

ADAPTIVE PARTICLE TECHNIQUES AND RARE EVENT ESTIMATION

FRÉDÉRIC CÉROU¹ AND ARNAUD GUYADER²

Abstract. The estimation of rare event probability is a crucial issue in areas such as reliability, telecommunications, aircraft management. In complex systems, analytical study is out of question and one has to use Monte Carlo methods. When rare is really rare, which means a probability less than 10^{-9} , naive Monte Carlo becomes unreasonable. A widespread technique consists in multilevel splitting, but this method requires enough knowledge about the system to decide where to put the levels at hand. This is unfortunately not always possible. In this paper, we propose an adaptive algorithm to cope with this problem: the estimation is asymptotically consistent, costs just a little bit more than classical multilevel splitting and has the same efficiency in terms of asymptotic variance. In the one dimensional case, we prove rigorously the a.s. convergence and the asymptotic normality of our estimator, with the same variance as with other algorithms that use fixed crossing levels. In our proofs we mainly use tools from the theory of empirical processes, which seems to be quite new in the field of rare events.

Résumé. L'estimation de la probabilité d'un événement rare est un problème crucial dans des domaines tels que la fiabilité, les télécommunications, le contrôle aérien. Dans des systèmes complexes, l'étude analytique est hors de portée, et on doit utiliser une méthode de Monte Carlo. Lorsque l'événement est vraiment rare, disons ayant une probabilité plus petite que 10^{-9} , une approche Monte Carlo naïve ne marche pas. Une technique courante consiste à utiliser des niveaux de branchement, mais cette méthode nécessite une connaissance suffisante du système pour choisir où mettre les différents niveaux. Cela n'est malheureusement pas toujours possible. Dans cet article, nous proposons un nouvel algorithme adaptatif pour résoudre ce problème : l'estimateur est asymptotiquement consistant, est juste un peu plus coûteux que la méthode multi-niveaux classique, et a la même efficacité en terme de variance asymptotique. Dans le cas unidimensionnel, nous montrons rigoureusement la convergence presque sûre et la normalité asymptotique de notre estimateur, avec la même variance que les autres algorithmes utilisant des niveaux fixés. Les preuves utilisent des outils issus des processus empiriques, une approche qui semble nouvelle dans le champ des événements rares.

INTRODUCTION

Let $(X_t)_{t \geq 0}$ be a strong Markov process with values in \mathbb{R} . Suppose that $X_0 = x_0 > 0$, the origin 0 is an attractive point and $M \gg 0$ is a barrier that the process is very unlikely to reach. We would like to estimate the probability that (X_t) reaches M before coming back to 0 : this is our rare event.

For the sake of simplicity, we suppose that the trajectories are continuous. The point 0 is “attractive” means that if we define

$$T_0 = \inf\{t \geq 0 : X_t = 0\},$$

¹ IRISA / INRIA Campus de Beaulieu 35042 Rennes Cédex, France. Frederic.Cerou@inria.fr

² Université de Rennes 2, Campus de Villejan, 35043 Rennes Cedex, France. Arnaud.Guyader@uhb.fr

the stopping time T_0 is such that : $\mathbb{E}[T_0] < +\infty$.

Since (X_t) tends to decrease to 0, it is clear that if we denote

$$T_M = \inf\{t \geq 0 : X_t = M\},$$

we have

$$\mathbb{P}(T_M < T_0) \approx 0.$$

Let us consider the situation when this very small probability $\mathbb{P}(T_M < T_0) = \alpha$ is strictly positive. We want to get an estimation $\hat{\alpha}$ of α .

When the process $(X_t)_{t \geq 0}$ is not simple, the usual way to cope with this kind of problem is to use Monte Carlo techniques. The most natural one is to simulate n i.i.d. trajectories of the process, to count those who reach M before 0, and to compute the ratio. Unfortunately, when the event is really rare, this is completely unrealistic. Another idea, called Importance Sampling, is to simulate trajectories with respect to another reference probability, so that the rare event becomes less rare, and then to correct the estimation via the importance function (which corresponds to a Radon-Nikodym derivative). But it is a difficult problem to find another suitable reference probability, especially when the system in study is very complex.

In such a situation, a classical method is the one of multilevel splitting : the idea dates back to 1951 with the work of Kahn and Harris in the setting of particle transmission [11]. Roughly speaking, the principle is to multiply the trajectories that approach the event of interest and to let the other die. It is much more simple to implement than Importance Sampling but it requires some *a priori* knowledge on the system also, precisely : when and how many times shall one split the trajectories ? Hereafter we propose a new method, called adaptive multi-level splitting, which uses the splitting idea, but where the splitting levels are determined during the simulations.

1. THE ALGORITHM

The structure of the algorithm is the following : to approach the barrier M of interest, we proceed in several steps. Except the first and the last ones, all steps are equivalent. In what follows, n denotes the number of particles which are simulated, and k denotes the number of particles that we do not throw away from one step to another.

- Step 1 : simulate n i.i.d. trajectories $(X_t^j)_{t \geq 0}$ according to the law of the process $(X_t)_{t \geq 0}$ and with common initial condition

$$\forall j \in \{1, \dots, n\} \quad X_0^j = x_0 > 0.$$

Wait until all trajectories have reached 0 : for the j -th particle, this requires time T_0^j , with $\mathbb{E}[T_0^j] = \mathbb{E}_{x_0}[T_0] < +\infty$. Denote

$$S_{n,j}^1 = \sup_{0 \leq t \leq T_0^j} X_t^j,$$

and sort the sample $(S_{n,1}^1, \dots, S_{n,n}^1)$ in increasing order : $S_{n,(1)}^1 \leq \dots \leq S_{n,(n-k)}^1 \leq \dots \leq S_{n,(n)}^1$. Keep in memory the quantity (see Figure 1) $\hat{q}_1 = S_{n,(n-k)}^1$.

- Step 2 : Keep $(S_{n,(n-k+1)}^1, \dots, S_{n,(n)}^1)$ unchanged, but denote them simply $(S_{n,n-k+1}^2, \dots, S_{n,n}^2)$. Simulate $(n-k)$ trajectories $(X_t^j)_{t \geq 0}$ from initial point \hat{q}_1 . Wait until all these $(n-k)$ trajectories have reached 0 : for the j -th particle, this requires time T_0^j , with $\mathbb{E}[T_0^j] = \mathbb{E}_{\hat{q}_1}[T_0] < +\infty$. For each $j \in \{1, \dots, n-k\}$, denote

$$S_{n,j}^2 = \sup_{0 \leq t \leq T_0^j} X_t^j,$$

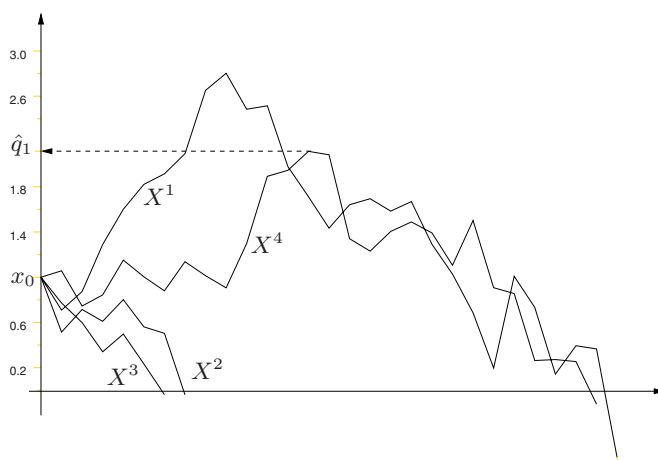


FIGURE 1. The first step of the algorithm, with $n = 4$ and $k = 1$.

and sort the sample $(S_{n,1}^2, \dots, S_{n,n-k}^2, S_{n,n-k+1}^2, \dots, S_{n,n}^2)$ in increasing order : $S_{n,(1)}^2 \leq \dots \leq S_{n,(n-k)}^2 \leq \dots \leq S_{n,(n)}^2$. Keep in memory the quantity $\hat{q}_2 = S_{n,(n-k)}^2$.

- ...
- Repeat the procedure until $\hat{q}_{\hat{N}+1} \geq M$. Among the sample $(S_{n,1}^{\hat{N}}, \dots, S_{n,n}^{\hat{N}})$, there is a proportion $\hat{r} > 0$ of them that are actually bigger than M .
- Compute the probability estimate. If we denote $p = \frac{k}{n}$, then the estimate of the rare event is simply

$$\hat{\alpha}_n = \hat{r} p^{\hat{N}}.$$

2. CONSISTENCY AND ASYMPTOTIC NORMALITY

At the end of step l , it is clear that the random variables $(S_{n,j}^l)_{1 \leq j \leq n}$ are i.i.d. according to the following law :

$$S_{n,j}^l \sim \mathcal{D}(\sup_{0 \leq t \leq T_0} X_t^{x_0} \mid \sup_{0 \leq t \leq T_0} X_t^{x_0} \geq \hat{q}_{l-1}),$$

with the convention that $\hat{q}_0 = x_0$. Thanks to the strong Markov property of the process and the continuity of its trajectories, we can write it a little bit simpler :

$$S_{n,j}^l \sim \mathcal{D}(\sup_{0 \leq t \leq T_0} X_t^{\hat{q}_{l-1}} \mid \hat{q}_{l-1}).$$

Let us also denote $S = \sup_{0 \leq t \leq T_0} X_t^{x_0}$, and F its distribution function. We suppose also that F is continuous (note that this property is not implied by the continuity of trajectories). Before stating the theorem, we sum up the assumptions on our model.

Hypothesis (\mathcal{H}). The strongly Markov process $(X_t)_{t \geq 0}$ starts from $x_0 > 0$, with 0 as an attractive point. $(X_t)_{t \geq 0}$ has time-continuous trajectories and the distribution function F of the random variable $S = \sup_{0 \leq t \leq T_0} X_t^{x_0}$ is continuous.

Theorem 1. *Under assumption (\mathcal{H}) , we have :*

$$\hat{\alpha}_n \xrightarrow[n \rightarrow \infty]{a.s.} \alpha$$

and :

$$\sqrt{n}(\alpha - \hat{\alpha}_n) \xrightarrow[n \rightarrow +\infty]{\mathcal{D}} \mathcal{N}(0, \sigma^2),$$

with :

$$\sigma^2 = \alpha^2 \left(N \frac{1-p}{p} + \frac{1-r}{r} \right).$$

We refer the reader to [3] for the proof.

3. NUMERICAL EXAMPLE

We have implemented this algorithm with trajectories following a Brownian process with drift. Noting B_t a Brownian motion, the process studied is $X_t = B_t + \mu t$, with $\mu < 0$. The drift was taken to have a motion going quickly to 0. This process clearly satisfies assumption \mathcal{H} . Moreover, it is simple enough so that analytical results are well-known about it.

We note $H_{a,b} = H = \min \{s > 0 : X_s \notin [a, b]\}$. The expression of the probability of reaching b before a starting from $x \in [a, b]$ is given by [2]:

$$\mathbb{P}_x(X_H = b) = e^{\mu(b-x)} \frac{\sinh((x-a)|\mu|)}{\sinh((b-a)|\mu|)}$$

Let us compare this to our numerical results. The first problem we had to solve was that we were considering a continuous process, which is impossible in computing. We had to choose a step δt and consider the process at every $k \times \delta t$. This step has to be small enough to avoid clipping the process, which could introduce a bias in the estimation. This error due to the time discretization is not easy to quantify, but it is not specific to our adaptive method, and one would be faced with the same difficulty using a continuous time model with any kind of level splitting algorithm, such as the ones discussed later. We will illustrate first the a.s. convergence. We ran our algorithm on the above example with parameters $a = 0$, $b = 12$, $\mu = -1$, $x_0 = 1$, such that the rare event probability is $\alpha \approx 2.412 \times 10^{-10}$. Figure 2 gives the relative error as a function of n the number of particles. For $n = 20,000$ we have an error as low as 5%. Then we illustrate the asymptotic normality. As we need to run the algorithm many times to estimate the law of the estimator, we chose a setting where α is not very small, but about 0.1244. Estimating of the same probability α 1000 times gives the histogram of Figure 3. This confirms the fact that the distribution of the estimating values tends asymptotically towards a Gaussian distribution.

4. COMPARISON WITH EXISTING ALGORITHMS

4.1. Complexity and variance

Now we want to have an idea about the complexity of the algorithm. To this aim let us consider the following simplification : we suppose that the simulation of a trajectory $(X_t^x)_{0 \leq t \leq T_0}$, between its starting point $x > 0$ and the first time T_0 it hits 0, has a cost C which is approximately constant, independent of the initial condition x .

For the first step, with the assumption above, the cost for simulating n trajectories is linear in n . The finding of the maxima of the n trajectories is also linear in n . The sorting of these maxima is, in expectation, in $O(n \log n)$. Thus the first step of the algorithm has an expected complexity in $O(n \log n)$. For n large enough, there is a finite number of steps, that is $(N+1)$, with $N = \lfloor \frac{\log \alpha}{\log p} \rfloor$. Finally the total cost of the algorithm is in $O(n \log n)$

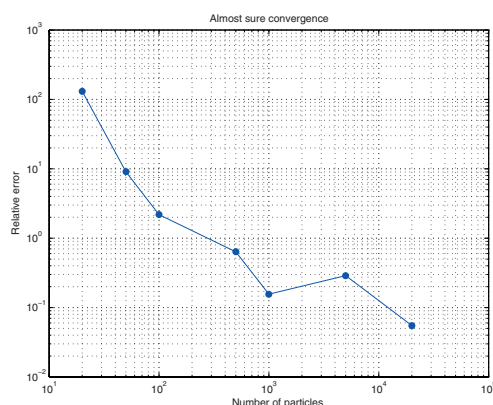


FIGURE 2. Almost sure convergence of the estimator as a function of n .

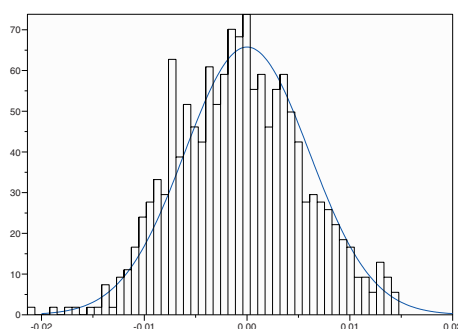


FIGURE 3. Histogram of the differences between γ and its estimations, with $n = 1000$, for 1000 instances. The curve represents the limit Gaussian distribution.

operations.

We are now interested in the precision of the estimator. We have seen above that the variance of $\hat{\alpha}$ is

$$\sigma^2 = \alpha^2 \left(N \frac{1-p}{p} + \frac{1-r}{r} \right),$$

with $p < r < 1$. If for simplicity we suppose that $\alpha = p^N$, then the normalized variance is

$$\frac{\sigma^2}{\alpha^2} = N \frac{1-p}{p} \approx \frac{\log \alpha}{\log p} \cdot \frac{1-p}{p}.$$

In real life applications, the only parameter that is fixed *a priori* is the small probability α to estimate. So the question is : what is the optimal choice for p ? A straightforward study of the variance for p varying between α and 1 proves that σ^2 decreases when p goes to one.

This result is intuitively clear and merely says that if we want a precise estimate for α , we just have to put a lot of intermediate levels. But, of course, the complexity of the algorithm is then increasing since the number of levels is $\frac{\log \alpha}{\log p}$. So the choice of the parameter p , or equivalently the choice of the number N of levels, depends on what we want : a precise estimate or a quick algorithm.

4.2. Classical multilevel splitting

A usual way for estimating rare event probability is the multilevel splitting algorithm. The splitting idea is widespread in Monte Carlo methods, see for instance [10], p.131. Its application to rare event estimation is first due to Kahn and Harris in the setting of particle transmission [11]. In 1970, Bayes proposed to apply it in the field of waiting queues [1]. This idea was rediscovered twenty years later by Villén-Altamirano and Villén-Altamirano [13] : this is what they call the RESTART method (REpetitive Simulation Trials After Reaching Thresholds) with applications in telecommunication traffic and reliability. In a simplified context, the idea has been theoretically studied by Glasserman *et al.* in several papers [7][8][9], and more recently by Lagnoux [12]. We refer the reader to [9] for a precise discussion and many references about splitting algorithms and their applications.

We can describe the algorithm in the simplified form of these last authors on our Markov process example. Let us denote A the event “reaching M before 0, starting from x_0 ”, then consider $A = A_N \subset A_{N-1} \subset \dots \subset A_1$ an increasing sequence of events. For us, this is equivalent to considering a decreasing sequence of levels $M = M_N > M_{N-1} > \dots > M_1 > x_0$, each M_i being the threshold between A_{i-1} and A_i . Let $p_1 = \mathbb{P}(A_1)$ and $p_{i+1} = \mathbb{P}(A_{i+1}|A_i)$. These probabilities are bigger than $\alpha = \mathbb{P}(A)$ and thus easier to estimate. Moreover, since the process is Markov, the following product decomposition holds :

$$\alpha = p_1 p_2 \dots p_N.$$

In the version of Lagnoux, the idea of splitting is to simulate n paths starting from x_0 , to duplicate R_1 times those who have reached M_1 before 0 (which happens with probability p_1), then to duplicate R_2 times those who have reached M_2 before 0 starting from M_1 (which happens with probability p_2), etc. (see Figure 4).

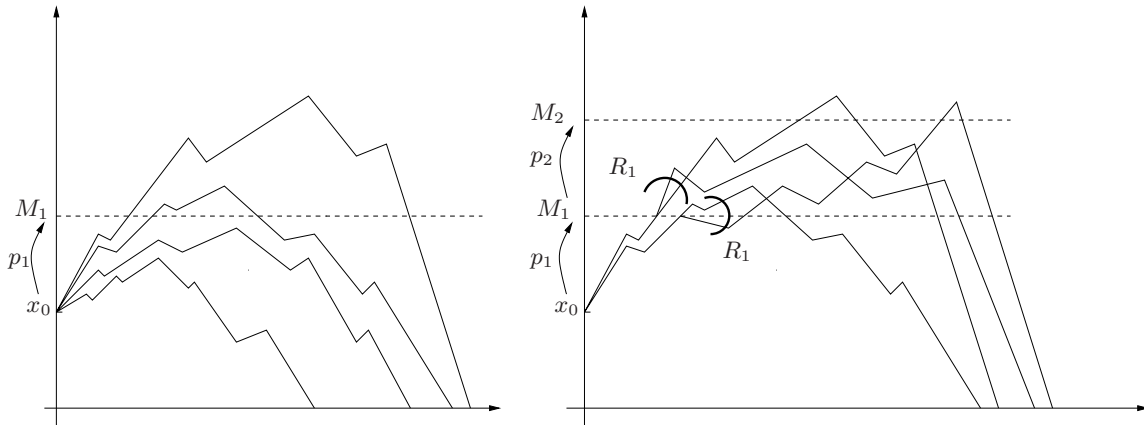


FIGURE 4. Multilevel splitting as seen by Lagnoux, with $n = 4$ and $R_1 = 2$.

An unbiased estimator of α is

$$\tilde{\alpha} = \frac{n_A}{n R_1 \dots R_N},$$

where n_A is the number of trajectories that have reached M before 0. The complexity of the algorithm is this time in $O(n)$, which is less than for the adaptive multilevel splitting algorithm.

As Glasserman *et al.* noticed, one can also see the multilevel splitting method in terms of branching processes. Suppose there is initially one ancestor, his offspring has a binomial distribution $\mathcal{B}(R_1, p_1)$, each child has himself an offspring with binomial distribution $\mathcal{B}(R_2, p_2)$, etc.. Compare to the version of Lagnoux, we see that it is the same idea, there is only a shift of one step in the splitting.

Suppose that if the number N and the levels (M_1, \dots, M_N) are fixed, we can compute the probabilities (p_1, \dots, p_N) , and *vice versa*. Then, if the sequence (R_1, \dots, R_N) is fixed also, we can compute the variance of the estimator $\tilde{\alpha}$: either through direct calculus [12], or via the theory of branching processes [9]. The natural question that arises then is : what is the best choice for these sequences, in terms of estimator variance and complexity ? Glasserman *et al.* have shown that asymptotically when $\alpha \rightarrow 0$, the best thing we have to do is to take all the p_i 's equal to $p = \alpha^{1/N}$ and all the R_i 's equal to $R = 1/p$ (if this quantity is not an integer, just randomize the algorithm so that $\mathbb{E}[R] = 1/p$). Lagnoux has shown the same result without any asymptotics on α . In terms of branching processes, this result says that the best compromise between complexity and variance is reached in the critical regime. No surprise in this story : if $R > 1/p$, the variance is smaller than simple Monte Carlo, but the number of paths will explode¹, and if $R < 1/p$, there is no substantial gain in estimation compared to naive Monte Carlo.

In relation with this last point, we could have described our algorithm in a slightly more general way, that means : with parameters $(k_1, n_1), \dots, (k_N, n_N)$ instead of the same couple (k, n) at each step. Anyway, when n and k go to infinity so that the ratio k/n goes to p , it is clear that we get closer and closer to the classical splitting algorithm with regular levels. Thus, the result obtained by preceding authors on the classical splitting algorithm shows that the best thing we have to do in our case is to keep the same couple (k, n) at each step.

For the sake of simplicity, let us suppose again that $\alpha = p^N$, with $R = \frac{1}{p}$ an integer. Then the variance s^2 of the estimator $\tilde{\alpha}$ in classical multilevel splitting is just the same² as for our $\hat{\alpha}$:

$$\frac{s^2}{\alpha^2} = N \frac{1-p}{p}.$$

The difference between the variances of $\hat{\alpha}$ and $\tilde{\alpha}$ is the following : since the adaptive version needs interactions between the n trajectories, its variance is asymptotic when n goes to infinity. This is not the case with classical splitting, where the trajectories are independent. On the other hand, the above formula is only a best case for classical multilevel splitting (regular levels), whereas it is always asymptotically granted in the adaptive version, as long as k/n is kept fixed. In other words, in terms of estimators fluctuations, classical multilevel splitting will never perform better than adaptive multilevel splitting.

Indeed, in real life applications, it is unfortunately impossible to have regular levels: the systems are usually so complex that any analytic calculus about them is just out of question. In this context, our adaptive algorithm is very useful : the levels $(\hat{q}_1, \dots, \hat{q}_N)$ are determined during the algorithm and they are in fact approximations of the true quantiles (M_1, \dots, M_N) . So, at the expense of a multiplying factor $\log n$ in the complexity, we have an algorithm that is really suitable for applications.

Concerning the issue of fixing the levels, the authors discovered *a posteriori* the thesis of Garvels [6]. In section 3.3.1., he proposes to estimate them during a trial run, in the same way as we do, and then to apply classical

¹in this case, most of the computing time is spent with highly correlated trajectories

²Note that the formula p.589 in [9] seems to be different, but this is only due to the first step of their algorithm.

splitting. He writes : “Care has to be taken that enough samples are used determining the thresholds, otherwise the thresholds may become biased and force the real simulation in the wrong area. A good rule of thumb is to devote 10% of the simulation effort to a trial run which will determine all the thresholds.” To our knowledge, he does not mention that both tasks can be treated simultaneously, with a consistency result and an “optimal” asymptotic variance.

4.3. Multilevel splitting as a Feynman-Kac model

Another variant of multilevel splitting has been recently proposed by Del Moral [4], [5]. Like in classical multilevel splitting, it requires to fix the thresholds before beginning simulations (in other words, it is not adaptive). Anyway, one of the benefits of this version is to connect multilevel splitting techniques with interacting particle systems approximations of Feynman-Kac distributions. This last topic has been intensively studied since the beginning of the 90’s, see [4], so that precise and general results can be transposed in the field of rare event analysis.

We can briefly describe the algorithm in the context of the above mentioned Markov process: suppose the number N and the levels $M_1 < \dots < M_N = M$ are fixed, like in classical splitting. At time 0, the algorithm starts with n independent copies of the process X that are stopped when they reach 0 or M_1 , whatever occurs the first. The particles which reach 0 before M_1 are killed and randomly redistributed among those having reached the first level, producing offsprings. If the whole system is killed, the algorithm is stopped. Otherwise, with the offsprings, there are still n particles at the first level M_1 . In a second step, the n particles in the first level evolve according to the same rule of the process X . Here again particles which reach 0 before M_2 are killed and for each killed one we randomly choose one of the particles that have reached the second level and add an offspring to it. Then the process goes on to the next level, and so on until the rare event is reached.

For comparison, the cost is the same as for classical splitting, in $O(n)$. The asymptotic normality of the estimator is proved, with the same variance as before, *i.e.* $N \cdot \frac{1-p}{p}$ (by adapting for instance theorem 12.2.2 in [4] in our framework).

REFERENCES

- [1] A. J. Bayes. Statistical techniques for simulation models. *Australian Comput. J.*, 2:180–184, 1970.
- [2] A. N. Borodin and P. Salminen. *Handbook of Brownian Motion – Facts and Formulae*. Birkhäuser Verlag Basel Boston Berlin, 1996.
- [3] Frédéric Cérou and Arnaud Guyader. Adaptive multilevel splitting for rare event analysis. *Stochastic Analysis and its Applications*, 25(2):417–443, 2007.
- [4] P. Del Moral. *Feynman-Kac formulae, Genealogical and interacting particle systems with applications*. Probability and its Applications. Springer-Verlag, New York, 2004.
- [5] Pierre Del Moral and Josselin Garnier. Genealogical particle analysis of rare events. *Ann. Appl. Probab.*, 15(4):2496–2534, 2005.
- [6] M.J.J. Garvels. *The splitting method in rare event simulation*. Thesis, University of Twente, Twente, May 2000. <http://www.math.utwente.nl/dos/sor/AiO's/Garvels/Thesis>.
- [7] P. Glasserman, P. Heidelberger, P. Shahabuddin, and T. Zajic. A large deviations perspective on the efficiency of multilevel splitting. *IEEE Trans. Automat. Control*, 43(12):1666–1679, 1998.
- [8] P. Glasserman, P. Heidelberger, P. Shahabuddin, and T. Zajic. A look at multilevel splitting. In *Monte Carlo and quasi-Monte Carlo methods 1996 (Salzburg)*, volume 127 of *Lecture Notes in Statist.*, pages 98–108. Springer, New York, 1998.
- [9] P. Glasserman, P. Heidelberger, P. Shahabuddin, and T. Zajic. Multilevel splitting for estimating rare event probabilities. *Oper. Res.*, 47(4):585–600, 1999.
- [10] J. M. Hammersley and D. C. Handscomb. *Monte Carlo methods*. Methuen & Co. Ltd., London, 1965.
- [11] H. Kahn and T.E. Harris. Estimation of particle transmission by random sampling. *National Bureau of Standards Appl. Math. Series*, 12:27–30, 1951.
- [12] Agnès Lagnoux. Rare event simulation. *Probab. Engrg. Inform. Sci.*, 20(1):45–66, 2006.
- [13] M. Villén-Altamirano and J. Villén-Altamirano. RESTART : a straightforward method for fast simulation of rare events. In Jeffrey D. Tew, Mani S. Manivannan, Deborah A. Sadowski, and Andrew F. Seila, editors, *Proceedings of the 1994 Winter Simulation Conference, Orlando 1994*, pages 282–289, December 1994.