesis $(\mathrm{HR_k})$, proposition (4.1) holds and we have with equation (4.23) that

$$\sup_{x\in\mathbb{R}} |F^k(x) - \tilde{\mathbb{F}}^k(x)| \xrightarrow[N\to+\infty]{a.s.} 0 \tag{4.24}$$

If we take for x in the left hand side of equation (4.24) the random variable $M_k^{(q_p^N)}$, we have that

$$F^k(M_k^{(q_p^N)}) \xrightarrow[N\to+\infty]{a.s.} 1-p.$$

Since $(F^k)^{-1}$ is continuous at point q, we have the first part of the theorem

$$\tilde{S}_{k+1} = M_k^{(q_p^N)} \xrightarrow[N\to+\infty]{a.s.} (F^k)^{-1}(1-p) = S_{k+1}.$$

♠ *Step 2.* Convergence of $\tilde{\mu}_{k+1}$ to μ_{k-1}.

First of all, for a continuous and bounded function f, the empirical measure $\tilde{\mu}_{k+1}$ is rewritten in terms of $\tilde{\gamma}_{k+1}$ as follows

$$\tilde{\mu}_{k+1}(f) = \frac{\frac{1}{N}\sum_{i=1}^N f(\tilde{\xi}_{k+1}^i)\mathbb{1}_{\phi(\tilde{\xi}_{k+1}^i)\ge\tilde{S}_{k+1}}}{\frac{1}{N}\sum_{i=1}^N \mathbb{1}_{\phi(\tilde{\xi}_{k+1}^i)\ge\tilde{S}_{k+1}}} \tag{4.25}$$

$$= \frac{\tilde{\gamma}_{k+1}(f)}{\tilde{\gamma}_{k+1}(1)}, \tag{4.26}$$

where $\tilde{\xi}^i_{k+1} = \tilde{Z}^i_k(\tilde{T}^i_k)$. Denoting

$$\tilde{g}_{k+1}(z) = \mathbb{1}_{\phi(z)\geq \tilde{S}_{k+1}},$$

and $\tilde{\eta}_{k+1}$ the empirical measure

$$\tilde{\eta}_{k+1} = \frac{1}{N}\sum_{i=1}^{N} \delta_{(\tilde{X}^i_{\tilde{T}^i_k}, \tilde{T}^i_k)},$$

we obtain that

$$\tilde{\gamma}_{k+1}(f) = \mathscr{S}^N(\tilde{\eta}_{k+1})(\tilde{g}_{k+1} f), \tag{4.27}$$

where $\mathscr{S}^N(\eta)$ denotes the empirical distribution function associated to the measure η

$$\mathscr{S}^N(\eta) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\zeta^i}, \text{ where } \zeta^i \ i.i.d. \text{ with distribution } \eta.$$

The problem is now reduced to the demonstration of

$$\tilde{\gamma}_{k+1}(f) \xrightarrow[N\to+\infty]{a.s.} \gamma_{k+1}(f), \tag{4.28}$$

with $\gamma_{k+1}(f) = \mu_k Q_{k+1}(g_{k+1} f)$.
For that purpose, we decompose (4.27) as follows

$$\tilde{\gamma}_{k+1}(f) = [\mathscr{S}^N(\tilde{\eta}_{k+1})(\tilde{g}_{k+1} f) - \tilde{\eta}_{k+1}(\tilde{g}_{k+1} f)] + \tilde{\eta}_{k+1}(\tilde{g}_{k+1} f). \tag{4.29}$$

Mimicking the proof of [Crisan and Doucet, 2002, lemma 2] and taking into account the independence of $\tilde{\xi}^i_{k+1}$ conditionally to the $\mathscr{F}^N$ σ-field, with $\mathscr{F}^N = \sigma(\tilde{\xi}^i_{k+1},\ i \in I_k)$ we have that

$$\begin{aligned}
&\mathbb{E}\left(\left[\mathscr{S}^N(\tilde{\eta}_{k+1})\tilde{g}_{k+1} f - \tilde{\eta}_{k+1}(\tilde{g}_{k+1} f)\right]^4 | \mathscr{F}^N\right) \\
&= \frac{1}{N^4}\mathbb{E}\left(\left[\sum_{i=1}^{N} \tilde{\eta}_{k+1}(\tilde{g}_{k+1} f) - (\tilde{g}_{k+1} f)(\tilde{\xi}^i_{k+1})\right]^4 | \mathscr{F}^N\right) \\
&= \frac{1}{N^4}\sum_{i=1}^{N} \mathbb{E}\left(\left[\tilde{\eta}_{k+1}(\tilde{g}_{k+1} f) - (\tilde{g}_{k+1} f)(\tilde{\xi}^i_{k+1})\right]^4 | \mathscr{F}^N\right) \\
&= \frac{6}{N^4}\sum_{1\leq i_1\neq i_2\leq N} \mathbb{E}\left(\left[\tilde{\eta}_{k+1}(\tilde{g}_{k+1} f) - (\tilde{g}_{k+1} f)(\tilde{\xi}^{i_1}_{k+1})\right]^2 \left[\tilde{\eta}_{k+1}(\tilde{g}_{k+1} f) - (\tilde{g}_{k+1} f)(\tilde{\xi}^{i_2}_{k+1})\right]^2 | \mathscr{F}^N\right) \\
&\leq \frac{2^4\|f\|^4 + 2^4\|f\|^4 \times 3N(N-1)}{N^4} \\
&\leq 48\|f\|^4 \frac{1}{N^2}.
\end{aligned} \tag{4.30}$$

Hence,

$$\mathbb{E}\left(\sum_{N=1}^{\infty}\left[\mathscr{S}^N(\tilde{\eta}_{k+1})(\tilde{g}_{k+1} f) - \tilde{\eta}_{k+1}(\tilde{g}_{k+1} f)\right]^4\right) \leq 48\|f\|^4 \sum_{N=1}^{\infty}\frac{1}{N^2} < \infty, \tag{4.31}$$

which implies with the Borel Cantelli lemma the following almost sure convergence

$$\mathscr{S}^N(\tilde{\eta}_{k+1})(\tilde{g}_{k+1} f) - \tilde{\eta}_{k+1}(\tilde{g}_{k+1} f) \xrightarrow[N\to+\infty]{a.s.} 0. \tag{4.32}$$

In order to show (4.28), with (4.29) and (4.32) it is sufficient to show that

$$\tilde{\eta}_{k+1}(\tilde{g}_{k+1} f) \xrightarrow[N\to+\infty]{a.s.} \mu_k Q_{k+1}(g_{k+1} f). \tag{4.33}$$

We first recall that, with proposition 3.1, we have

$$\eta_{k+1} = \mu_k Q_{k+1}.$$

Now, we remark that

$$T_{k+1} \leq T \qquad \text{if, and only if} \qquad \phi(Z^k(T_{k+1} \wedge T)) \geq S_{k+1},$$

hence

$$\begin{aligned}\eta_{k+1}(g_{k+1}f) &= \int \mathbb{1}_{\phi(z)\geq S_{k+1}} f(z) \mathrm{d}\mathbb{P}_{Z^k(T_{k+1}\wedge T)}(z) \\ &= \mathbb{E}\left(\mathbb{1}_{\phi(Z^k(T_{k+1}\wedge T))\geq S_{k+1}} f(Z^k(T_{k+1}\wedge T)\right) \\ &= \mathbb{E}\left(\mathbb{1}_{T_{k+1}\leq T} f(Z^k(T_{k+1}))\right).\end{aligned}$$

Similarly

$$\tilde{T}_{k+1} \leq T \qquad \text{if, and only if} \qquad \phi(\tilde{Z}^k(\tilde{T}_{k+1} \wedge T)) \geq \tilde{S}_{k+1},$$

hence

$$\tilde{\eta}_{k+1}(\tilde{g}_{k+1}f) = \mathbb{E}\left(\mathbb{1}_{\tilde{T}_{k+1}\leq T} f(\tilde{Z}^k(\tilde{T}_{k+1}))\right).$$

We recall that the convergence in distribution

$$(\tilde{T}_{k+1}, \tilde{Z}^k(\tilde{T}_{k+1})) \Longrightarrow (T_{k+1}, Z^k(T_{k+1}))$$

holds if $\tilde{T}_{k+1} \Longrightarrow T_{k+1}$. This last condition is given with the following lemma.

Lemma 4.2. *The following convergence in distribution holds*

$$\tilde{T}_{k+1} \underset{N\to+\infty}{\overset{Law}{\Longrightarrow}} T_{k+1}.$$

Proof. For any positive real number a, we have that

$$\begin{aligned}\mathbb{P}(\tilde{T}_{k+1} \leq a) &= \mathbb{P}\left(\sup_{t\leq a} \phi(\tilde{Z}^k(t)) \geq \tilde{S}_{k+1}\right) \\ &= \int \mathbb{P}\left(\sup_{t\leq a} \phi(\tilde{Z}^k(t)) \geq s\right) \mathrm{d}\mathbb{P}_{\tilde{S}_{k+1}}(s) \\ &= \int \mathbb{P}\left(\sup_{t\leq a} \phi(\tilde{Z}^k(t)) \geq s\right) - \mathbb{P}\left(\sup_{t\leq a} \phi(Z^k(t)) \geq s\right) \mathrm{d}\mathbb{P}_{\tilde{S}_{k+1}}(s) \\ &\qquad + \int \mathbb{P}\left(\sup_{t\leq a} \phi(Z^k(t)) \geq s\right) \mathrm{d}\mathbb{P}_{\tilde{S}_{k+1}}(s).\end{aligned}$$

We conclude using the uniform convergence of the cumulative distribution function of $\sup_{t\leq a} \phi(\tilde{Z}^k(t))$ given in the proof of proposition 4.1 and the convergence of $\tilde{S}_{k+1}$ toward S_{k+1}. □

Now we remark that the set where the application $(t,z) \mapsto \mathbb{1}_{t\leq T} f(z)$ is discontinuous is the set $\{t = T\}$. Hereby, the continuous mapping theorem [Billingsley, 1999, Chap. 1, Section 2] implies that

$$\tilde{\eta}_{k+1}(\tilde{g}_{k+1}f) \longrightarrow \eta_{k+1}(g_{k+1}f), \tag{4.34}$$

as long as $\mathbb{P}(T_{k+1} = T) = 0$, which is included in the hypotheses.

Combining the convergences given in equations (4.34) and (4.32), and using the decomposition (4.29) we have that for any continuous and bounded function f

$$\tilde{\gamma}_{k+1}(f) \underset{N\to+\infty}{\overset{a.s.}{\longrightarrow}} \gamma_{k+1}(f),$$

and

$$\tilde{\mu}_{k+1}(f) \xrightarrow[N\to+\infty]{a.s.} \mu_{k+1}(f),$$

which ends the proof. □

From this theorem, we deduce the following expected corollary on the convergence of the estimated conditional probabilities the estimated rare event probability.

Corollary 4.1. *Assume the conditions of theorem 4.1 hold. Then the estimated conditional probabilities and the estimated rare event probability almost surely converge*

$$\frac{|I_k|}{N} \xrightarrow[N\to+\infty]{a.s.} \mathbb{P}(T_k \le T | T_{k-1} \le T),$$

and

$$\widehat{P}_B \xrightarrow[N\to+\infty]{a.s.} \mathbb{P}(T_B \le T).$$

Proof. For the conditional probabilities it is sufficient to write that

$$\frac{|I_k|}{N} = \frac{\tilde{\gamma}_k(1)}{\tilde{\gamma}_{k-1}(1)},$$

and to use the convergences given in theorem 4.1. For the rare event probability estimation, we first remark that the number $\tilde{m}$ of created thresholds with algorithm 4.1 almost surely converges to the theoretical number m of thresholds defined by

$$\mathbb{P}(T_B \le T) = \gamma_m(1).$$

Indeed, using the fact that $\tilde{S}_k \xrightarrow[N\to+\infty]{a.s.} S_k$, for all $k \ge 1$, it exists an integer k^* and N_0 such that

$$\mathbb{P}(\tilde{S}_{k^*} \le S < \tilde{S}_{k^*+1},\ \forall N \ge N_0) = 1.$$

To conclude, we remark that $\mathbb{P}(T_B \le T)$ is estimated by $\tilde{\gamma}_m(1)$. □

4.4 Adaptive splitting algorithm for extreme quantile estimation

4.4.1 The algorithm

Given a real α into the real interval $(0,1)$ which is supposed to be very small, the aim of this section is to give numerical approximations of the $(1-\alpha)$-quantile of the supremum of the real random variable $\phi(Z(t))$ over time T, where T can be as well deterministic as random. Namely, we want to compute $\mathfrak{q}_\alpha$ such as

$$\mathfrak{q}_\alpha = \inf\left\{\mathfrak{q},\ \mathbb{P}\left(\sup_{0\le t\le T} \phi(Z(t)) \le \mathfrak{q}\right) \ge 1-\alpha\right\}.$$

If α is very small, the quantile $\mathfrak{q}_\alpha$ cannot be efficiently estimated with the usual Monte Carlo method.

The $\mathfrak{q}_\alpha$ number may be understood as the threshold that gives a risk equals to α in term of threshold exceedance of the process $\{\phi(Z(t)),\ t \ge 0\}$ over time T. For example, if

$$x \mapsto \mathbb{P}(\sup_{0\le t\le T} \phi(Z(t)) \le x)$$

is continuous, then q_α is such that

$$\alpha = \mathbb{P}\left(\sup_{0\le t\le T}\phi(Z(t)) \ge \mathrm{q}_\alpha\right).$$

To estimate q_α, we suppose that a sequence of thresholds $S_1 < \ldots < S_k < S_{k+1}$ is given with

$$\mathbb{P}\left(\sup_{0\le t\le T}\phi(Z(t)) > S_k\right) > \alpha \quad \text{and} \quad \mathbb{P}\left(\sup_{0\le t\le T}\phi(Z(t)) > S_{k+1}\right) \le \alpha. \tag{4.35}$$

With such a configuration, the quantile q_α is thus within S_k and S_{k+1}

$$S_k \le \mathrm{q}_\alpha < S_{k+1}.$$

Consider now the real number r defined as follows

$$r = \mathbb{P}\left(\sup_{0\le t\le T}\phi(Z(t)) \ge \mathrm{q}_\alpha | \sup_{0\le t\le T}\phi(Z(t)) \ge S_k\right). \tag{4.36}$$

Using the fact that $S_k \le \mathrm{q}_\alpha$, we remark that

$$\begin{aligned} r &= \frac{\mathbb{P}(\sup_{0\le t\le T}\phi(Z(t)) \ge \mathrm{q}_\alpha)}{\mathbb{P}(\sup_{0\le t\le T}\phi(Z(t)) \ge S_k)} \\ &= \frac{\alpha}{\mathbb{P}(\sup_{0\le t\le T}\phi(Z(t)) \ge S_k)}. \end{aligned}$$

The problem is now reduced to the estimation of the r-quantile of the real random variable $\sup_{0\le t\le T}\phi(Z(t))$ under the probability measure $\mathbb{P}\left(\cdot|\sup_{0\le t\le T}\phi(Z(t)) \ge S_k\right)$.

Hence, if we have a sample $m_k^1, \ldots, m_k^N$ of $\sup_{0\le t\le T}\phi(Z(t))|\sup_{0\le t\le T}\phi(Z(t)) \ge S_k$, we can thus estimate q_α with

$$\mathrm{q}_\alpha \approx m_k^{(q_r^N)},$$

where $m_k^{(1)} < \ldots < m_k^{(N)}$ is the ordered sample and q_r^N is defined in equation (4.8).

From this considerations, we deduce that we need the two following things :

1. a sequence of thresholds $S_1 < \ldots < S_{k+1}$,
2. a sample of the random variable $\sup\phi(Z(t))|\sup\phi(Z(t)) \ge S_k$.

These two objects are indeed approximated by the adaptive algorithm 4.1. To obtain the correct length of the sequence $S_1 < \ldots < S_{k+1}$, it is in fact sufficient to run algorithm 4.1 and to change the loop condition $\tilde{S}_k < B$ into $\prod_{l=1}^k \widehat{p}_l > \alpha$. Indeed, remarking that for all $k \ge 1$

$$\mathbb{P}\left(\sup_{0\le t\le T}\phi(Z(t)) > S_k\right) = \prod_{l=1}^k \mathbb{P}(T_l \le T | T_{l-1} \le T),$$

equation (4.35) gives that

$$\prod_{l=1}^k \mathbb{P}(T_l \le T | T_{l-1} \le T) > \alpha \quad \text{and} \quad \prod_{l=1}^{k+1} \mathbb{P}(T_l \le T | T_{l-1} \le T) \le \alpha.$$

Besides, we do not know how to simulate exactly from the distribution of the random variable $\sup\phi(Z(t))|\sup\phi(Z(t)) \ge S_k$, but we still have the following approximation given at equation (4.9)

$$\mu_k(\mathrm{d}z) \approx \frac{1}{|I_k|}\sum_{i\in I_k}\delta_{\tilde{\xi}_k^i}(\mathrm{d}z),$$

where $\tilde{\xi}_k^i = (\tilde{X}^i(\tilde{T}_k^i), \tilde{T}_k^i)$. That is why we do not work with a sample $m_k^1, \ldots, m_k^N$, but with a sample $M_k^1, \ldots, M_k^N$, where for each $i \in 1, \ldots, N$, M_k^i is the maximum of the trajectory $t \mapsto \phi(\widehat{Z}_i(t))$ defined in equation (4.11) where the initial distribution of $\{\widehat{Z}_i(t),\ t \geq 0\}$ is $1/|I_k| \sum_{i \in I_k} \delta_{\tilde{\xi}_k^i}(\mathrm{d}z)$.

These considerations lead to the algorithm 4.2.

Algorithm 4.2: Adaptive extreme quantile estimation

Data: $\alpha \in (0,1)$ (small) real number, number N of trajectories to estimate a threshold and to estimate conditional probability, expected proportion p of trajectories that reach a threshold starting from the previous one, final time T

Result: $\widehat{\mathrm{q}}_\alpha \approx \mathrm{q}_\alpha$ with $\mathbb{P}(\sup_{0 \leq t \leq T} \phi(Z(t)) \geq \mathrm{q}_\alpha) = \alpha$.

Initialization

  Lines 1-7 of algorithm 4.1.

  Set $\widehat{P}_1 = 1$.

while $\widehat{P}_k > \alpha$ **do**

  */*Estimation of the threshold*/*

  Lines 9-16 of algorithm 4.1.

  */*Conditional entrance distribution and conditional probability estimations*/*

  Lines 18-25 of algorithm 4.1.

  Set

$$\widehat{P}_{k+1} = \prod_{l=1}^{k+1} \widehat{p}_l$$

  */*Iteration*/*

  $k \leftarrow k+1$

Estimation

  Draw N trajectories of the process $\{Z(t),\ t \geq 0\}$ with initial distribution

$$\frac{1}{|I_k|} \sum_{i \in I_k} \delta_{e_k^i}(\mathrm{d}z),$$

  Set $M_k^{(1)} \leq \ldots \leq M_k^{(N)}$ the ordered maxima of these trajectories, $\widehat{r} = \dfrac{\alpha}{\widehat{P}_k}$ and

$$\widehat{\mathrm{q}}_\alpha = M_k^{(q_{\widehat{r}}^N)}.$$

4.4.2 Almost sure convergence of the estimated quantile

The following convergence result holds for the estimation of the extreme quantile q_α with algorithm 4.2.

Proposition 4.2. *Assume the conditions of theorem 4.1 hold. Assume also that F_k, the cumulative distribution function of $\sup \phi(Z(t)) | \sup \phi(Z(t)) \geq S_k$ is such that $(F_k)^{-1}$ is defined and continuous at point $1-r$. Then we have*

$$\widehat{q}_\alpha \xrightarrow[N \to +\infty]{a.s.} q_\alpha.$$

Proof. Algorithm 4.2 gives $\widehat{r}$, an approximation of equation (4.36) as follows

$$\widehat{r} = \frac{\alpha}{\widehat{P}_{k+1}}$$

and corollary 4.1 gives that

$$\widehat{r} \xrightarrow[N\to+\infty]{a.s.} \frac{\alpha}{P_{k+1}}, \tag{4.37}$$

where $P_{k+1} = \mathbb{P}(\sup_{0\le t\le T}\phi(Z(t)) \ge S_{k+1})$.
The definition of $\widehat{\mathfrak{q}}_\alpha$ gives that

$$\tilde{\mathbb{F}}^k(\widehat{\mathfrak{q}}_\alpha) = \tilde{\mathbb{F}}^k(M_k^{(q_{\widehat{r}}^N)}) = \frac{1}{N}([(1-\widehat{r})\times N]+1),$$

consequently

$$\tilde{\mathbb{F}}^k(\widehat{\mathfrak{q}}_\alpha) \xrightarrow[N\to+\infty]{a.s.} 1-r.$$

Theorem 4.1 with proposition 4.1 give that

$$\|\tilde{\mathbb{F}}^k - \tilde{F}^k\| \xrightarrow[N\to+\infty]{a.s.} 0,$$

and thus

$$F^k(\widehat{\mathfrak{q}}_\alpha) \xrightarrow[N\to+\infty]{a.s.} 1-r.$$

Using the hypotheses on $(F^k)^{-1}$, it follows that

$$\widehat{q}_\alpha \xrightarrow[N\to+\infty]{a.s.} (F^k)^{-1}(1-r),$$

and since $1-r = F^k(\mathfrak{q}_\alpha)$, we have

$$\widehat{\mathfrak{q}}_\alpha \xrightarrow[N\to+\infty]{a.s.} \mathfrak{q}_\alpha.$$

□

4.5 Adaptive splitting algorithm for the whole cumulative distribution function

4.5.1 The algorithm

We give here a way to estimate in a fully adaptive manner and in one run of a modified version of algorithm 4.1 the whole cumulative distribution function F of $\sup_{0\le t\le T}\phi(X(t))$:

$$F(x) = \mathbb{P}\left(\sup_{0\le t\le T}\phi(X(t)) \le x\right).$$

As usual, we focus on the extreme values of F, that is to say for F close to 1.

Assume that the threshold $\tilde{S}_k$ is created by algorithm 4.1. From this threshold, N trajectories of $\{Z(t),\ t\ge 0\}$ are sampled according to the empirical initial distribution

$$\frac{1}{|I_k|}\sum_{i\in I_k}\delta_{\tilde{\xi}_k^i}(\mathrm{d}z),$$

as detailed in equation (4.9) and we denote by $M_k^1,\dots,M_k^N$ the maxima of the image by ϕ of these trajectories until time T. We denote p the conditional probability $\mathbb{P}(T_k \le T | T_{k-1} \le T)$ which is supposed to be constant with respect to k. According to the adaptive algorithm 4.1, the next threshold $\tilde{S}_k$ equals to $M_k^{(q_p^N)}$, where q_p^N is defined in equation (4.8), and we have

$$\tilde{S}_k \le M_k^{(1)} \le \dots \le M_k^{(q_p^N)} \le \tilde{S}_{k+1}.$$

Now, we can construct an estimate of F at the points $M_k^{(l)}$ for $l = 1,\dots,q_p^N$. To this end, we make use of the identity

$$\mathbb{P}\left(\sup_{0\le t\le T}\phi(X(t)) \ge M_k^{(l)}\right) = \mathbb{P}\left(\sup_{0\le t\le T}\phi(X(t)) \ge M_k^{(l)} | \sup_{0\le t\le T}\phi(X(t)) \ge \tilde{S}_k\right) \times \mathbb{P}\left(\sup_{0\le t\le T}\phi(X(t)) \ge \tilde{S}_k\right),$$

and the fact that $\mathbb{P}(\sup_{0\le t\le T}\phi(X(t)) \ge \tilde{S}_k)$ is estimated with algorithm 4.1 by

$$\mathbb{P}\left(\sup_{0\le t\le T}\phi(X(t)) \ge \tilde{S}_k\right) \approx \widehat{P}_k,$$

where $\widehat{P}_k = \prod_{j=1}^k \widehat{p}_j$. Hence, the estimation

$$\mathbb{P}\left(\sup_{0\le t\le T}\sup\phi(X(t)) \ge M_k^{(l)} | \sup\phi(X(t)) \ge \tilde{S}_k\right) \approx \frac{N-l}{N}$$

leads to

$$\mathbb{P}\left(\sup_{0\le t\le T}\phi(X(t)) \ge M_k^{(l)}\right) \approx \frac{N-l}{N} \times \widehat{P}_k.$$

In other words, we have the estimate

$$F(M_k^{(l)}) \approx 1 - \frac{N-l}{N} \times \widehat{P}_k.$$

With such a method, we can estimate F at q_p^N points between two consecutive thresholds. Notice that we do not use the values $M_k^{(q_p^N+1)} \le \dots \le M_k^{(N)}$ because we prefer focusing on the values of the quantile of F_k that are in a reasonable order of magnitude. Algorithm 4.3 sums up the described procedure.

4.5.2 Almost sure convergence results

We take the notations of algorithm 4.3. The following convergence result holds. (4.3).

Proposition 4.3. *Assume the hypotheses of theorem 4.1. Assume also that F_k are homeomorphisms for each k. Then, for any sequence l_N such as*

$$l_N/N \xrightarrow[N\to+\infty]{} 1-r \in (0,1-p),$$

the following convergence for the point-wise estimation of the curve of $F(x) = \mathbb{P}(\sup\phi(Z(t)) \le x)$ holds

$$\widehat{x}_k^{l_N} = M_k^{l_N} \xrightarrow[N\to+\infty]{a.s.} x_{k,r} \quad \text{and} \quad \widehat{F}_k^{l_N} = \widehat{P}_k \times \frac{N-l_N}{N} \xrightarrow[N\to+\infty]{a.s.} F(x_{k,r}),$$

where $x_{k,r}$ is the $\left(1-P^k\times(1-r)\right)$-quantile of F.

Algorithm 4.3: Adaptive estimation of the cumulative distribution function of $\sup \phi(Z)$

Data: α (small) real number, number N of trajectories to estimate a threshold and to estimate conditional probability, expected proportion p of trajectories that reach a threshold starting from the previous one, final time T

Result: A point wise estimation of the curve $(x, F(x))$

Initialization

  Lines 1-7 of algorithm 4.1.

  Set $\widehat{P}_1 = 1$.

while $\widehat{P}_k > \alpha$ **do**

  */*Estimation of the threshold*/*

  Lines 9-15 of algorithm 4.1.

  Set

$$(\hat{x}_k^1, \ldots, \hat{x}_k^{q_p^N}) = (M_k^{(1)}, \ldots, M_k^{(q_p^N)}),$$

  and

$$(\widehat{F}_k^1, \ldots, \widehat{F}_k^{q_p^N}) = (\widehat{P}_k \times \frac{N-1}{N}, \ldots, \widehat{P}_k \times \frac{q_p^N}{N}).$$

  */*Conditional entrance distribution and conditional probability estimations*/*

  Lines 18-25 of algorithm 4.1.

  Set

$$\widehat{P}_{k+1} = \prod_{l=1}^{k+1} \widehat{p}_l$$

  */*Iteration*/*

  $k \leftarrow k+1$

Estimation

  The curve $(x, F(x))$ of F is estimated over $k \times q_p^N$ points

$$\{(\hat{x}_j^l, \widehat{F}_j^l),\ 1 \le j \le k,\ 1 \le l \le q_p^N\}.$$

Proof. We first remark that

$$\tilde{\mathbb{F}}^k(M_k^{(l_N)}) = \frac{l_N}{N} \xrightarrow[N\to+\infty]{} 1 - r.$$

Theorem 4.1 with proposition 4.1 give that

$$\|\tilde{\mathbb{F}}^k - \tilde{F}^k\| \xrightarrow[N\to+\infty]{a.s.} 0,$$

consequently

$$F_k(M_k^{(l_N)}) \xrightarrow[N\to+\infty]{a.s.} 1 - r.$$

Since F_k is bijective, we have that

$$M_k^{l_N} \xrightarrow[N\to+\infty]{a.s.} (F_k)^{-1}(1-r) = x_{k,r}.$$

where $x_{k,r}$ is the $(1-r)$-quantile of F_k. Then, the following identity

$$\mathbb{P}\left(\sup_{0\le t\le T} \phi(Z(t)) \ge x_{k,r}\right) = \mathbb{P}\left(\sup_{0\le t\le T} \phi(Z(t)) \ge S_k\right) \times \mathbb{P}\left(\sup_{0\le t\le T} \phi(Z(t)) \ge x_{k,r} \mid \sup_{0\le t\le T} \phi(Z(t)) \ge S_k\right)$$

shows that $x_{k,r}$ is also the $\big(1-P_k\times(1-r)\big)$-quantile of F.
Finally, the convergence

$$\widehat{F}_k^{l_N}=\widehat{P}_k\times\frac{N-l_N}{N}\xrightarrow[N\to+\infty]{a.s.}F(x_{k,r})$$

is shown at corollary 4.1. □

4.6 Discussion

4.6.1 Comparison with existing adaptive splitting algorithms

The proposed adaptive algorithm 4.1 has several advantages on the adaptive algorithms proposed in [Garvels, 2000] and [Cérou and Guyader, 2007]. In particular,

- it gives an approximation of the distribution of the entrance time and entrance state into the intermediate regions, in particular into the rare event set. Determining approximations of μ_k is necessary if we are interested in the average time the process stay in the rare event set. This is a crucial issue for example in [Wadman et al., 2013]. The authors address the question of estimating the average time a solar cell is exposed to a high quantity of sun,
- it gives approximations of the distributions γ_k and η_k defined in (3.2) and (3.4),
- it can handle the cases T random *and* T deterministic,
- it can handle the case of time non-homogeneous $\{X(t),\ t\geq 0\}$ process,
- the bias due to the fact that the conditional probability value is fixed in [Garvels, 2000, Cérou and Guyader, 2007] disappears in algorithm 4.1 since the conditional probability value is re-estimated at each threshold.
- at the end of the algorithm, we have a set of trajectories that are approximately sampled conditionally to the rare event probability B is reached. Such an approximation is no longer possible in [Garvels, 2000] and [Cérou and Guyader, 2007].

The multidimensional adaptive algorithm evoked in [Cérou et al., 2006a] proposes to resample the trajectories where their initial starting point is randomly chosen amongst the highest maxima of the trajectories sampled at each the previous stage. Hence, this algorithm gets rid of the intermediate regions. It requires the trajectories to be sampled up to final time T. In algorithm 4.1, the total computation time can be highly reduced if we consider a number N' of trajectories to estimate a threshold which is much smaller than the number N to estimate the conditional probability and the entrance distribution. A good rule of thumb is to consider N' equals $N/10$ or even lower. Consequently, only a number N' plus a fraction of N trajectories are sampled until final time T with algorithm 4.1, whereas in [Cérou et al., 2006a], as well as in [Garvels, 2000, Cérou and Guyader, 2007], all the N trajectories until final time T. Notice that a bias can also appear in [Cérou et al., 2006a] for the same reasons mentioned above.

As far as the threshold selection question is addressed in [Wadman et al., 2013], our method is more efficient in the following way. We can imagine a scenario where, for a given constant conditional probability value p between each consecutive threshold, the first thresholds are highly spaced and the distance between the last ones become smaller and smaller. In that case,

the initial set of fixed threshold required by the method of [Wadman et al., 2013] cannot estimate with sufficient reliability the true ones. The algorithm may even stop.

Independently from the author of the present manuscript and the people who have worked with him, the same algorithm has been derived in a more physical context in [Kratzer et al., 2013]. The overlapping of these two studies has nevertheless some advantages. First, it shows an other field of application which confirm the interest in studying this algorithm. Second, we addressed here theoretical results and we extended the approach of [Kratzer et al., 2013] computing extreme quantiles.

4.6.2 Reducing the computation time

As already discussed in section 4.6.1 we can consider two different sizes of trajectories. However, according to our experiment, if we are interested in estimating the extreme quantile with algorithm 4.2 or 4.3, to consider a bigger set of trajectories for the threshold estimation and a smaller one for conditional probability value and the entrance distribution gives a higher variance in the produced estimated quantile. This is due to the fact that the diversity is impoverished.

For further reductions of the total computation time in algorithms 4.1, 4.2 and 4.3 it is possible to use some truncation techniques as the ones detailed in [L'Ecuyer et al., 2006]. It consists in stopping the evolution of a trajectory when the sample path drops a fixed number of thresholds below the threshold where it originated from. This method can be especially relevant if T is infinite, since we have to sample trajectories over time T to estimate thresholds.

4.7 Numerical experiments

We test our algorithms on the standard one-dimensional Brownian bridge denoted by $\{X_t,\, 0 \leq t \geq 1\}$. It is a time non-homogeneous process and it solves the following stochastic differential equation

$$\mathrm{d}X_t = \frac{X_t}{t-1}\mathrm{d}t + \mathrm{d}W_t, \quad \text{with } X_0 = 0.$$

We can show that $X_1 = 0$ and

$$P_S = \mathbb{P}(T_S \leq 1) = \mathbb{P}(X_t \geq S, \text{ for some } t \leq 1) = \exp(-2S^2). \tag{4.38}$$

The sampled paths of the Brownian bridge are obtained with the Euler scheme [Kloeden and Platen, 1992, chap 9].

First, we investigate the influence of the conditional probability value p in algorithm 4.1. More precisely, we make p varying into the real interval $[0.1, 0.9]$, and we compute the time relative variance product (see section 3.1) for the estimation of $\mathbb{P}(T_S \leq 1)$ with $S = 3.1$. With equation (4.38) we get that

$$\mathbb{P}(T_S \leq 1) \approx 4.50.10^{-9}.$$

Figure 4.1 presents the summary of the estimates for every $p \in [0.1, 0.9]$. We set a number $N = 1000$ of trajectories to estimate the conditional probability values, the entrance distributions into each region and the thresholds. We compute also the time relative variance product for different numbers N' of trajectories to estimate the thresholds and a set of $N = 1000$ trajectories to estimate the conditional probability values and the entrance distributions into the

created regions, as discussed in sections 4.6.1 and 4.6.2. Each estimation is performed over 50 retrials.
We observe that the time relative variance product increases with the number of thresholds, that is to say with the conditional probability value p. This is mainly due to the fact that the number of re-sampling step increases. We remark also that the time relative variance product is smaller for the small values of N'. This is not surprising, since the number of trajectories that have to be sampled up to final T decreases with N', as discussed in section 4.6.1. In all cases, algorithm 4.1 is not sensitive to the value of p, provided it is lower that 0.5.

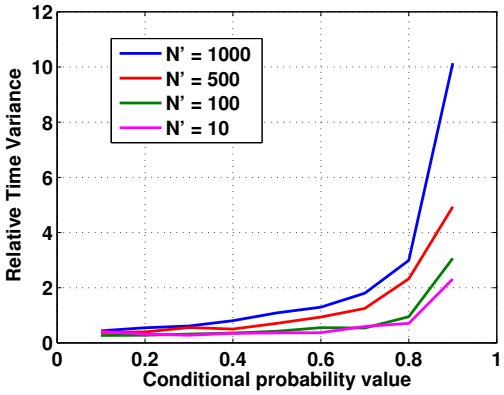
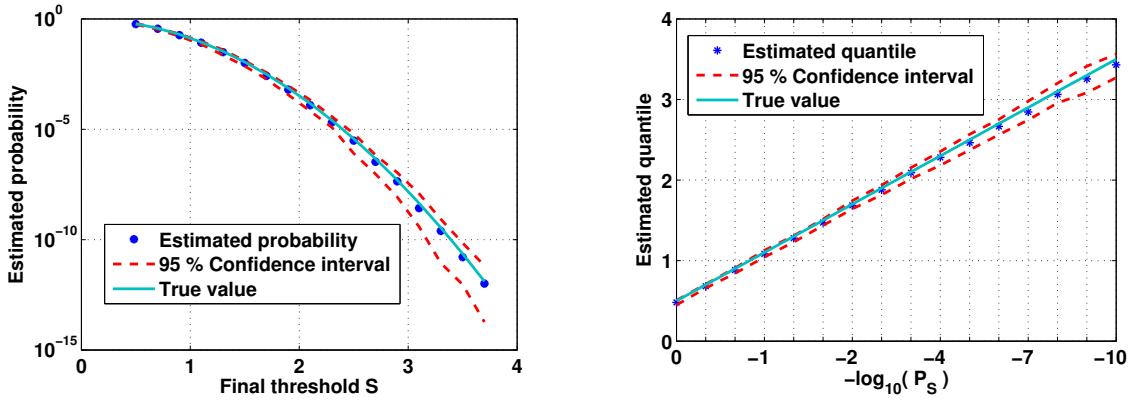


Figure 4.1: Estimation of the bias with the corrected and standard regions.

We set $p = 0.5$, $N = 1000$ and $N' = 100$. We present in figure 4.2.a) the results of the estimation of (4.38) as a function of S with algorithm 4.1. In figure 4.2.b) we show the results of the estimation of the extreme quantile q_α such as

$$\mathbb{P}(T_{q_\alpha} \leq 1) = \alpha$$

as a function of α with algorithm 4.2. Each estimation is performed over 50 retrials.



a) b)

Figures 4.2: Estimation of the rare event probability and extreme quantile with algorithms 4.1 and 4.2.

We set again $p = 0.5$, $N = 1000$ and $N' = 100$. In figure 4.2 we present the estimation of the whole survival function, defined by

$$S \mapsto P_S = \mathbb{P}(X_t \geq S, \text{ for some } 0 \leq t \leq 1).$$

To see how precise is the estimation for large value of S, we plot the $\log_{10}$ of the estimate. The important point here is that the whole curve of 4.2 was obtained with only one run of the adaptive splitting algorithm. It gives all the rare event probability values for any given threshold S. We also obtain the value $\mathfrak{q}_\alpha$ for which $\mathbb{P}(X_t \geq \mathfrak{q}_\alpha \text{ for some } t \leq 1) = \alpha$ for any given α value.

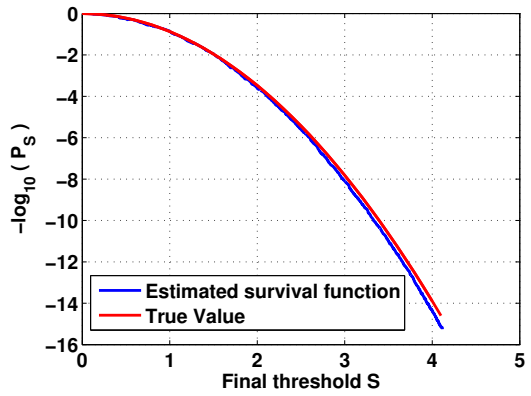


Figure 4.2: Estimation of the whole survival function with algorithm 4.3.

Conclusions of the chapter

Contributions

- Elaboration of an adaptive splitting algorithm for the estimation of a rare event that can occur during the evolution of a Markov process up to any time horizon (random or deterministic), provided the rare event probability set is defined as a threshold exceedance of a real valued function,
- Extension of the proposed adaptive splitting algorithm in order to estimate extreme quantile of the law of the supremum of the Markov process up to final evolution time,
- Extension of the proposed adaptive splitting algorithm to compute the whole survival function in only one run of the algorithm,
- Validation of the proposed algorithms with almost sure convergence results for the approximation of the entrance distribution into the intermediate regions, the estimated rare event probability and the estimated extreme quantile.

Limitations and links with the next chapter

- If we look very carefully at the plots 4.2.a), 4.2.b) and 4.2 we observe that the estimated quantities are slightly biased. This is due to the discretization of the path of the process during the implementation of the algorithm. We discuss this point in the next chapter.
- The created intermediate regions are defined in a non-optimal way. The problem of the estimation of the optimal regions (with regards to a minimization of the variance in the estimation) is addressed in the next chapter.

CHAPTER 5

Reduction of the bias and the variance in the splitting algorithm

In this chapter

- We highlight a bias induce by the discretization of the sample paths the splitting algorithm.
- We correct this bias using a deformation of the intermediate regions.
- We propose two estimation methods of the optimal regions in the splitting algorithm. One is based on a partial differential equation approach, the other one on the discretization of the state space of the process.

5.1 Introduction

We consider a Markov process $\{X(t),\ t \geq 0\}$ with càdlàg trajectories and state space E. Given a set B and a stopping time T we are interested in the estimation of the probability

$$\mathbb{P}(X(t) \in B,\ \text{for some } t \leq T) \tag{5.1}$$

with the splitting algorithm 3.1. We address here two distinct topics on the precision improvement in the estimation of the probability (5.1) with the splitting algorithm 3.1. The first one looks at the bias and the second one investigates variance reductions with the optimal regions defined in section 3.2.4.

In a first part, we show that for a process $\{X(t),\ t \geq 0\}$ with continuous state space, the practical implementation of the splitting algorithm may be biased, although it is not the case from a theoretical point of view. This is due to the discretization of the trajectories during the implementation of the splitting algorithm. This problem was quickly raised in [Cérou and Guyader, 2007], but no correction procedure has been given. In the first section, we highlight this bias and propose a correction procedure based on some deformation of the intermediate region of the splitting algorithm.

In a second part, we are interested in optimal regions that can (and should) be used in the splitting algorithm. It is shown that choosing the intermediate regions required by the splitting algorithm in a very particular way can reduce the variance in the estimation. See sections 1.2.2 for a state of the art of the optimal region and 3.2.4 for a formal description. The gain using the optimal regions in the case of a process with continuous trajectories has not been studied in the scientific literature. For a simple case, we propose two methods to estimate the optimal regions. The first one is based on a partial differential equation characterisation of the optimal regions. The second one uses a grid of the product space of the time axis and the state space of the process. In that case, the optimal regions are somehow recursively known starting from the nearest point of the grid to the rare event probability set.

5.2 Highlighting and correction of a bias in the splitting algorithm

5.2.1 Statement of the problem

The standard splitting algorithm (3.1) is theoretically unbiased. This result can be found in [Cérou et al., 2006b] and can be readily adapted to the framework of this thesis given in section 3.2. But in practice, it may be possible that the trajectories of the process $\{X(t),\ t \geq 0\}$ are known only at some discrete instant time $X(t_1),\dots,X(t_j),\dots$ and the entrance into a intermediate region can occur between two consecutive discrete times. The probability of interest can thus be underestimated. Although this problem is well known for some Monte Carlo estimation [Baldi, 1995, Gobet and Menozzi, 2010], it has never been highlighted in the use of the splitting algorithm. Here, we analyse and give a correction procedure for this bias.

5.2.2 Analysis of a simple case

5.2.2.1 The reachability problem with a discretized process

To estimate with simple Monte Carlo the probability $\mathbb{P}(\exists t \leq T,\ X(t) \in B)$, we proceed as follows. A number N of independent copies of trajectories of $X(t)$ are sampled up to final time T. We denote by $t \mapsto X^1(t),\dots,t \mapsto X^N(t)$ these trajectories and the probability of interest is estimated by

$$\mathbb{P}(\exists t \leq T,\ X(t) \in B) \approx \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{\exists t \leq T,\ X^i(t) \in B}.$$

But due to the discretization of the process during its implementation, the observed events are not $\{\exists t \leq T,\ X(t) \in B\}$ but $\{\exists j \in \{1,,\dots,,J\},\ X(t_j) \in B\}$, for a certain $J \in \mathbb{N}$ that can be random if T is random. Depending on the process $\{X(t),\ t \geq 0\}$ the following proper inclusion may hold

$$\left\{\exists j \in \{1,,\dots,,J\},\ X(t_j) \in B\right\} \subsetneq \{\exists t \leq T,\ X(t) \in B\},$$

and a bias appears in the Monte Carlo estimation

$$\mathbb{E}\left(\frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{\exists j \in \{1,,\dots,,J\},\ X^i(t_j) \in B}\right) = \mathbb{P}(\exists j \in \{1,,\dots,,J\},\ X(t_j) \in B) < \mathbb{P}(\exists t \leq T,\ X(t) \in B). \tag{5.2}$$

Remark 5.1. *We may interpret the inequality (5.2) as follows. Even asymptotically, or with a huge number N of trajectories, the Monte Carlo estimation of the probability $\mathbb{P}(\exists t \leq T,\ X(t) \in B)$ can be biased.*

Remark 5.2. *The inequality (5.2) is not longer correct if we consider for example pure jump process. In that case, the knowledge of $X(t_1),\dots,X(t_J)$ can imply the knowledge of $X(t)$ for all $t \in [0,T]$, provided the t_j are well chosen.*

The figure (5.1) sums up these ideas is the case $B = [S,+\infty)$.

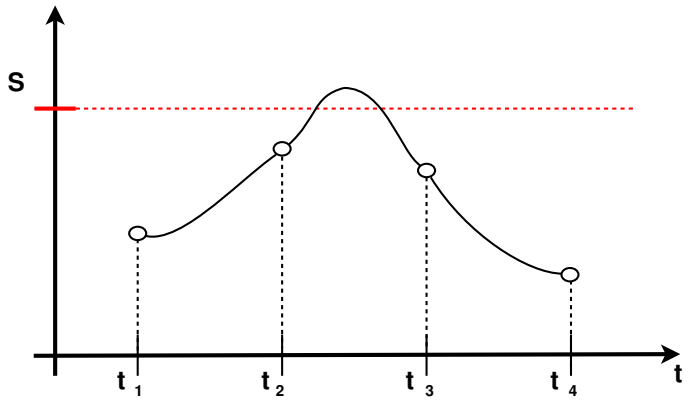


Figure 5.1: The discretization of the process misses the set $B = [S, +\infty)$.

5.2.2.2 Ideas for a correction of the bias

To correct this bias, we propose to use the deformation of the set B derived in [Gobet and Menozzi, 2010]. It consists in considering some supersets of B instead of B itself to compensate the underestimation of the probability. At each time t, these supersets of B should depend on the underlying randomness of the random variable $X(t)$. Thus, at each time step t_j we test if the sequence $\{X(t_j),\ j = 1,2,\ldots\}$ has reached a set $B'(t_j)$, with

$$B'(t_j) \supset B.$$

In a very simplified setting, the method can be understood with figure (5.2), where $B' = [S', +\infty)$ for a well chosen $S' < S$.

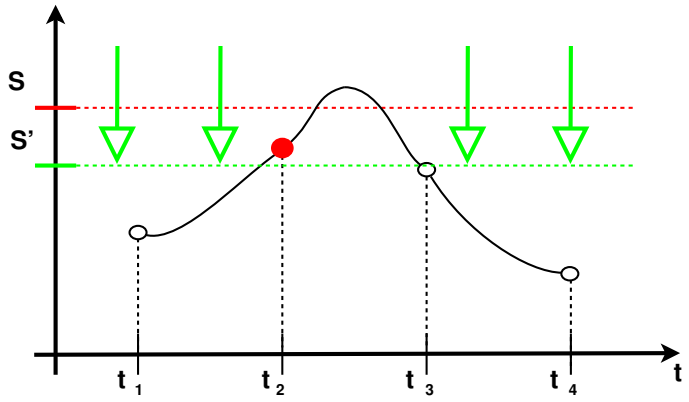


Figure 5.2: The discretization process $\{X(t),\ t \geq 0\}$ reaches the threshold S' but not S.

5.2.3 Bias reduction in the case of a d-dimensional diffusion

In this section the process $\{X(t),\ t \geq 0\}$ is assumed to be a d-dimensional diffusion process, and is thus denoted $\{X_t,\ t \geq 0\}$ as usual. Its dynamic is given by

$$\mathrm{d}X_t = b(t, X_t)\mathrm{d}t + \sigma(t, X_t)\mathrm{d}W_t, \tag{5.3}$$

where $\{W_t,\ t \geq 0\}$ is a d'-dimensional Brownian motion and $b : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d$ and $\sigma : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^{d \times d'}$ satisfy the usual Lipschitz continuity condition that ensures the existence of a unique

strong solution [Karatzas and Shreve, 1991, chap 5].
In practice, we can approximate the diffusion (5.3) by its Euler scheme [Kloeden and Platen, 1992, chap 9]. To achieve this, let δ by the time step and $t_j = j\delta$, for $j \in \mathbb{N}$. The solution of (5.3) at time t_j is then approximated by

$$X^{\delta}_{t_{n+1}} = X^{\delta}_{t_n} + b(t_j, X^{\delta}_{t_n})\delta + \sigma(t_j, X^{\delta}_{t_n})\varepsilon_j, \tag{5.4}$$

where ε_j are *i.i.d.* standard d'-dimensional Gaussian vectors.

Assume given a subset A of $\mathbb{R}^d \times [0, +\infty)$. In this thesis we are interested to the first time the $\mathbb{R}^d \times [0, +\infty)$ valued process $\{Z(t) = (X(t), t),\ t \geq 0\}$ enters a subset A of $\mathbb{R}^d \times [0, +\infty)$. The authors of the bias correction procedure we are based on [Gobet and Menozzi, 2010] are interested in the first time the process $\{X(t),\ t \geq 0\}$ reaches a set $A(t)$. The first observation is that a subset A of $\mathbb{R}^d \times [0 + \infty)$ can be decomposed as follows

$$A = \bigcup_{t \geq 0} A(t) \times \{t\},$$

where the sections are simply defined with $A(t) = \{x \in \mathbb{R}^d, (x, t) \in A\}$. It is clear that $(x, t) \in A$ if, and only if, $x \in A(t)$ and the point of view of the manuscript of checking the first time the process $\{Z(t),\ t \geq 0\}$ reaches A is the same that the point of view of checking the first time the process $\{X(t),\ t \geq 0\}$ reaches $A(t)$.

In [Gobet and Menozzi, 2010], the author proposes to use a superset of $A(t)$ instead of $A(t)$ to cancel the bias of the Monte Carlo estimation. It depends on the time step δ and is denoted in that case by $A^{\delta}(t)$. It is defined as follows

$$A^{\delta}(t) = \left\{x \in \mathbb{R}^d,\ \mathrm{d}(x, A(t)) \leq c_0\sqrt{\delta}|n(x, t) * \sigma(t, x)|\right\}, \tag{5.5}$$

where d is the Euclidean distance, c_0 is a constant and $n(x, t)$ is the inward normal vector at the closest point of x on the boundary of $A(t)$. The notation $n(x, t)*$ stands for the transpose of a matrix $n(x, t)$ and $|n(x, t) * \sigma(t, x)|$ is the Euclidean norm of the row vector $n(x, t) * \sigma(t, x)$. The constant c_0 equals to

$$c_0 = -\xi(1/2)/\sqrt{2\pi} \approx 0.5826. \tag{5.6}$$

To reduce the bias, we have thus to use the Monte Carlo estimation method with $A^{\delta}(t)$ instead of $A(t)$.

5.2.4 Application of the bias correction in the splitting algorithm

In the splitting algorithm, we estimate several reachability probabilities. The more of reachability probabilities we estimate, the greater the bias can be. We propose to use the method derived in section 5.2.2 to correct this bias. The framework is the following. The process under consideration is the strong solution of the stochastic differential equation given in equation (5.3). The solution is approximated in the time t_j with its Euler scheme as in (5.4). Given a deterministic or random final time horizon T and a subset B or $\mathbb{R}^d$, we are interested in the estimation of

$$\mathbb{P}(T_B \leq T), \quad \text{with } T_B = \inf\{t,\ X_t \in B\},$$

with the splitting algorithm 3.1.

We assume given the supersets

$$A_k,\ k = 1, \ldots, m$$

of $B \times [0, +\infty)$ required by the splitting algorithm. The sequence $\{A_k,\ k = 1, \dots, m\}$ can for example be constructed with algorithm 4.1. We recall that the process $\{Z_t,\ t \geq 0\}$ is defined by

$$Z_t = (X_t, t),$$

and we define accordingly the discrete time Markov chain $\{Z^\delta(t_j)\ j = 1, 2, \dots\}$ by

$$Z^\delta(t_j) = (X^\delta_{t_j}, t_j).$$

To deal with the discrete time Markov chain $\{Z^\delta(t_j)\ j = 1, 2, \dots\}$ we define the sequence of discrete stopping times $\tau_k, k = 1, \dots, m$ as follows

$$\tau_1 = \min\left\{t_j,\ Z^\delta(t_j) \in A_1\right\} = \inf\left\{t_j,\ X^\delta_{t_j} \in A_1(t_j)\right\},$$

and for $k \geq 2$,

$$\tau_k = \min\left\{t_j \geq \tau_{k-1},\ Z^\delta(t_j) \in A_k\right\} = \inf\left\{t_j \geq \tau_{k-1},\ X^\delta_{t_j} \in A_k(t_j)\right\},$$

where the section $A_k(t)$ are defined with

$$A_k(t) = \{x \in \mathbb{R}^d,\ (x, t) \in A_k\}$$

for all $t \geq 0$ and not only for the discrete time t_j. Notice also that the relation

$$A_1(t) \supset \dots \supset A_m(t)$$

is still valid for all $t \geq 0$. Indeed, if $x \in A_{k-1}(t)$, then $(x, t) \in A_{k-1} \supset A_k$ and then $x \in A_k(t)$.

In the same way, we define τ, the final horizon time for the evolution of the discrete time Markov chain $Z^\delta(t_j)$, with

$$\tau = \max\{t_j,\ t_j \leq T\}.$$

Remark that for $k = 1, \dots, m$ we have $\tau_k \in \{j\delta,\ j \in \mathbb{N}\}$ and that τ can be random if T is random.

The implementation of the splitting algorithm 3.1 gives actually an estimation of the events

$$\mathbb{P}\left(\exists t_j \in \{\tau_{k-1}, \dots, \tau\},\ Z^\delta(t_j) \in A_k | \tau_{k-1} \leq \tau\right),$$

instead of $\mathbb{P}(\exists t,\ T_{k-1} \leq t \leq T_k,\ Z_t \in A_k | T_{k-1} \leq T)$, with $T_k = \inf\{t,\ Z_t \in A_k\}$. Accordingly, the trajectories of the Markov chain $\{Z^\delta(t_j),\ j = 1, 2, \dots\}$ are sampled up to time $\tau_k \wedge \tau$ at each iteration of the algorithm. As explained in section 5.2.2, this can bias the estimation. The bias can be corrected using the sequence τ^δ_k, for $k = 1, \dots, m$ of stopping times defined for $k = 1$ with

$$\tau^\delta_1 = \min\left\{t_j \geq 0,\ X^\delta(t_j) \in A^\delta_1(t_j)\right\},$$

and for $k \geq 2$ with

$$\tau^\delta_k = \min\left\{t_j \geq \tau^\delta_{k-1},\ X^\delta(t_j) \in A^\delta_k(t_j)\right\},$$

where the sets $A^\delta_k(t)$ are defined with equation (5.5) by

$$A^\delta_k(t) = \left\{x \in \mathbb{R}^d,\ \mathrm{d}(x, A_k(t)) \leq c_0\sqrt{\delta}|n(x, t) * \sigma(t, x)|\right\}. \tag{5.7}$$

Notice that the relation

$$A^\delta_1(t) \supset \dots \supset A^\delta_m(t)$$

is not necessarily correct since the vector $n(x,t)$ is computed in a point that depends on x but also on the considered set.

To put the above described procedure in a nutshell, to correct the bias in the splitting algorithm the trajectories of the discrete-time Markov $\{Z^\delta(t_j),\ j=1,2,\ldots\}$ chain are stopped at time τ_k^δ instead of τ_k.

The splitting algorithm with correction of the bias is detailed in algorithm 5.1. Notice that τ may depend on the considered trajectory but is still denoted by τ not to overload the writing.

Algorithm 5.1: The splitting algorithm with bias correction

Data: A sequence $A_1 \subset \cdots \subset A_m$, a number N of trajectories to estimate conditional probabilities, a time step δ in the Euler scheme

Result: $\widehat{p}_B^\delta \approx \mathbb{P}(T_B \leq T)$

Initialization

  for $i = 1,\ldots,N$ **do**

    Set $\tau_0^{\delta,i} = 0$ and sample $Z_0^{\delta,i}$ independently from law π_0

    Set $e_0^i = Z_0^i$

  Set $I_0 = \{1,2,\ldots,N\}$

for $k = 1,\ldots,m-1$ **do**

  */*Conditional entrance distribution and conditional probability estimations*/*

  for $i = 1,\ldots,N$ **do**

    Choose randomly and uniformly a subscript $j_i \in J_k$

    Sample a trajectory of the Markov chain $\{Z^\delta(t_j),\ j=1,2,\ldots\}$ starting from $Z^{\delta,j_i}(\tau^{\delta,j_i})$ at time τ^{δ,j_i} until time $\tau_k^{\delta,j_i} \wedge \tau$, where

$$\tau_k^{\delta,i} = \min\left\{t_j,\ X^{\delta,j_i}(t_j) \in A_k^\delta(t_j)\right\}$$

    Set $e_{k+1}^i = Z^{\delta,j_i}\left(\tau_k^{\delta,j_i} \wedge \tau\right)$

  Set $I_{k+1} = \left\{i,\ (\tau_k^{\delta,i} \wedge \tau) < \tau\right\}$ and

$$\widehat{p}_{k+1}^\delta = \frac{|I_{k+1}|}{N}$$

Estimation

$$\mathbb{P}(T_B \leq T) \approx \prod_{l=1}^{m} \widehat{p}_l^\delta$$

5.2.5 The case of an importance function

Here we suppose that the sets A_k are defined with a real valued function ϕ as follows

$$A_k = \left\{(x,t) \in \mathbb{R}^d \times \mathbb{R}^+,\ \phi(x,t) \geq S_k\right\}$$

for a well chosen sequence of thresholds S_k. The sections and the deformed sections are defined with

$$A_k(t) = \left\{x \in \mathbb{R}^d,\ \phi(x,t) \geq S_k\right\} \tag{5.8}$$

and

$$A_k^\delta(t) = \left\{x \in \mathbb{R}^d,\ \mathrm{d}(x, A_k(t)) \leq c_0\sqrt{\delta}|n(x,t) * \sigma(t,x)|\right\}$$

respectively.

We investigate the existence of some thresholds $S_k^\delta \leq S_k$ such as $A_k^\delta(t)$ can be simply written with

$$A_k^\delta(t) = \left\{x \in \mathbb{R}^d,\ \phi(x,t) \geq S_k^\delta\right\}. \tag{5.9}$$

If such a sequence of thresholds exists, the correction of the bias is easily performed. Indeed, we have to replace the sequence S_k by S_k^δ in the splitting algorithm 3.1. Otherwise, we have to check for each iteration time $j\delta$ whether the random variable $Z^\delta(t_j)$ is in $A_k^\delta(t_j)$, defined in equation (5.7).

5.2.5.1 The one dimensional case

We assume to be in the simple case where $d = 1$ and $\sigma(t,x) = \sigma_0$ is a constant function. The evolution of $\{X_t,\ t \geq 0\}$ stands thus as follows

$$\mathrm{d}X_t = b(t, X_t)\mathrm{d}t + \sigma_0 \mathrm{d}W_t, \tag{5.10}$$

where $\{W_t,\ t \geq 0\}$ is a one-dimensional standard Brownian motion. We assume also that

$$\phi(x,t) = x$$

for all $x \in \mathbb{R}$ and we have that the region $A_k(t)$ defined in equation 5.8 equals to

$$A_k(t) = \left\{x \in \mathbb{R},\ \phi(x,t) \geq S_k\right\} = [S_k, +\infty).$$

We show here the existence of a threshold S_k^δ that defines a time-independent superset A_k^δ as in equation (5.9). It is sufficient to remark that, for every set A_k, the inward normal vector $n(x,t)$ is indeed a scalar for every (x,t) equals to

$$n(x,t) = -1.$$

Hence,

$$|n(x,t) * \sigma(t,x)| = |-\sigma_0| = \sigma_0.$$

Consequently,

$$A_k^\delta(t) = \left\{x \in \mathbb{R},\ \mathrm{d}(x, A_k(t)) \leq c_0\sqrt{\delta}\sigma_0\right\},$$

where the distance $\mathrm{d}(x, A_k(t))$ equals to

$$\mathrm{d}(x, A_k(t)) = \inf\left\{|x-y|,\ \text{for } y \geq S_k\right\} = (S_k - x)^+$$

and we obtain

$$A_k^\delta(t) = \left\{x \in \mathbb{R},\ (S_k - x)_+ \leq c_0\sqrt{\delta}\sigma_0\right\}.$$

We are thus in the case illustrated in figure 5.2. Hence, we propose to consider S_k^δ instead of S_k to correct the bias in the splitting algorithm with

$$S_k^\delta = S_k - c_0\sqrt{\delta}\sigma_0$$

and

$$\tau_k^\delta = \inf\left\{t \geq 0,\ X^\delta(t_j) \geq S_k^\delta\right\}.$$

5.2.5.2 The multidimensional case

Unfortunately, results of section 5.2.5.1 are no longer true in a more general setting. We show a counterexample. Let $\{X(t),\ t \geq 0\}$ be the 2-dimensional standard Brownian motion

$$X_t = W_t,$$

with $X_0 = 0$. We set

$$\phi(x,t) = \phi((x_1,x_2),t) = |x_1| - x_2^2$$

and

$$\phi_0(x) = \phi(x,t).$$

We are interested in estimating the probability

$$\mathbb{P}\left(\sup_{0\leq t\leq 1} \phi_0(X_t) \geq 1\right),$$

with Monte Carlo. It can be viewed as the setting of the splitting algorithm with only one threshold. Accordingly, we define the sets A and $A(t)$ with

$$A = \left\{(x,t) \in \mathbb{R}^2 \times \mathbb{R}^+,\ \phi_0(x) \geq 1\right\},$$

and

$$A(t) = \left\{x \in \mathbb{R}^2,\ \phi_0(x) \geq 1\right\}.$$

Even though $A(t)$ does not depend on t, we keep the notation of the present chapter for the sake of consistency.

Now, we proof by contradiction that a threshold S^δ such that

$$A^\delta(t) = \{x \in \mathbb{R}^2,\ \phi_0(x) \geq S^\delta\} \tag{5.11}$$

does not exist. We assume that such a threshold S^δ exists.

Remarking that $|n(x,t)| = 1$, we have with equation (5.5) that

$$A^\delta(t) = \left\{x,\ \mathrm{d}(x,A(t)) \leq c_0\sqrt{\delta}\right\}.$$

If we consider $x' = (1 - c_0\sqrt{\delta}, 0)$, we have $p(x') = (1,0)$, where p is the orthogonal projection on the set $A(t)$ and $\mathrm{d}(x', A(t)) = c_0\sqrt{\delta}$. Consequently $x' \in A^\delta(t)$, which implies with equation (5.11) that

$$S^\delta \leq (1 - c_0\sqrt{\delta})^2. \tag{5.12}$$

On the other hand, if $x'_\varepsilon = (1 - c_0\sqrt{\delta} - \varepsilon, 0)$, for $\varepsilon > 0$, we still have $p(x'_\varepsilon) = (1,0)$, and $\mathrm{d}(x'_\varepsilon, A(t)) = c_0\sqrt{\delta} + \varepsilon$. Hence $x'_\varepsilon \notin A^\delta(t)$ and equation (5.11) gives that

$$S^\delta > (1 - c_0\sqrt{\delta} + \varepsilon)^2,\ \forall \varepsilon > 0.$$

As a result $S^\delta \geq (1 - c_0\sqrt{\delta})^2$, and equation (5.12) leads to

$$S^\delta = (1 - c_0\sqrt{\delta})^2.$$

After that, if we consider x'' such that $p(x'') = (2,1)$ and $d(x'', A(t)) = c_0\sqrt{\delta}$. We have that

$$n(t,x'') = -\frac{\nabla\phi_0(x'')}{\|\nabla\phi_0(x'')\|} = \frac{1}{\sqrt{5}}(-1,2),$$

where $\nabla\phi_0$ is the gradient vector of ϕ_0. Thus, $x'' = p(x'') + c_0\sqrt{\delta}n(t,x'')$, as shown in figure 5.3. And if we set $\delta = 0.01$, we obtain that $\phi_0(x'') \approx 0.87$ and $S^\delta \approx 0.89$. The contradiction follows since x'' was supposed to belong to $A^\delta(t) = \{x,\ \phi_0(x) \geq S^\delta\}$.

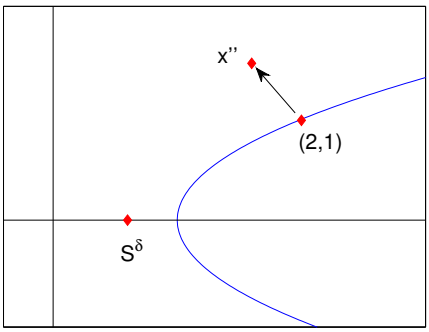


Figure 5.3: Configuration for the counterexample in the multidimensional case.

5.2.6 Numerical experiments

The procedure for the bias reduction proposed in section 5.2 is tested on the Brownian bridge $\{X_t, 0 \leq t \leq 1\}$, which is solution of the following real stochastic differential equation

$$\begin{cases} X_0 = 0 \\ \mathrm{d}X_t = \dfrac{X_t}{t-1}\mathrm{d}t + \mathrm{d}W_t. \end{cases} \tag{5.13}$$

We can show that $X_1 = 0$ almost surely and that

$$\mathbb{P}(T_S \leq 1) = \mathbb{P}(\sup_{0\leq t\leq 1} X_t \geq S) = \exp^{-2S^2}. \tag{5.14}$$

The sampled paths of the Brownian bridge are obtained with the Euler scheme [Kloeden and Platen, 1992, chap 9].

The nested regions required by the splitting algorithm are time-independent here and defined by $A_k = \{(x,t) \in \mathbb{R}\times\mathbb{R}^+,\ x > S_k\}$. The sequence S_k is chosen with equation (5.14) such that $\mathbb{P}(T_k \leq 1 | T_{k-1} \leq 1) = 0.5$. Notice that sequence could be empirically given using the adaptive splitting algorithm 4.1.

For a time step δ, the solution of (5.13) is approximated at each $t_j = j\delta$ by its Euler scheme with

$$X^\delta_{t_{j+1}} = X^\delta_{t_j} + X^\delta_{t_j}/(t_j - 1)\delta + \sqrt{\delta}\epsilon_j,$$

where ϵ_j are *i.i.d* centered Gaussian random variables with variance 1.

We are here in the framework of section 5.2.5.1. Consequently, to reduce the bias, we use the sequence $S^\delta_k,\ k = 1,\ldots,m$ instead of the sequence $S_k,\ k = 1,\ldots,m$ defined by

$$S_k - c_0\sqrt{\delta},$$

where c_0 is defined in (5.6). Then, the corrected regions used in algorithm 5.1 are given with $A^\delta_k(t) = \{x,\ x \geq S^\delta_k\}$ and do not depend on time.

Final threshold S		2	2.6	3.2	3.8	5
Theoretical probability		3.35e-4	1.34e-6	1.28e-9	2.90e-13	1.93e-22
Standard thresholds, $\delta = 0.01$	Bias	52%	65%	99%	202%	150%
	Run time (sec.)	0.11	0.18	0.27	0.38	0.65
Standard thresholds, $\delta = 0.001$	Bias	15%	23%	24%	28%	80%
	Run time (sec.)	0.35	0.55	0.80	1.1	1.9
Standard thresholds, $\delta = 0.0001$	Bias	2%	6%	12%	19%	12%
	Run time (sec.)	2.7	4.3	6.2	8.5	14.1
Modified thresholds, $\delta = 0.01$	Bias	3%	8%	8%	19%	22%
	Run time (sec.)	0.12	0.20	0.30	0.42	0.72

Table 5.1: Bias and run time for standard splitting and splitting with corrected thresholds.

For any H-sample $(\widehat{P}_1,\dots,\widehat{P}_H)$ of an estimation of a probability P, the estimated bias $b(\widehat{P}_1,\dots,\widehat{P}_H)$ is defined here as the average relative distance between the estimated and true value as follows

$$b(\widehat{P}_1,\dots,\widehat{P}_H) = \frac{|P - \frac{1}{H}\sum_{i=1}^{H}\widehat{P}_i|}{\frac{1}{H}\sum_{i=1}^{N}\widehat{P}_i}.$$

Each estimation is performed over $H = 200$ retrials.

Some bias computations are presented in table 5.1 and figure 5.4 presents the bias as a function of the final threshold S. We observe that the bias using the modified thresholds with a time step of 0.01 is of the same order of magnitude than the standard splitting algorithm with a time step of 0.0001. Moreover, for each threshold between $S = 2$ and $S = 5$ the run time is 20 times lower using the time step $\delta = 0.01$ with modified regions and $\delta = 0.0001$ with standard regions.

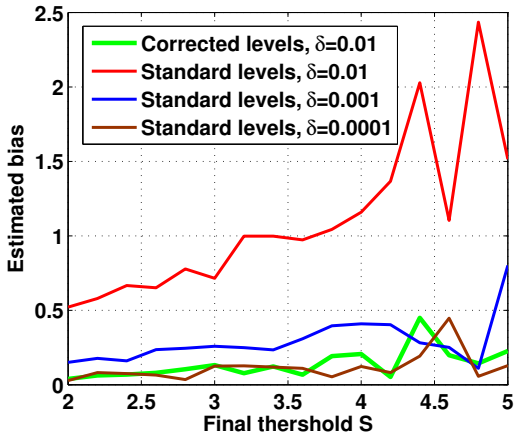


Figure 5.4: Estimation of the bias with the corrected and standard regions.

In figure 5.5 we compare the run time of the splitting algorithm 3.1 and the splitting algorithm with corrected intermediates region 5.1. The computation time is the same for the splitting algorithm with standard and corrected regions, both if they have the same time step $\delta = 0.01$. Furthermore, it is worth noting that as it can be seen in figure 5.6, changing the time

step does not affect the relative standard deviation.

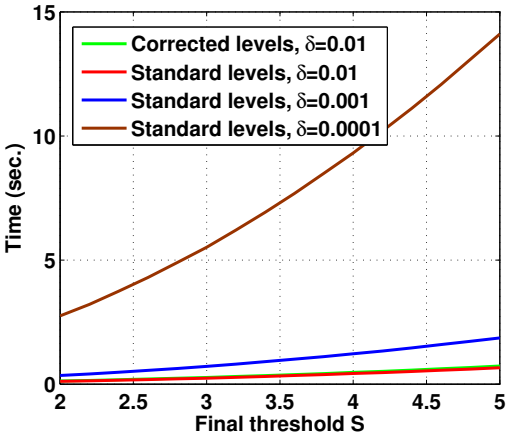


Figure 5.5: Estimation of the computation time in the estimation of (5.14) with the corrected and standard regions.

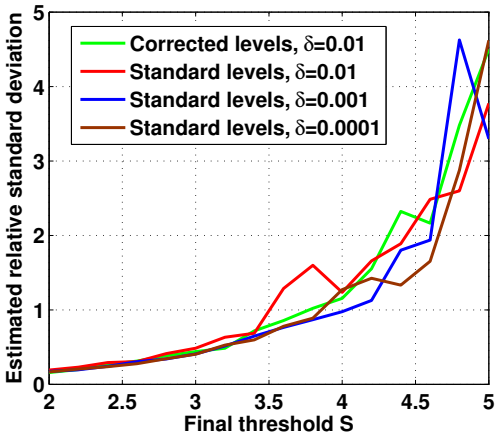


Figure 5.6: Estimation of the relative standard deviation in the estimation of (5.14) with the corrected and standard regions.

As a conclusion, using S^δ and the algorithm with corrected intermediate regions 5.1 instead of S and standard splitting algorithm 3.1 reduces the bias for every threshold S with the same computation time and the variance is not impacted.

5.3 Variance minimisation via the optimal regions

In this section, the final horizon time T is deterministic. Denoting $T_B = \inf\{t \geq 0,\ X_t \in B\}$, the probability of interest is rewritten as follows

$$\mathbb{P}(T_B \leq T). \tag{5.15}$$

We are looking here for methods to estimate the optimal regions that can be used in the splitting algorithm. The scientific literature related to the estimation of optimal regions in the

splitting algorithm is focused in the two papers [Garvels et al., 2002] and [Dean and Dupuis, 2009]. The first one gives a method in the case of a discrete state space for the process $\{X(t),\ t \geq 0\}$, with random final time T. In the second article, the problem of finding optimal regions is replaced by solving a variational equation.

The first method is not tractable in the case of continuous state space and deterministic final time T. As far as the second method is concerned, solving a variational equation can face delicate computation problems. Moreover, the lack of comparison of the variance between the use of optimal and non optimal regions in [Dean and Dupuis, 2009] makes the proposed approach performances impossible to evaluate .

We propose here two methods to estimate and use the optimal regions in the splitting algorithm in the case of a deterministic final time T. The first one is based on a partial differential equation, and the second one uses a grid of the state space to give estimates of some reachability probability in a sequential manner.

5.3.1 Description of the optimal regions

As reminded in section 3.2, the asymptotic theoretical variance in the estimation of the rare event probability (5.15) with the splitting algorithm is minimal if the boundaries of the nested regions $A_k \subset E \times \mathbb{R}^+$, for $k = 1, \ldots, m$ are chosen equal to A_k^* with

$$\left\{(x,s) \in E \times \mathbb{R}^+,\ \mathbb{P}(X_t \in B,\ \text{for some } s \leq t \leq T | X(s) = x) = c_k\right\}, \tag{5.16}$$

for an increasing sequence of real numbers $c_k \in (0,1)$. In that case, the variance takes the form (see section 3.2.4)

$$V_m^* = \sum_{k=1}^{m} (1/p_k - 1).$$

Remark 5.3. *The asymptotic variance using the regions characterized with equation (5.16) depends only on the sequence of conditional probability values. In particular, it does not depend on the dimension of the state space of the process.*

Remark 5.4. *The quantity V_m^* is the lower bound for the asymptotic variance in the case of the splitting algorithm.*

The knowledge of the optimal regions (5.16) implies the determination of

$$\mathbb{P}(X_t \in B,\ \text{for some } s \leq t \leq T | X(s) = x)$$

for all $s \geq 0$ and for all $x \in E$. These quantities are unknown. If we have sharp estimates of the regions A_k^*, we can have an estimation of the probability we want to estimate with the splitting algorithm $\mathbb{P}(T_B \leq T)$. Thus, the simulation budget for the estimation of A_k^* should not be to large, otherwise we could lose the benefit of estimate them. We would rather use more trajectories with non optimal regions, as for example those constructed with the adaptive algorithm 4.1. Nevertheless, even rough estimates of the optimal regions A_k^* could lead to an important reduction of the variance in the estimation of the probability (5.15). Thus, we propose hereafter two methods for the estimation of the optimal regions in the framework of the present section.

5.3.2 Estimation of the optimal regions

5.3.2.1 A partial differential equation approach

Here, the process $\{X(t),\ t \geq 0\}$ under consideration is assumed to be the solution of the following d-dimensional stochastic differential equation, and is thus rewritten $\{X_t,\ t \geq 0\}$. Its dynamic is given by

$$\mathrm{d}X_t = b(t, X_t)\mathrm{d}t + \sigma(t, X_t)\mathrm{d}W_t, \tag{5.17}$$

where $\{W_t,\ t \geq 0\}$ is a d'-dimensional Brownian motion and $b : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d$ and $\sigma : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^{d \times d'}$. We suppose that b and σ satisfy some conditions that ensure the almost sure existence and uniqueness of a strong solution [Karatzas and Shreve, 1991, chap 5].

As usual, we denote by $t \mapsto X_t^{x,s}$ the solution of (5.17) that starts at x at time s. We give here a characterization of the optimal regions A_k^* of equation (5.16) with a deterministic partial differential equation. Following [Lamberton, 2009, chap 4], for every $f : \mathbb{R}^d \times [0, T] \longrightarrow \mathbb{R}$ such that $f(x, t) \leq C(1 + |x|^k)$ for some positive constants C and k, we first define the function F defined on $\mathbb{R}^d \times [0, T]$ by

$$F(x, t) = \sup_{\tau \in \mathcal{T}_{t,T}} \mathbb{E}(f(X_\tau^{x,t}, \tau)),$$

where $\mathcal{T}_{t,T}$ is the set of all the stopping time τ such that

$$\mathbb{P}(\tau \in [t, T]) = 1.$$

The function F satisfies the variational inequality

$$\begin{cases} \max(DF, f - F) = 0 \\ F(., T) = f(., T), \end{cases} \tag{5.18}$$

where the operator D is defined by

$$D = \partial_t + \frac{1}{2}\sum_{i=1}^{d}\sum_{j=1}^{d} a_{i,j}(t, x)\partial_{i,j}^2 + \sum_{i=1}^{d} \mu_i(t, x)\partial_i,$$

with $a(t, x) = \sigma(t, x) \times \sigma(t, x)'$ and if we set the $\{0, 1\}$-valued function f by

$$f(x, t) = \mathbb{1}_{x \in B},$$

it follows the identity

$$\sup_{\tau \in \mathcal{T}_{t,T}} \mathbb{E}(f(X_\tau^{x,t}, \tau)) = \sup_{\tau \in \mathcal{T}_{t,T}} \mathbb{P}(X_\tau^{x,t} \in B) = \mathbb{P}(X_{\tau_B^{x,t}}^{x,t} \in B) = \mathbb{P}(\tau_B^{x,t} < T),$$

where

$$\tau_B^{x,t} = \inf\{s \geq t,\ X_s^{x,t} \in B\} \wedge T.$$

Using the particular expression of the function f, the variational inequality (5.18) is equivalent to the following partial differential equation

$$\begin{cases} DF = 0 \\ F(x, T) = \mathbb{1}_{x \in B}. \end{cases}$$

Hence, the optimal regions A_k^* are rewritten as follows

$$A_k^* = \left\{(x, t), F(x, t) = p_k\right\},$$

for some given $p_k \in [0, 1]$.
The approach can be extended to jump-diffusion models [Zhang, 1997] and diffusion models with regime switching [Khaliq and Liu, 2009].

5.3.2.2 General case

In this section $\{X(t),\ t \geq 0\}$ is a Markov process with càdlàg trajectories and T is a deterministic time and we assume its trajectories to be continuous. The process $\{X(t),\ t \geq 0\}$ is not necessary a solution of a diffusion equation. The idea is to consider a grid into the state space of the Markov process $\{Z(t),\ t \geq 0\} = \{(X(t), t),\ t \geq 0\}$, namely into the space $E \times \mathbb{R}^+$. Then, at each point of (x, s) of the grid, we estimate the probability that the process $\{X(t),\ t \geq 0\}$ starting from x at time s reaches the rare set B before time T. After that, we can either use a contouring algorithm to obtain directly from this estimates an approximation of the optimal regions A_k^*, or we can check in which region is the process using an interpolation in between the points of the grid.

For the sake of the understanding of the method, we discuss on a one dimensional case where $X(t) \in \mathbb{R}$, for all $t \geq 0$ for which we want to estimate the probability

$$\mathbb{P}(T_B \leq T), \quad \text{where } B = [S, +\infty).$$

We suppose $X(0) = 0$ and $S > 0$. After that, we divide the time interval $[0, S]$ into H sub-intervals, and the interval $[0, T]$ into G sub-intervals. This implies the existence of real numbers $x_{n_1}, n_1 = 0, \ldots, H+1$ such that $x_0 = 0 < x_1 < \cdots < x_H < x_{H+1} = S$ and $t_{n_2}, n_2 = 0, \ldots, G+1$ such that $t_0 = 0 < t_1 < \cdots < t_G < t_{G+1} = T$. Next, we set α_{n_1,n_2} the following points

$$\alpha_{n_1,n_2} = (x_{n_1}, t_{n_2}) \in \mathbb{R}^2.$$

The set $\{\alpha_{n_1,n_2},\ n_1 = 0, \ldots, H+1, , n_2 = 0, \ldots, G+1\}$ determines a grid in the $[0, S] \times [0, T]$ rectangle.

For $x \in [0, S]$ and $s \in [0, T]$, let $T_y^{x,s}$ denotes the stopping time defined by

$$T_y^{x,s} = \inf\{t \geq s,\ X(t) \geq y | X(s) = x\}.$$

For a fixed time s and two fixed point x, y such that $0 < x < y < S$, since the trajectories of the process are continuous, we have the following identity

$$\left\{T_B^{x,s} \leq T\right\} = \bigcup_{j=1}^{G} \left\{t_j \leq T_y^{x,s} \leq t_{j+1},\ T_B^{y,T_y^{x,s}} \leq T\right\}. \tag{5.19}$$

Namely, reaching the set B before time T starting at place x at time s is achieved by passing through the point y at a certain time, denoted by $T_y^{x,s}$, between t and T.

Now recall that our goal is to compute $\mathbb{P}(T_B^{x,s} \leq T)$ at each point $\alpha_{n_1,n_2} = (x_{n_1}, t_{n_2})$. With equation (5.19) we obtain that

$$\mathbb{P}(T_B^{x_{n_1},t_{n_2}} \leq T) = \sum_{j=1}^{G} \mathbb{P}\left(t_j \leq T_{x_{n_1+1}}^{x_{n_1},t_{n_2}} \leq t_{j+1},\ T_B^{x_{n_1},T_{x_{n_1+1}}^{x_{n_1+1},t_{n_2}}} \leq T\right). \tag{5.20}$$

Although this equation contains a lot of variables, its interpretation is straightforward. Indeed, the event

$$\left\{T_B^{x_{n_1},t_{n_2}} \leq T\right\} = \left\{\text{ starting at } x_{n_1} \text{ at time } t_{n_2} \text{ and reaching } B \text{ before time } T\right\}$$

equals to the event

$$\begin{aligned}\{&\text{ starting at } x_{n_1} \text{ at time } t_{n_2}, \text{ reaching } x_{n_1+1} \text{ at time } T_{x_{n_1+1}}^{x_{n_1},t_{n_2}},\\ &\qquad \text{and then reaching } B \text{ starting at } x_{n_1+1} \text{ at time } T_{x_{n_1+1}}^{x_{n_1},t_{n_2}}\},\end{aligned}$$

which is the right hand side of equation (5.20). Notice also that the time $T^{x_{n_1},t_{n_2}}_{x_{n_1+1}}$ is the first time the process $\{X(t),\ t \geq 0\}$ starting at x_{n_1} at time t_{n_2} equals to x_{n_1+1}.

After, we write that

$$\mathbb{P}(T_B^{x_{n_1},t_{n_2}} \leq T) = \sum_{j=1}^{G} \mathbb{P}\left(T_B^{x_{n_1+1},T^{x_{n_1},t_{n_2}}_{x_{n_1+1}}} \leq T | t_j \leq T^{x_{n_1},t_{n_2}}_{x_{n_1+1}} \leq t_{j+1}\right) \mathbb{P}\left(t_j \leq T^{x_{n_1},t_{n_2}}_{x_{n_1+1}} \leq t_{j+1}\right),$$

and we approximate $T^{x_{n_1},t_{n_2}}_{x_{n_1+1}}$ by t_j on the event $t_j \leq T^{x_{n_1},t_{n_2}}_{x_{n_1+1}} \leq t_{j+1}$, which leads to

$$\mathbb{P}(T_B^{x_{n_1},t_{n_2}} \leq T) \approx \sum_{j=1}^{G} \mathbb{P}(T_B^{x_{n_1+1},t_j} \leq T)\mathbb{P}(t_j \leq T^{x_{n_1},t_{n_2}}_{x_{n_1+1}} \leq t_{j+1}). \tag{5.21}$$

We deduce from equation (5.21) a recursive way to estimate the probability that the process starting at a place and a time of the grid reaches B before time T. To see this, we first define the sequence $p(n_1,n_2)$, for $n_1 = 1,\dots,H$ and $n_2 = 1,\dots,G$ by

$$p(n_1,n_2) = \mathbb{P}(T_B^{x_{n_1},t_{n_2}} \leq T).$$

Equation (5.21) can then be rewritten as

$$p(n_1,n_2) \approx \sum_{j=1}^{G} p(n_1,j)\mathbb{P}(t_j \leq T^{x_{n_1},t_{n_2}}_{x_{n_1+1}} \leq t_{j+1}). \tag{5.22}$$

It is now sufficient to estimate the quantities $\mathbb{P}(t_j \leq T^{x_{n_1},t_{n_2}}_{x_{n_1+1}} \leq t_{j+1})$. We estimate the whole set of probability values

$$\left\{\mathbb{P}(t_j \leq T^{x_{n_1},t_{n_2}}_{x_{n_1+1}} \leq t_{j+1}),\ j = 1,\dots,G\right\}$$

with a single run of Monte Carlo as follows. A number N' of trajectories of the process $\{X(t),\ t \geq 0\}$ starting from x_{n_1} at time t_{n_2} is sampled until final time T. We denote $t \mapsto X^i(t)$ these trajectories and for all $j = 1,\dots,G$ and we have the following estimates

$$\mathbb{P}(t_j \leq T^{x_{n_1},t_{n_2}}_{x_{n_1+1}} \leq t_{j+1}) \approx \frac{1}{N'}\sum_{i=1}^{N'} \mathbb{1}_{X^i(t)=x_{n_1+1},\ \text{for some } t\in[t_j;t_{j+1}]}.$$

In the following, we denote these probability values by

$$q(j,n_1,n_2) = \mathbb{P}(t_j \leq T^{x_{n_1},t_{n_2}}_{x_{n_1+1}} \leq t_{j+1}) \tag{5.23}$$

The procedure can be summed up as follows. We first set some notations. The estimates of $p(n_1,n_2)$ are denoted by $\widehat{p}(n_1,n_2)$, and those of $\mathbb{P}(t_j \leq T^{x_{n_1},t_{n_2}}_{x_{n_1+1}} \leq t_{j+1})$ given in equation (5.23) are denoted by $\widehat{q}(j,n_1,n_2)$. Notice that we only need to know $\widehat{q}(j,n_1,n_2)$ since $\widehat{p}(n_1,n_2)$ are recursively given using equation (5.22) by

$$\widehat{p}(n_1,n_2) = \sum_{j=1}^{H} \widehat{p}(n_1,j)\widehat{q}(j,n_1,n_2). \tag{5.24}$$

Finally we initialize the recursion (5.24). We recall that $\widehat{p}(n_1,n_2)$ is an estimate of $\mathbb{P}(X(t) \geq S$, for some $t \leq T | X(t_{n_2}) = x_{n_1})$ and that $t_{n_{G+1}} = T$ and $x_{n_{H+1}} = S$. Consequently we set

$$\widehat{p}(n_1,n_{G+1}) = \begin{cases} 0 & \text{if } n_1 = 0,\dots,H, \\ 1 & n_1 = H+1, \end{cases}$$

and

$$\widehat{p}(n_{H+1}, n_2) = 1, \ \forall n_2 = 0, \dots, H.$$

For the sake of conciseness, the above detailed procedure to estimate the probabilities

$$p(x,s) = \mathbb{P}(X(t) \geq B, \text{ for some } t \leq T | X(s) = x)$$

at each point (x_{n_2}, t_{n_1}) of the grid is written in algorithm 5.2.

Algorithm 5.2: Estimation of $\mathbb{P}(T_B^{x,t} \leq T)$ at each point of the grid (x_{n_1}, t_{n_2})

Data: rare event probability set B, process $\{X(t),\ t \geq 0\}$ with continuous trajectories, grid $\{\alpha_{n_1,n_2},\ n_1 = 0, \dots, H+1, , n_2 = 0, \dots, G+1\}$

Initialization
  for $n_2 = 1, \dots, G+1$ **do**
    Set $\widehat{p}(n_{H+1}, n_2) = 1$.
  for $n_2 = 1, \dots, G+1$ **do**
    Set $\widehat{p}(n_1, n_{G+1}) = \begin{cases} 0 & n_1 = 0, \dots, H, \\ 1 & \text{if } n_1 = H+1, \end{cases}$
  Set $n_1 = G+1$
while $n_1 > 0$ **do**
  for $n_2 = 1, \dots, G$ **do**
    Sample N' trajectories $t \mapsto X^i(t)$ starting from x_{n_1} at time t_{n_2}.
    for $j = 1, \dots, G$ **do**
      Set $\widehat{q}(j, n_1, n_2) = \frac{1}{N'} \sum_{i=1}^{N'} \mathbb{I}_{X^i(t) = x_{n_1+1}, \text{ for some } t \in [t_j; t_{j+1}]}$.
  Set $n_1 = n_1 - 1$.
  for $n_2 = 1, \dots, G+1$ **do**
    Set $\widehat{p}(n_1, n_2) = \sum_{j=1}^{G} \widehat{p}(n_1, j) \widehat{q}(j, n_1, n_2)$.

Result: $\mathbb{P}\left(T_B^{t_{n_1}, x_{n_2}} \leq T\right) \approx \widehat{p}(n_1, n_2),\ n_1 = 0, \dots, H+1, , n_2 = 0, \dots, G+1$

5.3.3 Numerical experiments

Here the process under consideration is the drifted Brownian motion, defined by

$$\mathrm{d}X_t = -\nu \mathrm{d}t + \mathrm{d}W_t.$$

This process has been extensively studied, in particular we have the close form expression for the probability of reaching a threshold S before time deterministic time T starting from a place $x \in \mathbb{R}$ at time $s \in \mathbb{R}^+$ [Deelstra, 1994]

$$\mathbb{P}(T_S^{x,s} \leq T) = 1 - F\left((S-x)/\sqrt{T-s}\right) + \nu\sqrt{T-s} + e^{-2\nu(S-x)} F\left(\nu\sqrt{T-s}\right) - (S-x)/\sqrt{T-s},$$

where F is the cumulative distribution function of the standard one-dimensional Gaussian random variable, $F(x) = 1/\sqrt{2\pi} \int_{-\infty}^{x} \exp(-t^2/2)\mathrm{d}t$.

We set $\nu = 1$. We are interested in estimating the probability

$$\mathbb{P}(T_8 \leq 5),$$

with the splitting algorithm. Using the `contour` function of MATLAB, we have obtained the optimal regions for the splitting algorithm in figure 5.7 where $S = 8$ and $T = 5$. More precisely, we drew the time dependent curves

$$A_k^* = \left\{(x,t) \in \mathbb{R} \times \mathbb{R}^+, \mathbb{P}(T_S^{x,s} \leq T) = p_k\right\},$$

for each $p_k = 10^{-k}$ and the negative integers onto each curve are equals to $\log_{10}(p_k)$. Namely, the probability of reaching S before T starting from any x and any t such that $(x,t) \in A_k^*$ equals to 10^{-k}.

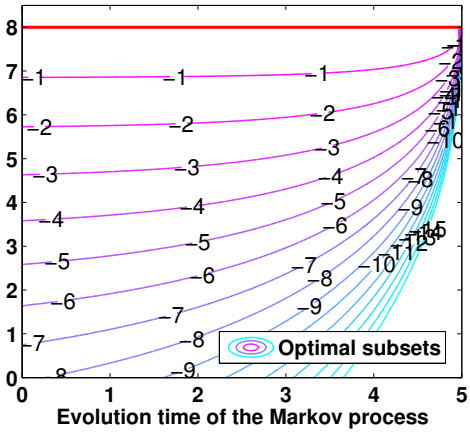


Figure 5.7: The optimal time dependent regions for the drifted Brownian motion.

Using the Runge-Kutta method [Demailly, 2012, chap 7] and the `contour` function of MATLAB, we get the approximations of the optimal regions in figure 5.8.

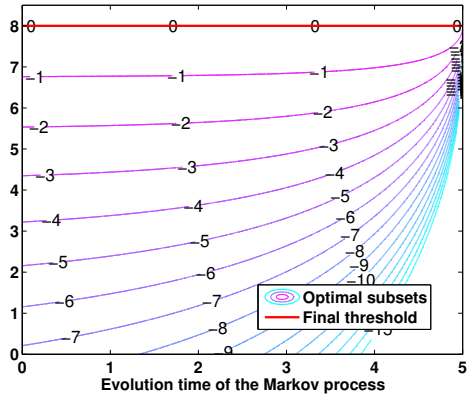


Figure 5.8: Estimation of the optimal regions with the Runge Kutta method.

We have also estimated the optimal regions A_k^* with algorithm 5.2. The grid $\{\alpha_{n_1,n_2},\ n_1 = 0,\ldots,H+1,,n_2 = 0,\ldots,G+1\}$ is constructed in the following way. The real intervals $[0,S]$ and $[0,T]$ are divided into n_1 and n_2 equally spaced intervals respectively. For both of the intervals $[0,S]$ and $[0,T]$, the length of each sub-interval equals to 0.2. To estimate the probabilities $q(j,n_1,n_2)$ defined in equation (5.23), we use a small set of $N' = 10$ trajectories. A typical sample of the estimated regions with algorithm 5.2 is presented in figure 5.9 for which the `contour`

function of MATLAB was used.

We denote here the non-optimal regions $A_k = \{(x,t),\ x \geq S_k\}$ and its associate hitting time

$$T_k = \inf\{t,\ X_t \geq S_k\}.$$

In the same way we define T_k^* with

$$T_k^* = \inf\{t,\ (X_t,t) \in A_k^*\}.$$

The expected value of the conditional probabilities $\mathbb{P}(T_k^* \leq T | T_{k-1}^* \leq T)$ and $\mathbb{P}(T_k \leq T | T_{k-1} \leq T)$ are equal to 0.5 in every splitting run. Notice that this value is different in figures (5.7),(5.8) and (5.9) for the sake of plot clarity.

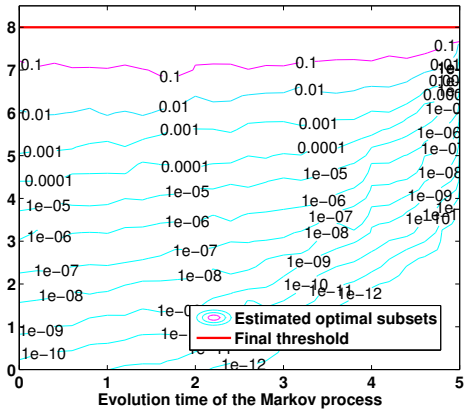


Figure 5.9: Estimation of the optimal regions with the algorithm 5.2.

For final thresholds S between 4 and 12 and final horizon time $T = 5$, estimation of the relative standard deviation of the splitting algorithm with different regions are compared in figure 5.10. The number N of trajectories sampled at each threshold equals to $N = 1000$. To compute V_m^*, the lower bound of the relative variance variance, we use the fact that $p_k = 0.5$. Hence, taking into account that $p_1 \cdots p_m = \mathbb{P}(T_B \leq T)$ the number of threshold m equals to $m = \log(\mathbb{P}(T_B \leq T))/\log(0.5)$. Finally, using the $\sqrt{N}$ convergence rate given by the central limit theorem 3.1, the red curve in figure 5.10 is actually the curve

$$S \mapsto \sqrt{\frac{1}{N}\frac{\log \mathbb{P}(T_B \leq T)}{\log(0.5)}(1/0.5-1)}.$$

We observe that performances of the partial differential equation method and the grid method of algorithm 5.2 both reach the lower bound V_m^*. The splitting with non-optimal regions is the splitting where the regions are defined with the non-optimal importance function $x \mapsto x$. The variance with the splitting with non optimal regions increases exponentially.

In the table 5.2, we make varying the number of trajectories used in the splitting algorithm. The table consists of the relative standard deviation in the estimation with the splitting algorithm with the theoretical optimal regions A_k^* and the splitting with non optimal regions. The number N is the number of sampled trajectories at each threshold. Letters o and n-o stand

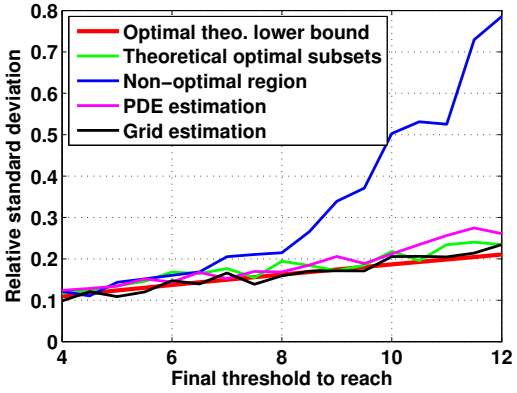


Figure 5.10: Comparison of the different estimation methods of the optimal regions.

for optimal regions and non-optimal regions. For example, for $B = 12$ and $N = 500$, the splitting with optimal regions gives a 34% relative standard deviation whereas the splitting with non-optimal region gives a 48% relative standard deviation with ten times more trajectories $N = 5000$. When the probability becomes very rare, using the optimal regions is very interesting from a budget point of view.

	S	5		9		12	
	Theoretical probability	$2.66.10^{-5}$		$7.52.10^{-10}$		$4.75.10^{-14}$	
		n-o	o	n-o	o	n-o	o
N	100	65%	38%	125%	44%	340%	69%
	500	24%	16%	52%	23%	148%	34%
	1000	20%	14%	40%	21%	99%	23%
	2500	10%	9%	26%	10%	59%	16%
	5000	9%	6%	19%	9%	48%	10%

Table 5.2: Trajectory number influence on the relative standard deviation in the splitting algorithm with optimal (o) and non-optimal (n-o) regions.

Conclusions of the chapter

Contributions

- Highlighting and correction of the bias in the splitting algorithm. The bias due to the discretization of the trajectories of the process under consideration is corrected using bigger well chosen time dependent regions. However, we have shown that in the case of intermediate regions defined by threshold exceedance of a real valued function, the bigger set to consider is not necessarily defined as threshold exceedance of the same function. Thus, additional computation are required to take into account the corrected bigger set region. In any case, we have to keep in mind that the discretization step may lead to underestimated probabilities.

- Elaboration of two methods to compute the optimal regions in the case of a deterministic final horizon time T. The first one is based on a PDE characterisation of the optimal importance function, and the second one is more general and only requires the trajectories of the process to be continuous. We numerically showed that the two proposed method reach the highest performances in term of minimization of the variance. However, as already given methods for the estimation of the optimal regions [Garvels et al., 2002, Dean and Dupuis, 2009], the proposed methods suffer from the dimension.

Links with the next chapter

- A huge computation time may be required to estimate optimal regions in dimension greater or equal than 2. On the other hand, defining the regions in a non-optimal manner is very easy with the algorithm 4.1 proposed in chapter 4. In the next chapter, we will investigate improvements of the variance in the splitting algorithm using non-optimal regions. We will suppose the intermediate regions already given, for example with algorithm 4.1, and we will show that some variance reduction can be obtained using different selections of trajectories.

CHAPTER 6

Elaboration of an interacting particle system algorithm for rare event with non-negative potential functions and applications

In this chapter

- We elaborate an interacting particle system algorithm with non negative weights for the approximation of some Feynman-Kac distribution and a special focus on the estimation of rare event probability
- We derive a splitting algorithm that gives more importance to the trajectories that reach an intermediate region the earliest in order to reduce the variance in the estimated probability.

6.1 Introduction and motivations

From a formal point of view, both the splitting algorithm 3.1 and the weighted redistribution algorithm 3.2 use interacting particle techniques for the approximation of Feynman-Kac distributions. Their ultimate aims are to numerically compute the probability of an event which can be characterized by a function of the final state of a Markov chain. Namely, given a fixed integer m and a Markov chain $\{\Xi_k,\, k = 0,\dots,m\}$, the splitting and the weighted redistribution algorithm gives estimates of

$$\mathbb{P}(\Xi_m \in A).$$

Such a characterisation is obvious for the weighted redistribution algorithm. As far a the splitting is concerned, we mention the framework derived in section 3.2. We consider the Markov chain $\{\Xi_k = (X(T_k), T_k),\, k = 0,\dots,m\}$, where T_k is the first entrance time of the process $\{(X(t), t),\, t \geq 0\}$ into the k-th intermediate region. We are interested in the event $\{T_m \leq T\}$, and it can be rewritten as follows

$$\{T_m \leq T\} = \{\Xi_m \in A_T^{\mathrm{sp}}\},$$

with $A_T^{\mathrm{sp}} = \pi^{-1}\{[0,T]\}$ where π is the projection on the last coordinate. Moreover, such a characterisation is still valid for static splitting algorithm [Cérou et al., 2012].

However, these algorithms differ from their respective potential functions. The splitting actually uses $\{0,1\}$-valued potential functions, whereas the weighted redistribution algorithm requires its potential functions to be positive. Nevertheless, the similarity of these two algorithms in their fundamental functioning and the probability they aim at estimating leads to study a more general version of an interacting particle system (IPS) for rare event estimation that includes the splitting and the weighted redistribution algorithm.

6.2 An IPS algorithm for rare event estimation with non-negative weights

6.2.1 Framework

Assume given a Markov chain $\{\Xi_k,\, k=0,\dots,m\}$, where for all $k=0,\dots,m$, $\Xi_k \in E_k$. For a fixed integer m and a given set A, we are interested in estimating the probability of the event $\mathbb{P}(\Xi_m \in A)$. Its value is assumed to be very small.

We adopt a path-wise point of view. We denote by $\{Y_k,\, k=0,\dots,m\}$ the historical process of the Markov chain $\{\Xi_k,\, k=0,\dots,m\}$:

$$Y_k = \Xi_{0:k} = (\Xi_0, \Xi_1, \dots, \Xi_k)\,.$$

The state space of the random variable Y_k is denoted by $F_k = E_0 \times \cdots \times E_k$.
We are interested in giving some approximations of

- the probability of the rare event
$$\mathbb{P}(\Xi_m \in A), \tag{6.1}$$
- the law of the paths of the Markov chain $\{\Xi_k,\, k=0,\dots,m\}$ given the event $\Xi_m \in A$,
$$\mathbb{P}\left(\Xi_{0:m} \in \mathrm{d}x_{0:m} | \Xi_m \in A\right). \tag{6.2}$$

The transition kernels of the Markov chain $\{\Xi_k,\, k=0,\dots,m\}$ are denoted by

$$Q_k(x_{k-1}, \mathrm{d}x'_k) = \mathbb{P}(\Xi_k \in \mathrm{d}x'_k | \Xi_{k-1} = x_{k-1}),$$

and the Markov chain Y_k is characterized by the following transition kernels

$$\begin{aligned} Q_k^\bullet(y_{k-1}, \mathrm{d}y'_k) &= Q_k^\bullet(x_0, \dots, x_{k-1}, \mathrm{d}x'_0, \dots, \mathrm{d}x'_k) \\ &= \delta_{(x_0,\dots,x_{k-1})}(\mathrm{d}x'_0, \dots, \mathrm{d}x'_{k-1}) Q_k(x_{k-1}, \mathrm{d}x'_k). \end{aligned}$$

6.2.2 Feyman-Kac distributions associated to the paths of the Markov chain

The objective of this section is to derive numerical approximations of the following unnormalised and normalised Feynman-Kac distributions defined on the path space

$$\langle \gamma_k^\bullet, f\rangle = \mathbb{E}[f(Y_k) \prod_{p=1}^{k} g_p^\bullet(Y_p)] \quad \text{and} \quad \langle \mu_k^\bullet, f\rangle = \frac{\langle \gamma_k^\bullet, f\rangle}{\langle \gamma_k^\bullet, 1\rangle}, \tag{6.3}$$

where some potential functions $g_p^\bullet$ take their values in $[0, +\infty)$. The distributions $\gamma_k^\bullet$ and $\mu_k^\bullet$ are linked with (6.1) and (6.2) as explained hereafter.

The unnormalized and normalized distributions satisfy the fundamental recursive equation, which can be shown using conditioning and Markov properties

$$\gamma_k^\bullet = g_k^\bullet(\gamma_{k-1}^\bullet Q_k^\bullet) = g_k^\bullet(\mu_{k-1}^\bullet Q_k^\bullet)\langle \gamma_k^\bullet, 1\rangle, \quad \text{and} \quad \mu_k^\bullet = g_k^\bullet(\mu_{k-1}^\bullet Q_k^\bullet). \tag{6.4}$$

For the sake of clarity, we also define the distribution $\eta_k^\bullet$ with

$$\eta_k^\bullet = \mu_{k-1}^\bullet Q_k^\bullet.$$

The distribution $\eta_k^\bullet$ satisfies the identity

$$\langle \gamma_k^\bullet, f\rangle = \langle \eta_k^\bullet g_k^\bullet, f\rangle \langle \gamma_{k-1}^\bullet, f\rangle,$$

whose proof is similar to the one of proposition 3.1 and the following decomposition product holds

$$\langle \gamma_k^\bullet, f\rangle = \prod_{p=1}^{k} \langle \eta_p^\bullet, g_p^\bullet f\rangle. \tag{6.5}$$

6.2.2.1 Potential function factorisation

The key idea is the following. We propose the following factorisation for the potential function $g_k^\bullet$

$$g_k^\bullet(y_k) = a_k^\bullet(y_k) b_k^\bullet(y_k), \tag{6.6}$$

where for all $y_k \in F_k$

$$a_p^\bullet(y_p) > 0 \quad \text{and} \quad b_p^\bullet(y_p) \in \{0,1\}.$$

Roughly speaking, if for a given $y_k = (x_0, \ldots, x_k)$ we have $b_k(y_k) = 0$, the trajectory $(x_0, \ldots, x_k)$ is considered to be useless and is discarded. On the other hand, if $b_k^\bullet(y_k) = 1$ the trajectory is kept and is weighted with $a_k^\bullet(y_k)$.

Besides, in order to handle with quantities (6.1) and (6.2) we make the assumption that

$$\prod_{k=1}^{m} b_k^\bullet(y_k) = \mathbb{1}_{x_m \in A},$$

where $y_k = (x_1, \ldots, x_k)$.

6.2.2.2 Interpretation of the rare event probability

To connect the quantities (6.1) and (6.2) of interest with $\gamma_k^\bullet$ and $\mu_k^\bullet$, we use the function $T_k^\bullet f$ defined on the path space by

$$T_k^\bullet f(y_k) = T_k^\bullet f(x_0, \ldots, x_k) = f(y_k) \prod_{p=1}^{k} \frac{1}{a_p^\bullet(y_p)}. \tag{6.7}$$

For the particular choice $f \equiv 1$, we obtain

$$\mathbb{P}(\Xi_m \in A) = \langle \gamma_m^\bullet, T_m^\bullet 1\rangle. \tag{6.8}$$

and the law of the trajectories of the Markov chain given the rare event is reached at final time m is characterized by

$$\begin{aligned} \mathbb{E}\left(f(\Xi_{0:m}) | \Xi_m \in A\right) &= \frac{\mathbb{E}\left(f(\Xi_{0:m})\mathbb{1}_{\Xi_m \in A}\right)}{\mathbb{P}(\Xi_m \in A)} \\ &= \frac{\langle \gamma_m^\bullet, T_m^\bullet f\rangle}{\langle \gamma_m^\bullet, T_m^\bullet 1\rangle}. \end{aligned} \tag{6.9}$$

6.2.3 Interacting path-particle system for the approximation of the Feynman-Kac distributions

We take into account the recursive equation (6.4) to derive an interacting particle system for the estimation of (6.8) and (6.9).

We are interested in approximations of the normalized distribution $\mu_{k-1}^{\bullet}$ with the following form

$$\mu_{k-1}^{\bullet,N} = \sum_{i=1}^{N} w_{k-1}^{i} \delta_{\xi_{k-1}^{\bullet,i}} , \quad \text{with} \sum_{j=1}^{N} w_{k-1}^{j} = 1,$$

where, for all $i = 1,\dots,N$, the particle $\xi_{k-1}^{\bullet,i}$ is a random trajectory

$$\xi_{k-1}^{\bullet,i} = (\xi_{0,k-1}^{i},\dots,\xi_{k-1,k-1}^{i}).$$

Furthermore, with equation (6.4) we obtain

$$\mu_{k-1}^{\bullet,N} Q_k^{\bullet}(\mathrm{d}y'_k) = \sum_{i=1}^{N} w_{k-1}^{i} Q_k^{\bullet}(\xi_{k-1}^{\bullet,i}, \mathrm{d}y'_k), \tag{6.10}$$

which yields to the following particle approximation of $\eta_k^{\bullet} = \mu_{k-1}^{\bullet} Q_k^{\bullet}$

$$\eta_k^{\bullet,N} = \frac{1}{N} \sum_{i=1}^{N} \delta_{\xi_k^{\bullet,i}} \tag{6.11}$$

where, independently for all $i = 1,\dots,N$, the trajectory $\xi_k^{\bullet,i}$ is sampled from the finite mixture distribution (6.10).

In practice, the transition from $\mu_{k-1}^{\bullet,N}$ to $\eta_k^{\bullet,N}$ is obtained as follows. Independently for all $i = 1,\dots,N$

1. a path $\widehat{\xi}_{k-1}^{\bullet,i} = (\widehat{\xi}_{0,k-1}^{i},\dots,\widehat{\xi}_{0,k-1}^{i})$ is selected amongst the current population $(\xi_{k-1}^{\bullet,1},\dots,\xi_{k-1}^{\bullet,N})$, according to their respective weights $(w_{k-1}^{1},\dots,w_{k-1}^{N})$.
2. the random path $\xi_k^{\bullet,i} = (\xi_{0,k}^{i},\dots,\xi_{k,k}^{i})$ is sampled from the distribution $Q_k^{\bullet}(\widehat{\xi}_{k-1}^{\bullet,i}, \mathrm{d}y'_k)$: in other words, we set $\xi_{p,k}^{i} = \widehat{\xi}_{p,k-1}^{i}$ for all $p = 0,\dots,k-1$, and the random variable $\xi_{k,k}^{i}$ is sampled from the distribution $Q_k(\xi_{k-1,k}^{i}, \mathrm{d}x'_k)$.

This recursive construction can be summarized by the following classical selection mutation scheme

$$\begin{pmatrix} \xi_{k-1}^{\bullet,1} \\ \vdots \\ \vdots \\ \xi_{k-1}^{\bullet,N} \end{pmatrix} \xrightarrow[(w_{k-1}^{1},\dots,w_{k-1}^{N})]{\text{Selection}} \begin{pmatrix} \widehat{\xi}_{k-1}^{\bullet,1} \\ \vdots \\ \vdots \\ \widehat{\xi}_{k-1}^{\bullet,N} \end{pmatrix} \xrightarrow[Q_k^{\bullet}(\widehat{\xi}_{k-1}^{\bullet,i}, \mathrm{d}y'_k)]{\text{Mutation}} \begin{pmatrix} \xi_{k}^{\bullet,1} \\ \vdots \\ \vdots \\ \xi_{k}^{\bullet,N} \end{pmatrix} \tag{6.12}$$

With the recursive equation (6.4), we obtain the following particle approximation of the unnormalized distribution $\gamma_k^{\bullet}$:

$$\gamma_k^{\bullet,N} = g_k^{\bullet} \eta_k^{\bullet,N} \langle \gamma_{k-1}^{\bullet,N}, 1 \rangle = \left(\frac{1}{N} \sum_{i=1}^{N} g_k^{\bullet}(\xi_k^{\bullet,i}) \delta_{\xi_k^{\bullet,i}} \right) \langle \gamma_{k-1}^{\bullet,N}, 1 \rangle.$$

From there, we deduce the useful approximations

- the approximation of the normalization constant as a recursive equation

$$\langle \gamma_k^{\bullet,N}, 1\rangle = \left(\frac{1}{N}\sum_{i=1}^{N} g_k^{\bullet}(\xi_k^{\bullet,i})\right)\langle \gamma_{k-1}^{\bullet,N}, 1\rangle,$$

- the particle approximation of the normalized distribution $\mu_k^{\bullet,N}$ as a recursive equation

$$\mu_k^{\bullet,N} = \frac{\gamma_k^{\bullet,N}}{\langle \gamma_k^{\bullet,N}, 1\rangle} = \sum_{i=1}^{N} \frac{g_k^{\bullet}(\xi_k^{\bullet,i})}{\sum_{j=1}^{N} g_k^{\bullet}(\xi_k^{\bullet,j})}\delta_{\xi_k^{\bullet,i}} = \sum_{i=1}^{N} w_k^i \delta_{\xi_k^{\bullet,i}},$$

  which implicitly defines the weights w_k^i as follows

$$w_k^i = \frac{g_k^{\bullet}(\xi_k^{\bullet,i})}{\sum_{j=1}^{N} g_k^{\bullet}(\xi_k^{\bullet,j})}, \quad i = 1,\ldots,N. \tag{6.13}$$

Then, the following estimates of the quantities (6.8) and (6.9) hold

$$\mathbb{P}(\Xi_m \in A) \approx \langle \gamma_m^{\bullet,N}, T_m^{\bullet} 1\rangle,$$

and

$$\mathbb{E}\left(f(\Xi_{0:m})| \in \Xi_m \in A\right) \approx \frac{\langle \gamma_m^{\bullet,N}, T_m^{\bullet} f\rangle}{\langle \gamma_m^{\bullet,N}, T_m^{\bullet} 1\rangle}.$$

To analyse the asymptotic fluctuation of the above described particle approximation, we first define the life time of the set of particles $\{\xi_k^{\bullet,i},\ i = 1,\ldots,N\}$ with

$$\tau^N = \inf\left\{k \geq 0,\ \sum_{i=1}^{N} g_k^{\bullet}(\xi_k^{\bullet,i}) = 0\right\},$$

for which the exponential lower bound holds [Del Moral, 2004, theo 7.4.1]

$$\mathbb{P}(\tau^N \leq m) \leq a_m \exp(-b_m N).$$

The approximation of the rare event probability defined on the *good* set $\{\tau^N > m\}$ converges in law, as detailed in the following theorem [Del Moral, 2004, chap. 9], [Cappé et al., 2005, chap. 9],

Theorem 6.1.

$$\sqrt{N}\frac{\langle \gamma_m^{\bullet,N} - \gamma_m^{\bullet}, f\rangle}{\langle \gamma_m^{\bullet}, 1\rangle} \mathbb{I}_{\tau^N > m} \xrightarrow[N\to+\infty]{} \mathcal{N}(0, V_m^{\bullet}(f)),$$

where the the asymptotic variance equals to

$$V_m^{\bullet}(f) = \sum_{k=0}^{m} \frac{\mathrm{var}(g_k^{\bullet} R_{k+1:m}^{\bullet} f, \eta_k^{\bullet})}{\langle \eta_k^{\bullet}, g_k^{\bullet} R_{k:m}^{\bullet} 1\rangle},$$

and the $R_{k+1:m}^{\bullet}$ functions equal to

$$R_{k+1:m}^{\bullet} f(x_{0:k}) = \mathbb{E}\left(f(\Xi_{0:k}) \prod_{p=k+1}^{m} g_p^{\bullet}(\Xi_{0:k})|\Xi_{0:k} = x_{0:k}\right).$$

Others asymptotic results on the approximation of the normalizing constant can be found in [Le Gland, 2007].

6.2.4 Links with existing algorithms

6.2.4.1 The splitting algorithm

The splitting aims at estimating quantities such as $\mathbb{P}(X(t) \in B, \text{ for some } 0 \leq t \leq T)$. And in that case the Markov chain $\{\Xi_k,\ k = 0,\dots,m\}$ and the $g_k^\bullet$ functions are defined as follows (see section 3.2)

$$\Xi_k = (X_{T_k \wedge T}, T_k \wedge T) \quad \text{and} \quad g_k^\bullet(\Xi_{0:k}) = \mathbb{I}_{(X_{T_k \wedge T}, T_k \wedge T) \in A_k}. \tag{6.14}$$

As a result, we have

$$a_k^\bullet = 1, \quad b_k^\bullet = g_k^\bullet \quad \text{and} \quad T_m^\bullet f \equiv 1,\ \forall f \tag{6.15}$$

and the following expression for the rare event probability holds

$$\mathbb{P}(X(t) \in B, \text{ for some } 0 \leq t \leq T) = \mathbb{P}(T_m \leq T) = \gamma_m^\bullet(1).$$

Moreover, the distribution $\eta_k^\bullet$ gives that

$$\langle \eta_k^\bullet, g_k^\bullet \rangle = \frac{\langle \gamma_k^\bullet, 1 \rangle}{\langle \gamma_{k-1}^\bullet, 1 \rangle} = \mathbb{P}(T_k \leq T | T_{k-1} \leq T),$$

and equation (6.5) gives the usual Bayes decomposition product formula of the splitting algorithm

$$\mathbb{P}(T_m \leq T) = \prod_{k=1}^{m} \mathbb{P}(T_k \leq T | T_{k-1} \leq T).$$

Finally, it is worth mentioning that the estimation of $\eta_k^\bullet$ given at equation (6.11) gives the following well known estimation for the conditional probability value $\mathbb{P}(T_k \leq T | T_{k-1} \leq T)$

$$\begin{aligned} \mathbb{P}(T_k \leq T | T_{k-1} \leq T) &= \langle \eta_k^\bullet, g_k^\bullet \rangle \\ &\approx \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}_{\xi_k^{\bullet,i} \in A_k} \\ &= \frac{N_k}{N}, \end{aligned}$$

where N_k is the number of trajectories of the process $\{(X_t, t),\ t \geq 0\}$ that have reached the region A_k before time T, starting from the region A_{k-1}.

6.2.4.2 The weighted redistribution algorithm

For the weighted redistribution algorithm, the goal is to give some numerical approximations of $\mathbb{P}(\Xi_n \in A)$ for a Markov chain $\{\Xi_k,\ k = 0,\dots,m\}$, and a region A of the state space of Ξ_m. The original paper that details the weighted redistribution algorithm only requires that $g_k^\bullet(x_{0:k}) > 0$. Hence, denoting $y_k = (x_0,\dots,x_k)$ for $k = 1,\dots,m$ we have in this case :

- for $k = 1,\dots,m-1$

$$a_k^\bullet(y_k) > 0 \quad \text{and} \quad b_k^\bullet(y_k) = 1 \tag{6.16}$$

- and for $k = m$

$$a_m^\bullet(y_m) = 1 \quad \text{and} \quad b_m^\bullet(y_m) = \mathbb{I}_{y_m \in A}.$$

Remark 6.1. *The interest in rewriting the splitting and the weighted redistribution algorithm is not only purely mathematics. In particular, we have extended the weighted redistribution algorithm to the case of non-negative potential function.*

6.3 Application to the previous IPS algorithm to a splitting algorithm with important sampling on the entrance time

6.3.1 Framework and motivations

We assume $\{X(t),\ t \geq 0\}$ to be a Markov process with state space E and càdlàg trajectories. The law of $X(0)$ is denoted by η_0. We assume B to be a region of E which the process $\{X(t),\ t \geq 0\}$ is very unlikely to reach. We denote $A = B \times [0, +\infty)$ and we assume given a sequence of nested supersets of A

$$A_1 \supset \cdots A_{m-1} \supset A_m = A,$$

which can be given, for example, with the adaptive algorithm 4.1 if B is characterized with a threshold exceedance of a real valued function. We set also $T_k = \inf\{t \geq 0,\ (X(t), t) \in A_k\}$. Given a *deterministic* final time T we are interested in the event

$$\mathbb{P}(X(t) \in B \ \text{ for some } t \leq T).$$

Defining the region A_k with a non-optimal importance function ϕ is most of the time the easiest way to proceed, and the adaptive algorithm given in chapter 4 is a very simple way to give some regions A_k. The question addressed here is how can we improve the accuracy of the splitting algorithm without modifying the importance function ?

One of the goal of the splitting algorithm 3.1 is to simulate trajectories conditionally to the event $\{T_B \leq T\}$, or equivalently to sample trajectories that reach the rare event probability set before time T. As far as the trajectories that did not reach B are concerned, either they were not selected at one of the re-sample steps, or they did not succeed to reach an intermediate region. We focus on the last eventuality. These trajectories were not relevant because they could not enable to create trajectories that reach B. We should not have selected them and have tried to anticipate the fact that they do not manage to reach B.

We remind that in this section we have to work with a deterministic horizon time T. Intuitively, trajectories that have reached the region A_k early are more likely to reach the rare set $B \times [0, +\infty)$ than trajectories that have reached the region A_k later on. These observations are illustrated in figure 6.1. The red circles represent the crossing points between the trajectories and an intermediate threshold. Trajectories starting from the later crossing point are useless.

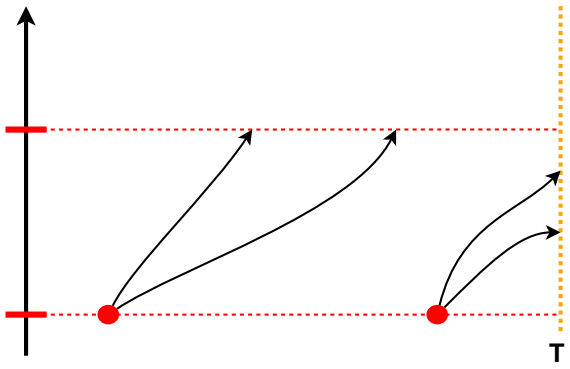


Figure 6.1: The trajectories that reach the intermediate region the latest cannot enable to create trajectories that reach the rare even probability set.

We propose thus to give more importance to the earliest trajectories, selecting them mostly than the others. To be more specific, at each re-sample step of the trajectories, we give more importance to the trajectories that have reached a region the earliest. After that, a weighting step is required not to bias the estimated probability.

6.3.2 A Feynman-Kac distribution framework

6.3.2.1 Feynman-Kac interpretation of the rare event probability

The key idea is to decompose the potential function $g_k(\Xi_k) = g_k(X(T_k \wedge T), T_k \wedge T)$ defined by

$$g_k(x) = g_k(\chi, t) = \mathbb{I}_{(\chi,t)\in A_k}$$

as follows

$$g_k = g_k^{\text{imp}}\, g_k^{\text{red}},$$

where the function g_k^{red} is used to re-sample the successful trajectories, and where the function g_k^{imp} is used to measure the importance of trajectories with regard to the estimations, in other words to weight trajectories. In this context, we propose the following factorization

$$g_k^{\text{imp}}(\chi, t) = \frac{1}{a_k(t)} \quad \text{and} \quad g_k^{\text{red}}(\chi, t) = a_k(t)\, \mathbb{I}_{(\chi,t)\in A_k}, \tag{6.17}$$

with a function $t \mapsto a_k(t)$ that should be non-increasing, in order to select preferably the trajectories that reach the region A_k early. Figure 6.2 emphasizes this idea.

Remark 6.2. *Other a_k functions can be considered. The a_k functions can not only depend on the time variable, but also on the space variable. Indeed, we can be interested in giving more importance to some space regions, setting $a_k(\chi, t)$ depending on space variable χ or space-time variable (χ, t). Consequently, the below described algorithm can be understood as a led exploration of the state space coupling with a rare event estimation. Such considerations are relevant for example if T is no longer deterministic.*

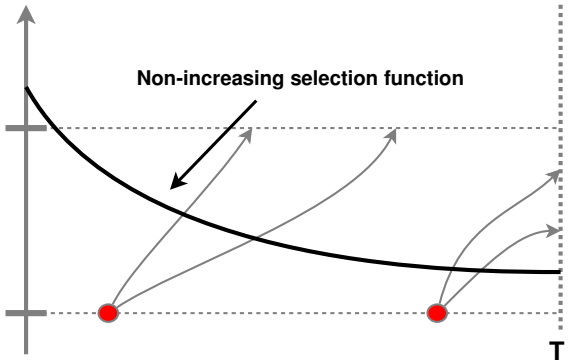


Figure 6.2: A non-increasing potential function makes more likely the selection of the earliest crossing region trajectories.

We remind the Feynman-Kac interpretation of the standard splitting algorithm 3.1. To this purpose, we set $\Xi_k = (X_{T_k}, T_k)$ and the following unnormalised and normalised distributions

$$\langle \gamma_k, f\rangle = \mathbb{E}\left(f(\Xi_k)\prod_{p=0}^{k} g_k(\Xi_k)\right) = \mathbb{E}[f(X_{T_k}, T_k)\, \mathbb{I}_{T_k\le T}], \quad \text{and} \quad \langle \mu_k, f\rangle = \frac{\langle \gamma_k, f\rangle}{\langle \gamma_k, 1\rangle}.$$

So that

$$\langle \gamma_k, 1\rangle = \mathbb{P}(T_k \leq T)$$

and

$$\mu_k(\mathrm{d}\chi, \mathrm{d}t) = \mathbb{P}(X_{T_k} \in \mathrm{d}\chi, T_k \in \mathrm{d}t | T_k \leq T)$$

is the entrance distribution into the region A_k. The unnormalised Feynman-Kac distribution γ_k can be rewritten in the following way

$$\langle \gamma_k, f\rangle = \mathbb{E}\left[f(\Xi_k)\prod_{p=1}^{k} g_p(\Xi_p)\right] = \mathbb{E}\left[\left(f(\Xi_k)\prod_{p=1}^{k} g_p^{\mathrm{imp}}(\Xi_p)\right)\prod_{p=1}^{k} g_p^{\mathrm{red}}(\Xi_p)\right],$$

and we can decide to group the potentials g_p^{imp} with the test function f.

Now, we can make use of the framework of section 6.2 to figure out the idea of the decomposition formula (6.17). We denote

$$\Xi_k = (T_k \wedge T, X_{T_k \wedge T}) \quad \text{and} \quad Y_k = \Xi_{0:k}$$

and (6.3) gives that

$$\langle \gamma_k^\bullet, f\rangle = \mathbb{E}[f(Y_k)\prod_{p=1}^{k} g_p^\bullet(Y_p)] \quad \text{and} \quad \langle \mu_k^\bullet, f\rangle = \frac{\langle \gamma_k^\bullet, f\rangle}{\langle \gamma_k^\bullet, 1\rangle}.$$

In that case, the potential functions $g_p^\bullet$ take the following form

$$g_p^\bullet(y_p) = g_p^\bullet(x_0, \ldots, x_p) = g_p^{\mathrm{red}}(x_p). \tag{6.18}$$

Moreover, we have with (6.6) and (6.17) that

$$a_k^\bullet(y_k) = a_k(t_k) \quad \text{and} \quad b_k^\bullet(y_k) = \mathbb{I}_{(\chi_k, t_k)\in A_k},$$

where $y_k = (x_0, \ldots, x_k)$ and $x_p = (\chi_p, t_p) \in E \times [0, +\infty)$ for $p = 1, \ldots, k$.

To connect the two distributions γ_k and $\gamma_k^\bullet$, we use the function $T_k^\bullet f$ defined at equation (6.7) and equals here to

$$T_k^\bullet f(y_k) = T_k^\bullet f(x_0, \ldots, x_k) = f(x_k)\prod_{p=1}^{k} g_p^{\mathrm{imp}}(x_p),$$

so that

$$\langle \gamma_k, f\rangle = \mathbb{E}[f(\Xi_k)\prod_{p=1}^{k} g_p(\Xi_p)] = \mathbb{E}[T_k^\bullet f(Y_k)\prod_{p=1}^{k} g_p^\bullet(Y_p)] = \langle \gamma_k^\bullet, T_k^\bullet f\rangle.$$

In particular, with $f \equiv 1$, we obtain

$$\langle \gamma_k^\bullet, T_k^\bullet 1\rangle = \langle \gamma_k, 1\rangle = \mathbb{P}(T_k \leq T).$$

However, the normalizing constant $\langle \gamma_k^\bullet, 1\rangle$ does not seem to have any useful probabilistic interpretation.

6.3.2.2 Associated interacting particle system

The path particle system approximation It is now possible to re-interpret these formulae with the potential functions g_k^{red} and g_k^{imp} in the following way. To this end, let $\xi_k^i = \xi_{k,k}^i$ denote the final state of the trajectory $\xi_k^{\bullet,i} = (\xi_{0,k}^i, \ldots, \xi_{k,k}^i)$. We get with equation (6.18) that

$$g_k^{\bullet}(\xi_k^{\bullet,i}) = g_k^{\mathrm{red}}(\xi_k^i),$$

and we use the interacting particle system algorithm derived in section 6.2.3 in the framework of the present section. In particular, we deduce the following recursive formula for the approximated normalising constant

$$\langle \gamma_k^{\bullet,N}, 1 \rangle = \left(\frac{1}{N} \sum_{i=1}^{N} g_k^{\mathrm{red}}(\xi_k^i) \right) \langle \gamma_{k-1}^{\bullet,N}, 1 \rangle.$$

Then, if we set

$$v_k^i = T_k^{\bullet} 1(\xi_k^{\bullet,i}),$$

we remark also with equation (6.12) that

$$v_k^i = \prod_{p=1}^{k} g_p^{\mathrm{imp}}(\xi_{p,k}^i) = g_k^{\mathrm{imp}}(\xi_{k,k}^i) \prod_{p=1}^{k-1} g_p^{\mathrm{imp}}(\xi_{p,k}^i) = g_k^{\mathrm{imp}}(\xi_{k,k}^i) \prod_{p=1}^{k-1} g_p^{\mathrm{imp}}(\widehat{\xi}_{p,k-1}^i),$$

and the following estimates for the probabilities $\mathbb{P}(T_k \leq T)$ holds

$$\begin{aligned} P_k^N &= \left(\frac{1}{N} \sum_{i=1}^{N} g_k^{\mathrm{red}}(\xi_k^i) v_k^i \right) \langle \gamma_{k-1}^{\bullet,N}, 1 \rangle \\ &= \left(\frac{1}{N} \sum_{i=1}^{N} g_k^{\mathrm{red}}(\xi_k^i) v_k^i \right) \prod_{p=1}^{k-1} \frac{1}{N} \sum_{i=1}^{N} g_p^{\mathrm{red}}(\xi_p^i). \end{aligned}$$

The associated mixed population particle system The path particle system associated to the Feynman-Kac distribution defined of the path space of the Markov chain $\{\Xi_k,\ k = 1, \ldots, m\}$ requires to keep in memory N paths of the Markov chain. Such a way to proceed is actually a trick that disappears if we consider only the current state of the Markov chain with the cumulative weight associated to each path.

Here finally comes the following recursive non path-wise implementation, in terms of a mixed population, where for all $i = 1, \ldots, N$ the particle

$$\Pi_k^i = (\xi_k^i, v_k^i)$$

is a pair (position, auxiliary weight), evolving in the following way. Independently, for all $i = 1, \ldots, N$

1. a pair $\widehat{\Pi}_{k-1}^i = (\widehat{\xi}_{k-1}^i, \widehat{v}_{k-1}^i)$ is selected among the current population $(\Pi_{k-1}^1, \ldots, \Pi_{k-1}^N)$, according to their respective weights $(w_{k-1}^1, \ldots, w_{k-1}^N)$,
2. the variable ξ_k^i is sampled with the distribution $Q_k(\widehat{\xi}_{k-1}^i, \mathrm{d}z')$, and we set $v_k^i = g_k^{\mathrm{imp}}(\xi_k^i)\ \widehat{v}_{k-1}^i$,
3. according to equation (6.13), the normalized weight is defined as

$$w_k^i \propto g_k^{\mathrm{red}}(\xi_k^i).$$

6.3.3 The combined importance sampling and splitting algorithm

This procedure is rewritten below in algorithm 6.1 in terms of the original continuous time Markov process $\{X(t)\,,\, t \in [0,\infty)\}$ and in the special case $g_k^{\text{imp}}(x,t) = \dfrac{1}{a_k(t)}$ and $g_k^{\text{red}}(x,t) = a_k(t)\mathbb{1}_{\phi(x,t)\geq S_k}$. The algorithm and the produced estimates are referred with *I2S*, which abbreviates the mixture importance sampling and splitting used in this context.

Algorithm 6.1: I2S : a combined importance sampling and splitting algorithm

Data: $\{X(t),\ t \geq 0\}$ a càdlàg process with initial distribution η_0, a deterministic final time T, a rare event probability set B and a sequence of nested supersets $\{A_k,\ k = 1,\dots,m\}$ of $B \times [0,+\infty)$.

Initialization

  Set $J_0 = \{1,\dots,N\}$ and $\widehat{\theta}_0 = 0$.

  for $i = 1,\dots,N$ **do**

    Set $T_0^i = 0$ and sample X_0^i independently from law η_0.

    Set $v_0^i = 1/N$ and $w_0^i = 1/N$ the initial importance weights and re-sampling weights.

for $k = 1,\dots,m$ **do**

  for $i = 1,\dots,N$ **do**

    Select an index $j_i \in J_{k-1}$ from the weighted law $\displaystyle\sum_{j\in J_{k-1}} w_{k-1}^j \delta_j$.

    Set $\widehat{X}_{k-1}^i = X_{k-1}^I$, $\widehat{T}_{k-1}^i = T_{k-1}^I$ and $\widehat{v}_{k-1}^i = v_{k-1}^I$.

    Sample a path $t \mapsto X^i(t)$ starting from state $\widehat{X}_{k-1}^i$ at time $\widehat{T}_{k-1}^i$ until time $T_k^i \wedge T$,

    where $T_k^i = \inf\{t \geq \widehat{T}_{k-1}^i : (X^i(t),t) \in A_k\}$.

    Set the importance weights v_k^i and the re-sampling weights w_k^i as

$$v_k^i = \frac{1}{a_k(T_k^i)}\,\widehat{v}_{k-1}^i \quad \text{and} \quad w_k^i = a_k(T_k^i)\,\mathbb{1}_{T_k^i \leq T}.$$

  Normalize the re-sampling weights.

  Set $J_k = \{i = 1,\dots,N\,,\, T_k^i \leq T\}$ and define

$$\widehat{\theta}_k = \left(\frac{1}{N}\sum_{i\in J_k} a_k(T_k^i)\right)\widehat{\theta}_{k-1}$$

  and

$$\widehat{P}_k^{\text{I2S}} = \frac{\displaystyle\sum_{i\in J_k} a_k(T_k^i)\,v_k^i}{\displaystyle\sum_{i\in J_k} a_k(T_k^i)}\,\widehat{\theta}_k$$

Estimation

$$\mathbb{P}(T_B \leq T) \approx \widehat{P}_m^{\text{I2S}}$$

6.3.4 Central limit theorem

This section gives central limit theorem for the estimation of the rare event probability. It is given in terms of the Feynman-Kac distributions and with a more explicit formulation.

As usual, we first define the life time of the algorithm with

$$\tau^N = \inf\left\{ k \geq 0,\ \sum_{i=1}^{N} \mathbb{I}_{T_k^i \leq T} = 0 \right\},$$

for which the exponential lower bound holds [Del Moral, 2004, theo 7.4.1]

$$\mathbb{P}(\tau^N \leq m) \leq a_m \exp(-b_m N).$$

Theorem 6.2. *Assume the a_p functions to be bounded. The central limit theorem for the relative error of the estimator $\widehat{P}_m^{I2S}$ given by the I2S algorithm 6.1 stands as follows*

$$\sqrt{N}\,\frac{\widehat{P}_m^{I2S} - P_B}{P_B}\,\mathbb{I}_{\tau^N > m} \underset{N\to+\infty}{\overset{Law}{\Longrightarrow}} \mathcal{N}(0, V_m),$$

where

$$V_m = \left[\frac{\langle \eta_0, (R_{1:m}1)^2 \rangle}{\langle \eta_0, R_{1:m}1 \rangle^2} - 1 \right] + \sum_{k=1}^{m} \left[\langle \gamma_{k-1}^{red}, 1 \rangle \frac{\langle \gamma_k^{imp}, g_k^{red} (R_{k+1:m}1)^2 \rangle}{\langle \gamma_k, R_{k+1:m}1 \rangle^2} - 1 \right].$$

Here, the functions $R_{k+1:m} f$ function equals to

$$\begin{aligned} R_{k+1:m} f(z) &= \mathbb{E}[f(Z_m) \prod_{p=k+1}^{m} g_p(Z_p) | Z_k = z] \\ &= \mathbb{E}[f(Z_m)\, \mathbb{I}_{T_B \leq T} | Z_k = z], \end{aligned}$$

namely, denoting $z = (x, t)$ we have

$$R_{k+1:m} f(x,t) = \mathbb{E}[f(X(T_B), T_B)\, \mathbb{I}_{T_B \leq T} | X(t) = x].$$

The explicit formulation of the variance is also obtained in term of the a_p functions and

$$u_B(\chi, t) = \mathbb{P}(T_n \leq T | X_t = \chi).$$

is given by

$$V_m = \sum_{k=1}^{m} \left[\frac{\mathbb{E}\left(\prod_{p=0}^{k-1} a_p(T_p) | T_{k-1} \leq T \right) \mathbb{E}\left(u_B(X(T_k), T_k)^2 \prod_{p=0}^{k-1} \frac{1}{a_p(T_p)} | T_{k-1} \leq T \right)}{p_k \langle \mu_k, u_B \rangle^2} - 1 \right]$$

Proof. When importance sampling is added to the splitting algorithm, the variance takes the form [Le Gland, 2007]:

$$V_m = \frac{\mathrm{var}(R_{1:m}1, \eta_0)}{\langle \eta_0, R_{1:m}1 \rangle^2} + \sum_{k=1}^{m} C_k + \sum_{k=1}^{m} D_k,$$

where

$$C_k = \frac{\langle \gamma_{k-1}^{\mathrm{imp}}, 1 \rangle \langle \gamma_{k-1}^{\mathrm{red}}, 1) \rangle}{\langle \gamma_{k-1}, 1 \rangle^2} \frac{\mathrm{var}(g_k\, R_{k+1:m}1, \mu_{k-1}^{\mathrm{imp}} Q_k)}{\langle \mu_{k-1}, R_{k:m}1 \rangle^2}$$

and

$$D_k = \frac{\langle \gamma_{k-1}^{\text{imp}}, 1\rangle \langle \gamma_{k-1}^{\text{red}}, 1\rangle}{\langle \gamma_{k-1}, 1\rangle^2} \frac{\langle \mu_{k-1}^{\text{imp}}, R_{k:m} 1\rangle^2}{\langle \mu_{k-1}, R_{k:m} 1\rangle^2} - 1.$$

The distribution μ_k^{red} and μ_k^{imp} are the normalised distributions associated with γ_k^{red} and γ_k^{imp}, namely

$$\mu_k^{\text{red}} = \frac{\gamma_k^{\text{red}}}{\langle \gamma_k^{\text{red}}, 1\rangle} \quad \text{and} \quad \mu_k^{\text{imp}} = \frac{\gamma_k^{\text{imp}}}{\langle \gamma_k^{\text{imp}}, 1\rangle},$$

and following [Le Gland, 2007], the unnormalised distributions γ_k^{red} and γ_k^{imp} are defined as follows

$$\begin{aligned} \langle \gamma_k^{\text{red}}, f\rangle &= \mathbb{E}[f(Z_k) \prod_{p=1}^{k} g_p^{\text{red}}(Z_p)], \\ \langle \gamma_k^{\text{imp}}, f\rangle &= \mathbb{E}[f(Z_k) \prod_{p=1}^{k} \left(g_p^{\text{imp}}(Z_p)\right)^2 \prod_{p=1}^{k} g_p^{\text{red}}(Z_p)]. \end{aligned} \tag{6.19}$$

In the original paper [Le Gland, 2007], γ_k^{imp} is denoted as $\gamma_k^{\square}$, and it is rewritten here for the sake of clarity.
The non-negative distributions γ_k^{red} and γ_k^{imp} satisfy the recurrent relations

$$\gamma_k^{\text{red}} = \gamma_{k-1}^{\text{red}} R_k^{\text{red}} \quad \text{and} \quad \gamma_k^{\text{imp}} = \gamma_{k-1}^{\text{imp}} R_k^{\text{imp}},$$

where the non-negative kernels R_k^{red} and R_k^{imp} are defined by

$$R_k^{\text{red}}(z, \mathrm{d}z') = g_k^{\text{red}}(z')\, Q_k(z, \mathrm{d}z')$$

and

$$R_k^{\text{imp}}(z, \mathrm{d}z') = (g_k^{\text{imp}}(z'))^2\, g_k^{\text{red}}(z')\, Q_k(z, \mathrm{d}z').$$

To compute the V_m term, first notice that

$$\langle \mu_{k-1}^{\text{imp}} Q_k, g_k R_{k+1:m} 1\rangle = \langle \mu_{k-1}^{\text{imp}}, R_{k:m} 1\rangle,$$

which results from the the calculation below

$$\begin{aligned} \langle \gamma_{k-1}^{\text{imp}} Q_k, g_k R_{k+1:m} 1\rangle &= \mathbb{E}[Q_k(g_k R_{k+1:m})(X_k) \prod_{p=0}^{k-1} g_p^{\text{imp}}(X_p)] \\ &= \mathbb{E}\left[\mathbb{E}\left[g_k(X_k)\mathbb{E}[\prod_{p=k+1}^{n} g_p(X_p)|X_k]X_{k-1}\right] \prod_{p=0}^{k-1} g_p^{\text{imp}}(X_p)\right] \\ &= \mathbb{E}\left[\mathbb{E}[\prod_{p=k}^{n} g_p(X_p)|X_{k-1}] \prod_{p=0}^{k-1} g_p^{\text{imp}}(X_p)\right] \\ &= \langle \gamma_{k-1}^{\text{imp}}, R_{k:m} 1\rangle, \end{aligned}$$

and we obtain the following identity

$$\begin{aligned} \operatorname{var}(g_k R_{k+1:m} 1, \mu_{k-1}^{\text{imp}} Q_k) &= \langle \mu_{k-1}^{\text{imp}} Q_k, \left(g_k R_{k+1:m} 1\right)^2\rangle - \langle \mu_{k-1}^{\text{imp}} Q_k, \left(g_k R_{k+1:m} 1\right)\rangle^2 \\ &= \langle \mu_{k-1}^{\text{imp}} Q_k, \left(g_k R_{k+1:m} 1\right)^2\rangle - \langle \mu_{k-1}^{\text{imp}}, R_{k:m} 1\rangle^2. \end{aligned}$$

From this equality and using the fact that

$$g_k^2 = g_k,$$

we obtain

$$\begin{aligned} C_k + D_k &= \frac{\langle \gamma_{k-1}^{\text{imp}}, 1\rangle \langle \gamma_{k-1}^{\text{red}}, 1\rangle}{\langle \gamma_{k-1}, 1\rangle^2} \frac{\langle \mu_{k-1}^{\text{imp}} Q_k, g_k (R_{k+1:m}1)^2\rangle}{\langle \mu_{k-1}, R_{k:m}1\rangle^2} - 1 \\ &= \langle \gamma_{k-1}^{\text{red}}, 1\rangle \frac{\langle \gamma_{k-1}^{\text{imp}} Q_k, g_k (R_{k+1:m}1)^2\rangle}{\langle \gamma_{k-1}, R_{k:m}1\rangle^2} - 1. \end{aligned}$$

Finally, notice that for any bounded measurable function f

$$\begin{aligned} \langle \gamma_{k-1}^{\text{imp}} Q_k, g_k f\rangle &= \int_{z'} g_k(z') f(z') \int_z \gamma_{k-1}^{\text{imp}}(dz)\, Q_k(z, dz') \\ &= \int_{z'} (g_k(z'))^2 f(z') \int_z \gamma_{k-1}^{\text{imp}}(dz)\, Q_k(z, dz'), \quad \text{since } g_k^2 = g_k \\ &= \int_{z'} g_k^{\text{red}}(z') f(z') \int_z \gamma_{k-1}^{\text{imp}}(dz) (g_k^{\text{imp}}(z'))^2 g_k^{\text{red}}(z')\, Q_k(z, dz') \\ &= \int_{z'} g_k^{\text{red}}(z') f(z') \int_z \gamma_{k-1}^{\text{imp}}(dz)\, R_k^{\text{imp}}(z, dz') \\ &= \int_{z'} g_k^{\text{red}}(z') f(z')\, \gamma_k^{\text{imp}}(dz') \\ &= \langle \gamma_k^{\text{imp}}, g_k^{\text{red}} f\rangle. \end{aligned}$$

Thus,

$$C_k + D_k = \langle \gamma_{k-1}^{\text{red}}, 1\rangle \frac{\langle \gamma_k^{\text{imp}}, g_k^{\text{red}} (R_{k+1:m}1)^2\rangle}{\langle \gamma_k, R_{k+1:m}1\rangle^2} - 1$$

which is the expression given in the theorem.

To have the explicit formulation in terms of the a_p functions, we recall that

$$\langle \gamma_k, R_{k+1:m}1\rangle = \langle \gamma_k, 1\rangle \langle \mu_k, R_{k+1:m}1\rangle = \mathbb{P}(T_k \le T) \langle \mu_k, u_B\rangle.$$

The Feynman-Kac distributions given in the expression of V_m in the theorem are defined by

$$\langle \gamma_k, f\rangle = \mathbb{E}[f(Z_k) \prod_{p=1}^{k} g_p(Z_p)] = \mathbb{E}[f(Z_k) \prod_{p=1}^{k} \mathbb{1}_{T_k \le T}],$$

$$\langle \gamma_k^{\text{red}}, f\rangle = \mathbb{E}[f(Z_k) \prod_{p=1}^{k} g_p^{\text{red}}(Z_p)] = \mathbb{E}[f(Z_k) \prod_{p=1}^{k} a_p(T_p) \mathbb{1}_{T_k \le T}]$$

and

$$\langle \gamma_k^{\text{imp}}, f\rangle = \mathbb{E}[f(Z_k) \prod_{p=1}^{k} (g_p^{\text{imp}}(Z_p))^2 \prod_{p=1}^{k} g_p^{\text{red}}(Z_p)] = \mathbb{E}[f(Z_k) \prod_{p=1}^{k} \frac{1}{a_p(T_p)} \mathbb{1}_{T_k \le T}],$$

and we have

$$\langle \gamma_k^{\text{imp}}, g_k^{\text{red}} f\rangle = \mathbb{E}[f(Z_k) \prod_{p=1}^{k-1} \frac{1}{a_p(T_p)} \mathbb{1}_{T_k \le T}].$$

In particular, we obtain the following equalities

$$\langle \gamma_{k-1}^{\text{red}}, 1\rangle = \mathbb{E}[\prod_{p=1}^{k-1} a_p(T_p) \mathbb{1}_{T_{k-1} \le T}],$$

$$\langle \gamma_k^{\text{imp}}, g_k^{\text{red}}(R_{k+1:m}1)^2 \rangle = \mathbb{E}[(u_B(X(T_k), T_k))^2 \prod_{p=1}^{k-1} \frac{1}{a_p(T_p)} \mathbb{1}_{T_k \leq T}],$$

and if we use these expressions in V_m, we obtain the following expression

$$\begin{aligned} &\langle \gamma_{k-1}^{\text{red}}, 1 \rangle \frac{\langle \gamma_k^{\text{imp}}, g_k^{\text{red}}(R_{k+1:m}1)^2 \rangle}{\langle \gamma_k, R_{k+1:m}1 \rangle^2} - 1 \\ &\qquad = \frac{\mathbb{E}[\prod_{p=1}^{k-1} a_p(T_p) \mathbb{1}_{T_{k-1} \leq T}] \mathbb{E}[(u_B(X(T_k), T_k))^2 \prod_{p=1}^{k-1} \frac{1}{a_p(T_p)} \mathbb{1}_{T_k \leq T}]}{\mathbb{P}(T_B \leq T)^2} - 1 \end{aligned},$$

which is the expression we gave in [Jacquemart et al., 2013].

To find the more explicit expression in term of the a_p functions, we remark that

$$\mathbb{E}[\prod_{p=1}^{k-1} a_p(T_p) \mathbb{1}_{T_{k-1} \leq T}] = \mathbb{E}[\prod_{p=1}^{k-1} a_p(T_p) | T_{k-1} \leq T] \mathbb{P}(T_{k-1} \leq T),$$

and

$$\begin{aligned} &\mathbb{E}[(u_B(X(T_k), T_k))^2 \prod_{p=1}^{k-1} \frac{1}{a_p(T_p)} \mathbb{1}_{T_k \leq T}] \\ &\qquad = \mathbb{E}[(u_B(X(T_k), T_k))^2 \prod_{p=1}^{k-1} \frac{1}{a_p(T_p)} | T_k \leq T] \mathbb{P}(T_k \leq T), \end{aligned}$$

and we recall that

$$\mathbb{P}(T_B \leq T) = \mathbb{P}(T_k \leq T) \langle \mu_k, u_B \rangle,$$

which implies that

$$\begin{aligned} &\frac{\mathbb{E}[\prod_{p=1}^{k-1} a_p(T_p) \mathbb{1}_{T_{k-1} \leq T}] \mathbb{E}[(u_B(X(T_k), T_k))^2 \prod_{p=1}^{k-1} \frac{1}{a_p(T_p)} \mathbb{1}_{T_k \leq T}]}{\mathbb{P}(T_B \leq T)^2} - 1 \\ &= \mathbb{E}[\prod_{p=1}^{k-1} a_p(T_p) | T_{k-1} \leq T] \mathbb{E}[(u_B(X(T_k), T_k))^2 \prod_{p=1}^{k-1} \frac{1}{a_p(T_p)} | T_k \leq T] \frac{\mathbb{P}(T_{k-1} \leq T) \mathbb{P}(T_k \leq T)}{\mathbb{P}(T_k \leq T)^2 \langle \mu_k, u_B \rangle^2} - 1 \\ &= \frac{\mathbb{E}[\prod_{p=1}^{k-1} a_p(T_p) | T_{k-1} \leq T] \mathbb{E}[(u_B(X(T_k), T_k))^2 \prod_{p=1}^{k-1} \frac{1}{a_p(T_p)} | T_k \leq T]}{p_k \langle \mu_k, u_B \rangle^2} - 1, \end{aligned}$$

because the conditional probability p_k equals to

$$p_k = \mathbb{P}(T_k \leq T | T_{k-1} \leq T) = \frac{\mathbb{P}(T_k \leq T)}{\mathbb{P}(T_{k-1} \leq T)}.$$

□

Remark 6.3. *If the weight functions a_p are constant and equal to $a_p \equiv 1$ then the re-sampling among the successful trajectories is uniform. In that case, the weighted splitting algorithm 6.1 and the standard splitting algorithm 3.1 are strictly the same algorithm. It is worth noting that*

the expression of the asymptotic variance given in theorem 6.2 if $a_p \equiv 1$ is the same as the one of the standard splitting given in section 3.2

$$\begin{aligned}
&\frac{\mathbb{E}[\prod_{p=1}^{k-1} a_p(T_p)|T_{k-1} \le T]\mathbb{E}[(u_B(X(T_k),T_k))^2 \prod_{p=1}^{k-1} \frac{1}{a_p(T_p)}|T_k \le T]}{p_k \langle \mu_k, u_B\rangle^2} - 1 \\
&= \frac{\mathbb{E}[(u_B(X(T_k),T_k))^2|T_k \le T]}{p_k \langle \mu_k, u_B\rangle^2} - 1 \\
&= \frac{\langle \mu_k, u_B^2\rangle}{p_k \langle \mu_k, u_B\rangle^2} - 1 \\
&= [\frac{1}{p_k} - 1] + \frac{1}{p_k}[\frac{\langle \mu_k, u_B^2\rangle}{\langle \mu_k, u_B\rangle^2} - 1].
\end{aligned}$$

We see here the same expression of the asymptotic variance given by the standard splitting algorithm.

6.4 Numerical experiments

The algorithm is tested on the Brownian bridge $\{X_t, 0 \le t \le 1\}$, which is solution of the following real stochastic differential equation

$$\mathrm{d}X_t = \frac{X_t}{t-1}\mathrm{d}t + \mathrm{d}W_t,$$

with $X_0 = 0$. We can show that $X_1 = 0$ almost surely and that

$$\mathbb{P}(T_S \le 1) = \mathbb{P}(\sup_{0\le t\le 1} X_t \ge S) = \exp^{-2S^2}. \tag{6.20}$$

The sampled paths of the Brownian bridge are obtained with the Euler scheme [Kloeden and Platen, 1992, chap 9].

The intermediate regions A_k are simply defined here with

$$A_k = \{(\chi, t), \chi \ge S_k\}$$

for a sequence of thresholds that can easily be chosen adaptively, as explained in chapter 4, or using the formula (6.20). Here, the expected conditional probability values p_k that the process reaches S_k starting from S_{k-1} equals to 0.5. We can also notice that the estimation of thresholds can efficiently be done with a small number of particles. Here, 10 would be sufficient as numerically shown in section 4.7. However the sequence of thresholds S_k was chosen using the formula (6.20).

The number m of intermediate regions is related to the probability of interest $P_S = \mathbb{P}(T_S < T)$ with the formula $P_S = \prod_{k=1}^{m} p_k = p^m$. Thus, $m = \log P / \log p$ and the total number of trajectories of the Markov chain used the algorithm is then in $\mathcal{O}(m \times N)$. Besides, the multinomial re-sampling used at each step can be in $\mathcal{O}(N)$. Neglecting the computation of the importance weights and the re-sampling weights, the complexity of the algorithm is then related to the rarity of the event and equals to $\mathcal{O}(m \times N)$. This result stands for I2S algorithm 6.1 and standard splitting algorithm 3.1 alike.

The $a_p(t)$ functions have to be soft enough to preserve some diversity. According to our experiments, an exponential selection is too strong and does not ensure variance reduction anymore. Furthermore, an easy tuning is required to proceed to several tests. For all these reasons, the a_p functions are chosen in the parametric family defined by

$$a_p(t) = a_p^\alpha(t) = \frac{1}{t^\alpha}, \tag{6.21}$$

for $\alpha \geq 0$. Notice that $\alpha = 0$ corresponds to the standard splitting case.

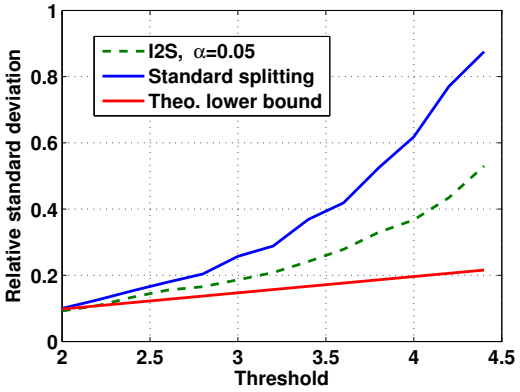
For any real N-sample $(x_1, \ldots, x_N)$, the accuracy of an estimator is measured by its relative standard deviation (*rSTD*), defined as the ratio of the empirical standard deviation to the empirical mean (see section 3.1).

I2S VS splitting algorithm The I2S and the splitting algorithms are unbiased. In figure 6.3a, we compare the rSTD given by the probability estimation of equation (6.20) with standard splitting algorithm, and I2S algorithm (algorithm 6.1). We choose $\alpha = 0.05$. The I2S algorithm gives better rSTD for every final threshold S.
We compare I2S rSTD with the lower bound for the rSTD in the estimation with the splitting algorithm detailed in section 3.2.4. To compute the lower bound V_m^* of the relative variance, we use the fact that $p_k = 0.5$. Hence, taking into account that $p_1 \cdots p_m = \mathbb{P}(T_S \leq T)$ the number of threshold m equals to $m = \log(\mathbb{P}(T_S \leq T))/\log(0.5)$. Finally, using the $\sqrt{N}$ convergence rate given by the central limit theorem 3.1, the red curve in figure 5.10 is in fact the curve

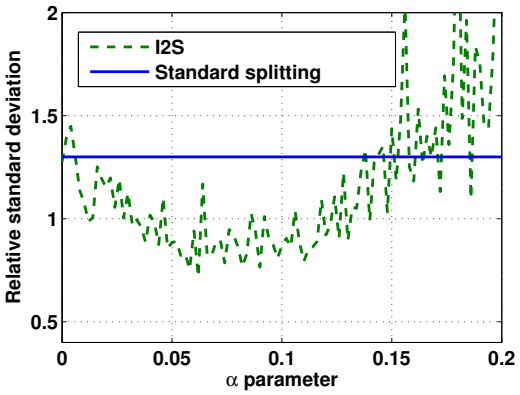
$$S \mapsto \sqrt{\frac{1}{N}\frac{\log \mathbb{P}(T_B \leq T)}{\log(0.5)}(1/0.5 - 1)}.$$

We see that using a non-optimal importance function (equivalently some non-optimal regions) we are about halfway between the performances of the splitting algorithm with "naive" importance function $x \mapsto x$ and the splitting with optimal region. Table 6.2 gives additional results. We observe in particular that the run time is the same using the standard splitting algorithm 3.1 or the I2S algorithm.

In figure 6.3b, we present the evolution of the rSTD in function of parameter α. The rSTD of the splitting algorithm is independent of the α value and is drawn with the red line. We observe that I2S algorithm gives a better rSTD than the splitting for $\alpha \in]0;0,15[$. An optimum value of α with regard to the minimization of the rSTD can be found. However, the values $\alpha > 0.15$ gives greater rSTD than the standard splitting algorithm. The results of figures 6.3b was obtained with $N = 1000$ particles and over 250 retrials for each α. To understand the gain in probability estimation via I2S algorithm, table 6.1 compares rSTD of I2S algorithm with parameters $S = 4$, $\alpha = 0.05$ and $N = 5000$ and rSTD of standard splitting for different values of N. Estimations are obtained over 250 retrials. We need at least twice as the number of trajectories in the splitting algorithm to reach the same accuracy than the I2S algorithm.



(a) Splitting and I2S algorithms.



(b) I2S rSTD with different functions $a_p^{\alpha}(t)$.

Figures 6.3: Comparison of I2S algorithm and the splitting

	I2S	Standard splitting			
N	5000	5000	7500	10000	12500
rSTD	0.36	0.61	0.49	0.47	0.35
time (sec.)	5.55	5.37	8.00	10.66	13.30

Table 6.1: Convergence of standard splitting.

S	$\mathbb{P}(T_S \leq 1)$	Opt rSTD	Non-optimal splitting		I2S algorithm	
			estimate	time (sec.)	estimate	time (sec.)
2	$3.4.10^{-4}$	8.9%	$2.93.10^{-4} \pm 10\%$	2.15	$3.00.10^{-4} \pm 9.2\%$	2.12
2.8	$1.5.10^{-7}$	11%	$1.27.10^{-7} \pm 20\%$	3.85	$1.25.10^{-7} \pm 17\%$	4.02
3.4	$9.1.10^{-11}$	12%	$6.72.10^{-11} \pm 37\%$	5.60	$6.94.10^{-11} \pm 24\%$	5.68
4	$1.3.10^{-14}$	13%	$8.64.10^{-15} \pm 62\%$	7.50	$9.90.10^{-15} \pm 36\%$	7.63

Table 6.2: Comparison of the rSTD given by optimal standard splitting.

Conclusion of the chapter

Contributions

- An original interacting particle system with non-negative potential functions for the approximation of rare event probability and for the construction of paths of the Markov chain approximately distributed given the rare event is reached. In particular, we constructed an unified framework for the splitting algorithm 3.1, the static splitting algorithm [Cérou et al., 2006a] and the weighted redistribution algorithm.
- An extension to the Del Moral and Granier algorithm to the case of non-negative potential functions.
- I2S algorithm, an original modification of the splitting algorithm that takes into account how early an intermediate threshold is reached.
- A central limit theorem for the asymptotic fluctuations of the estimated probability with the I2S algorithm.
- We numerically conclude that the rSTD can be improved without modifying the importance function.

Limitations and links with the next chapter

- Although some numerical results have shown that the I2S algorithm gives better variance than the standard splitting algorithm, the practical side has not been deeply studied and an automatic tuning parameter is still an open question,
- In the same line of thinking, the theoretical framework derived in this chapter for the splitting algorithm 3.1 and the weighted redistribution algorithm 3.2 does not give any indications for the tuning of the latter mentioned algorithm. This question will be addressed in the next chapter.
- For the moment, we have tried to improve the splitting algorithm with optimal importance function as in chapter 5 of combining it with some importance sampling as in the present chapter. In the next chapter, we want also to find an other algorithm that can be used instead of the splitting algorithm for the estimation of the probability $\mathbb{P}(T_S \leq T)$.

CHAPTER 7

Elaboration of an automatic procedure for parameter selection in the weighted redistribution algorithm and extension to hitting time probability estimation

In this chapter

- We make the tuning of the weighted redistribution algorithm automatic.
- We derive a procedure that works for the estimations of exceedance probability and probability density function alike.
- We extend the weighted redistribution algorithm to compute some hitting time probabilities.

7.1 Introduction

Motivation In section 1.2, we presented a state of the art for estimation methods to compute probabilities such that

$$\mathbb{P}(X(t) \in B \text{ for some } t \leq T) \tag{7.1}$$

for $\{X(t),\ t \geq 0\}$ a continuous-time continuous-state space stochastic process. The splitting algorithm 3.1, and especially its adaptive version given in chapter 4 is a very simple way to estimate such probabilities. The question initially addressed in this chapter is :

Is the splitting algorithm the only algorithm, easy tunable, to compute (7.1) ?

We will see in section 7.3 that the weighted redistribution algorithm presented in chapter 3.3 answers the question. Nevertheless, a tuning is required and there is no automatic procedure proposed in the scientific literature. See section 1.3 for a state of the art of the weighted redistribution algorithm.

This chapter is organised as follows. We first propose an algorithm to find appropriate selection functions to be used in the weighted redistribution algorithm in its original framework (detailed in section 3.3). We detail procedures for the estimation of threshold exceedance probabilities and tail probability density function (p.d.f.) point-wise estimation. Then, we show how the weighted redistribution algorithm can be used to give estimations of the probability (7.1). The proposed methods are first tested on estimation problems that were already addressed with the weighted redistribution algorithm in the scientific literature. Then, we numerically compare the splitting and the weighted redistribution algorithm for the estimation of the probability (7.1).

7.2 Adaptive version of the weighted redistribution algorithm

Reminder Let $\{X_k,\ k = 0,\dots,m\}$ be a Markov chain. We remind that the goal of the weighted redistribution algorithm is to estimate

$$\mathbb{P}(V(X_m) \in A) \tag{7.2}$$

for m a fixed integer, V a real valued function and A a subset of $\mathbb{R}$. For each $k = 1,\dots,m$, we denote E_k the state space of the random variable X_k. We remind also that the weighted redistribution algorithm consists of a set of N random sequences $(\widehat{X}^i_{0:m},\ 1 \le i \le N)$ for which the recursive construction can be summed up as follows

$$\begin{pmatrix} \widehat{X}^{(1)}_{0:k} \\ \vdots \\ \vdots \\ \widehat{X}^{(N)}_{0:k} \end{pmatrix} \xrightarrow{\text{Selection}} \begin{pmatrix} \check{X}^{(1)}_{0:k} \\ \vdots \\ \vdots \\ \check{X}^{(N)}_{0:k} \end{pmatrix} \xrightarrow{\text{Mutation}} \begin{pmatrix} \widehat{X}^{(N)}_{0:k+1} \\ \vdots \\ \vdots \\ \widehat{X}^{(N)}_{0:k+1} \end{pmatrix}$$

See section 3.3 for further details.

7.2.1 Hypothesis for the selection functions

We discuss here the choice of the selection functions G_k introduced in section 3.3 to be used in the weighted redistribution algorithm. A usual way to proceed is to take the G_k functions equal for all $k = 1,\dots,m$ and to assume they belong to a parametric family. All the examples encountered in the scientific literature assume the parameter defined on the G_k functions is unidimensional [Carmona and Crepey, 2010, Carmona et al., 2009, Garnier and Del Moral, 2006, Giesecke et al., 2010]. Also, we can reduce choices to non-negative ones. More formally, we suppose that

$$G_k \in \{G_k^\alpha,\ \alpha \in \mathbb{R}^+\}. \tag{7.3}$$

Without loss of generality, we assume that the G_k^α functions are increasing functions of α in the following sense

$$\forall y_k \in F_k,\ \left[\alpha \le \alpha' \Rightarrow G_k^\alpha(y_k) \le G_k^{\alpha'}(y_k)\right], \tag{7.4}$$

where $y_k = (x_0,\dots,x_k)$ and $F_k = E_0 \times \cdots \times E_k$. Namely, the higher is the value α, the stronger is the selection.

Based on large deviation theory, and assuming that $E_0 = \ldots = E_n$, the original article [Del Moral and Garnier, 2005] proposes to use the following family of potential functions

$$G_k^\alpha(y_k) = G_k^\alpha(x_1,\dots,x_k) = \exp[\alpha\,(V(x_k) - V(x_{k-1}))]. \tag{7.5}$$

All the selection functions proposed in the scientific literature fulfils equations (7.3) and (7.4). This remark is valid for exponential selection functions [Carmona and Crepey, 2010, Carmona et al., 2009, Garnier and Del Moral, 2006, Giesecke et al., 2010]. and also for the other selection functions proposed in [Carmona and Crepey, 2010]. In the following, all the considered selection functions are assumed to fulfil (7.3) and (7.4).

7.2.2 Estimation of threshold exceedance probability

The rare event under consideration is characterized here by some threshold exceedance of the real valued random variable $V(X_m)$, for some fixed positive integer m. If B is a real number, we denote the rare set of interest by

$$A_B = [B, +\infty)$$

and the corresponding rare event we wish to estimate its probability stands as follows

$$\{V(X_m) \geq B\}.$$

7.2.2.1 Heuristic for the choice of a good parameter

The algorithm 3.2 gives as outputs N trajectories $\{\widehat{X}^{(N)}_{0:m}, 1 \leq i \leq N\}$ that are approximately sampled from the distribution defined with equation (3.9). The procedure is based on two following ideas.

Ideas for the choice of good parameters

1. If none of these trajectories have a terminal value such that $V(\widehat{X}^i_m) \geq B$ for $i = 1,\dots,N$, the probability is estimated by 0. Consequently, the selection was too weak, and then we have to increase the α parameter. See figure 7.1a.

2. If every trajectory is such that $V(\widehat{X}^i_m) \geq B$ for $i = 1,\dots,N$, the selection was too strong. As a result, the diversity is impoverished, which increases the variance in the estimation. That is why the α parameter have to be reduced. See figure 7.1b.

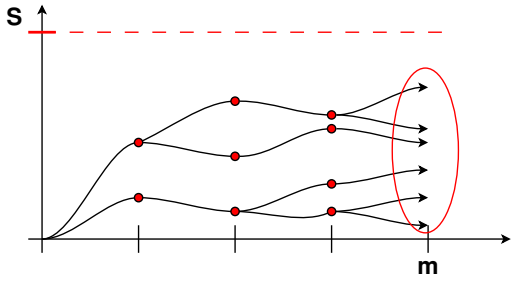
Thus, an appropriate α should be such that only a part of the trajectories reaches the rare set A_B. To take into account this remark, let $\pi^N_B(\alpha)$ and $\Pi^N_B(\alpha)$ denote the proportion of elements into the set $\{V(\widehat{X}^i_m), i = 1,\dots,N\}$ that pass below and above the threshold B at the end of the algorithm 3.2 :

$$\pi^N_B(\alpha) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{V(\widehat{X}^i_m)\leq B} \quad \text{and} \quad \Pi^N_B(\alpha) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{V(\widehat{X}^i_m)\geq B}.$$

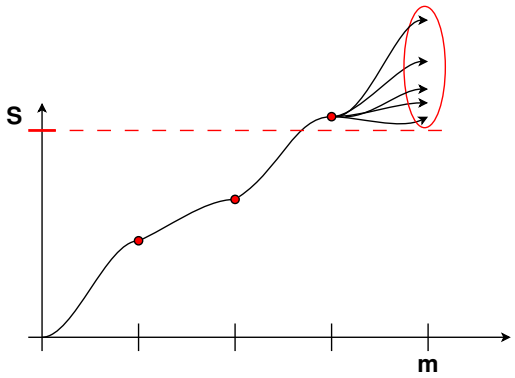
According to the two observations above, we expect that a good α parameter, denoted by $\hat{\alpha}$, is such that $\pi^N_B(\hat{\alpha})$ and $\Pi^N_B(\hat{\alpha})$ are in some reasonable order of magnitude. Namely, the $\hat{\alpha}$ parameter should satisfy these two conditions

$$\pi^N_B(\hat{\alpha}) > p_0 \quad \text{and} \quad \Pi^N_B(\hat{\alpha}) > p_0, \quad \text{for some } p_0 > 0. \tag{7.6}$$

The bound p_0 of the proportion of trajectories that passes above and below the threshold B belongs thus to $(0, 1/2)$. So that, we can measure if the selection was too weak or too strong for a given α. It is now sufficient to compute the quantities $\pi^N_B(\alpha)$ and $\Pi^N_B(\alpha)$.



(a) Selection too weak : none of the trajectories reaches the set at final time.



(b) Selection too strong : every trajectory reaches the set at final time. The diversity is impoverished.

Figure 7.1: Illustration of the heuristic.

7.2.2.2 Adaptive version of the weighted redistribution algorithm

To find an $\hat{\alpha}$ parameter that satisfies the inequalities (7.6), we propose the following method. Given two parameters $\alpha_* < \alpha^*$ such as $\pi_B^N(\alpha_*) = 0$ and $\Pi_B^N(\alpha^*) = 1$, we make a dichotomy into the real interval $[\alpha_*; \alpha^*]$ to obtain a good $\hat{\alpha}$ parameter. Algorithm 7.1 details this dichotomy. As a result, algorithm 7.1 gives a way for selecting an α parameter which fulfils conditions given by inequalities (7.6) for a given p_0. Moreover, it gives an estimate of the rare event probability (7.2) with an appropriate α parameter. Notice that the condition "$\pi_B^N(\alpha) < p_0$ or $\Pi_B^N(\alpha) < p_0$" into the **while** loop is the negation of (7.6).

7.2.3 Estimation of probability density function tail

In this paragraph, we are interested in estimating the p.d.f. tail of the real valued random variable $V(X_m)$, for some fixed positive integer m.

Algorithm 7.1: Adaptive estimations of parameter and rare event probability

Data: Threshold B and rare event probability set A_B, number N of trajectories in algorithm 7.1, α_* and α^* such that $\pi_B^N(\alpha_*) = 0$ and $\Pi_B^N(\alpha^*) = 1$, $p_0 \in (0, 1/2)$

Result: Parameter $\widehat{\alpha}$ and $\widehat{P}(A_B, \widehat{\alpha})$ an estimation of $\mathbb{P}(V(X_m) \geq B)$

```
Set π = 0 and Π = 1
while π < p_0 or Π < p_0 do
    Set α = (α_* + α^*)/2.
    Run algorithm 3.2 to compute P̂(A_B, α), π_B^N(α) and Π_B^N(α).
    Set π = π_B^N(α) and Π = Π_B^N(α)
    if Π > π then
        Set α^* = α.
    else
        Set α_* = α.
```

Result: $\hat{\alpha} = \alpha$; $\mathbb{P}(V(X_m) \geq B) \approx \widehat{P}(A_B, \widehat{\alpha})$

7.2.3.1 Discrete state spaces

Here, the state space of the Markov chain $\{X_k,\ k = 1, \ldots, m\}$ is assumed to be discrete. Thus, the rare event is assumed to be of the form

$$A^{\mathrm{d}}(b) = \{V(X_m) = b\},$$

for large values of b. Given some $\alpha \geq 0$, the probability of the event $A^{\mathrm{d}}(b)$ is then estimated with algorithm 3.2 by

$$\mathbb{P}(A^{\mathrm{d}}(b)) = \mathbb{P}(V(X_m) = b) \approx \hat{P}(A^{\mathrm{d}}(b), \alpha).$$

As pointed out in [Carmona and Crepey, 2010], the α that should be retained is the one giving the greatest number of trajectories such that $V(\widehat{X}_m^i) = b$ for $i = 1, \ldots, N$. Algorithm 7.1 can be readily adapted to this case. For this purpose, one has to run algorithm 7.1 with the following functions

$$\pi_b^{\mathrm{d},N}(\alpha) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}_{V(\widehat{X}_m^i)<b} \quad \text{and} \quad \Pi_b^{\mathrm{d},N}(\alpha) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}_{V(\widehat{X}_m^i)>b}.$$

instead of $\pi_B^N(\alpha)$ and $\Pi_B^N(\alpha)$, respectively. Thus, we expect that a good α parameter gives small values of $\pi_b^{\mathrm{d},N}(\alpha)$ and $\Pi_b^{\mathrm{d},N}(\alpha)$.

7.2.3.2 Continuous state spaces

We assume here that the random variable $V(X_m)$ have a density with respect to the Lebesgue measure. First of all, the rare event we want to estimate is defined by

$$A^{\mathrm{c}}(b) = \{V(X_m) \in [b, b + \delta_b)\},$$

for some large value of b, and small δ_b. After that, the p.d.f. p of $V(X_m)$ is assumed to be continuous, so that

$$p(b) = \lim_{\delta_b \to 0} \frac{1}{\delta_b} p_{\delta_b}(b), \quad \text{where} \quad p_{\delta_b}(b) = \mathbb{P}\left(V(X_m) \in [b, b + \delta_b)\right).$$

Consequently, for a given α, the p.d.f. of $V(X_m)$ evaluated in b is then estimated by algorithm 3.2 with

$$p(b) \approx \frac{1}{\delta_b} \hat{P}(A^{\mathrm{c}}(b), \alpha).$$

Mimicking the discrete case, the functions under consideration to use algorithm 7.1 are

$$\pi_b^{c,N}(\alpha) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{V(\widehat{X}_m^i)\leq b} \quad \text{and} \quad \Pi_b^{c,N}(\alpha) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{V(\widehat{X}_m^i)\geq b+\delta_b},$$

instead of $\pi_B^N(\alpha)$ and $\Pi_B^N(\alpha)$, respectively.

7.3 An alternative to the splitting algorithm via the weighted redistribution algorithm

The weighted redistribution algorithm is written to compute the probability that a critical event occurs at the final time m of the evolution of a Markov chain. It is nevertheless possible to compute the probability that a critical event occurs at some time $k \leq m$ of a Markov chain $\{X'_k,\ k = 1,\dots,m\}$. Namely to compute

$$\mathbb{P}(V'(X'_k) \geq B, \text{ for some } k = 1,\dots,m).$$

To have a critical event that occurs at terminal time m, we consider the variable

$$M_k = \max\left\{V'(X'_l),\ 0 \leq l \leq k\right\}$$

and we obtain

$$\mathbb{P}(M_m \geq B) = \mathbb{P}(V'(X'_k) \geq B, \text{ for some } k = 1,\dots,m). \tag{7.7}$$

To fit with the weighted redistribution algorithm framework, the underlying sequence must be a Markov chain. Since $\{M_k,\ k = 1,\dots,m\}$ is no longer a Markov chain, we have to consider

$$X_k = (X'_k, M_k),$$

which is a Markov chain. Indeed, we have

$$X_{k+1} = \left(X'_{k+1}, \max(M_k, V'(X'_{k+1}))\right).$$

Since the conditional law of X'_{k+1} given the past depends only on the present state X'_k, the above formulation shows that the conditional law of X_{k+1} given the past depends only on the present state $X_k = (X'_k, M_k)$. That is why $\{X_k,\ k = 1,\dots,m\}$ is a Markov chain. And rigorously, the score function to be use instead of V' in algorithm 3.2 is

$$V(X_k) = M_k.$$

To finish this subsection, it is worth noting that the weighted redistribution algorithm can now be used to compute some rare event that can occur during the evolution of a Markov process such as

$$\mathbb{P}(V(X(t)) \geq B, \text{ for some } t \in [0,T]),$$

where T is a *deterministic* time. To this end, we set a sequence t_k such as

$$0 = t_0 < t_1 < \cdots < t_m = T$$

and we consider the sequence $\{X'_k,\ k = 1,\dots,m\}$ with

$$X'_k = X(t_k).$$

To finish, we make use of the equation (7.7).

Consequently, we have derived an other method to compute hitting time probability (7.1) in the case of deterministic final time T instead of the splitting algorithm 3.1.

7.4 Numerical examples

We first study a toy case, the Gaussian random walk. We address next two more complex industrial problem dealing with optical fibre and financial engineering. After that, we compare the results of the adaptive splitting algorithm and the adaptive weighted redistribution algorithm.

According to the section 3.3, we will make use of the exponential selection functions

$$G_k(x_1,\dots,x_k) = \exp(\alpha(V(x_k) - V(x_{k-1}))), \tag{7.8}$$

for some $\alpha \geq 0$.

When $\alpha = 0$, the trajectory selection in algorithm 3.2 is uniform. As a result, the weighted redistribution algorithm is equivalent to the standard Monte Carlo method. Consequently, the rare set will not be reached in practice with a reasonable sample size. That is why we set

$$\alpha_* = 0$$

to initialize algorithm 7.1. On the other hand, in the three cases of study below, we observe that a large arbitrary value that ensures too strong selections is achieved if α is greater than 50. Namely, we have observed that

$$\Pi_b^N(\alpha) = \Pi_b^{\mathrm{d},N}(\alpha) = \Pi_b^{\mathrm{c},N}(\alpha) = 1, \quad \text{if } \alpha \geq 50.$$

Besides, notice that it could happen several retrials of the dichotomy performed in algorithm 7.1 give the same result. To avoid such a repetition, α^* is randomly uniformly chosen in the real interval $[50,100]$. As a result, the diversity on the output parameter $\hat{\alpha}$ of algorithm 7.1 is increased. Finally, we also set the mild condition $p_0 = 0.2$. As a conclusion, the dichotomy performed in algorithm 7.1 will be thus initialized for every run in every situation with

$$\alpha_* = 0, \quad \alpha^* \sim \text{Uniform}[50,100], \quad p_0 = 0.2. \tag{7.9}$$

The number of trajectories N in the adaptive weighted redistribution algorithm 7.1 is $N = 5000$. Each estimation of algorithm 7.1 well be performed over 200 retrials.

7.4.1 Gaussian random walk

Here, the Markov chain under consideration is $X_{k+1} = X_k + W_k$, where W_k are *i.i.d.* centered Gaussian random variable with variance 1, and $X_0 = 0$. For m a positive integer, we are interested in the distribution tail of the random variable X_m. By the way, notice that the real valued random variable X_m is a centered Gaussian random variable with variance m. In that case, the V function defined in equation 3.7 is reduced to identity and the selection functions are reduced to

$$G_k^\alpha(y_k) = \exp(\alpha(x_k - x_{k-1})), \quad \text{where } y_k = (x_1,\dots,x_k).$$

Exceedance probability estimation
We first compute the probability that X_m exceeds a certain threshold S

$$\mathbb{P}(X_m \geq B), \text{ for } B \text{ large}.$$

Let P_B^{Gauss} and $\hat{P}_B^{\text{Gauss}}$ denote the theoretical and estimated value of $\mathbb{P}(X_m \geq B)$ for the Gaussian random walk. We set m =15. Table 7.1 sums up the results of the probability estimation with

algorithm 7.1. The $\hat{\alpha}$ parameter is the average of the α values given by algorithm 7.1. The average number of run of algorithm 3.2 required by algorithm 7.1 is denoted by κ. The average total number of particle is therefore $\kappa \times N$. The survival function $B \mapsto \mathbb{P}(X_m \geq B)$ is estimated with the adaptive weighted redistribution algorithm 7.1, and results are plotted in figure 7.2a.

B	P_B^{Gauss}	$\hat{P}_B^{\text{Gauss}}$	$\hat{\alpha}$	κ
15	$5.37.10^{-5}$	$5.35.10^{-5} \pm 11\%$	1.07	7.08
17	$5.68.10^{-6}$	$5.69.10^{-6} \pm 14\%$	1.21	7.21
20	$1.21.10^{-7}$	$1.24.10^{-7} \pm 26\%$	1.43	7.14
22	$6.72.10^{-9}$	$6.73.10^{-9} \pm 30\%$	1.57	7.14
25	$5.41.10^{-11}$	$5.41.10^{-11} \pm 42\%$	1.78	7.18

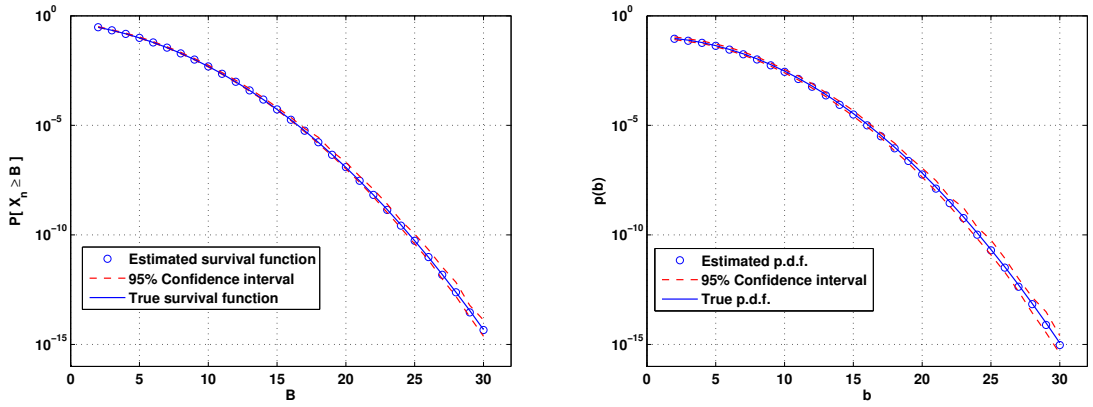
Table 7.1: Results of the estimation of the probability with adaptive weighted redistribution for the Gaussian sum.

Probability density function tail estimation

To compute the p.d.f. tail of the real valued random variable X_m with the method of section 7.2.3, we set $\delta_b = 0.2$. Estimated p.d.f. is plotted in figure 7.2b. In the case of p.d.f. tail estimate, it is shown in [Del Moral and Garnier, 2005] that the optimal α in term of minimization of the variance is $\alpha_{\text{opt}} = B/(m-1)$. Table 7.2 compares this optimal value with the mean $\hat{\alpha}$ of the values produced by the proposed approach. It shows excellent agreement.

B	15	17	20	22	25
α_{opt}	1.07	1.23	1.44	1.58	1.82
$\hat{\alpha}$	1.07	1.21	1.43	1.57	1.79

Table 7.2: Comparison with the theoretical optimal α parameter in the case of p.d.f. tail estimation.



(a) Survival function

(b) Probability density function

Figures 7.2: Survival function and p.d.f. estimation on the Gaussian random walk

Validation of the heuristic

We set $m = 15$, and we use a number of $N = 5000$ particles. Figure 7.3 presents an illustration of the heuristic worked out in section 7.2.2 when the threshold to reach is set at $B = 15$, which induces a theoretical probability equals to $\mathbb{P}(X_m \geq B) \approx 5.376.10^{-5}$. Green dotted line is the proportion of trajectories that have a terminal value greater than B as a function of α, namely $\Pi_B^N(\alpha)$. On the other hand, the blue plain line represents the relative standard deviation for the estimation of $\mathbb{P}(X_m \geq B)$ with algorithm 3.2 as a function of α. For this case, we observe that the variance is minimised when $\Pi_B^N(\alpha)$ is between 10% and 60%, which corresponds to a value of α between 0.8 and 1.3. It validates the results obtained in table 7.1 and the proposed strategy described in section 7.2.2.1. Plotting figure 7.3 is not necessary in practice, and should not be done.

The convex shape of the curve in figure 7.3 can be linked to the Jourdain-Lelong method [Jourdain and Lelong, 2009]. It gives a method to compute optimal importance sampling measures in the case of Gaussian random vectors. The method is based on a convex optimisation problem for the choice of some optimal parameters. See the examples provided [Jourdain and Lelong, 2009, section 2] for further details on the Markov chain case. However, for the case of rare event probability estimation, the Jourdain-Lelong method fails since the sample they use to reduce the variance cannot reach the rare event probability set.

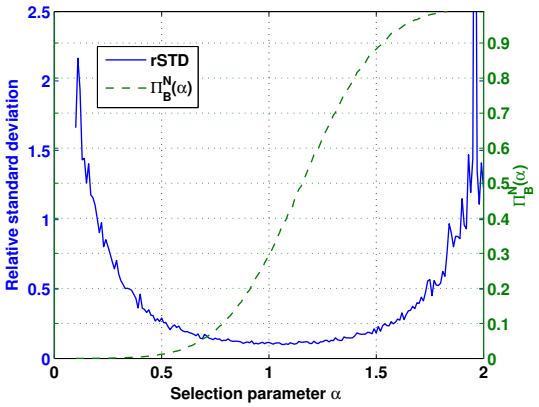


Figure 7.3: Proportion of particle above B at final time and relative standard deviation.

7.4.2 Outage probability in optical fiber

Statement of the problem

In this section we address the problem of the outage probability estimation due to polarization mode dispersion (PMD) in optical fibre. See [Garnier and Del Moral, 2006] for a review of PMD model. This is the first rare event estimation problem dealt with the weighted redistribution algorithm 3.2 in the scientific literature [Del Moral and Garnier, 2005, Garnier and Del Moral, 2006]. This problem has been treated with importance sampling, see for example [Biondini et al., 2002], and with some sequential Monte Carlo methods in [Johansen et al., 2006]. Both methods require the user to have sufficiently knowledge on the studied system. Furthermore, an accurate tuning is necessary to get reliable results. As said before, the main advantage of the weighted redistribution algorithm 3.2 is that we only have to know the law of the studied process, and it can be applied to more complicated situations.

The optical fiber light transmission quality can be measured by the difference in transit time for the light launched into the fast axis and the light launched in the slow axis. In a perfect optical fiber the light has the same speed on the fast and the low axis. But in practice, a delay may occur and can reduce the quality of the transmission. See figure 7.4.

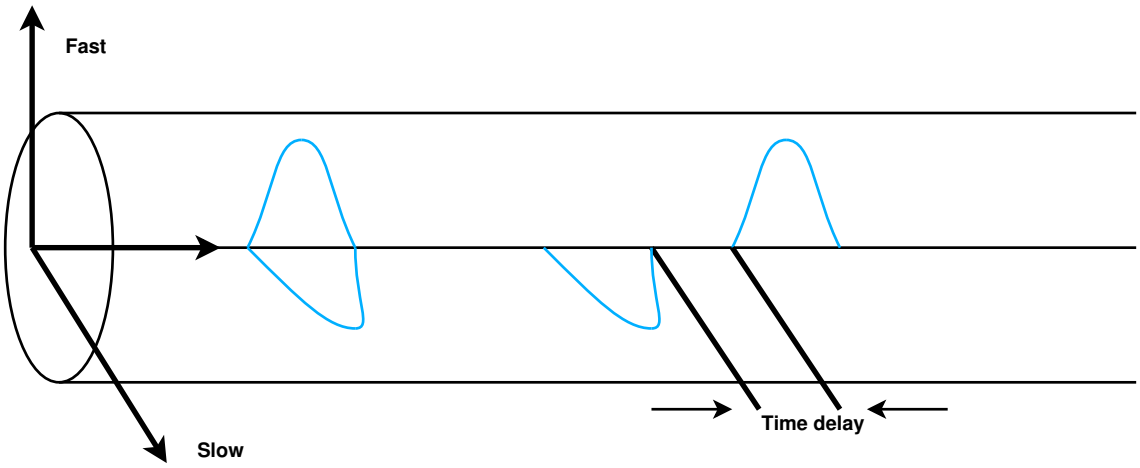


Figure 7.4: Delay in optical fiber.

The rare event under consideration corresponds to a pulse spreading beyond a threshold value. The pulse spreading in a randomly birefringent fibre is characterised by the square differential group delay (DGD)

$$\tau = |\hat{r}|^2,$$

where $\hat{r}$ is the PMD vector which is solution of

$$\hat{r}_z = \omega\Omega(z) \times \hat{r} + \Omega(z),$$

where Ω is a white noise process.
The model under consideration takes into account the concatenation of birefringent elements with piecewise constant vectors Ω. The segments junctions give a discrete model for the PMD vector $\hat{r}$. After the k^{th} section, the PMD vector is obtained with the equation

$$\hat{r}_k = R_n \hat{r}_{k-1} + \sigma\Omega_k,$$

where σ is the DGD per section, $\Omega_k = (\cos(\theta_k), \sin(\theta_k), 0)^T$, and $R_n = R(\theta_n, \phi_n)$ is the rotation matrix through an angle θ_k about the axis Ω_k:

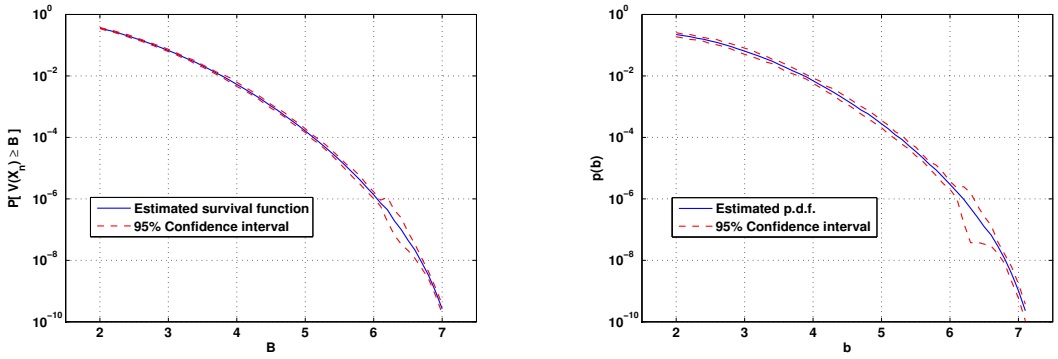
$$R_k = \begin{pmatrix} \cos(\theta_k)^2 + \sin(\theta_k)^2\cos(\phi_k) & \cos(\theta_k)\sin(\theta_k)(1-\cos(\phi_k)) & \sin(\theta_k)\sin(\phi_k) \\ \cos(\theta_k)\sin(\theta_k)(1-\cos(\phi_k)) & \sin(\theta_k)^2 + \cos(\theta_k)^2\cos(\phi_k) & -\cos(\theta_k)\sin(\phi_k) \\ -\sin(\theta_k)\sin(\phi_k) & \cos(\theta_k)\sin(\phi_k) & \cos(\phi_k) \end{pmatrix}$$

The angles ϕ_k and θ_k are considered as random variable. More precisely, ϕ_k and θ_k are *i.i.d.* uniformly distributed in $(0, 2\pi)$. If the fiber is modelled as the concatenation of m segments, the random variable variable that we want to estimate the distribution tail is $|\hat{r}_m|$. The sequence $(\hat{r}_k)_{k\in\mathbb{N}}$ is a Markov chain. Thus, we have $V(\hat{r}_m) = |\hat{r}_m|$ and the selection function used in algorithm 3.2 are

$$G_k^\alpha((|\hat{r}_1|, \ldots, |\hat{r}_k|) = \exp(\alpha(|\hat{r}_k| - |\hat{r}_{k-1}|).$$

B	$\hat{P}_B^{\mathrm{PMD}}$	$\hat{\alpha}$	κ
5	$1.58.10^{-4} \pm 9\%$	4.38	4.24
5.5	$1.82.10^{-5} \pm 9\%$	5.28	4.00
6	$1.33.10^{-6} \pm 15\%$	7.90	3.30
6.2	$4.13.10^{-7} \pm 23\%$	10.3	2.86
6.5	$6.62.10^{-9} \pm 30\%$	38	1.00
7	$2.17.10^{-10} \pm 65\%$	37	1.00

Table 7.3: Results of the estimation of the probability with adaptive weighted redistribution algorithm for the PMD.



(a) Survival function

(b) Probability density function

Figures 7.5: Survival function and p.d.f. estimation of the outage probability

Threshold exceedance probability estimation

The probability of the outage rare event that we want to estimate takes the form

$$P_B^{\mathrm{PMD}} = \mathbb{P}(|\hat{r}_m| > B).$$

The number of segments is set to $m = 20$. Table 7.3 sums up the results of the estimation of the probability with algorithm 7.1. Quantities $\hat{P}_m^{\mathrm{PMD}}$, $\hat{\alpha}$ and κ respectively denote the estimation of the probability P_m^{PMD}, the average of $\hat{\alpha}$ given by algorithm 7.1 and the average runs number of algorithm 3.2 required by algorithm 7.1. The survival function $B \mapsto \mathbb{P}(|\hat{r}_m| \geq B)$ is estimated with algorithm 7.1, and results are plotted in figure 7.5a.

Probability density function tail estimation

To compute the p.d.f. tail of the real valued random variable $|\hat{r}_m|$ as detailed in section 7.2.3, we set $\delta_b = 0.1$. Estimated p.d.f. is plotted in figure 7.5b.

Validation of the heuristic

We set $B = 6$. Figure 7.6 presents an estimation of the relative standard deviation for probability estimation of $\mathbb{P}(|\hat{r}_m| \geq 6)$ with the weighted redistribution algorithm as a function of α. Blue plain line represents the estimations of the relative standard deviation in the estimation of $\mathbb{P}(|\hat{r}_m| \geq 6)$ with the weighted redistribution algorithm as a function of α. The green dotted line is $\Pi_B^N(\alpha)$ as a function of α. For this case, we observe also that the variance is minimised when $\Pi_B^N(\alpha)$ is between 10% and 60%. In the case of outage probability, such a percentage corresponds to a value of α between 4 and 8. Due to costly computation time, figure 7.6 is not

plotted in practice but it validates the proposed strategy.

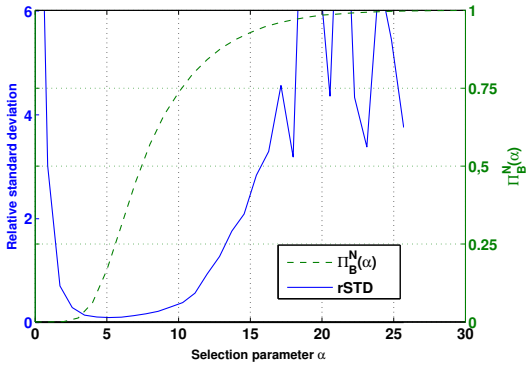


Figure 7.6: Proportion of particle above B at final time and relative standard deviation.

Comparison with the existing approach

In the original article [Del Moral and Garnier, 2005], the authors proposed to select, for each given value of α, which run of the weighted redistribution algorithm gives the minimal value of the relative standard deviation (rSTD). See section 3.1 for the definition of the rSTD. Even though one run of the weighted redistribution algorithm is sufficient to compute an estimate of the rSTD, it is not possible to anticipate in which real interval could be an appropriate α parameter. Consequently, this manner to proceed can be quite costly. Besides, the authors did not give any additional precision on the number of tests that have been carried out.

We compare our results with the ones of [Del Moral and Garnier, 2005] on the p.d.f. estimation tail. Authors of [Del Moral and Garnier, 2005] proposed to use $\alpha = 2$ and $\alpha = 6$. In table 7.4, we present the relative standard deviation obtained with our method and with the weighted redistribution algorithm 3.2 (non adaptive) with $\alpha = 2$ and $\alpha = 6$. The ratio computed in the table 7.4 is defined as follows. First, denote as S_2 and S_6 the relative standard deviation with the weighted redistribution algorithm with $\alpha = 2$ and $\alpha = 6$. Denote also S_{dicho} the relative standard deviation in the estimation with our method. Then we set

$$\text{Ratio} = \max\left(\frac{S_{\text{dicho}} \times \kappa}{S_2}, \frac{S_{\text{dicho}} \times \kappa}{S_6}\right),$$

where κ is the average number of run of the weighted redistribution algorithm required by algorithm 7.1. The ratio can be interpreted as the maximum number of tests that it is allowed to proceed for the choice of an appropriate α with the weighted redistribution algorithm 3.2. If we have planed to make more tests, we should have preferably make use of the adaptive weighted redistribution algorithm. Results and promising. In a fully adaptive way, we reduced up to ten time times the number of runs of the weighted redistribution algorithm . Anyway, selecting an appropriate α with brute force may lead to much more runs.

7.4.3 Loss distributions of credit portfolio

Statement of the problem

The weighted redistribution algorithm 3.2 recently became popular for the computation of rare

B	4	4.5	5	5.5	6	6.5	7
rSTD for IPS with $\alpha = 2$	13%	12%	22%	41%	70%	230%	⌣
rSTD for IPS with $\alpha = 6$	81%	36%	25 %	20%	14%	15%	47%
rSTD for IPS Adapt.	6%	12%	15%	19%	21%	45%	121%
κ	4.8	4.6	4.5	3.9	3.6	3.3	1.4
Ratio	3.0	4.9	3.1	3.7	5.7	9.9	3.7

Table 7.4: Comparison of the relative standard deviation in the estimation of the p.d.f. tail.

credit portfolio loss distributions. The literature on this topic is mainly focused in these three papers [Carmona and Crepey, 2010, Carmona et al., 2009, Giesecke et al., 2010]. The method worked out in section 7.2.2.1 can be applied to all models considered in these papers. For the sake of conciseness, the model under consideration in the present paper is the one described in [Carmona et al., 2009].

We considers a portfolio of $H = 125$ firms. Dynamics of the asset values of these H firms are given by the following system of stochastic differential equations

$$\mathrm{d}S_n(t) = rS_n(t)\mathrm{d}t + \sigma_n\sigma(t)S_n(t)\mathrm{d}W_n(t),\ n = 1,\dots,H, \tag{7.10}$$

where the common stochastic volatility factor $\sigma(t)$ is solution of the one dimensional stochastic equation

$$\mathrm{d}\sigma(t) = \nu(\overline{\sigma} - \sigma(t))\mathrm{d}t + \gamma\sqrt{\sigma(t)}\mathrm{d}W(t). \tag{7.11}$$

In these equations, $W_n(t)$ and $W(t)$ are one dimensional Weiner processes, starting from 0 at $t = 0$, and correlated as follows

$$\mathrm{d}\langle W_n, W_{n'}\rangle_t = \rho_{n,n'}\mathrm{d}t \quad \text{and} \quad \mathrm{d}\langle W_n, W\rangle_t = \rho_\sigma \mathrm{d}t,$$

where ρ_σ does not depend on n.

For every deterministic boundary $t \mapsto \Theta_n(t)$, $n = 1,\dots,H$, the time of default is given by

$$\tau_n = \inf\{t,\ S_n(t) \le \Theta_n(t)\},$$

and the portfolio loss function at time t is given by

$$L(t) = \sum_{n=1}^{H} \mathbb{1}_{\tau_n \le t}, \quad t \ge 0.$$

We are interested in the following in the behaviour of the distribution tail of the $\{0,\dots,H\}$-valued random variable $L(T)$, for a given time T.

The parameters involved are those of the original paper [Carmona et al., 2009], and are reminded in table 7.5. For the sake of the simplicity, the Θ_n functions are constant. The continuous time model is discretized using a time step $\Delta_t = (1/20)$ years for equation (7.10), and $\delta_t = 10^{-3}$ years for equation (7.11), as chosen in the original paper [Carmona et al., 2009]. The Markov chain used in the weighted redistribution algorithm is thus the following one

$$\left\{X_k = \left(\sigma(k\Delta_t), (S_n(k\Delta_t))_{1\le n\le H}, \left(\min_{0\le l\le k} S_n(l\Delta_t)\right)_{1\le n\le H}\right),\ k \ge 0\right\}.$$

r	σ_n	$\rho_{n,n'}$	ν	$\overline{\sigma}$	γ	ρ_σ	Θ_n
0.06	0.4	0.1	3.5	0.4	0.7	-0.06	36

Table 7.5: Parameters for equations (7.10) and (7.11).

B	$\hat{P}_B^{\text{Loss}}$	$\hat{\alpha}$	κ
40	$1.04.10^{-5} \pm 55\%$	1.08	7.84
45	$1.84.10^{-6} \pm 59\%$	1.13	7.90
50	$3.42.10^{-7} \pm 98\%$	1.20	8.14
55	$4.75.10^{-8} \pm 79\%$	1.26	7.82
60	$4.23.10^{-9} \pm 124\%$	1.40	8.40

Table 7.6: Results of the probability estimation of P_B^{Loss} with the adaptive weighted redistribution algorithm for credit portfolio losses.

Consequently, the dimension of the state space is $\dim(\mathbb{R}^{2H+1}) = 251$. Following [Carmona et al., 2009], the V function is defined as follows

$$V(X_k) = \sum_{n=1}^{H} \log\left(\min_{0\le l\le k} S_n(l\Delta_t)\right),$$

and the potential functions G_k required by algorithm 3.2 are given in equation (7.8). The adaptive weighted redistribution algorithm 7.1 is initialized with equations (7.9) at each run.

Threshold exceedance probability estimation

Here, we are interested in estimating the survival function of the random variable $L(T)$

$$B \mapsto \mathbb{P}(L(T) \ge B) \quad \text{and} \quad n \mapsto \mathbb{P}(L(T) = B), \quad \text{for } B = 1, \dots, H.$$

The maturity is set at $T = 1$ year, so that $m = 20$. Table 7.6 sums up the results of the estimation of the probability with algorithm 7.1. Quantities $\hat{P}_B^{\text{Loss}}$, $\hat{\alpha}$, and κ denote respectively the estimation of the probability

$$P_B^{\text{Loss}} = \mathbb{P}(L(T) \ge B),$$

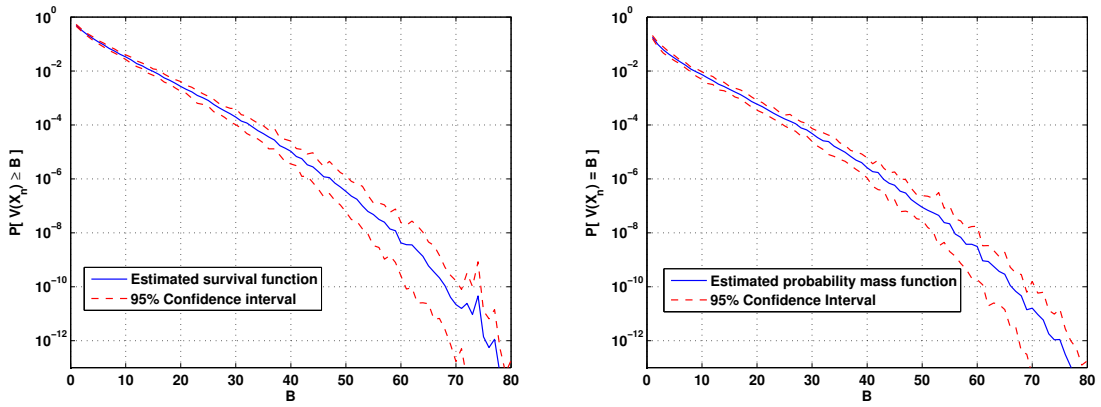
the average of $\hat{\alpha}$ given by algorithm 7.1 and the average number of runs of the algorithm 3.2. The survival function $B \mapsto \mathbb{P}(L(T) \ge B)$ is estimated with algorithm 7.1, and results are plotted in figure 7.7a.

Probability mass function tail estimation

Here we compute the probability mass function for $L(T)$, namely the quantities

$$B \mapsto \mathbb{P}(L(T) = B), \quad n = 0, \dots, H.$$

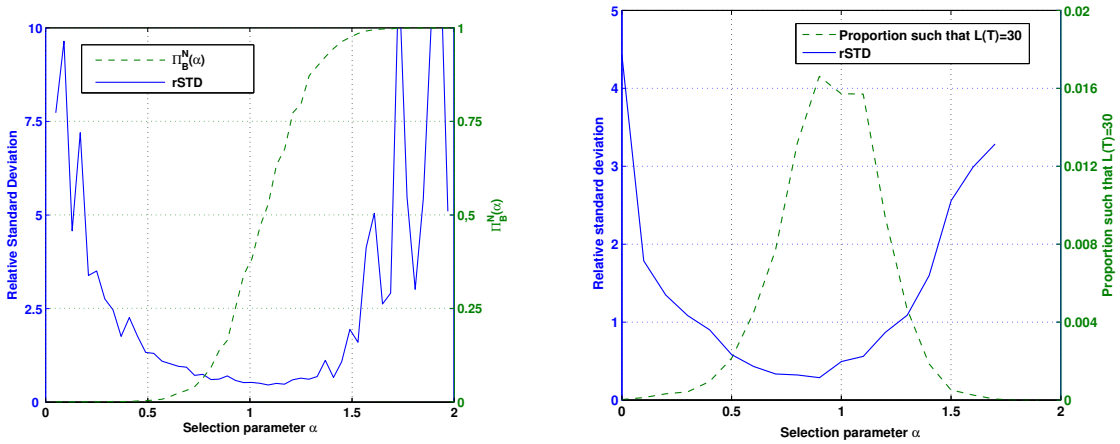
To achieve this goal, we apply the procedure described in section 7.2.3 for the discrete case. Figure 7.7b presents the estimated probability mass function computed with the procedure detailed in section 7.2.3.



(a) Survival function

(b) Probability mass function

Figures 7.7: Survival function and probability mass function estimation of the portfolio losses.



(a) Proportion of particles above B at final time and relative standard deviation.

(b) Proportion of particles such that $L(T) = 30$ at final time and relative standard deviation.

Figures 7.8: Validation of the heuristic for credit portfolio losses probability estimation.

Validation of the heuristic

We set $B = 30$. Figure 7.8a presents an estimation of the relative standard deviation for the estimation of the probability $\mathbb{P}(L(T) \geq 30)$ with the weighted redistribution algorithm as a function of α. Blue plain line represents estimates of relative standard deviation in the estimation of $\mathbb{P}(L(T) \geq 30)$ with the weighted redistribution algorithm, as a function of α. Green dotted line is $\Pi_B^N(\alpha)$ as a function of α. For this case, we observe that the variance is minimised when $\Pi_B^N(\alpha)$ is between 35% and 75%, which corresponds to a value of α between 1 and 1.2. Due to the high dimension of the problem, rSTD estimated values in table 7.6 remains high, but is the best that the weighted redistribution algorithm can do in the present case, as it can be observed in figure 7.8a.

We set again $B = 30$. It is shown in figure 7.8b that the best $\hat{\alpha}$ is the one for which the number of trajectories such that $V(X_m) = B$ is maximum. Green dotted line stands for proportion of trajectories hitting n at terminal time, and blue plain line stands for the estimated relative standard deviation in the estimation of $\mathbb{P}(L(T) = B)$. It thus validates observations detailed in section 7.2.3 for the discrete case. Figures 7.8a and 7.8b are of course not plotted in practice if

N	30000	50000	80000
$\hat{P}_{55}^{\text{Loss}}$	$3.84.10^{-8} \pm 143\%$	$3.90.10^{-8} \pm 124\%$	$2.90.10^{-8} \pm 110\%$

Table 7.7: Estimation of $\hat{P}_{55}^{\text{Loss}}$ with the standard weighted redistribution algorithm for different values of N and random α parameter.

we want to use algorithm 7.1.

Comparison with the existing method

Authors of articles related to this problem [Carmona and Crepey, 2010, Carmona et al., 2009] did not make any comment on the number of tests that was carried out to find an appropriate α. It seems to have been selected by brute force. Then, their results are illustrated with some costly three dimensional plots. Consequently, the gain in the total computational cost that can be obtained between the method of the present paper and the choices made in [Carmona and Crepey, 2010, Carmona et al., 2009] cannot be quantified. Nevertheless, we present what would imply a bad choice for α in the case of the estimation of the survival function. We focus on the estimation $\hat{P}_B^{\text{Loss}}$ of $P_B^{\text{Loss}} = \mathbb{P}(L(T) \geq B)$, with $B = 55$. The adaptive weighted redistribution algorithm detailed in 7.1 gives a relative standard deviation equals to 79%, for an average budget of $\kappa \times N = 7.9 \times 5000 = 39500$ trajectories as shown in table 7.6. We obtain also an α equals to $\alpha = 1.26$. In the table 7.7, α is uniformly randomly chosen into the real interval $[0.5, 2]$

$$\alpha \sim \text{Uniform } [0.5, 2].$$

We obtain thus numerical conclusions on the efficiency of our method. In particular the choice of a good α parameter is very sensitive. Even with twice as the total budget of trajectories used in our method, the weighted redistribution algorithm with an α parameter not so far than the one given with our method leads to a higher rSTD.

7.4.4 Adaptive splitting and adaptive weighted redistribution algorithm comparison

Let $\{X_t, 0 \leq t \leq 1\}$ be the Brownian bridge, solution of the following stochastic differential equation

$$\begin{cases} X_0 = 0 \\ \mathrm{d}X_t = \dfrac{X_t}{t-1}\mathrm{d}t + \mathrm{d}W_t. \end{cases} \tag{7.12}$$

We can show that $X_1 = 0$. We are interested here in estimating the probability

$$\mathbb{P}(X_t \geq B, \text{ for some } t \leq 1) = \exp^{-2B^2}, \tag{7.13}$$

for some given threshold B.

We use a the Euler scheme [Kloeden and Platen, 1992, chap 9] for the implementation of the trajectories of $\{X_t, 0 \leq t \leq 1\}$ with a time step $\delta = 0.001$. The Markov chain resulting from this discretization is denoted by

$$\{X^{\delta}(t_j),\ j = 1, \ldots, 1/\delta\}.$$

Adaptive splitting algorithm The nested regions required by the splitting algorithm are time-independent here and defined by $A_k = \{(x,t) \in \mathbb{R} \times \mathbb{R}^+,\ x > S_k\}$. The sequence S_k is empirically given at each run of the adaptive splitting algorithm 4.1.
For the threshold selection, we use a set of 20 trajectories for each threshold. Approximations of the entrance distribution into the created thresholds and approximations of the conditional probability value are performed using a set of $N = 1000$ trajectories at each threshold.

Adaptive weighted redistribution algorithm The Markov chain $\{X_k,\ k = 1,\dots,m\}$ under consideration in the weighted redistribution algorithm is defined with

$$X_k = \left(X^\delta(t_{50\times k}), \max\left\{X^\delta(t_{50\times l}),\ l \le k\right\}\right).$$

Roughly speaking, the selection step is performed every 50 iterations of the Markov chain $\{X^\delta(t_j),\ j = 1,\dots,1/\delta\}$.
We use the standard exponential selection function defined by

$$G_k(x_0,\dots,x_k) = \exp[\alpha(V(x_k) - V(x_{k-1}))]$$

and the V function equals to

$$V(x_k) = x_k^{(2)},$$

where $x_k = (x_k^{(1)}, x_k^{(2)})$ and $x_k^{(2)}$ is the second coordinate of x_k. Hence, the probability of interest (7.13) can be rewritten as follows

$$\mathbb{P}(X_t \ge B,\ \text{for some } t \le 1) = \mathbb{P}(V(X_m) \ge B).$$

We use a set of $N = 1000$ trajectories in the adaptive weighted redistribution algorithm 7.1.

For both methods, the bias is negligible. We compare thus the adaptive splitting algorithm 4.1 and the adaptive weighted redistribution algorithm 7.1 in term of time relative variance product. It is defined in section 3.1 by the product of the relative standard deviation by the average run time of the algorithm.
For thresholds B from 2 to 4.5, we present in figure 7.9 the time relative variance product for the adaptive splitting algorithm 4.1 and the adaptive weighted redistribution algorithm 7.1. We numerically conclude with this example that the adaptive weighted redistribution algorithm gives better time relative variance product than adaptive splitting for every threshold B. Notice that the adaptive weighted redistribution algorithm consists of several runs of the weighted redistribution algorithm.

To understand the gain in using the adaptive weighted redistribution algorithm instead of the splitting algorithm, we present additional results in table 7.8. We compare for different thresholds B the rSTD of the adaptive weighted redistribution algorithm (rSTD WR), the rSTD of the adaptive splitting algorithm (rSTD Split) and the average run time of the algorithms for the estimation of the probability (7.13). Even though the adaptive splitting algorithm gives better rSTD, its run time increases with B. The run time of the adaptive weighted redistribution algorithm is pretty much constant for every threshold B, and is from 3 to 10 times lower than the one of the adaptive splitting algorithm.

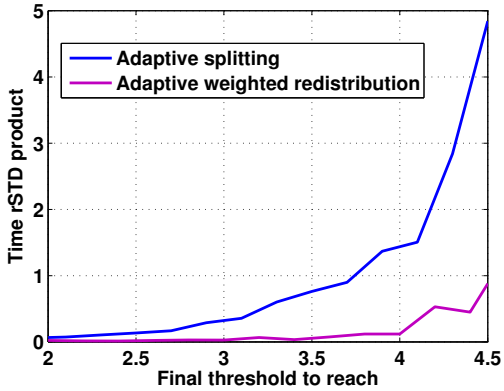


Figure 7.9: Relative time variance product for the adaptive splitting algorithm 4.1 and the adaptive weighted redistribution algorithm 7.1.

B	Theo. proba.	rSTD WR	rSTD Split	WR time (sec.)	Split time (sec.)
2	3.35e-4	0.39	0.16	0.14	0.36
2.2	6.25e-5	0.45	0.22	0.14	0.43
2.4	9.93e-6	0.71	0.23	0.13	0.50
2.6	1.34e-6	1.56	0.28	0.17	0.57
2.8	1.54e-7	1.02	0.33	0.14	0.65
3	1.53e-8	1.14	0.44	0.14	0.76
3.2	1.26e-9	1.79	0.54	0.16	0.82
3.4	9.10e-11	1.63	0.64	0.18	0.93
3.6	5.53e-12	2.10	0.90	0.17	1.03
3.8	2.83e-13	2.97	0.95	0.17	1.13
4	1.26e-14	4.77	1.74	0.18	1.25

Table 7.8: rSTD and run time comparison with adaptive weighted redistribution algorithm and adaptive splitting algorithm.

Conclusion of the chapter

Contributions

- We proposed a way to select valuable selection functions for the implementation of the weighted redistribution algorithm. The proposed method is fully adaptive.
- We detailed estimation procedures for probability of threshold exceedance and p.d.f. tail. Some numerical tests are performed on a toy case and on the two main problems addressed with weighted redistribution algorithm in the scientific literature. We have shown that our method outperformed the existing ones. In addition, a study on a toy case demonstrated that the chosen parameters with our method are very close to the optimal ones.
- It is worth noting that the total number of iterations in our procedure does not increase as the probability to estimate becomes smaller. Few calls to the weighted redistribution algorithm are needed to achieve procedure, it would have been nice to reduce this number as much as possible.
- We proposed to use the weighted redistribution algorithm with the cumulative supremum of a real functional of a process $\{X(t),\ t \geq 0\}$ to numerically compute hitting time probabilities. With this new method, the time relative variance product for the estimation of rare event probability is strongly reduced.

Limitations and links with the next chapter

- We did not address any theoretical justification for the proposed adaptive weighted redistribution algorithm.
- The proposed hitting time probability estimation method is valid only if the final horizon time is deterministic. Besides, we did not address the tuning of the selection step frequency. It can be adapted to reduce the variance of probability estimates as the parameter α.
- With the adaptive splitting algorithm derived in the chapter 4, we have now two new adaptive methods to compute the probability of conflict in air traffic management. In the next chapter, we will use these two algorithms to compute some conflict probability.

CHAPTER 8

Applications of the proposed algorithms to conflict probability estimation in air traffic management

In this chapter

- We give a very simple model for aircraft trajectories with stochastic processes.
- We address the problem of conflict probability estimation and extreme quantile estimation in the case where the conflict is thought as separation distance between aircraft.
- We derive real valued functions that characterise the conflict zones defined in section 3.4. With such a characterization, adaptive splitting algorithms derived in section 4 and the adaptive weighted redistribution 7.1 can be implemented.
- We give numerical experiments for rare event probabilities with algorithms 4.1,4.2, and 7.1 and extreme quantiles with the algorithm 4.3 .

8.1 Introduction

A conflict can be thought as a violation of a minimum separation distance between aircraft. It can also be characterized using the relative speed and relative rate of climbing between aircraft as detailed in section 3.4. The goal of this chapter is to validate the proposed algorithms of this thesis and to compare them with the existing ones for the conflict probability estimation. See section 1.4 for a state of the art of the conflict probability estimation. We will focus on adaptive splitting algorithms for rare event estimation 4.1, for extreme quantile estimation 4.2 and for the whole cumulative distribution function estimation 4.3 and the adaptive weighted redistribution algorithm 7.1.

All the above mentioned algorithms require the event of interest to be written in term of threshold exceedance of a real valued function. Such a characterization is obvious for the min-

imum separation distance. This will be detailed in section 8.3. For the more advanced conflict zones defined in section 3.4, we will derive score functions in section 8.4. First of all, we will detail the aircraft trajectory model that will be used in the following.

Results of this chapter are partially published in [Jacquemart and Morio, 2013].

8.2 Aircraft trajectory model

Dynamic of an aircraft The considered model for aircraft trajectories is very simple. In particular it enables many runs for each algorithm for a reasonable computation time.

An aircraft trajectory will be well modelled by a stochastic process in continuous time [Watkins and Prandini, 2005]. A first way is to include randomness to the flight mechanics equations describing the aircraft motion [Blom et al., 2003]. An other method [Paielli and Erzberger, 1999, Paielli and Erzberger, 1997, Hu et al., 1999] consists in superimposing a stochastic component to the nominal aircraft motion. This latter idea will be used.

More precisely, we assume that the three-dimensional position of an aircraft at time t is given by the following drifted Brownian motion

$$\mathrm{d}x_t = v\mathrm{d}t + \sigma_t \mathrm{d}W_t, \tag{8.1}$$

where v is a three-dimensional speed vector, σ_t is a squared correlation matrix and W_t is a standard three-dimensional Brownian motion. The initial position is given by some x_0.

The cross-correlation between the along-track and cross-track error is small enough to be considered as null. The vertical position error can be negligible when compared to along-track and cross-track position error [Paielli and Erzberger, 1997]. If we assume that the aircraft has a constant speed v in a coordinate system where the abscissa axis is parallel to aircraft trajectory, we have $x_t = (x_t^{(a)}, x_t^{(c)}, x_t^{(\mathrm{v})})$ where

$$\begin{cases} \mathrm{d}x_t^{(a)} = v\mathrm{d}t + g^a(t)\mathrm{d}W_t^{(a)} \\ \mathrm{d}x_t^{(c)} = g^c(t)\mathrm{d}W_t^{(c)} \\ \mathrm{d}x_t^{(\mathrm{v})} = 0 \end{cases} \tag{8.2}$$

where $W_t^{(a)}$ et $W_t^{(c)}$ are two independent standard one-dimensional Brownian motions, with $W_0^{(a)} = W_0^{(c)} = 0$ and $x_t^{(a)}$, $x_t^{(c)}$, $x_t^{(\mathrm{v})}$ denote respectively the along-track, cross-track and vertical position.

As commonly supposed [Prandini et al., 2011, Blom et al., 2007b], the relative mean square error in the distance between the expected and real position of an aircraft is linear with the time t. The slope equals to $r_a = 0.25$ nmi/min (nautical miles per minute) in the along track direction and $r_c = 0.2$ nmi/min in the cross-track direction.

The objective is now to choose properly some functions g^a and g^c so that

$$x_t^{(a)} = x_0^{(a)} + vt + \int_0^t g^a(s)\mathrm{d}W_s^{(a)} \sim \mathcal{N}\left(vt, (r_a t)^2\right) \tag{8.3}$$

$$x_t^{(c)} = x_0^{(c)} + \int_0^t g^c(s)dW_s^{(c)} \sim \mathcal{N}\left(0, (r_c t)^2\right). \tag{8.4}$$

Thanks to Itô's isometry, it is straightforward to check that the two following functions satisfy the problem

$$g^a(t) = r_a\sqrt{2t}, \quad g^c(t) = r_c\sqrt{2t}. \tag{8.5}$$

Finally, if the speed vector v has angles θ_1 and θ_2 with the horizontal and the vertical axis, the aircraft position is obtained applying the corresponding rotation matrix.

Multiple aircraft model If we consider n_a aircraft, an air traffic scenario will be modelled by

$$X_t = (X_t^1, \ldots, X_t^{n_a}) \in \mathbb{R}^{N_A}, \tag{8.6}$$

where $N_A = 3 \times n_a$. The correlation error between aircraft are supposed to be null.

8.3 The conflict is thought as a separation distance between aircraft below a threshold

8.3.1 The quantities to be estimated

The rare event probability We assume the air traffic scenario to involve n_a aircraft. The positions of these aircraft at time t are given by equation (8.6) with the vector $X_t = (X_t^1, \ldots, X_t^{n_a})$. If the positions of each aircraft are assumed to belong to $\mathbb{R}^3$ then $X_t \in \mathbb{R}^{N_A}$ where $N_A = 3n_A$.

We are first interested in estimating the probability of the event that aircraft being too close from each other during a fixed flight duration. Namely, for a given separation distance $\delta_{\text{sep}} \geq 0$ and a given deterministic time T, we are first interested in simulating the event

$$B^{i,j} = \left\{ \|X_t^i - X_t^j\| \leq \delta_{\text{sep}}, \text{ for some } t \leq T \right\}, \tag{8.7}$$

where X_t^i and X_t^j denote the position of aircraft i and j.

We could also be interested in the event that the aircraft i being too close from any other aircraft

$$B^i = \bigcup_{j \neq i} B^{i,j}. \tag{8.8}$$

Finally, we consider the event that at least two aircraft being too close each other during a fixed time duration flight

$$B = \bigcup_{i=1}^{n_A} B^i. \tag{8.9}$$

The minimum separation distance that ensures a given probability On the other hand, we address the problem of finding the minimum separation distance between aircraft that the plan should fulfil to ensure that the probability of being closer that this distance during the whole duration flight is below a given probability value.

More precisely, let p_0 a (small) real number within the interval $(0,1)$. If we are only interested in a two aircraft scenario, we want first to estimate the distance $\delta_{\text{sep}}^{i,j}(p_0)$ such that

$$\mathbb{P}\left(\|X_t^i - X_t^j\| \leq \delta_{\text{sep}}^{i,j}(p_0), \text{ for some } t \leq T \right) = p_0. \tag{8.10}$$

Quantities linked to equation (8.8) and (8.9) are also of interest. Hence, we define he distances $\delta^i_{\text{sep}}(p_0)$ and $\delta_{\text{sep}}(p_0)$ with

$$\mathbb{P}\left(\|X^i_t - X^j_t\| \leq \delta^i_{\text{sep}}(p_0), \text{ for some } t \leq T \text{ and some } 1 \leq j \leq n_A \text{ with } j \neq i\right) = p_0$$

and

$$\mathbb{P}\left(\|X^i_t - X^j_t\| \leq \delta_{\text{sep}}(p_0), \text{ for some } t \leq T \text{ , some } 1 \leq i \leq n_A \text{ and some } 1 \leq j \neq i \leq n_A\right) = p_0. \tag{8.11}$$

8.3.2 Score functions for the algorithms

Several algorithms studied in this manuscript require that the event of interest are written in terms of threshold exceedance of a score function. It is in particular the case for the adaptive splitting algorithms 4.1,4.2,4.3 and the adaptive weighted redistribution algorithm 7.1.

It is straightforward that the event (8.7) can be rewritten with the $\phi^{i,j}$ function defined by

$$\phi^{i,j} : \begin{cases} (\mathbb{R}^3)^{n_A} \longrightarrow \mathbb{R}^+ \\ x = (x^1, \dots, x^{n_A}) \longmapsto \|x^i - x^j\|, \end{cases} \tag{8.12}$$

where $\|\|$ denotes the Euclidean norm in $\mathbb{R}^3$.
We have for any $\delta_{\text{sep}} \geq 0$ that

$$B^{i,j} = \left\{ \inf_{0 \leq t \leq T} \phi^{i,j}(X_t) \leq \delta_{\text{sep}} \right\}, \tag{8.13}$$

and $\delta^{i,j}(p_0)$ is nothing but the p_0-quantile of the real valued random variable $\inf_{0\leq t\leq T} \phi^{i,j}(X_t)$

$$p_0 = \mathbb{P}\left(\inf_{0 \leq t \leq T} \phi^{i,j}(X_t) \leq \delta_{i,j}(p_0) \right). \tag{8.14}$$

In the same way, with the events (8.8) and (8.9) we use

$$\phi^i : \begin{cases} (\mathbb{R}^3)^{n_A} \longrightarrow \mathbb{R}^+ \\ \qquad x \longmapsto \min\left\{ \phi^{i,j}(x),\ 1 \leq j \neq i \leq n_A \right\}, \end{cases} \tag{8.15}$$

and

$$\phi : \begin{cases} (\mathbb{R}^3)^{n_A} \longrightarrow \mathbb{R}^+ \\ \qquad x \longmapsto \min\left\{ \phi^{i}(x),\ 1 \leq i \leq n_A \right\}. \end{cases} \tag{8.16}$$

We can thus rewrite the events B^i and B as follows

$$B^i = \left\{ \inf_{0 \leq t \leq T} \phi^i(X_t) \leq \delta_{\text{sep}} \right\}, \tag{8.17}$$

and

$$B = \left\{ \inf_{0 \leq t \leq T} \phi(X_t) \leq \delta_{\text{sep}} \right\}, \tag{8.18}$$

hence, for all $p_0 \in (0,1)$, $\delta^i(p_0)$ and $\delta(p_0)$ are the p_0-quantiles of $\inf_{0\leq t\leq T} \phi^i(X_t)$ and $\inf_{0\leq t\leq T} \phi(X_t)$ respectively.

8.3.3 Implementation of the algorithm and numerical results

The considered flight plan consists in a two aircraft scenario. The duration flight is 20 minutes. The speed of the aircraft is set to 20 nmi/min. It is presented in figure 8.1. The panel of the left present a top view of the two aircraft trajectories. The panel of the right present the separation distance between aircraft as a function of the time. For both panels, the green curves are the expected position and the expected separation distance. The blue curves are a realisation of the random trajectory of the aircraft, given with equation (8.1).

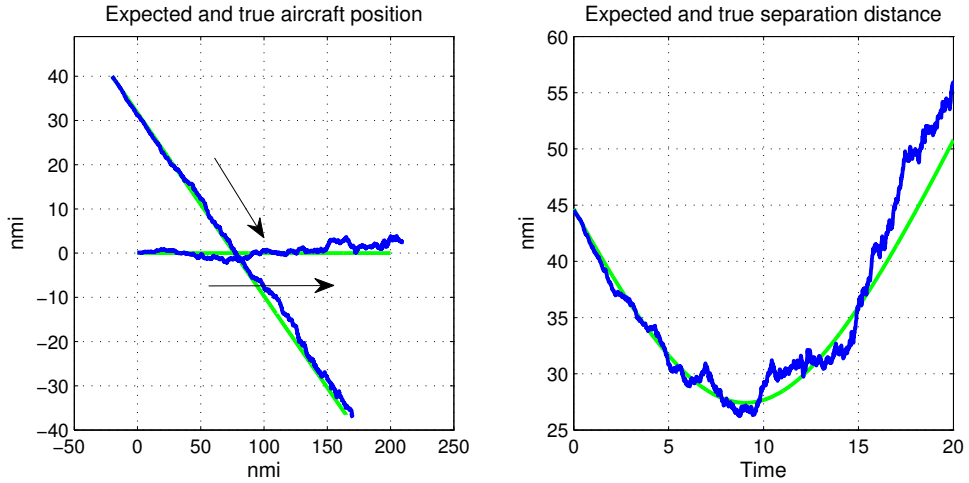


Figure 8.1: Aircraft positions and expected separation distance.

For each algorithm, the stochastic process (8.1) is implemented using the Euler scheme with a time step of $\delta = 0.05$. Every estimation is performed over 50 retrials. With the notation of the chapter, the position of the aircraft at time t are given with the vector

$$X_t = (X_t^1, X_t^2).$$

Then the quantities of interest are the rare event probability

$$\mathbb{P}\left(\inf_{0\le t\le T} \phi^{1,2}(X_t) \le \delta_{\text{sep}}\right) \tag{8.19}$$

and, for any given $p_0 \in (0,1)$, the p_0-quantile $\delta_{\text{sep}}^{1,2}(p_0)$ of the random variable $\inf_{0\le t\le T} \phi^{1,2}(X_t)$:

$$\mathbb{P}\left(\inf_{0\le t\le T} \phi^{1,2}(X_t) \le \delta_{\text{sep}}^{1,2}(p_0)\right) = p_0. \tag{8.20}$$

The adaptive splitting algorithm for rare event estimation For the adaptive splitting algorithms 4.1, 4.2 and 4.3, we use a (small) number of 10 trajectories to estimate a new threshold and a number of $N = 500$ trajectories to estimates entrance distributions and conditional probabilities. The expected probability of reaching a threshold starting from the previous one is equals to 0.5.

For every minimum separation distance δ_{sep} between 20 and 0.2 we compute the probability (8.19) that the two aircraft being at a distance δ_{sep} from each other before final time $T = 20$ minutes. The estimation is done with the adaptive splitting algorithm 4.1 and the splitting

(non adaptive) algorithm 3.1. For the algorithm 3.1, the thresholds are empirically chosen with algorithm 4.1 using a number $N = 10000$ for threshold estimation and $N = 10000$ for entrance distribution estimation. Then we run algorithm 3.1 with a number $N = 500$ trajectories. Results are presented in table 8.2.

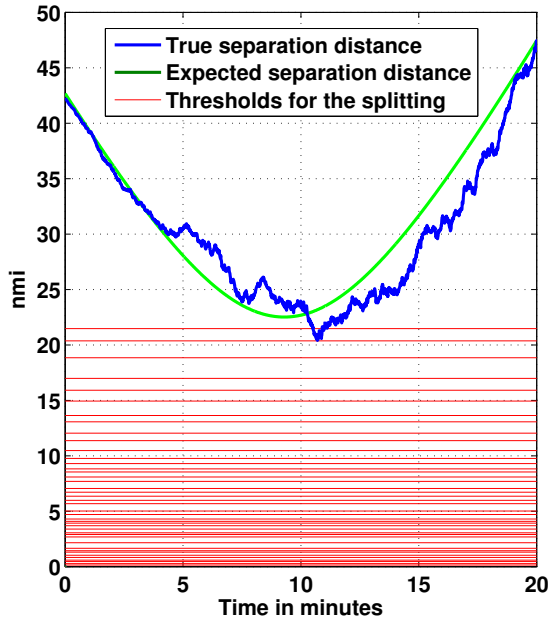


Figure 8.2: Threshold created by adaptive the splitting.

We present in figure 8.2 the threshold created by the adaptive splitting algorithm 3.1. They correspond to 0.5^k-quantiles of the random variable $\inf_{0\leq t\leq 20}\phi^{1,2}(X_t)$ for $k = 1,2,\ldots$. As reminded in section 1.2.2, with a deterministic time T, adaptive methods of splitting algorithm of [Garvels, 2000] and [Cérou and Guyader, 2007] fails. Moreover, spaces between consecutive thresholds is decreasing. As a result, the adaptive method of [Wadman et al., 2013] also fails (see section again 1.2.2).

The adaptive splitting algorithm for extreme quantile estimation For every probability value $p_0 \in (0,1)$, we compute the distance such that

$$\mathbb{P}\left(\inf_{0\leq t\leq T}\phi^{1,2}(X_t) \leq \delta^{1,2}_{\text{sep}}(p_0)\right) = p_0$$

with the adaptive splitting algorithm for extreme quantile estimation 4.2. Namely, we compute quantiles of the real valued random variable $\inf_{0\leq t\leq T}\phi^{1,2}(X_t)$. Results are presented in table 8.1. There is, to our knowledge, no algorithm in the scientific literature for the computation extreme quantiles in this context.

Adaptive splitting algorithm for the whole cumulation distribution function Figure 8.3 presents a typical run of the algorithm 4.3 for the estimation of the whole cumulative distribution function (c.d.f.) of the random variable $\inf_{0\leq t\leq T}\phi^{1,2}(X_t)$. Namely, we estimate the function

$$\delta \longmapsto \mathbb{P}\left(\inf_{0\leq t\leq T}\phi^{1,2}(X_t) \leq \delta\right).$$

Probability	Estimated quantile
10^{-2}	11.8±10%
10^{-3}	9.45±12%
10^{-4}	7.40±16%
10^{-5}	5.80±19%
10^{-6}	4.36±25%
10^{-7}	3.10±39%
10^{-8}	1.99±50%
10^{-9}	0.83±140%
10^{-10}	0.35±110

Table 8.1: Estimated minimum separation distance for a given probability value.

We compare the estimated survival function with the mean of estimates of probabilities $\mathbb{P}\left(\inf_{0\le t\le T}\phi^{1,2}(X_t)\le\delta_{\text{sep}}\right)$ already given in figure 8.3. The good point here is that the whole survival function is estimated with one run of a slightly modified version of the adaptive splitting algorithm 4.1. Consequently, the run time of algorithm 4.3 is the same as the one of 4.1.

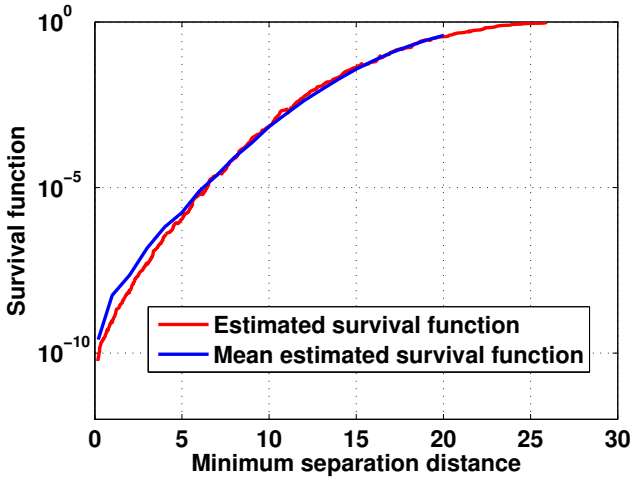


Figure 8.3: Estimated survival function with algorithm 4.3.

The adaptive weighted redistribution algorithm For the adaptive weighted redistribution algorithm 7.1 the Markov chain $\{X_k,\ k=1,\dots,m\}$ under consideration is defined by

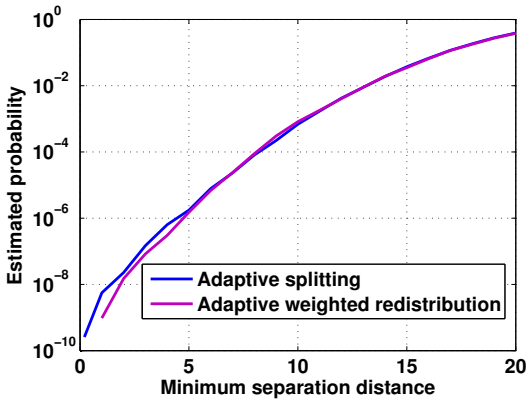
$$X_k=\left\{X^\delta(t_{5\times k}),\max\left\{\phi^{1,2}(X^\delta(t_{5\times l}),l\le k\right\}\right\},\tag{8.21}$$

where $\{X^\delta(t_k),\ k\le 0\}$ is the result of the Euler scheme for a time step δ for the process $\{X_t,\ t\ge 0\}$ as discussed in section 7.3.
We use the exponential selection function

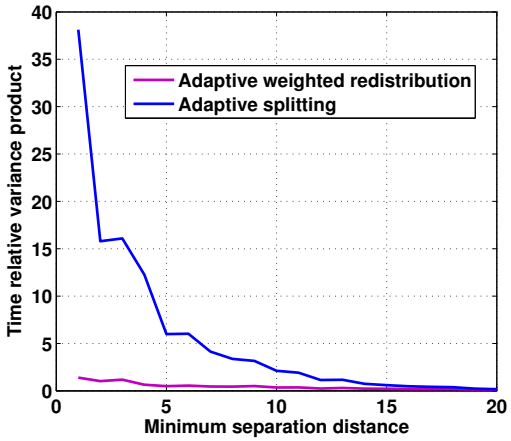
$$G_k(x_0,\dots,x_k)=\exp[\alpha(V(x_k)-V(x_{k-1}))]$$

where $V(x_k)=x_k^{(2)}$, for $x_k=(x_k^{(1)},x_k^{(2)})$ and $x_k^{(2)}$ is the second coordinate of x_k when x_k takes the form given in equation (8.21). The probability of interest (7.13) can thus be rewritten as follows

$$\mathbb{P}\left(\inf_{0\le t\le T}\phi^{1,2}(X_t)\le\delta_{\text{sep}}\right)=\mathbb{P}(V(X_m)\le\delta_{\text{sep}}).$$



(a) Mean estimated probability.



(b) Time relative variance product.

Figures 8.4: Comparison of the adaptive splitting and adaptive weighted redistribution algorithm.

We use a set of $N = 1000$ trajectories. We set $p_0 = 0.2$ and α^* and α_* that ensures too strong a too weak selection are equals to $\alpha^* = 10 + u$ where $u \sim \mathscr{U}[0,10]$ and $\alpha_* = 0$. Adding randomness to α^* enable to increase the diversity on the output parameter $\widehat{\alpha}$ of algorithm 7.1. See section 7.4 for additional remarks.

We compare the adaptive weighted redistribution algorithm 7.1 and the adaptive splitting algorithm for rare event estimation 3.1 in figures 8.4. In the left panel, we show the mean estimates of the probability (8.19). On the right panel, we compare the time relative variance product (RTV). The adaptive weighted redistribution algorithm RTV is up to 40 times faster than the one of the adaptive splitting algorithm. To understand the gain using the weighted redistribution algorithm, table 8.2 presents some additional results. The adaptive splitting algorithm gives better relative standard deviation, but its run time is up to 60 times higher to the run time of the weighted redistribution algorithm. Remark also that the run time of the adaptive weighted redistribution algorithm does not increase as the probability becomes smaller. Moreover, notice that the adaptive weighted redistribution algorithm consists of several runs of the weighted redistribution algorithm 7.1.

δ_{sep}	rSTD WR	rSTD Split	WR time (sec.)	Split time (sec.)
20	0.21	0.05	0.12	0.81
15	0.30	0.13	0.13	1.60
10	0.69	0.42	0.12	2.70
5	1.24	1.25	0.13	4.73
0.2	4.60	2.57	0.13	8.10

Table 8.2: rSTD and run time comparison with adaptive weighted redistribution algorithm and adaptive splitting algorithm.

8.4 The conflict zones take into account the relative velocity

8.4.1 Quantities of interest

The rare event probability We assume gain that the air traffic scenario involve n_a aircraft. The positions of these aircraft at time t are given by equation (8.6) with the vector $X_t = (X_t^1, \dots, X_t^{n_a})$. If the positions of each aircraft are assumed to belong to $\mathbb{R}^3$ then $X_t \in \mathbb{R}^{N_A}$ where $N_A = 3n_A$.

For aircraft i and j and $k = 1, \dots, 5$, the conflict zones $D_k^{i,j}$, D_k^i and D_k are presented in section 3.4. We recall that a pair (i, j) of aircraft is in conflict of level k at time t, if and only if

$$\begin{aligned} X_t \in D_k^{i,j} = \{x \in \mathbb{R}^{N_A}, |y^{i,j}(x) + \Delta v^{i,j}(x)| \le d_k, \text{ and} \\ |z^{i,j}(x) + \Delta r^{i,j}(x)| \le h_k, \text{ for some } \Delta \in [0, \Delta_k]\}, \end{aligned}$$

where, $v^{i,j}(X_t)$ is the relative speed and $r^{i,j}(X_t)$ is the relative rate of climbing of aircraft i and j. Next, D_k^i and D_k are defined with

$$D_k^i = \bigcup_{j \neq i} D_k^{i,j}, \tag{8.22}$$

and

$$D_k = \bigcup_{i=1}^{n_a} D_k^i. \tag{8.23}$$

For $i \neq j$ in $\{1, \dots, n_A\}$ and $k \ge 1$, we are interested in estimating the probability of the events that aircraft being in conflict during a fight duration of time T

$$\left\{X_t \in D_k^{i,j}, \text{ for some } 0 \le t \le T\right\}, \tag{8.24}$$

$$\left\{X_t \in D_k^i, \text{ for some } 0 \le t \le T\right\} \tag{8.25}$$

and

$$\{X_t \in D_k, \text{ for some } 0 \le t \le T\}. \tag{8.26}$$

The conflict zone that reaches a rare event probability value On a second thought, we are looking at some triplet (Δ, d, h) for which the probability of being in conflict relatively to this triplet is fixed. More precisely, for any triplet (Δ, d, h) we first define the corresponding conflict zones in the same way of equation (8.27) with

$$\begin{aligned} D^{i,j}(\Delta, d, h) = \{x \in \mathbb{R}^{N_A}, |y^{i,j}(x) + \delta v^{i,j}(x)| \le d, \text{ and} \\ |z^{i,j}(x) + \Delta r^{i,j}(x)| \le h, \text{ for some } \delta \in [0, \Delta]\}. \end{aligned} \tag{8.27}$$

We define also $D^i(\Delta, d, h)$ and $D(\Delta, d, h)$ by

$$D^i(\Delta, d, h) = \bigcup_{j \neq i} D^{i,j}(\Delta, d, h)$$

and

$$D(\Delta, d, h) = \bigcup_{i=1}^{n_A} D^i(\Delta, d, h).$$

Then, given a (small) probability value numbers $p_0 \in (0,1)$, we are looking at some triplets $\left(\Delta^{i,j}(p_0), d^{i,j}(p_0), h^{i,j}(p_0)\right)$ such as

$$\mathbb{P}\left(X_t \in D^{i,j}\left(\Delta^{i,j}(p_0), d^{i,j}(p_0), h^{i,j}(p_0)\right), \text{ for some } 0 \leq t \leq T\right) = p_0, \tag{8.28}$$

In the same way, we are also interested in the estimation of some triplets $\left(\Delta^i(p_0), d^i(p_0), h^i(p_0)\right)$ and $\left(\Delta(p_0), d(p_0), h(p_0)\right)$ such as

$$\mathbb{P}\left(X_t \in D^i\left(\Delta^i(p_0), d^i(p_0), h^i(p_0)\right), \text{ for some } 0 \leq t \leq T\right) = p_0 \tag{8.29}$$

and

$$\mathbb{P}\left(X_t \in D\left(\Delta(p_0), d(p_0), h(p_0)\right), \text{ for some } 0 \leq t \leq T\right) = p_0, \tag{8.30}$$

8.4.2 Characterization of conflict zones with real valued functions and continuous deformation between conflict zones

The goal of this subsection is to determine some functions

$$\psi^{i,j} : (\mathbb{R}^3)^{n_A} \mapsto \mathbb{R}$$

that satisfies the two following points

1. it exists $u_1^{i,j}, \ldots, u_5^{i,j} \in \mathbb{R}$ with

$$\mathbb{P}\left(\sup_{0 \leq t \leq T} \psi^{i,j}(X_t) \geq u_k^{i,j}\right) = \mathbb{P}\left(X_t \in D_k^{i,j}, \text{ for some } 0 \leq t \leq T\right), \quad \forall k = 1, \ldots, 5,$$

2. for all $p_0 \in (0,1)$, it exists $u^{i,j}(p_0)$ such that

$$\mathbb{P}\left(\sup_{0 \leq t \leq T} \psi^{i,j}(X_t) \geq u^{i,j}(p_0)\right) = p_0.$$

Namely, 1. means the functions are compliant with the air traffic management rules and with 2. we can construct zones for any given probability.

With such $\psi^{i,j}$ functions, the construction of functions ψ^i and ψ with $\psi^i, \psi : (\mathbb{R}^3)^{n_A} \mapsto \mathbb{R}$ such that

1. it exists $u_1^i, \ldots, u_5^i \in \mathbb{R}$ and $u_1, \ldots, u_5 \in \mathbb{R}$ with $\forall k = 1, \ldots, 5$

$$\mathbb{P}\left(\sup_{0 \leq t \leq T} \psi^i(X_t) \geq u_k^i\right) = \mathbb{P}\left(X_t \in D_k^i, \text{ for some } 0 \leq t \leq T\right),$$
$$\mathbb{P}\left(\sup_{0 \leq t \leq T} \psi(X_t) \geq u_k\right) = \mathbb{P}\left(X_t \in D_k, \text{ for some } 0 \leq t \leq T\right)$$

2. and for all $p_0 \in (0,1)$, it exists $u^i(p_0)$ and $u(p_0)$ such that

$$\mathbb{P}\left(\sup_{0\le t\le T} \psi^i(X_t) \ge u^i(p_0)\right) = p_0,$$
$$\mathbb{P}\left(\sup_{0\le t\le T} \psi(X_t) \ge u(p_0)\right) = p_0$$

will immediately follow, as explain hereafter.

8.4.2.1 Linear interpolation of parameter for the construction of intermediate zones

Here, we propose a continuous deformation between consecutive subsets into the sequence $D_k^{i,j}$, $k = 1,\dots,5$, for some fixed i, j. Then, continuous deformations between consecutive zones in the sequences D_k^i or D_k immediately follow thanks to equations (8.22) and (8.23). With such a way to proceed, we will be able to reach some space zones with a given reachability probability. Furthermore, these zones will be included between two consecutive zones in the sequences $D_k^{i,j}$, D_k^i or D_k.

First of all, to avoid the reachability probability of the first considered set $D_1^{i,j}$ to be too small, we assume the existence of d_0, h_0 and Δ_0 such that the probability of the set

$$\begin{aligned} D_0^{i,j} = \{x \in \mathbb{R}^{N_A}, |y^{i,j}(x) + \Delta v^{i,j}(x)| \le d_0, \text{ and} \\ |z^{i,j}(x) + \Delta r^{i,j}(x)| \le h_0, \text{ for some } \Delta \in [0,\Delta_0]\}. \end{aligned} \tag{8.31}$$

is greater than $1-\varepsilon$, for some small $\varepsilon > 0$. We define also as usual the sets D_0^i and D_0 with

$$D_0^i = \bigcup_{j\neq i} D_0^{i,j} \quad \text{and} \quad D_0 = \bigcup_{i=1}^{n_A} D_0^i.$$

The main idea is to consider k as the function $k(u) = u$, for u into the real interval [0,5]. Let us define the functions $d_{k(u)}$, $h_{k(u)}$ and $\Delta_{k(u)}$, for u restricted to $[0,5)$, by

$$\begin{aligned} d_{k(u)} &= (u - [u])d_{k([u]+1)} + (1 - u + [u])d_{k([u])}, \\ h_{k(u)} &= (u - [u])h_{k([u]+1)} + (1 - u + [u])h_{k([u])}, \\ \Delta_{k(u)} &= (u - [u])\Delta_{k([u]+1)} + (1 - u + [u])\Delta_{k([u])}. \end{aligned}$$

For $u = 5$, we denote $d_{k(5)} = d_5$, $h_{k(5)} = h_5$ and $\Delta_{k(5)} = \Delta_5$. Remark that if $u \in [s, s+1]$, for some $s \in \{0,\dots,4\}$ we can write $u = v + s$, with $v = u - [u] \in [0,1)$. Consequently, the simpler expressions for $d_{k(u)}$, $h_{k(u)}$ and $\Delta_{k(u)}$ follow

$$\begin{aligned} d_{k(u)} &= v d_{k(s+1)} + (1-v)d_{k(s)}, \\ h_{k(u)} &= v h_{k(s+1} + (1-v)h_{k(s)}, \\ \Delta_{k(u)} &= v\Delta_{k(s+1)} + (1-v)\Delta_{k(s)}. \end{aligned}$$

Hence, it clearly appears that each consecutive value in the sequences d_k, h_k and Δ_k, $k = 0,\dots,5$ is connected by segments. See figure 8.5.

Then we define $D_{k(u)}^{i,j}$ the region of $\mathbb{R}^{N_A}$ characterised by $d_{k(u)}$, $h_{k(u)}$ and $\Delta_{k(u)}$ in the following way

$$\begin{aligned} D_{k(u)}^{i,j} = \{x \in \mathbb{R}^{N_A}, |y^{i,j}(x) + \Delta v^{i,j}(x)| \le d_{k(u)}, \text{ and} \\ |z^{i,j}(x) + \Delta r^{i,j}(x)| \le h_{k(u)}, \text{ for some } \Delta \in [0,\Delta_{k(u)}]\}. \end{aligned} \tag{8.32}$$

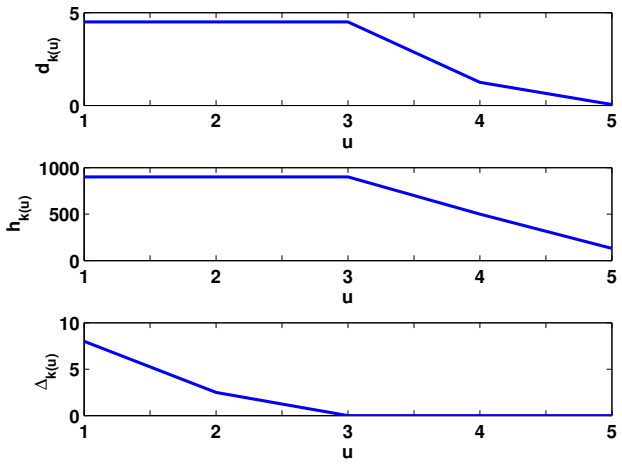


Figure 8.5: Linear interpolation between parameters.

It is clear that

$$D^{i,j}_{k(u)} \subset D^{i,j}_{k(u')} \text{ if } u \geq u'. \tag{8.33}$$

For the sake of notation, the conflict zones $D^{i,j}_{k(u)}$, for $u = 0,\dots,5$ are still denoted $D^{i,j}_{k}$ for $k = 0,\dots,5$.

We denote also the conflict zone $D^{i}_{k(u)}$ and $D_{k(u)}$ by

$$D^{i}_{k(u)} = \bigcup_{j\neq i} D^{i,j}_{k(u)}, \tag{8.34}$$

and

$$D_{k(u)} = \bigcup_{i=1}^{n_a} D^{i}_{k(u)}. \tag{8.35}$$

Definition 8.1. *We say that the pair (i,j) of aircraft is in conflict of level u if*

$$X_t = (X^1_t,\dots,X^{n_a}_t) \in D^{i,j}_{k(u)}.$$

Similarly, the aircraft i is said to be in conflict of level u if

$$X_t \in D^{i}_{k(u)}$$

and the scenario is said to be in conflict of level u if

$$X_t \in D_{k(u)}.$$

8.4.2.2 Score functions for the characterization of conflict zones

The goal here is to rewrite the three following events

$$\begin{aligned} &\left\{X_t \in D^{i,j}_{k(u)}, \text{ for some } t \in [0,T]\right\},\\ &\left\{X_t \in D^{i}_{k(u)}, \text{ for some } t \in [0,T]\right\},\\ &\left\{X_t \in D_{k(u)}, \text{ for some } t \in [0,T]\right\}, \end{aligned} \tag{8.36}$$

in terms of exceedance over thresholds of real valued functions.

For $i, j \in \{1, \dots, n_a\}$ with $i \neq j$ and $x = (x^1, \dots, x^{n_a})$ consider first the following real valued function

$$\psi^{i,j}(x) = \begin{cases} \sup\left\{u \in [0,5],\, x \in D^{i,j}_{k(u)}\right\} & \text{if } x \in D^{i,j}_0 \\ 0 & \text{if } x \notin D^{i,j}_0. \end{cases} \tag{8.37}$$

Notice that $D^{i,j}_{k(u)} \subset D^{i,j}_0$ for all $u \in [0,5]$.

Remark 8.1. *The $\psi^{i,j}$ functions are defined on $\mathbb{R}^{n_A}$, namely on the whole state space of the trajectories of $\{X_t,\ t \geq 0\}$. Hence, we can assign a score for any given realisation of $\{X_t,\ t \geq 0\}$ and the proposed algorithms such as 4.1,4.2,4.3 and 7.1 can be implemented.*

Using the definitions of $D^{i,j}_{k(u)}$ in equation (8.32) and $\psi^{i,j}$, we easily show the following result.

Proposition 8.1. *A pair (i, j) of aircraft is in conflict of level u at time t if, and only if,*

$$\psi^{i,j}(X_t) \geq u.$$

Moreover, the probability of interest (8.24) is rewritten in term of threshold exceedance of the random variable $\sup_{0\leq t\leq T} \psi^{i,j}(X_t)$ as follows

$$\mathbb{P}\left(X_t \in D^{i,j}_{k(u)},\ \textit{for some}\, 0 \leq t \leq T\right) = \mathbb{P}\left(\sup_{0\leq t\leq T} \psi^{i,j}(X_t) \geq u\right).$$

In the same way, defining ψ^i and ψ for $x \in \mathbb{R}^{n_A}$ with

$$\begin{aligned} \psi^i(x) &= \sup\left\{u \in [1,5],\, x \in D^i_{k(u)}\right\} \\ &= \sup_{j\neq i} \psi^{i,j}(x) \end{aligned} \tag{8.38}$$

and

$$\begin{aligned} \psi(x) &= \sup\left\{u \in [1,5],\, x \in D_{k(u)}\right\} \\ &= \sup_i \psi^i(x) \end{aligned} \tag{8.39}$$

leads to the following identities

$$\begin{aligned} \left\{X_t \in D^i_{k(u)}\right\} &= \bigcup_{j\neq i} \left\{\psi^{i,j}(X_t) \geq u\right\} \\ &= \{\psi^i(X_t) \geq u\} \end{aligned} \tag{8.40}$$

and

$$\begin{aligned} \left\{X_t \in D_{k(u)}\right\} &= \bigcup_i \left\{X_t \in D^i_{k(u)}\right\} \\ &= \left\{\psi(X_t) \geq u\right\}. \end{aligned} \tag{8.41}$$

The same kind of results than those given in proposition 8.1 can be obtained.

Proposition 8.2. *The probabilities of interest (8.25) and (8.26) are rewritten in term of threshold exceedance of the random variable $\sup_{0\leq t\leq T} \psi^i(X_t)$ and $\sup_{0\leq t\leq T} \psi(X_t)$ as follows*

$$\mathbb{P}\left(X_t \in D^i_{k(u)},\ \textit{for some}\, 0 \leq t \leq T\right) = \mathbb{P}\left(\sup_{0\leq t\leq T} \psi^i(X_t) \geq u\right),$$

and

$$\mathbb{P}\left(X_t \in D_{k(u)},\ \textit{for some}\, 0 \leq t \leq T\right) = \mathbb{P}\left(\sup_{0\leq t\leq T} \psi(X_t) \geq u\right).$$

Propositions 8.1 and 8.2 are the central point of the section 8.4. Indeed, all the algorithms that require a score function can now be used for the estimation of the rare event probability of conflict.

The conflict zones that have a given reachability probability We can also parametrize some conflict zones for some given level of risk. This latter point is achieved considering, for a given p_0 the quantities

$$u^{i,j}(p_0) = \inf\left\{u \geq 0,\ \mathbb{P}\left(\sup_{0\leq t\leq T} \psi^{i,j}(X_t) \geq u\right) \leq p_0\right\},$$

$$u^{i}(p_0) = \inf\left\{u \geq 0,\ \mathbb{P}\left(\sup_{0\leq t\leq T} \psi^{i}(X_t) \geq u)\right) \leq p_0\right\}$$

and

$$u(p_0) = \inf\left\{u \geq 0,\ \mathbb{P}\left(\sup_{0\leq t\leq T} \psi(X_t) \geq u\right) \leq p_0\right\}.$$

8.4.2.3 Effective calculation of the score functions

Here, we give some details on the explicit calculation of the score functions given in equation (8.37). We focus on the estimation that a given pair (i, j) of aircraft being in conflict. The probability characterization that the aircraft i be in conflict with any other aircraft will immediately follow with equation (8.38) and the probability that the whole scenario be in conflict is deduced with (8.39). This section contains very basic algebra and its reading can be omit for the reader who is not interested in implementing the algorithm in the case of conflict probability estimation.

Here, the quantity $\psi^{i,j}(x) = \sup\left\{u \in [0,5],\ x \in D^{i,j}_{k(u)}\right\}$ is calculated. For the sake of understanding, we explain the way to compute

$$\sup\left\{u \in [1,5],\ x \in D^{1,2}_{k(u)}\right\},$$

where we remind that

$$\begin{aligned} D^{1,2}_{k(u)} = \{x \in \mathbb{R}^{n_A},\ & |y^{i,j}(x) + \Delta v^{1,2}(x)| \leq d_{k(u)},\ \text{and} \\ & |z^{1,2}(x) + \Delta r^{1,2}(x)| \leq h_{k(u)},\ \text{for some } \Delta \in [0, \Delta_{k(u)}]\}. \end{aligned} \tag{8.42}$$

We first define the functions f and g as follows

$$f(\Delta, u) = \mid y^{1,2}(x) + \Delta v^{1,2}(x) \mid^2 - d^2_{k(u)},$$

$$g(\Delta, u) = \mid z^{1,2}(x) + \Delta r^{1,2}(x) \mid - h_{k(u)}.$$

Then, we denote by $m_f(u)$ and $m_g(u)$ their maxima over the real interval $[0, \Delta_{k(u)}]$:

$$m_f(u) = \sup\left\{f(\Delta, u),\ 0 \leq \Delta \leq \Delta_{k(u)})\right\},$$
$$m_g(u) = \sup\left\{g(\Delta, u),\ 0 \leq \Delta \leq \Delta_{k(u)})\right\}.$$

Finally, if we denote $u^{(1)}$ and $u^{(2)}$ the following real numbers

$$u^{(1)} = \sup\left\{u' \in [0,5],\ m_f(u') \leq 0\right\},$$

$$u^{(2)} = \sup\left\{u' \in [0,5],\ m_g(u') \leq 0\right\},$$

it is clear that for any fixed real number u we have

$$x \in D^{1,2}_{k(u)} \iff u^{(1)} \geq u \text{ and } u^{(2)} \geq u. \tag{8.43}$$

Thus, we obtain with equations (8.42) and (8.43) that

$$\sup\left\{u \in [0,5],\, x \in D^{1,2}_{k(u)}\right\} = \min(u^{(1)}, u^{(2)}).$$

The problem is thus reduced in finding the $u^{(1)}$ and $u^{(2)}$ bounds. First, remark that

$$\sup\left\{u \in [0,5],\, m_f(u) \leq 0\right\} = \max_{s=0,\dots,4} \sup_{0\leq v\leq 1} \left\{s+v,\, m_f(s+v) \leq 0\right\}.$$

It is thus sufficient to work on each segment. Let s be a fixed integer in $\{0,\dots,4\}$. Taking into account that f is a polynomial function of degree 2 with positive dominant coefficient, the quantity $m_f(s+v)$ is easily computed:

$$m_f(s+v) = \begin{cases} f(0, s+v) & \text{if } \Delta^* \leq 0 \\ f(\Delta^*, s+v) & \text{if } 0 \leq \Delta^* \leq \Delta_{k(s+v)} \\ f(\Delta_{k(s+v)}, s+v) & \text{otherwise,} \end{cases}$$

where $\Delta^* \in \mathbb{R}$ is such that $\partial/\partial_\Delta f(\Delta^*, u) = 0$, where $\partial/\partial_\Delta f$ is the partial derivative of f with respect to Δ. Note that Δ^* does not depend on u. The computation of u_1 can now be derived with very basic algebra and is not detailed here.

8.4.3 Implementation of the algorithms

We consider a three aircraft scenario. The position of the three aircraft at time t are given in the vector $X_t = (X_t^1, X_t^2, X_t^3)$ where $X_t^j \in \mathbb{R}^3$. The dynamic of each aircraft is given in section 8.2 and the proposed flight plan is given in figure 8.6. First, we are first interested in the event that the aircraft 1 being in conflict with aircraft 2 or 3. Namely, the event of interest is the following

$$\left\{X_t \in D_k^1,\ \text{for some } 0 \leq t \leq T\right\}, \tag{8.44}$$

where the flight duration time $T = 20$ minutes and the conflict zone number $k = 1,\dots,5$. The event can be rewritten in term of the score function ψ^1 defined in equation (8.38) as follows

$$\left\{X_t \in D_k^1,\ \text{for some } 0 \leq t \leq T\right\} = \left\{\sup_{0\leq t\leq T} \psi^1(X_t) \geq u_k\right\}.$$

To avoid estimates equals to zero as explained in section 8.4.2.1, the $D_0^{i,j}$ sets defined in equation (8.31) are characterized with $d_0 = 25$, $h_0 = 900$ and $\Delta_0 = 8$.

Probability estimation results are summed up in table 8.4 and details for the implementation are given below. The algorithm under consideration are the splitting algorithm 3.1 (non-adapt. splitting), the adaptive splitting algorithm 4.1 (adapt. splitting) and the adaptive weighted redistribution algorithm 7.1 (adapt. WR). Each estimation is performed over 50 retrials.

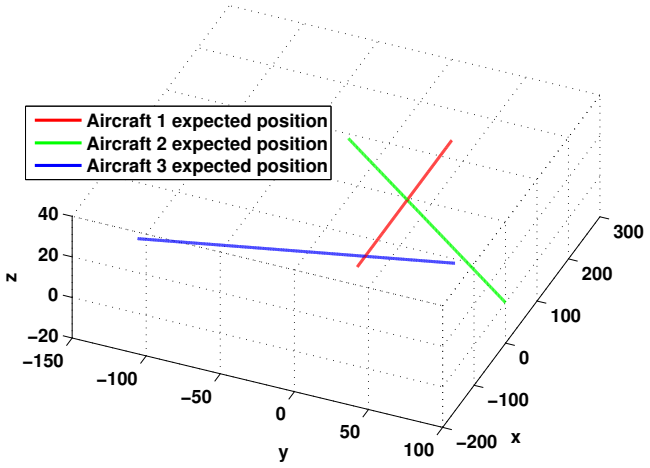


Figure 8.6: Proposed flight plan for the three aircraft scenario.

Splitting algorithm For the adaptive splitting algorithm 4.1 we use a set of $N = 1000$ trajectories to estimate conditional probabilities and entrance distributions. The thresholds are estimated using a set of 20 trajectories.
For the non adaptive splitting algorithm 3.1 we also use a set of $N = 1000$ trajectories at each threshold. The splitting algorithm 3.1 is performed with the additional intermediates zones used in [Rubino and Tuffin, 2009],[Blom et al., 2007a] and [Prandini et al., 2011]. See section 1.4 for details. We consider also the additional zone $D_0^1 = D_0^{1,2} \cup D_0^{1,3}$ in its implementation, otherwise the algorithm stops most of the time because none of the trajectories reaches the set D_1^1, starting from the initial distribution.

Weighted redistribution algorithm For the adaptive weighted redistribution algorithm 7.1 the Markov chain $\{X_k,\ k = 1,\dots,m\}$ under consideration is defined by

$$X_k = \left\{X^\delta(t_{5\times k}), \max\left\{\psi^1(X^\delta(t_{5\times l}), l \le k\right\}\right\}, \tag{8.45}$$

where $\{X^\delta(t_k),\ k \le 0\}$ is the result of the Euler scheme for a time step δ for the process $\{X_t,\ t \ge 0\}$ as discussed in section 7.3.
We use the exponential selection function

$$G_k(x_0,\dots,x_k) = \exp[\alpha(V(x_k) - V(x_{k-1}))]$$

where $V(x_k) = x_k^{(2)}$, for $x_k = (x_k^{(1)}, x_k^{(2)})$ and $x_k^{(2)}$ is the second coordinate of x_k when x_k takes the form given in equation (8.45). The probability of interest (7.13) can thus be rewritten as follows

$$\mathbb{P}\left(\inf_{0\le t\le T} \psi^1(X_t) \le \delta_{\text{sep}}\right) = \mathbb{P}(V(X_m) \le \delta_{\text{sep}}),$$

where m is the last iteration of the Markov chain $\{X_k^\delta,\ k = 1,\dots,m\}$. Finally, we set $N = 5000$, $p_0 = 0.2$, $\alpha_* = 0$ and $\alpha^* = 10 + u$, where u is uniformly distributed into the real interval $[0,10]$ as explained in section 7.4. According to our experiments, smaller values of N give most of time an estimate equals to zero.

Zone	non-adapt. splitting	adapt. splitting	adapt. WR
1	100%	100%	100
2	14%	100%	98
3	14%	100%	98
4	12%	100%	98
5	14%	100%	52

Table 8.3: Percentage of convergence of the splitting algorithm 3.1, the adaptive splitting algorithm 4.1 and the adaptive weighted redistribution algorithm 7.1.

Zone	non-adapt. splitting		adapt. splitting		adapt. WR	
	Estim.	Time (sec)	Estim.	Time (sec)	Estim.	Time (sec)
1	1.8e-1±5%	10	5.6e-1±4%	22	2.48e-1±6%	205
2	7.2e-4±54%	11	9.7e-4±76%	103	9.30e-4±70%	170
3	5.3e-4±65%	12	1.3e-4±100%	114	4.7e-4±65%	150
4	8.4e-5±116%	15	1.6e-5±163%	134	1.3e-5±150%	120
5	2.1e-7±127%	17	7.4e-7±460%	184	3.8e-7±240%	150

Table 8.4: Conflict probability estimation with the splitting algorithm 3.1, the adaptive splitting algorithm 4.1 and the adaptive weighted redistribution algorithm 7.1.

Percentage of convergence of the algorithms For the splitting algorithm (and even the adaptive splitting algorithm), it is possible that none of the trajectories sample from an intermediate region reaches the next intermediate region. Consequently, there is no estimation of the probability. For the adaptive weighted redistribution algorithm some estimations are equals to zero because none of the trajectories of the Markov chain reaches the set $V^{-1}(]-\infty, d_{\text{sep}}])$ and the dichotomy performed in algorithm 7.1 stops. Table (8.3) shows the percentage of convergence of the algorithms.

Rare event probability estimation Results for the estimation of the probability of the event 8.44 are summed up in the table 8.4. In this table, we compute mean estimates and rSTD for each algorithm above mentioned only for convergent runs. The non adaptive splitting gives better results but there less than 15% of the runs have converged. The adaptive weighted redistribution algorithm gives lower variance than the adaptive splitting algorithm but also may not converge, as seen in table 8.3

Estimation of conflict zones for a given probability value Next, we are interested in estimating some conflict zones that have a given probability value. To this purpose, we use the adaptive splitting algorithm for extreme quantile estimation 4.2. In that case, it is worth noting that we need the characterization with the ψ^1 function of the rare event of interest. Consequently, we need the continuous deformation between conflict zones given in section 8.4.2.1. Results are given in table 8.5. For a probability p, the estimated quantile u is an approximation of $u(p)$ such that

$$\mathbb{P}\left(\sup_{0\leq t\leq T}\psi^1(X_t)\geq u(p)\right)=p.$$

Then, the quantities $d(u)$, $\Delta(u)$ and $h(u)$ characterize the conflict zones $D^i_{k(u)}$ defined in equation (8.34). Some results have to be interpreted. We first recall that the continuous deformation between the conflict zone numbered with $k = 1,\dots,5$ relies on linear interpolation between parameters that characterize them. Then, if we look carefully the figure 8.5 we observe that, sometimes, two consecutive values of a parameter constant between two consecutive integer value of u. That is why the estimation of $\Delta(u)$, $h(u)$ given in table 8.1 can be exact, namely with variance 0. After that, we see that the estimated quantile (and the estimated quantities $d(u)$, $\Delta(u)$ and $h(u)$) cannot reach the probability 10^{-8}. Actually, the algorithm stops most of the time. The algorithm cannot go further $u = 6$ simply because the function ψ^1 is no longer defined for $u > 6$. Indeed, $u = 6$ is the collision case and it is meaningless to define ψ^1 for $u > 6$.

Probability	Quantile u	$d(u)$	$\Delta(u)$	$h(u)$
10^{-2}	1.69±2%	9.83±7%	8±0%	900±0%
10^{-3}	2.41±40%	6.54±70%	7.05±35%	879±7%
10^{-4}	4.04±22%	3.26±41%	1.61±176%	727±30%
10^{-5}	5.11±7%	1.70±68%	0±0%	519±33%
10^{-6}	5.72±6%	1.07±70%	0±0%	416±30%
10^{-7}	5.9±3%	0.85±30%	0±0%	377±21%
10^{-8}	⁄	⁄	⁄	⁄

Table 8.5: Estimated minimum separation distance for a given probability value.

Conclusion of the chapter

- We applied the adaptive splitting algorithm 4.1 and the adaptive weighted redistribution algorithm 7.1 to estimate conflict probability. In the case of minimum separation distance loss between aircraft, the adaptive splitting algorithm create thresholds for the splitting that characterized the intermediate regions. Thus the splitting is fully implementable without any tuning. Moreover, we observed that the adaptive weighted redistribution algorithm is up to 40 times faster than the splitting algorithm to reach same relative standard deviation. With the adaptive splitting algorithm for extreme quantile estimation we got minimum separation distance that aircraft should respect to ensure a given conflict probability.

- If the conflict zones take into account the relative speed and relative rate of climbing between aircraft, we detailed score functions to use in some algorithms proposed in this thesis. To this end, we used a continuous deformation between conflict zones thanks to a linear interpolation between parameters that characterize them. Thus, we could apply the adaptive splitting algorithm 4.1 and the adaptive weighted redistribution algorithm 7.1 to estimate conflict probability. We observed that the non-adaptive splitting algorithm used in [Rubino and Tuffin, 2009],[Blom et al., 2007a] and [Prandini et al., 2011] gives better relative standard deviation and lower computation time. However, the adaptive splitting algorithm gives a convergence percentage of 100% whereas the non-adaptive splitting algorithm has a convergence percentage equals to 14%. Moreover, with the score function characterisation of the conflict zones, we could give parameters for conflict zones that have a given reachability probability. Such a result does not exist in the scientific literature and may improve regulation rules in the air traffic management.

Part III

Assessment

Conclusions

Goal of this section
Here we synthesise contributions and limitations of the thesis. We give next some ideas of improvements and extensions.

Contributions

In this thesis, the main topic was the simulation and the estimation of the very small probability

$$\mathbb{P}(T_B \leq T),$$

where $T_B = \inf\{t \geq 0,\ X(t) \in B\}$ and $\{X(t),\ t \geq 0\}$ is a continuous-time continuous-state space stochastic process. We explored several techniques that are based on interacting particle system algorithms. More precisely, contributions can be decomposed into the following points.

The splitting algorithm We were first interested in the splitting algorithm. After the state of the art, we concluded that it is the most suitable algorithm to estimate the probability $\mathbb{P}(T_B \leq T)$ that a time continuous stochastic process $\{X(t),\ t \geq 0\}$ with continuous state space enter a rare set B before some time T. If the rare region can be rewritten with $B = \phi^{-1}([0, +\infty))$, we derived a method to estimate the intermediate region required by the splitting algorithm. This method is fully adaptive and does not require any tuning. Furthermore, it is more general than the ones given in [Garvels, 2000] and [Cérou and Guyader, 2007] and has smaller run time than the multidimensional one of [Cérou et al., 2012].
The good point is that the characterization for B as threshold exceedance of the ϕ function is not restrictive. Indeed, as detailed in section 1.2.4 all the problem studied in the scientific literature uses this characterization, except the one of the conflict probability estimation detailed in section 3.4. In section 8.4.2 we gave such a characterization for the latter mentioned problem.

Extreme quantile estimation We extended the adaptive splitting algorithm for rare even estimations to an adaptive algorithm to compute extreme quantiles of the law of the supremum over time T of the process $\{\phi(X(t)),\ t \leq 0\}$. In particular, this led to the determination of some conflict zones that depends on the relative speed of the each pair of aircraft and that have a given probability value to be reached. Moreover, these zones are compliant with the conflict zones already given by the air traffic management. If we extend again this method, we can have in one single run of the adaptive splitting algorithm the whole cumulative distribution function of $\sup_{0 \leq t \leq T} \phi(X(t))$, in particular in the rare regime.

Bias and variance in the splitting We highlighted a bias in the implementation of the splitting algorithm, even if it is theoretically unbiased. We gave a correction procedure for this bias. After

that we investigated on the optimal regions that could be used in the splitting algorithm. We show that they can reduce up to ten time the needed number of trajectories compare to the naive non optimal splitting algorithm. We gave two methods for the estimation of the optimal region for small dimension of the state space of $\{X(t),\ t \geq 0\}$.

A weighted splitting algorithm We created a unified framework for the splitting and the weighted redistribution algorithm. That is to say, we derived an original interacting particle system algorithm for rare event estimation where the potential functions can be non-negative. We showed that this method can be used to give more importance to some well chosen trajectories in the splitting. With such a method, the variance in the estimation of the rare event probability $\mathbb{P}(T_B \leq T)$ is about half way between the splitting with the naive, non optimal region and the splitting with the optimal region (most of the time unrealistic).

The weighted redistribution algorithm We gave a way to make automatic the implementation of the weighted redistribution algorithm, originally exposed in [Del Moral and Garnier, 2005]. We successfully tested our procedure in the field of optical fiber and financial engineering. These two problems ware addressed with the weighted redistribution algorithm in [Garnier and Del Moral, 2006] and [Carmona and Crepey, 2010, Carmona et al., 2009, Giesecke et al., 2010]. Furthermore, we showed that the weighted redistribution algorithm can be used instead of the splitting to estimate the probability $\mathbb{P}(T_B \leq T)$ if T is deterministic. We numerically showed that this new estimation method can reduce up to forty times the time relative variance product for some cases.

Limitations

Central limit theorem for the adaptive splitting Even though performances of the adaptive splitting algorithm are very close to the ones of the splitting algorithm in term of variance, no theoretical results have been derived for the asymptotic variance of the splitting algorithm.

Long run time for bias correction As far as the bias correction procedure is concerned, it can be very costly to implement. Indeed, we have to compute some distance to the boundary of a subset at every iteration time for every trajectory. However, we have to keep in mind that the splitting algorithm is biased. Consequently, the time step for the discretization of the stochastic process under consideration must not be too large to prevent from the bias.

Costly estimation of optimal region The proposed methods for the estimation of the optimal region suffer from dimension. This is not surprising because they require to know, for every instant time t and every state x, the probability that the process reaches the rare event probability set before time T starting at x at time t. An additional conclusion of this study is that trying to estimate and use the optimal region is too costly and very hard to implement for high dimensions. If we try to do it, we could lose the benefits of the adaptive splitting algorithm, which is this simplicity of implementation.

No tuning for the I2S algorithm For the combined splitting and sampling algorithm 6.1, the choice of the functions that weight the trajectories has not been completely handled. We only numerically showed that a good choice of such a function can lead to variance reduction. Further work should be done to enable this algorithm to be used in practice.

Lack of theoretical background for the adaptive weighted redistribution algorithm We did not address any theoretical justification for the use of the adaptive weighted redistribution algorithm. Moreover, we can use this algorithm only for deterministic final time T, not random.

Perspectives

A central limit theorem for the adaptive splitting algorithm In [Cerou et al., 2014], a central limit theorem has been derived for the adaptive splitting algorithm in the case of a static distribution. Adapting the proof in the case of algorithm 4.1 would for example require to work with differentiable manifolds on the space the trajectories of the process, which is un-tractable. A good starting point for the demonstration of a central limit theorem for the algorithm 4.1 would be, as in [Cerou et al., 2014], to use the coarea formula [Evans and Gariepy, 1992, Section 3.4] which states that for any integrable function f and any ϕ with the adequate properties

$$\int_{\mathbb{R}^d} f(z)\mathrm{d}z = \int_{-\infty}^{+\infty} \mathrm{d}u \int_{\phi^{-1}(u)} f(z)\frac{\mathscr{H}(\mathrm{d}z)}{|\phi'(z)|}, \tag{8.46}$$

where $\mathscr{H}(\mathrm{d}z)$ is the $(d-1)$-dimensional Hausdorff measure. We could think about a recursive proof as the one of the theorem 4.1. We take now the notations of theorem 4.1. If f equation in (8.46) equals to

$$f(z) = \tilde{q}_k(z) f(z) \mathbb{1}_{\phi(z)\geq S_k},$$

where $\tilde{q}_k(z)$ is the density of $\tilde{\eta}_k$, supposing it exists. Then we have that

$$\begin{aligned}\int_{\mathbb{R}^d} f(z)\mathrm{d}z &= \int_{-\infty}^{+\infty} \mathrm{d}u \int_{\phi^{-1}(u)} \tilde{q}_k(z) f(z) \mathbb{1}_{\phi(z)\geq S_k} \frac{\mathscr{H}(\mathrm{d}z)}{|\phi'(z)|} \\ &= \int_{S_k}^{+\infty} \mathrm{d}u \int_{\phi^{-1}(u)} f(z)\tilde{q}_k(z)\frac{\mathscr{H}(\mathrm{d}z)}{|\phi'(z)|}.\end{aligned}$$

Then, we have the same formula with the empirical threshold $\tilde{S}$, which lead to

$$\tilde{\eta}_k(\tilde{g}_k f) - \tilde{\eta}_k(g_k f) = \int_{\tilde{S}_k \wedge S_k}^{\tilde{S}_k \vee S_k} \mathrm{d}u \int_{\phi^{-1}(u)} f(z)\tilde{q}_k(z)\frac{\mathscr{H}(\mathrm{d}z)}{|\phi'(z)|}.$$

Hence,

$$\sqrt{N}\left(\tilde{\eta}_k(\tilde{g}_k f) - \tilde{\eta}_k(g_k f)\right) =)\sqrt{N}(\tilde{S}_k - S_k)\frac{1}{\tilde{S}_k - S_k}\int_{\tilde{S}_k \wedge S_k}^{\tilde{S}_k \vee S_k} \mathrm{d}u \int_{\phi^{-1}(u)} \tilde{q}_k(z) f(z)\frac{\mathscr{H}(\mathrm{d}z)}{|\phi'(z)|}.$$

Now, if we have the convergence in distribution

$$\sqrt{N}(\tilde{S}_k - S_k) \Rightarrow \mathscr{N}(0, v_k),$$

and assuming that the following convergence holds

$$\frac{1}{\tilde{S}_k - S_k}\int_{\tilde{S}_k \wedge S_k}^{\tilde{S}_k \vee S_k} \mathrm{d}u \int_{\phi^{-1}(u)} \tilde{q}_k(z) f(z)\frac{\mathscr{H}(\mathrm{d}z)}{|\phi'(z)|} \xrightarrow[N\to+\infty]{} \int_{\phi^{-1}(S_k)} \tilde{q}_k(z) f(z)\frac{\mathscr{H}(\mathrm{d}z)}{|\phi'(z)|}$$

then we obtain with Slustky's lemma the convergence of $\tilde{\eta}_k(\tilde{g}_k f) - \tilde{\eta}_k(g_k f)$.

On the weighted redistribution algorithm We limited the choice to the potential function that have to be used in the weighted redistribution algorithm to the choice of a real parameter, say α. We observed in figures 7.3,7.6 and 7.8 that the best parameter according to the heuristic derived in section 7.2.2.1 is the one that minimizes a convex function. As said in section 7.4.1, this is highly linked to the work of [Jourdain and Lelong, 2009]. We worked in this way. Instead of considering the minimisation problem of the empirical expectation proposed in [Jourdain and Lelong, 2009], we tried to deal with the approximation of the variance $\widehat{V}(\alpha)$ for the point wise probability density function estimation. This approximation is detailed in [Del Moral and Garnier, 2005, equation 3.18]. It depends on the set of sampled trajectories (with the selection mutation scheme), denoted by $\mathscr{S}(\alpha) = \{\widehat{X}^i_{0:m}(\alpha),\ i = 1,\dots,N\}$ and on the product of normalisation constants that depends itself on $\mathscr{S}(\alpha)$, denoted by $n(\alpha,\mathscr{S}(\alpha))$. Hence we can write

$$\widehat{V}(\alpha) = F\big(n(\alpha,\mathscr{S}(\alpha)),\mathscr{S}(\alpha)\big),$$

for a certain function F. One of the possible improvements to the tuning of the weighted redistribution algorithm instead of algorithm 7.1 is to consider the normalisation constant as a function $\alpha \mapsto n(\alpha,\mathscr{S}(\alpha_0))$ of the parameter α, and to consider as a constant the sampled trajectories $\mathscr{S}(\alpha_0)$, for some fixed α_0. We have numerically observed that the function

$$\alpha \longmapsto F\big(n(\alpha,\mathscr{S}(\alpha_0)),\mathscr{S}(\alpha_0)\big)$$

is a convex function of the α parameter. Moreover, for most of the tested cases, the minimizer α^* of this function corresponds to the parameter that minimize the variance $\widehat{V}(\alpha)$. As a result, we could use only one run of the weighted redistribution algorithm to find the good α parameter to use. We recall that the adaptive weighted redistribution 7.1 consists of several runs of the weighted redistribution algorithm 3.2. However, for some cases, this method fails when the probability to estimate becomes very small ($< 10^{-7}$). A way for further research could be to consider the decomposition of the potential function G_k, $k = 1,\dots,m$ on a basis (e.g. Fourier or polynomial) and to estimate their coefficients, instead of considering the potential functions to belong to a one dimensional parametrised family. This idea has been suggested by Emmanuel Gobet.

Extension of the proposed algorithms to stochastic hybrid systems The framework of this study is not completely suitable for stochastic processes with regime changes. See [Krystul and Blom, 2006] for a detailed framework. However, some algorithms can readily be modified to fit this framework. It is in particular the case for the adaptive splitting for rare event estimation 4.1, the adaptive splitting for extreme quantile estimation 4.2 and the adaptive splitting for the whole cumulative distribution 4.3. The nested subsets required by the algorithms of [Krystul, 2006, Krystul and Blom, 2006, Blom et al., 2007a, Krystul and Blom, 2005] and [Blom et al., 2006] can be estimated in the very same way algorithms 4.1, 4.2 and 4.3 do. We recall that the conflict zones considered by the latest mentioned papers can be characterized by a score function, as explained in section 8.4. Moreover, the correction procedure is still valid since, in each regime of a stochastic differential equation with regime changes, the dynamics is governed by a stochastic differential equation, as considered in the section 5.2. The implementation of an adaptive weighted redistribution algorithm adapted to stochastic differential equation with regime changes can also be done. By the way, this have to be linked with the island particle system theory derived in [Vergé et al., 2013].

Combination with filtering The proposed algorithms can be combined to filtering methods if the model needs to be corrected on-line. This may lead to less estimation errors due to model errors. Such applications could be performed for aircraft trajectory model [Ichard and Baehr, 2013], unmanned aircraft [Serle, 2014] and rocket navigation [Nordlund and Gustafsson, 2009].

Combination with island particle models Most of the proposed algorithm in this thesis are based on interacting system particle approximations of some Feynman-Kac measure that characterize the rare event probability. In [Vergé et al., 2013], it shown that such algorithms can be improved in term of variance reduction if we consider some "island" of particle. We did not work in that way, but algorithms that combine the ones of this thesis and the ones of [Vergé et al., 2013] should lead to further variance reduction.

General sum up

In this thesis we were interested in the estimation of the following probability

$$\mathbb{P}(X(t) \in B \,; \text{for some } t \leq T), \tag{8.47}$$

where $\{X(t),\ t \geq 0\}$ is a stochastic process, T is a deterministic or random stopping time and B is a subset of the state space of $\{X(t),\ t \geq 0\}$. The probability (8.47) is assumed to be very small. This estimation problem arises in nuclear engineering [Kahn and Harris, 1951], telecommunication network [Garvels and Kroese, 1998], renewal energy [Wadman et al., 2013] and air traffic management [Jacquemart and Morio, 2013],[Prandini et al., 2011]. The bibliography study detailed in section 1 showed that the estimation problem is most of the time performed with the splitting algorithm. Moreover, when the trajectories of $\{X(t),\ t \geq 0\}$ are continuous, or at least piece-wise continuous, the splitting seems to be the only choice for the user.

If the rare set B is such that $B = \{x,\ \phi(x) \geq S\}$ for some $S \in \mathbb{R}$ and some ϕ a real valued function, we proposed an adaptive splitting algorithm that requires no tuning for the estimation of the probability (8.47). This algorithm relies on the estimation of quantiles of the distribution of $\sup_{0\leq t\leq T} \phi(X(t))$. Consequently, we also derived an algorithm for the estimation of extreme quantile of the law of $\sup_{0\leq t\leq T} \phi(X(t))$. Eventually, we proposed an algorithm that gives in one run the whole cumulative distribution function the distribution of $\sup_{0\leq t\leq T} \phi(X(t))$.

We investigated the restrictiveness of the characterization of B with ϕ. The bibliography study and some computations on the conflict zone in air traffic management showed that all the problems considered in the scientific literature for the estimation of (8.47) [Kahn and Harris, 1951], [Garvels and Kroese, 1998], [Wadman et al., 2013], [Jacquemart and Morio, 2013], [Prandini et al., 2011] can be rewritten in such a threshold exceedance of a real valued function.

We showed that a bias can appear in the estimation with the splitting algorithm and we detailed a correction procedure. Then, for some simple cases, we gave estimations of the optimal regions that can be used in the splitting algorithm and probability estimation results with them. We observed that up to ten times more trajectories are needed to reach the performances of the splitting with the optimal region compared to the splitting with naive non optimal regions. However, for dimension greater than two it does not seems reasonable to use these optimal regions for computational cost reasons.

To avoid the use of the optimal regions for additional variance reduction with the splitting algorithm, we proposed a method that combines the splitting algorithm with an importance sampling on the entrance time into the intermediate regions. The idea is to consider some naive non optimal region (for example given by algorithm 4.1) and to use different redistribution functions. With such a way, the variances in the estimation of probability (8.47) are at halfway between the variance with the splitting with optimal regions and the splitting with the naive non-optimal regions. Furthermore, we gave a theoretical framework for an interacting

particle for rare event estimation that can use non negative weights.

Then, we looked for another algorithm that can be used for the estimation of (8.47) instead of the splitting algorithm. We proposed to deal with the weighted redistribution algorithm presented in [Del Moral and Garnier, 2005]. We detailed the Markov chain to use in the weighted redistribution for the estimation of (8.47). Then, we gave an adaptive method for the choice of some good parameters to be used in the weighted redistribution algorithm. We successfully tested this adaptive method on problems arising in optical fiber and financial engineering.

Then, we implemented the adaptive splitting algorithm and the adaptive weighted redistribution algorithm for the estimation of minimum separation loss probability between aircraft and the estimation of conflict zone violation probability. For the minimum separation distance probability estimation, the adaptive weighted redistribution algorithm reduces up to forty times the time relative variance product. For the advanced conflict zones, the adaptive weighted redistribution algorithm is less efficient than the splitting algorithm. However, the adaptive splitting algorithm converges at a 100% rates, whereas the non adaptive splitting algorithm traditionally used converges no more than 14% of the runs.

Finally, we gave an estimation of some conflict zones that depends on the relative speed and the relative rate of climbing of each pair of aircraft for a given conflict probability. This is made with the adaptive splitting algorithm for extreme quantile estimation 4.2. Such a result does not exist in the scientific literature and may improve regulation rules in the air traffic management.

Bibliography

[Ahamed et al., 2006] Ahamed, T., Borkar, V., and Juneja, S. (2006). Adaptive importance sampling technique for Markov chains using stochastic approximation. *Operations Research*, 54(3):489–504.

[Albin, 1990] Albin, J. (1990). On extremal theory for stationary processes. *Annals of Probability*, 18(1):92–128.

[Albin, 1998] Albin, J. (1998). On extremal theory for self-similary processes. *Annals of Probability*, 26(2):743–793.

[Albin and Sunden, 2008] Albin, J. and Sunden, M. (2008). On the asymptotic behaviour of Lévy processes, part i: Subexponential and exponential processes. *Stochastic Processes and their Applications*, 119(1):281–304.

[Aldous, 1989] Aldous, D. (1989). *Probability approximations via the Poisson clumping heuristic*. Springer-Verlag.

[Alexopoulos and Shultes, 2001] Alexopoulos, C. and Shultes, B. (2001). Estimating reliability measures for highly-dependable Markov systems, using balanced likelihood ratios. *IEEE Transactions on Reliability*, 50(3):265–280.

[Amrein and Kunsch, 2011] Amrein, M. and Kunsch, H. (2011). A variant of importance splitting for rare event estimation: Fixed number of successes. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 21(2):13.

[Baldi, 1995] Baldi, P. (1995). Exact asymptotics for the probability of exit from a domain and applications to simulation. *The Annals of Probability*, pages 1644–1670.

[Billingsley, 1999] Billingsley, P. (1999). *Convergence of probability measures*. John Wiley & Sons Inc, 2nd edition.

[Biondini et al., 2002] Biondini, G., Kath, W., and Menyuk, C. (2002). Importance sampling for polarization-mode dispersion. *Photonics Technology Letters, IEEE*, 14(3):310–312.

[Blanchet et al., 2009] Blanchet, J., Marty, C., and Lehning, M. (2009). Extreme value statistics of snowfall in the Swiss alpine region. *Water Resources Research*, 45(5):1–12.

[Blom et al., 2003] Blom, H., Bakker, G., Everdij, M., and Van der Park, M. (2003). Collision risk modeling of air traffic. In *Proceedings of European Control Conference*.

[Blom et al., 2007a] Blom, H., Bakker, G., and Krystul, J. (2007a). Probabilistic reachability analysis for large scale stochastic hybrid systems. In *Proceedings of the 46th IEEE Conference on Decision and Control, New Orleans, USA*, pages 3182–3189.

[Blom et al., 2006] Blom, H., Krystul, J., and Bakker, G. (2006). A particle system for safety verification of free flight in air traffic. In *Proceedings of the 45th IEEE Conference on Decision and Control*.

[Blom et al., 2007b] Blom, H., Krystul, J., Bekker, G., Klompstra, M., and Klein Obbink, B. (2007b). *Free flight collision risk estimtoin by sequential MC simulation, in Stochastic Hydrid Systems*, pages 248–282. Taylor and Francis group, CRC Press.

[Cappé et al., 2005] Cappé, O., Moulines, E., and Rydén, T. (2005). *Inference in hidden Markov models*. Springer.

[Carmona and Crepey, 2010] Carmona, R. and Crepey, S. (2010). Particle methods for the estimation of credit portfolio loss distributions. *International Journal of Theoretical and Applied Finance*, 13:577–602.

[Carmona et al., 2009] Carmona, R., Fouque, J.-P., and Vestal, D. (2009). Interacting particle systems for the computation of rare credit portfolio losses. *Finance and Stochastics*, 13(4):613–633.

[Castillo et al., 2005] Castillo, E., Hadi, A., and Sarabia, J. (2005). *Extreme Value and Related Models with Applications in Engineering and Science*. John Wiley and Sons.

[Cérou et al., 2012] Cérou, F., Del Moral, P., Furon, T., and Guyader, A. (2012). Sequential Monte Carlo for rare event estimation. *Statistics and Computing*, 22(3):795–808.

[Cérou et al., 2006a] Cérou, F., Del Moral, P., Guyader, A., Le Gland, F., Lezaud, P., and Topart, H. (2006a). Some recent improvements to importance splitting. In *6th International Workshop on Rare Event Simulation*, Bamberg, Germany.

[Cerou et al., 2014] Cerou, F., Del Moral, P., Guyader, A., and Malrieu, F. (2014). Fluctuation analysis of adaptive multilevel splitting. *arXiv preprint arXiv:1408.6366*.

[Cérou et al., 2005] Cérou, F., Del Moral, P., Le Gland, F., and Lezaud, P. (2005). Limit theorems for the multilevel splitting algorithm in the simulation of rare events. In *Winter Simulation Conference*, pages 682–691, Orlando, Florida, USA.

[Cérou et al., 2006b] Cérou, F., Del Moral, P., Le Gland, F., and Lezaud, P. (2006b). Genetic genealogical models in rare event analysis. *ALEA*, 1:181–203.

[Cérou and Guyader, 2007] Cérou, F. and Guyader, A. (2007). Adaptive multilevel splitting for rare event analysis. *Stochastic Analysis and Applications*, 25(2):417–443.

[Chan and Lai, 2011] Chan, H. and Lai, T. (2011). A sequential Monte Carlo approach to computing tail probabilities in stochastic models. *The Annals of Applied Probability*, 21(6):2315–2342.

[Crisan and Doucet, 2002] Crisan, D. and Doucet, A. (2002). A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on Signal Processing*, 50(3):736–746.

[Davison and Smith, 1990] Davison, A. and Smith, R. (1990). Models for exceedances over high thresholds. *Journal of the Royal Statistical Society*, 52(3):393–442.

[De Boer et al., 2007] De Boer, P., L'Écuyer, P., Rubino, G., and Tuffin, B. (2007). Estimating the probability of a rare event over a finite time horizon. In *Winter Simulation Conference*, pages 403–411, Washington, DC.

[De Haan and Ferreira, 2006] De Haan, L. and Ferreira, A. (2006). *Extreme value theory: an introduction*. Springer.

[Dean and Dupuis, 2009] Dean, T. and Dupuis, P. (2009). Splitting for rare event simulation: a large deviation approach to design and analysis. *Stochastic Processes and their Applications*, 119(2):562–587.

[Deelstra, 1994] Deelstra, G. (1994). Remarks on boundary crossing result for Brownian motion. *Blatter der DGVFM*, 21(4):449–456.

[Del Moral, 2004] Del Moral, P. (2004). *Feynman-Kac formulae: genealogical and interacting particle systems with applications*. Series: Probability & Applications Springer Verlag.

[Del Moral and Garnier, 2005] Del Moral, P. and Garnier, J. (2005). Genealogical particle analysis of rare events. *Annals of Applied Probability*, 15(4):2496–2534.

[Del Moral et al., 2011] Del Moral, P., Hu, P., and Wu, L. (2011). On the concentration properties of interacting particle processes. *Foundations and Trends in Machine Learning*, 3(3-4):225–389.

[Del Moral and Miclo, 2001] Del Moral, P. and Miclo, L. (2001). Genealogies and increasing propagation of chaos for Feynman-Kac and genetic models. *Annals of Applied Probability*, 11(4):1166–1198.

[Demailly, 2012] Demailly, J.-P. (2012). *Analyse numérique et équations différentielles (Nouvelle édition)*. Collection Grenoble Science.

[Deng et al., 2012] Deng, S., Giesecke, K., and Lai, T. (2012). Sequential importance sampling and resampling for dynamic portfolio credit risk. *Operations Research*, 60(1):78–91.

[Embrechts et al., 2011] Embrechts, P., Kluppelberg, C., and Mikosch, T. (2011). *Modelling extremal events for insurance and finance*. Springer Verlag.

[Ethier and Kurtz, 2009] Ethier, S. and Kurtz, T. (2009). *Markov processes: characterization and convergence*. John Wiley & Sons.

[Evans and Gariepy, 1992] Evans, L. and Gariepy, R. (1992). *Measure theory and fine properties of functions*. Studies in Advanced Mathematics. CRC press, Boca Raton, FL.

[Fouque and Tullie, 2002] Fouque, J.-P. and Tullie, T. (2002). Variance reduction for Monte Carlo simulation in a stochastic volatility environment. *Quantitative Finance*, 2(1):24–30.

[Garnier and Del Moral, 2006] Garnier, J. and Del Moral, P. (2006). Simulations of rare events in fiber optics by interacting particle systems. *Optics Communications*, 267(1):205–214.

[Garvels, 2000] Garvels, M. (2000). *The splitting in rare event estimation*. PhD thesis, University of Twente, Netherlands.

[Garvels and Kroese, 1998] Garvels, M. and Kroese, D. (1998). A comparision of RESTART implementation. pages 601–608.

[Garvels et al., 2002] Garvels, M., Van Ommeren, J., and Kroese, D. (2002). On the importance function in splitting simulation. *European Transactions on Telecommunications*, 13(4):363–371.

[Giesecke et al., 2010] Giesecke, K., Kakavand, H., Mousavi, M., and Takada, H. (2010). Exact and efficient simulation of correlated defaults. *SIAM Journal on Financial Mathematics*, 1(1):868–896.

[Glasserman et al., 1998] Glasserman, P., Heidelberger, P., Shahabuddin, P., and Zajic, T. (1998). A large deviations perspective on the efficiency of multilevel splitting. *IEEE Transactions on Automatic Control*, AC-43(12):1666–1679.

[Glasserman et al., 1999] Glasserman, P., Heidelberger, P., Shahabuddin, P., and Zajic, T. (1999). Multilevel splitting for estimating rare event probabilities. *Operations Research*, 47(4):585–600.

[Glynn and Iglehart, 1989] Glynn, P. and Iglehart, D. (1989). Importance sampling for stochastic simulations. *Management Science*, 35(11):1367–1392.

[Gnedenko, 1943] Gnedenko, B. (1943). Sur la distribution limite du terme maximum d'une serie aleatoire. *Annals of Mathematics*, 44(3):423–453.

[Gobet and Menozzi, 2010] Gobet, E. and Menozzi, S. (2010). Stopped diffusion processes: boundary corrections and overshoot. *Stochastic Processes and their Applications*, 120(2):130–162.

[Goyal et al., 1987] Goyal, A., Heidelberger, P., and Shahabuddin, P. (1987). Measure specific dynamic importance sampling for availability simulations. In *Winter Simulation Conference*, pages 351–357, Atlanta, Georgia.

[Goyal et al., 1992] Goyal, A., Shahabuddin, P., Heidelberger, P., Nicola, V., and Glynn, P. (1992). A unified framework for simulating Markovian models of highly dependable systems. *Computers, IEEE Transactions on Computers*, 41(1):36–51.

[Hammersley and Handscomb, 1964] Hammersley, J. M. and Handscomb, D. C. (1964). *Monte Carlo methods*. Springer.

[Heidelberger, 1995] Heidelberger, P. (1995). Fast simulation of rare events in queueing and reliability models. *ACM Transaction on Modeling and Computer Simulation*, 5(1):43–85.

[Hu et al., 1999] Hu, J., Lygeros, J., Prandini, M., and Sastry, S. (1999). Aircraft conflict prediction and resolution using Brownian motion. In *38 Conference on Decision and Control (CDC), Phoenix, Arizona*, pages 2438–2443.

[Ichard and Baehr, 2013] Ichard, C. and Baehr, C. (2013). Inference of a random environment from random process realizations: Formalism and application to trajectory prediction. In *ISIATM 2013, 2nd International Conference on Interdisciplinary Science for Innovative Air Traffic Management*.

[Jacquemart, 2014] Jacquemart, D. (2014). Adaptive particle methods for rare event simulation in a Markovian framework. In *MascotNum Annual Conference, ETH ZÃ¼rich (Switzerland)*.

[Jacquemart and Le Gland, 2014] Jacquemart, D. and Le Gland, F. (2014). Convergence of a two step adaptive multilevel splitting algorithm. In *Proceedings of 6th International Workshop on Rare Event Simulation RESIM, Tinbergen Institute, Amsterdam*.

[Jacquemart et al., 2013] Jacquemart, D., Le Gland, F., and Morio, J. (2013). A combined important splitting important sampling for rare event estimation. In *Winter Simulation Conference*, pages 94–100, Washingtown, DC.

[Jacquemart and Morio, 2013] Jacquemart, D. and Morio, J. (2013). Conflict probability estimation between aircraft with dynamic importance splitting. *Safety Science*, 51(1):94–100.

[Johansen et al., 2006] Johansen, A., Del Moral, P., and Doucet, A. (2006). Sequential Monte Carlo samplers for rare events. In *6th International Workshop on Rare Event Simulation, Bamberg*, pages 256–267.

[Jourdain and Lelong, 2009] Jourdain, B. and Lelong, J. (2009). Robust adaptive importance sampling for normal random vectors. *The Annals of Applied Probability*, 19(5):1687–1718.

[Juneja and Shahabuddin, 2006] Juneja, S. and Shahabuddin, P. (2006). Rare-event simulation techniques : An introduction and recent advances. *Handbooks in Operations Research and Management Science*, 13(06):291–350.

[Kahn and Harris, 1951] Kahn, H. and Harris, E. (1951). Estimation of particle transmission by random sampling. *National Bureau of Standards Applied Mathematical Series*, 12:27–30.

[Karatzas and Shreve, 1991] Karatzas, I. and Shreve, E. (1991). *Brownian motion and stochastic calculus*. Springer, 2 edition.

[Khaliq and Liu, 2009] Khaliq, A. Q. and Liu, R. (2009). New numerical scheme for pricing American option with regime-switching. *International Journal of Theoretical and Applied Finance*, 12(03):319–340.

[Kloeden and Platen, 1992] Kloeden, P. and Platen, E. (1992). *Numerical solution of stochastic differential equations*. Springer.

[Kotz and Nadarajah, 2000] Kotz, S. and Nadarajah, S. (2000). *Extreme Value Distributions: Theory and Applications*. Imperial College Press.

[Kratzer et al., 2013] Kratzer, K., Arnold, A., and Allen, R. (2013). Automatic, optimized interface placement in forward flux sampling simulations. *The Journal of chemical physics*, 138(16):164–112.

[Krystul, 2006] Krystul, J. (2006). *Modelling of stochastic hybrid systems with applications to accident risk assessment*. PhD thesis, University of Twente.

[Krystul and Blom, 2005] Krystul, J. and Blom, H. (2005). Sequential Monte Carlo simulation of rare event probability in stochastic hybrid systems. In *Proceedings of the 16th IFAC World Congress, Prague, Czech Republic*, pages 2179–2185.

[Krystul and Blom, 2006] Krystul, J. and Blom, H. (2006). Sequential Monte Carlo simulation for the estimation of small reachability probabilities for stochastic hybrid systems. In *2006 Second International Symposium on Communications, Control and Signal Processing*.

[Lagnoux, 2006] Lagnoux, A. (2006). *Analyses des modéles de branchements avec duplication des trajectoires pour l'etude des evenements rares*. PhD thesis, Univ. Toulouse III.

[Lamberton, 2009] Lamberton, D. (2009). Optimal stopping and American options. *Ljubljana Summer School on Financial Mathematics*.

[Le Gland, 2007] Le Gland, F. (2007). Combined use of importance weights and resampling weights in sequential Monte Carlo methods. *ESAIM: Proceedings*, 19:85–100 (electronic).

[L'Ecuyer et al., 2006] L'Ecuyer, P., Demers, V., and Tuffin, B. (2006). Splitting for rare-event simulation. In *Proceedings of the 2006 Winter Simulation Conference, Monterey*, pages 137–148.

[L'Écuyer et al., 2007] L'Écuyer, P., Demers, V., and Tuffin, B. (2007). Rare events, splitting, and quasi–Monte Carlo. *ACM Transactions on Modeling and Computer Simulation*, 17(2, Special issue honoring Perwez Shahabuddin). Article 9.

[L'Écuyer et al., 2009] L'Écuyer, P., Le Gland, F., Lezaud, P., and Tuffin, B. (2009). Splitting methods. In Rubino, G. and Tuffin, B., editors, *Monte Carlo Methods for Rare Event Analysis*, chapter 3, pages 39–61. John Wiley & Sons.

[L'Ecuyer and Tuffin, 2007] L'Ecuyer, P. and Tuffin, B. (2007). Effective approximation of zero-variance simulation in a reliability setting. In *European Simulation and Modeling Conference, Prague*.

[Massart, 1990] Massart, P. (1990). The tight constant in the dvoretzky-kiefer-wolfowitz inequality. *The Annals of Probability*, pages 1269–1283.

[Nakayama, 1996] Nakayama, M. (1996). General conditions for bounded relative error in simulations of highly reliable Markovian systems. *Advances in Applied Probability*, 28(3):687–727.

[Neves and Fraga A., 2004] Neves, C. and Fraga A., M. I. (2004). Reiss and Thomas' automatic selection of the number of extremes. *Computational Statistics & Data Analysis*, 47(4):689–704.

[Newton, 1994] Newton, N. (1994). Variance reduction for simulated diffusions. *SIAM journal on applied mathematics*, 54(6):1780–1805.

[Nordlund and Gustafsson, 2009] Nordlund, P.-J. and Gustafsson, F. (2009). Marginalized particle filter for accurate and reliable terrain-aided navigation. *Aerospace and Electronic Systems, IEEE Transactions on*, 45(4):1385–1399.

[Paielli, 1998] Paielli, R. (1998). Empirical test of conflict probability estimation. In *Proceedings of the 2nd USA/Europe Air Traffic Management Research and Development Seminar, Orlando 1998*.

[Paielli and Erzberger, 1997] Paielli, R. and Erzberger, H. (1997). Conflict probability estimation for free flight. *Journal of Guidance Control and Dynamics*, 20(3):588–596.

[Paielli and Erzberger, 1999] Paielli, R. and Erzberger, H. (1999). Conflict probability estimation generalized to non-level flight. *Air Traffic Control Quarterly*, 7(3):195–222.

[Pickands, 1975] Pickands, J. (1975). Statistical inference using extreme order statistics. *Annals of Statistics*, 3(1):119–131.

[Prandini et al., 2011] Prandini, M., Blom, H., and Bakker, G. (2011). Air traffic complexity and the interacting particle system method: An integrated approach for collision risk estimation. In *American Control Conference (ACC), 2011*, pages 2154–2159. IEEE.

[Reiss and Thomas, 1997] Reiss, R. D. and Thomas, M. (1997). *Statistical Analysis of Extreme Values*. Birkhauser.

[Rubino and Tuffin, 2009] Rubino, G. and Tuffin, B., editors (2009). *Rare Event Simulation using Monte Carlo Methods*. John Wiley and Sons.

[Sandmann, 2005] Sandmann, W. (2005). On optimal importance sampling for discrete-time Markov chains. *Second International Conference on the Quantitative Evaluation of Systems, Torino*.

[Serle, 2014] Serle, J. (2014). Drone Warfare, More than 2,400 dead as Obama's drone campaign marks five years. *The Bureau of Investigate Journalism*.

[Van der Vaart, 1988] Van der Vaart, A. (1988). *Asymptotic Statistics*, volume 3. Cambridge university press, Cambridge.

[Vergé et al., 2013] Vergé, C., Dubarry, C., Del Moral, P., and Moulines, E. (2013). On parallel implementation of sequential monte carlo methods: the island particle model. *Statistics and Computing*, pages 1–18. to appear.

[Villén-Altamirano and Villén-Altamirano, 1994] Villén-Altamirano, M. and Villén-Altamirano, J. (1994). RESTART: a straightforward method for fast simulation of rare events. In *Winter Simulation Conference*, pages 282–289.

[Villén-Altamirano and Villén-Altamirano, 2002] Villén-Altamirano, M. and Villén-Altamirano, J. (2002). Analysis of RESTART simulation: Theoretical basis and sensitivity study. *European Transactions on Telecommunications*, 13(4):373–385.

[Wadman et al., 2013] Wadman, W., Crommelin, D., and Frank, J. (2013). Applying a splitting technique to estimate electrical grid reliability. In *Winter Simulation Conference*, pages 577 – 588, Washington, D.C.

[Watkins and Prandini, 2005] Watkins, O. and Prandini, M. (2005). Probabilistic aircraft conflict detection. *Hybridge report*.

[Zhang, 1997] Zhang, X. (1997). Numerical analysis of American option pricing in a jump-diffusion model. *Mathematics of Operations Research*, 22(3):668–690.