

A TWO-STEP BRANCHING SPLITTING MODEL UNDER COST CONSTRAINT FOR RARE EVENT ANALYSIS

AGNÈS LAGNOUX-RENAUDIE,* *Institut de Mathématique de Toulouse*

Abstract

In this paper we consider the splitting method first introduced in rare event analysis. In this technique, the sample paths are split into R multiple copies at various stages to speed up the simulation. Given the cost, the optimization of the algorithm suggests taking all the transition probabilities to be equal; nevertheless, in practice, these quantities are unknown. We address this problem by presenting an algorithm in two phases.

Keywords: Splitting method; simulation; cost function; Laplace transform; Galton–Watson branching process; iterated function; rare event

2000 Mathematics Subject Classification: Primary 65U05

Secondary 44A10; 60J80

1. Introduction

The study of rare events is an important area in the analysis and prediction of major risks such as earthquakes, floods, air collision risks, etc. Two main approaches are used in the study of these major risks, the statistical analysis of the collected data and the modeling of the processes that led to the accident. The statistical analysis of the extreme values needs a long observation time, owing to the very low probability of the events considered. The modeling approach consists first in formalizing the system considered and then in using mathematical (see [1] and [16]) or simulation tools to obtain some estimates.

Analytical and numerical approaches are useful, but may require many simplifying assumptions. On the other hand, Monte Carlo simulation is a practical alternative when the analysis calls for fewer simplifying assumptions. Nevertheless, obtaining accurate estimates of rare event probabilities, say about 10^{-9} to 10^{-12} , using traditional techniques requires a huge amount of computing time.

Many techniques for reducing the number of trials in Monte Carlo simulations have been proposed, such as importance sampling (see, e.g. [5] and [10]) or trajectory splitting. In trajectory splitting we suppose that there exists some well-identifiable intermediate system states that are visited much more often than the target states themselves and behave as gateway states to reach the rare event. Thus, we consider a decreasing sequence of events L_i leading to the rare event L :

$$L = L_{M+1} \subset L_M \subset \cdots \subset L_1.$$

Then $P(L)$ is given by

$$P(L) = P(L | L_M) P(L_M | L_{M-1}) \cdots P(L_2 | L_1) P(L_1),$$

where, on the right-hand side, each conditioning event is ‘not rare’. For the applications we

Received 28 November 2007; revision received 28 April 2009.

* Postal address: Institut de Mathématique de Toulouse, Université Paul Sabatier, 31062 Toulouse, France.

Email address: lagnoux@cict.fr

have in mind, these conditional probabilities are in general not available explicitly. However, we do know how the particles evolve from level L_i to the next level L_{i+1} (e.g. Markovian behavior).

The principle of the algorithm is to at first simultaneously run several particles starting from level L_i ; after a while, some of these particles will have evolved ‘badly’, whereas the remaining particles will have evolved ‘well’, i.e. succeeded in reaching the threshold L_{i+1} . The ‘bad’ particles are then moved to the positions of the ‘good’ particles, and so on until L is reached. In this way, the more promising particles are favored. Unfortunately, this algorithm is hard to analyze directly because of the interaction introduced between the particles and may be difficult to apply. Examples of this class of algorithm can be found in [2] with the ‘go with the winners’ scheme, in [7] and [11] in the context of approximate counting, and in [4], [6], and [8] in a more general setting.

All the abovementioned algorithms nevertheless have a common base which is simpler to analyse and is called the branching splitting model. In the branching splitting technique we make a $\{0, 1\}$ Bernoulli trial to check whether or not the set event L_1 has occurred. In the case when L_1 has occurred we split the trial into R_1 Bernoulli subtrials, and, for each subtrial, we check whether or not event L_2 has occurred. This procedure is repeated at each level, until L is reached. If an event level is not reached, neither is L , and we stop the current retrial. Using N independent replications of this procedure, we have then considered $NR_1 \cdots R_M$ trials, taking into account, for example, the fact that if we fail to reach level L_i at the i th step, the $R_i \cdots R_M$ possible retrials also fail. Clearly, the particles reproduce and evolve independently.

An unbiased estimator of $P(L)$ is given by the quantity

$$\hat{P} = \frac{N_L}{N \prod_{i=1}^M R_i},$$

where N_L is the total number of trajectories having reached the set L . As this algorithm can be represented by N independent Galton–Watson branching processes $(Z_n)_n$, as in [12], the variance of \hat{P} can be derived, and depends on the probability transitions and on the mean numbers (m_i) of the particle successes at each level. Following the heuristics presented in [17] and [18], an optimal algorithm is derived by minimizing the variance of the estimator for a given budget (computational cost), defined as the expected number of trials generated during the simulation, where each trial is weighted by a cost function.

The optimization of the algorithm [12] suggests taking all the transition probabilities to be equal to a constant denoted by P_0 and taking the splitting numbers equal to the inverse of this constant. We then deduce the number of thresholds, M , and, finally, N is given by the cost. This result is not surprising since it means that the branching processes are critical Galton–Watson processes. In other words, optimal values are chosen in such a way as to balance the loss of variance from too little splitting and the exponential growth in the computational effort from too much splitting.

Some practical problems arise when applying the optimal algorithm to models issued from concrete problems. First, the optimal splitting number can be a noninteger. In [13], the author proposed three algorithms to address this problem. For the applications we have in mind, the thresholds L_i are fixed but the conditional probabilities are unknown (however, we know how the particles evolve from level L_i to the next level L_{i+1}). Moreover, we assume here that the conditional probabilities lie in some compact interval $[a, b] \subset (0, 1)$. This hypothesis is essential, as otherwise nothing can be done algorithmically. In practice, this hypothesis is implicitly assumed but not explicitly stated.

We propose here an algorithm in two phases, based on the branching splitting model. The first phase is a learning phase in which we sample ρ_N particles. The algorithm proceeds as in the classical branching splitting method with splitting numbers $(R_i^0)_{i=1,\dots,M}$ chosen arbitrarily at the start. In the second phase we run $N - \rho_N$ particles which we evolve as in the first phase, but with the splitting number estimators of the optimal splitting numbers $(R_i)_{i=1,\dots,M}$; the estimators are obtained during the first learning phase and follow the optimal rule given in [12]. Owing to the complexity of the formulae, when the cost C goes to ∞ , we will simply carry out an asymptotic study. Assuming that the transition probabilities lie in a compact interval implies that the particle cost is bounded from below and from above, which means that we can carry out the study when N goes to ∞ . A precise analysis shows that we will dedicate asymptotically $\mu_s C^{2/3}$ particles to the learning phase and $C/C_{\text{opt}} - \mu_s C^{2/3}$ particles to the second phase, where C_{opt} is a constant defined in Section 4.1 and μ_s is derived from the optimization of the algorithm, i.e. assuming that the number of particles generated during the learning phase behaves as $\mu_\alpha(C)C^{1-\alpha}$, where we will take $\alpha = \frac{1}{3}$. Moreover, we note that N is linear in C , so dedicating $\mu_s C^{2/3}$ particles to the first phase amounts to dedicating $\lambda_s N^{2/3}$ particles for some λ_s depending on μ_s .

The paper is organized as follows. In Section 2 we briefly recall the general settings and some previous results on the branching splitting model. In Section 3 we present some useful analytical results. In Section 4 we study the two-step branching splitting model and derive optimal parameters. In Section 5 we implement the two-step algorithm on an approximate Ornstein–Uhlenbeck process. Finally, in Section 6 we conclude and discuss the merits of this approach and potential directions for further research.

2. Previous results and general settings

2.1. Optimal branching splitting model

As mentioned in the introduction and following [12], we consider N independent Galton–Watson branching processes $(Z_n^{(i)})_{n \geq 0}$, $i = 1, \dots, N$, where, for each i , $Z_n^{(i)}$ is the number of particles derived from the i th particle ($Z_0^{(i)}=1$) that reached level L_n . Then, letting R_i be the sampling number at level i ,

$$\hat{P} := \frac{1}{N} \sum_{i=1}^N \tilde{Q}_i, \quad \text{where} \quad \tilde{Q}_i = \frac{Z_{M+1}^{(i)}}{R_1 \cdots R_M}.$$

To simplify the notation, we will consider only the case in which $N = 1$ in the following, i.e. we will consider the process $(Z_n)_{n \geq 0}$ with $Z_0 = 1$. We have the following relation:

$$Z_{n+1} = \sum_{j=1}^{Z_n} X_n^{(j)},$$

where, for each n , the random variables $(X_n^{(j)})_{j \geq 1}$ are independent and identically distributed (i.i.d.) with common law satisfying a binomial distribution with parameters (R_n, P_{n+1}) for $n \geq 1$ and a Bernoulli distribution with parameter P_1 for $n = 0$. For more details on Galton–Watson and branching processes, we refer the reader to [3], [9], and [15].

Let us introduce the following quantities:

$$\begin{aligned} r_0 &= 1, & r_i &= R_1 \cdots R_i, & i &= 1, \dots, M, \\ p_0 &= 1, & p_i &= P_1 \cdots P_i, & i &= 1, \dots, M + 1, \\ m_0 &= P_1, & m_i &= P_{i+1} R_i, & i &= 1, \dots, M, \end{aligned}$$

where r_i represents the weight of a particle having reached L_i , p_i is the probability of the particle reaching L_i starting from 0, and m_i is the mean number of offspring of an individual in the i th generation. Integrating N , the variance of \hat{P} is given by (see [12] for details)

$$\text{var}(\hat{P}) = \frac{P(L)^2}{N} \sum_{i=0}^M \left(\frac{1}{P_{i+1}} - \frac{1}{P_i} \right) \frac{1}{r_i}.$$

In [12], the three-step minimization of the variance of $P(L)$ for a given budget C , defined as the mean number of particles generated during the simulation, i.e.

$$C = N \sum_{i=0}^M r_i p_i,$$

leads to the optimal parameters of the algorithm: first we derive the optimal N and the optimal splitting numbers, R_i , then the optimal transition probabilities, P_i , and finally the optimal number of thresholds, M . In the case where the transition probabilities are fixed, the first step of the optimization leads to

$$\begin{aligned} R_i &= \sqrt{\frac{1}{P_i P_{i+1}}} \sqrt{\frac{1 - P_{i+1}}{1 - P_i}} \quad \text{for } i = 1, \dots, M, \\ N &= \frac{C \sqrt{1/P_1 - 1}}{\sum_{i=1}^{M+1} \sqrt{1/P_i - 1}}. \end{aligned}$$

Using these relations, the variance has the following expression:

$$\text{var}(\hat{P}) = \frac{P(L)^2}{N} \left(\frac{1}{P_1} - 1 \right) \sum_{i=0}^M r_i p_i = \frac{P(L)^2}{N^2} \left(\frac{1}{P_1} - 1 \right) C. \tag{2.1}$$

2.2. Settings

2.2.1. *The two-step branching splitting model.* In this paper, the transition probabilities are considered to be unknown but to belong to some interval $[a, b] \subset (0, 1)$. In practice, we can propose values for a and b using specific knowledge of the process or a previous learning simulation. Arbitrarily choose $M + 1$ numbers P_i^0 in $[a, b]$. Let

$$R_i^0 = \frac{1}{\sqrt{P_i^0 P_{i+1}^0}} \sqrt{\frac{1 - P_{i+1}^0}{1 - P_i^0}} \quad \text{for } i = 1, \dots, M,$$

and introduce the following notation:

$$r_i^0 = 1, \quad r_i^0 = R_1^0 \cdots R_i^0, \quad i = 1, \dots, M.$$

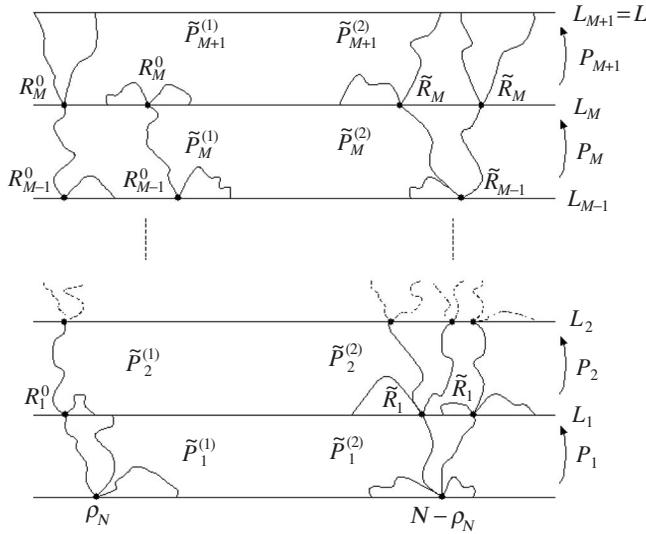


FIGURE 1: The two-step branching splitting model.

Remark 2.1. The analytical complexity of the calculation prevents us from making a precise analysis, and, when the cost C goes to ∞ , we are only able to derive asymptotic results. But the fact that the P_i s lie in the compact interval $[a, b] \subset (0, 1)$ implies that the ratio C/N is bounded from above and from below, allowing us to carry out an asymptotic study when N goes to ∞ . Note also that the total number of particles is in reality a random variable: in practice, a given budget is fixed at the start and we sample particles until the budget is consumed; hence, the randomness of N .

We propose here an algorithm in two phases: the first phase is a learning phase in which the splitting numbers are $(R_i^0)_{i=1, \dots, M}$ while the second phase uses estimators of $(R_i)_{i=1, \dots, M}$ obtained in the learning phase. More precisely, we have N particles that we split into two groups of ρ_N and $N - \rho_N$ particles. The two-step branching splitting model then proceeds as follows (see also Figure 1).

Phase 1: sample ρ_N particles. The particles having reached L_1 are split into R_1^0 subtrials that we evolve from L_1 . The particles having reached L_2 are split into R_2^0 subtrials that we evolve from L_2 . This procedure is repeated until L is reached. From step i ($i = 1, \dots, M+1$) we obtain an estimator $\hat{P}_i^{(1)}$ of P_i (the fraction of successful particles starting from L_{i-1}). To exploit all the information we possess, we improve the algorithm by replacing $\hat{P}_i^{(1)}$ with $\tilde{P}_i^{(1)} := a \vee \hat{P}_i^{(1)} \wedge b$ during the simulation, since $P_i \in [a, b]$. This substitution is advantageous because it prevents us from prematurely stopping the algorithm. Without this ability, the possibility of the particle dying remains (see Section 3). After a premature stop of the algorithm, we can start a new simulation. Nevertheless, we must take into account the cost induced by this first aborted phase, as we work for a given fixed effort. Now, for all $i = 1, \dots, M$, let

$$\tilde{R}_i = \frac{1}{\sqrt{\tilde{P}_i^{(1)} \tilde{P}_{i+1}^{(1)}}} \sqrt{\frac{1 - \tilde{P}_{i+1}^{(1)}}{1 - \tilde{P}_i^{(1)}}}$$

represent the splitting numbers of the second phase.

Phase 2: sample $N - \rho_N$ particles. The particles having reached L_1 are split into \tilde{R}_1 subtrials that we evolve from L_1 . The particles having reached L_2 are split into \tilde{R}_2 subtrials that we evolve from L_2 . This procedure is repeated until L is reached. From step i ($i = 1, \dots, M + 1$) we obtain an estimator $\hat{P}_i^{(2)}$ of P_i , which we change into $\tilde{P}_i^{(2)} := a \vee \hat{P}_i^{(2)} \wedge b$ during the simulation.

In the following, the truncated estimators will be denoted with a tilde and all others estimators will be denoted with a hat.

Remarks. 1. A first analytical survey leads us to restrict our attention to the case when

$$\rho_N \rightarrow +\infty \text{ as } N \rightarrow +\infty, \quad N - \rho_N \rightarrow +\infty \text{ as } N \rightarrow +\infty, \quad \text{and} \quad \rho_N = o(N - \rho_N).$$

2. Moreover, as mentioned in Remark 2.1, the calculation is too complex to make a precise analysis, so we carry out an asymptotic study when the cost C (and so when N) goes to ∞ . We aim at optimizing the algorithm with respect to the parameters. After analysis, we prove that the optimum is attained when

$$\left(\frac{\rho_N}{N}\right)^2 \approx \frac{1}{\rho_N} \text{ as } N \rightarrow \infty.$$

More precisely, to have a better understanding of what happens and to clarify the analysis, we restrict our attention to the case where ρ_N is asymptotically of the form $\lambda N^{1-\alpha}$ with λ depending on N , and we aim at optimizing the algorithm in terms of α , N , and λ . Then we prove that the optimum is attained for $\alpha = \frac{1}{3}$; indeed, for $\alpha < \frac{1}{3}$, $\lambda \rightarrow 0$ as $N \rightarrow \infty$, for $\alpha = \frac{1}{3}$, $\lambda \rightarrow \text{constant}$ as $N \rightarrow \infty$, and, for $\alpha = \frac{1}{3}$, $\lambda \rightarrow \infty$ as $N \rightarrow \infty$.

Let

$$\mathcal{F}_{2i} = \sigma(\tilde{P}_1^{(1)}, \dots, \tilde{P}_i^{(1)}, \tilde{P}_1^{(2)}, \dots, \tilde{P}_i^{(2)}) \quad \text{and} \quad \mathcal{F}_{2i+1} = \mathcal{F}_{2i} \vee \sigma(\tilde{P}_{i+1}^{(1)}),$$

and note that in the following we use the shorthand notation E_k for $E(\cdot \mid \mathcal{F}_k)$ and var_k for $\text{var}(\cdot \mid \mathcal{F}_k)$.

2.2.2. *Estimators of the P_i s.* To exploit all the information given during the simulation, we use both algorithms to estimate the transition probabilities: P_{i+1} is estimated by the fraction of successful particles in each generation in phase 1 and in phase 2. More precisely, for $i = 0, \dots, M$, P_{i+1} is estimated by

$$\begin{aligned} \tilde{P}_{i+1} &= \frac{\text{total number of successes in } L_{i+1}}{\text{total number of particles generated from } L_i} \\ &= \frac{(\lambda/N^\alpha)r_i^0 \tilde{p}_{i+1}^{(1)} + (1 - \lambda/N^\alpha)\tilde{r}_i \tilde{p}_{i+1}^{(2)}}{(\lambda/N^\alpha)r_i^0 \tilde{p}_i^{(1)} + (1 - \lambda/N^\alpha)\tilde{r}_i \tilde{p}_i^{(2)}} \end{aligned}$$

with

$$\begin{aligned} \tilde{p}_0^{(1)} &= 1, & \tilde{p}_i^{(1)} &= \tilde{P}_1^{(1)} \dots \tilde{P}_i^{(1)} & \text{for } i = 1, \dots, M + 1, \\ \tilde{p}_0^{(2)} &= 1, & \tilde{p}_i^{(2)} &= \tilde{P}_1^{(2)} \dots \tilde{P}_i^{(2)} & \text{for } i = 1, \dots, M + 1, \\ \tilde{r}_0 &= 1, & \tilde{r}_i &= \tilde{R}_1 \dots \tilde{R}_i & \text{for } i = 1, \dots, M. \end{aligned}$$

It corresponds to the minimal variance estimator among the estimators given by a linear combination of $\tilde{P}_i^{(1)}$ and $\tilde{P}_i^{(2)}$.

Finally, $P(L)$ is estimated by the product of these fractions:

$$\tilde{P} = \tilde{P}_1 \cdots \tilde{P}_{M+1} =: \tilde{p}_{M+1}.$$

Noting that

$$\tilde{P}_{i+1} = \frac{\tilde{p}_{i+1}^{(2)}}{\tilde{p}_i^{(2)}} \left(1 + \frac{\lambda}{N^\alpha} \left(\frac{\tilde{p}_{i+1}^{(1)} r_i^0}{\tilde{p}_{i+1}^{(2)} \tilde{r}_i} - 1 \right) \right) / \left(1 + \frac{\lambda}{N^\alpha} \left(\frac{\tilde{p}_i^{(1)} r_i^0}{\tilde{p}_i^{(2)} \tilde{r}_i} - 1 \right) \right),$$

we have the following result.

Proposition 2.1. $P(L)$ is estimated by

$$\tilde{P} = \tilde{p}_{M+1}^{(2)} \prod_{i=1}^{M+1} \left(1 + \frac{\lambda}{N^\alpha} \left(\frac{\tilde{p}_i^{(1)} r_{i-1}^0}{\tilde{p}_i^{(2)} \tilde{r}_{i-1}} - 1 \right) \right) / \prod_{i=1}^M \left(1 + \frac{\lambda}{N^\alpha} \left(\frac{\tilde{p}_i^{(1)} r_i^0}{\tilde{p}_i^{(2)} \tilde{r}_i} - 1 \right) \right). \tag{2.2}$$

3. Mathematical tools

In this section we present the technical tools necessary for the rest of the analysis: we give two results concerning the truncated estimators used in the algorithm. First a precise bound of the truncation probabilities of a sum of i.i.d. Bernoulli random variables is presented; then we state a result on the expectation of truncated estimators.

Lemma 3.1. Let \hat{P} be a random variable having the following expression:

$$\hat{P} = \frac{1}{N} \sum_{i=1}^N \text{Ber}_i,$$

where the Ber_i are i.i.d. Bernoulli random variables with parameter P .

1. Letting

$$l(x) = \log \left[\left(\frac{1-x}{1-P} \right)^{1-x} \left(\frac{x}{P} \right)^x \right],$$

we have

$$P(\hat{P} \leq a) \leq \exp\{-Nl(a)\}, \quad P(\hat{P} \geq b) \leq \exp\{-Nl(b)\}.$$

2. Let f and g be two functions defined on $(0, 1)$, let \mathcal{C}^2 be defined on $[a, b]$, and let $\tilde{P} = a \vee \hat{P} \wedge b$. Then

$$E(f(\tilde{P})) = f(P) + \frac{P(1-P)}{2N} f''(P) + o\left(\frac{1}{N}\right).$$

As a consequence,

$$\begin{aligned} \text{var}(f(\tilde{P})) &= \frac{P(1-P)}{N} f'(P)^2 + o\left(\frac{1}{N}\right), \\ \text{cov}(f(\tilde{P}), g(\tilde{P})) &= \frac{P(1-P)}{N} f'(P)g'(P) + o\left(\frac{1}{N}\right). \end{aligned}$$

Proof. See Appendix A.

We can easily generalize the results of Lemma 3.1 to functions of two variables by duplication.

4. Asymptotic optimal algorithm

4.1. Asymptotic expressions for the cost and the variance

In this section we derive the optimal algorithm by minimizing the variance of the estimator for a given simulation cost. As mentioned in Remark 2.1, we will carry out an asymptotic survey when N goes to ∞ and derive an asymptotic optimal algorithm by minimizing the variance for a given budget. First we give asymptotic expressions for the cost and the variance; then we deduce an asymptotic optimal algorithm and we end the section by explaining how to proceed practically.

We first need to derive asymptotic expressions of the (average) simulation cost and the variance of the estimator. An accurate analysis of the optimization problem shows that we must carry out a second-order asymptotic expansion for the learning phase (i.e. the second order in ρ_N) while we simply need the first order for the second phase (i.e. the first order in $N - \rho_N$ or, equivalently, in N). The proofs of the two following theorems are technical and complex, and are thus postponed to Appendices B and C.

Assuming that, asymptotically, ρ_N is of the form $\lambda N^{1-\alpha}$, the (average) cost (which is defined as the mean number of particles generated during the simulation) is given by

$$C = \lambda N^{1-\alpha} \sum_{i=0}^M R_0^i E(\tilde{p}_i^{(1)}) + (N - \lambda N^{1-\alpha}) \sum_{i=0}^M E(\tilde{r}_i \tilde{p}_i^{(2)}). \tag{4.1}$$

Remember that our aim is to derive an expression for the cost as the given budget goes to ∞ . We remark that

$$\frac{C}{N} = \frac{\lambda}{N^\alpha} \sum_{i=0}^M R_0^i E(\tilde{p}_i^{(1)}) + \left(1 - \frac{\lambda}{N^\alpha}\right) \sum_{i=0}^M E(\tilde{r}_i \tilde{p}_i^{(2)}),$$

and since $P_i \in [a, b] \subset (0, 1)$ for all $i = 1, \dots, M + 1$, the right-hand side is bounded above and below. As a consequence, we can verify that, as mentioned in the introduction,

$$N \rightarrow \infty \quad \text{while} \quad C \rightarrow \infty$$

and that C is linear in N . Now repeatedly using Lemma 3.1, we derive the following result.

Theorem 4.1. *We have*

$$C = N \left(C_{\text{opt}} + \frac{B_1}{\lambda N^{1-\alpha}} + B_2 \frac{\lambda}{N^\alpha} + o\left(\frac{1}{N^\alpha}\right) + o\left(\frac{1}{N^{1-\alpha}}\right) \right),$$

where

$$B_1 = \frac{1}{2} \sum_{i=1}^M r_i p_i \left(\frac{3/4}{P_1(1 - P_1)} - 2 + 2 \sum_{k=1}^{i-1} \frac{1 - P_{k+1}}{r_k^0 p_{k+1}} + \frac{3/4 - P_{i+1}}{r_i^0 p_{i+1}(1 - P_{i+1})} \right),$$

$$B_2 = \sum_{i=1}^M p_i (r_i^0 - r_i),$$

and C_{opt} is the (average) particle cost generated in the optimal model,

$$C_{\text{opt}} = \sum_{i=0}^M r_i p_i.$$

Proof. See Appendix B.

Using the notation introduced previously, the variance of the estimator is given as follows.

Theorem 4.2. *We have*

$$\text{var}(\tilde{P}) = \frac{P(L)^2}{N} \left(V_{\text{opt}} + \frac{1}{\lambda N^{1-\alpha}} A_1 + \frac{\lambda}{N^\alpha} A_2 + \frac{\lambda^2}{N^{2\alpha}} A_3 + o\left(\frac{1}{N^{1-\alpha}}\right) + o\left(\frac{1}{N^{2\alpha}}\right) \right),$$

where

$$A_1 = \frac{1}{2} \sum_{i=1}^M \left(\frac{P_{i+1} - 1/4}{r_i^0 r_i p_{i+1}^2} - \frac{1}{4} \frac{r_i p_i}{P_1^2} \right), \quad A_2 = \left(\frac{1}{P_1} - 1 \right) \sum_{i=1}^M p_i (r_i - r_i^0),$$

$$A_3 = \left(\frac{1}{P_1} - 1 \right) \sum_{i=1}^M \frac{p_i}{r_i} (r_i^0 - r_i)^2,$$

and V_{opt} is the renormalized particle variance of the optimal model,

$$V_{\text{opt}} = \sum_{i=0}^M \left(\frac{1}{p_{i+1}} - \frac{1}{p_i} \right) \frac{1}{r_i} = \left(\frac{1}{P_1} - 1 \right) C_{\text{opt}},$$

arising from (2.1).

Proof. See Appendix C.

4.2. Optimization of the algorithm

We remind the reader that our goal is to optimize the algorithm with respect to the parameters N , α , and λ . We first prove the following lemma.

Lemma 4.1. *Let F and G be two real-valued functions. Suppose that*

$$F(\lambda, N) \sim \frac{1}{N} + \frac{f_1(\lambda)}{N^{1+\alpha}} + \frac{f_2(\lambda)}{N^{1+2\alpha}} + \frac{f_3(\lambda)}{N^{2-\alpha}} \quad \text{as } N \rightarrow \infty,$$

$$G(\lambda, N) \sim N - N^{1-\alpha} f_1(\lambda) + N^\alpha f_4(\lambda) \quad \text{as } N \rightarrow \infty.$$

Then minimizing F for a fixed $G = G_f$ asymptotically amounts to minimizing

- $f_3 + f_4$ for $\alpha > \frac{1}{3}$,
- $f_2 - \alpha(f_1)^2$ for $\alpha < \frac{1}{3}$,
- $f_2 + f_3 + f_4 - (f_1)^2$ for $\alpha = \frac{1}{3}$,

with N being given by the equation $G = G_f$.

Proof. See Appendix D.

As a consequence of Lemma 4.1, we obtain the following proposition.

Proposition 4.1. *Minimizing the variance for a given cost leads to taking $\alpha = \frac{1}{3}$,*

$$\lambda_s = \left(\frac{C_{\text{opt}}(A_1 + B_1(1/P_1 - 1))}{2(A_2 B_2 + A_3 C_{\text{opt}})} \right)^{1/3},$$

and N is given by the cost.

Proof. From (2.1) and the expressions for A_2 and B_2 , the following relation is clearly satisfied:

$$A_2 C_{\text{opt}} + B_2 V_{\text{opt}} = 0.$$

As a consequence, the variance and the cost (after a renormalization by V_{opt} and C_{opt} , respectively) have the form of the functions F and G in Lemma 4.1, which allows us to state that minimizing the variance for a given cost C amounts to minimizing

- $(1/\lambda)(A_1/V_{\text{opt}} + B_1/\lambda C_{\text{opt}})$ for $\alpha > \frac{1}{3}$, which leads to $\lambda_s = +\infty$,
- $\lambda^2(A_3/V_{\text{opt}} + \alpha A_2 B_2/V_{\text{opt}} C_{\text{opt}})$ for $\alpha < \frac{1}{3}$, which leads to $\lambda_s = 0$,

•

$$\frac{A_1/\lambda + A_3\lambda^2}{V_{\text{opt}}} + \frac{B_1}{\lambda C_{\text{opt}}} + \lambda^2 \frac{A_2 B_2}{V_{\text{opt}} C_{\text{opt}}} \quad \text{for } \alpha = \frac{1}{3},$$

which leads to

$$\lambda_s = \left(\frac{C_{\text{opt}}(A_1 + B_1(1/P_1 - 1))}{2(A_2 B_2 + A_3 C_{\text{opt}})} \right)^{1/3}.$$

As a conclusion, the optimal parameters are $\alpha = \frac{1}{3}$, N is simply given by the cost, and

$$\lambda_s = \left(\frac{C_{\text{opt}}(A_1 + B_1(1/P_1 - 1))}{2(A_2 B_2 + A_3 C_{\text{opt}})} \right)^{1/3}.$$

Remark 4.1. We insist on the fact that the equation $A_2 C_{\text{opt}} + B_2 V_{\text{opt}} = 0$, which allows us to apply Lemma 4.1, follows directly from the particular choice of the R_i s, which are solutions of a particular optimization problem, and this leads to $\alpha = \frac{1}{3}$. Otherwise, we would not have had this relation between the constants A_2 and B_2 , and the optimal α would simply have been $\frac{1}{2}$, as is expected a priori. This emphasizes the fact that it is worth choosing the transition probabilities to be as close as possible to the optimal P_i s.

Remark 4.2. If the optimal R_i for the learning phase is chosen arbitrarily, we are led to the solution

$$\lambda_s = +\infty,$$

which confirms our intuition: if the optimal splitting numbers are used from the start then introducing an extra Monte Carlo stage cannot be advantageous.

4.3. Guidelines to proceed practically

In practice, we are given a fixed budget to consume during the simulation which amounts to fixing the total number of particles, N , generated during the simulation. We will sample $\rho_N = \lambda_s N^{1-\alpha}$ particles during the learning phase and $N - \rho_N$ particles during the second phase. The optimization analysis suggests that we take

$$\alpha = \frac{1}{3} \quad \text{and} \quad \lambda_s = \left(\frac{C_{\text{opt}}(A_1 + B_1(1/P_1 - 1))}{2(A_2 B_2 + A_3 C_{\text{opt}})} \right)^{1/3}.$$

Recall that the transition probabilities P_i are unknown and, thus, we cannot evaluate λ_s explicitly. Nevertheless, these conditional probabilities are bounded above and below; thus, we can determine a lower bound $\lambda_s(\text{min})$ (but also an upper bound $\lambda_s(\text{max})$) of λ_s . As a consequence, we will proceed as follows.

- *First phase: the learning phase.* We first sample $\lambda_s(\min)N^{2/3}$ particles, proceeding as in the classical branching splitting method with splitting numbers $(R_i^0)_{i=1,\dots,M}$ that have been chosen arbitrarily at the start. Then we estimate the transition probabilities, the optimal splitting numbers, and the optimal λ_s (denoted by λ_{est}). Finally, we continue to sample particles until the total number of trials generated during the first phase equals $\lambda_{\text{est}}N^{2/3}$.
- *Second phase.* We run $N - \rho_N$ particles that we evolve as in the first phase, but here we use estimators of $(R_i)_{i=1,\dots,M}$ obtained during the first learning phase and follow the optimal rule given in [12], as explained previously.

5. Approximate Ornstein–Uhlenbeck process

In this section we study the *approximate Ornstein–Uhlenbeck process* X , governed by the following stochastic differential equation:

$$dX_t = -\mu(X_t)X_t dt + \sigma(X_t) dW_t,$$

where W is a Brownian motion and the unknown drift is such that

$$0 < \mu^1 \leq \mu(x) \leq \mu^2 < \infty.$$

In the numerical application we will take the drift constant on disjoint intervals. Remember that the general Ornstein–Uhlenbeck process is governed by the following stochastic differential equation:

$$dX_t = -\mu X_t dt + \sigma dW_t. \tag{5.1}$$

The Ornstein–Uhlenbeck process is recurrent and excursions above large values are rare; this is also the case for the approximate Ornstein–Uhlenbeck process. Here, we want to estimate the probabilities that the approximate Ornstein–Uhlenbeck process reaches some high levels starting from $x > 0$ and before returning to 0. When the drift is known, we can easily determine these probabilities. But with an unknown drift, their estimation is very expensive in terms of the simulation. Here, since we only want to estimate the levels of excursions, there is no need to estimate μ , and the two-step branching splitting algorithm appears to be a promising and efficient way to solve the problem.

Before turning to the numerical application, we first recall the general settings and results on the general Ornstein–Uhlenbeck process.

5.1. Analytical study

We recall in this subsection some results on the standard Ornstein–Uhlenbeck process, governed by the stochastic differential equation

$$dX_t = -X_t dt + \sqrt{2} dW_t$$

(i.e. the drift μ is equal to 1 and the variance σ^2 is equal to 2). The results for the general Ornstein–Uhlenbeck process governed by (5.1) can be easily deduced by a change of variables.

The speed measure m is given by

$$m(dx) = \rho(x) dx = \exp\left\{-\frac{\mu x^2}{\sigma^2}\right\} dx,$$

and the infinitesimal generator \mathcal{L} is given by

$$\mathcal{L}(f)(x) = f''(x) - \mu x f'(x) = f''(x) + \frac{\sigma^2}{2} \frac{\rho'(x)}{\rho(x)} f'(x).$$

Let L_0, x , and L be such that $L_0 < x < L$. Suppose that $X_0 = x$, and note that $H(x) = \int_{L_0}^x dy/\rho(y)$. Let $T_{L_0,L}$ be the first leaving time of $[L_0, L]$:

$$T_{L_0,L} = \inf\{t \geq 0: X_t \notin [L_0, L]\}.$$

We can easily verify that $T_{L_0,L}$ is a stopping time. Noting that H solves $\mathcal{L}(H) = 0$, we have

$$\begin{aligned} P_x(X_{T_{L_0,L}} = L) &= \frac{H(x)}{H(L)}, \\ P_x(X_{T_{L_0,L}} = L_0) &= \frac{H(L) - H(x)}{H(L)}, \\ E_x(T_{L_0,L}) = M_{L_0,L}(x) &= - \int_{L_0}^x \frac{R(t)}{\rho(t)} dt + \frac{H(x)}{H(L)} \int_{L_0}^L \frac{R(t)}{\rho(t)} dt. \end{aligned}$$

5.1.1. *How to position the thresholds to have constant transitions.* The optimization of the algorithm suggests taking the transition probabilities to be equal, i.e. if the thresholds are denoted by L_i , we want to solve in terms of L_{i+1} once L_i has been fixed,

$$P_{L_i}(X_{T_{L_0,L_{i+1}}} = L_{i+1}) = \frac{H(L_i)}{H(L_{i+1})} = \theta \quad \text{for some } \theta.$$

Then, for large values of i , we aim at having

$$\begin{aligned} P_{L_i}(X_{T_{L_0,L_{i+1}}} = L_{i+1}) &= \theta, \\ \iff H(L_i) &= \theta H(L_{i+1}) \sim \theta[H(L_i) + (L_{i+1} - L_i)H'(L_i)], \\ \iff L_{i+1} - L_i &\sim \exp\left\{-\frac{L_i^2}{2}\right\} H(L_i) \left(\frac{1}{\theta} - 1\right), \\ \iff L_{i+1} - L_i &\sim \frac{1}{L_i} \left(\frac{1}{\theta} - 1\right) \quad \text{since } H(L_i) \sim \frac{\exp\{x^2/2\}}{x} \quad \text{as } x \rightarrow \infty. \end{aligned}$$

If, for example, $\theta = \frac{1}{2}$, the last equation above suggests taking

$$L_{i+1} = L_i + \frac{1}{L_i},$$

and we deduce an equivalent of L_n for large values of n ,

$$L_n \sim \sqrt{2n} \quad \text{as } n \rightarrow \infty,$$

from the following lemma.

Lemma 5.1. *Let $(u_n)_n$ be a real-valued sequence such that*

$$u_0 > 0 \quad \text{and} \quad u_{n+1} = u_n + \frac{1}{u_n^\alpha} \quad \text{for all } n \in \mathbb{N},$$

where $\alpha > -1$. Then $u_n \sim [n(1 + \alpha)]^{1/(\alpha+1)}$ as $n \rightarrow \infty$.

Proof. We give only a sketch of the proof of this classical analytical result. First, we know that, using monotonicity and a fixed point argument, (u_n) diverges. Then we look for a constant $\beta > 0$ such that $u_{n+1}^\beta - u_n^\beta$ converges. The result then follows easily.

5.1.2. *Cost of transition.* Now we deduce the asymptotic behavior of the cost of the transition, i.e. the cost of a particle starting at x and reaching the next level $(x + 1/x)$ or 0:

$$E_x(T_{0,x+1/x}) \sim (1 - e^{-1}) \log(x) \quad \text{as } n \rightarrow \infty.$$

To prove this result, note that

$$E_x(T_{0,x+1/x}) = - \int_0^x \frac{R(t)}{\rho(t)} dt + \frac{\int_0^{x+1/x} (R(t)/\rho(t)) dt}{\int_0^{x+1/x} (1/\rho(t)) dt} \int_0^x \frac{1}{\rho(t)} dt$$

and write $R(t) = R(\infty) - D(t)$ with $D(t) = \int_t^\infty \rho(u) du$. Multiplying each side of the above equation by $\int_0^{x+1/x} dt/\rho(t)$, we obtain

$$\begin{aligned} E_x(T_{0,x+1/x}) \int_0^{x+1/x} \frac{dt}{\rho(t)} &= \int_0^x \frac{dt}{\rho(t)} \int_0^{x+1/x} \frac{R(t)}{\rho(t)} dt - \int_0^{x+1/x} \frac{dt}{\rho(t)} \int_0^x \frac{D(t)}{\rho(t)} dt \\ &= - \int_0^x \frac{dt}{\rho(t)} \int_x^{x+1/x} \frac{D(t)}{\rho(t)} dt + \int_x^{x+1/x} \frac{dt}{\rho(t)} \int_0^x \frac{D(t)}{\rho(t)} dt. \end{aligned}$$

Using classical estimates, we obtain the required result.

Remark 5.1. The above result justifies our hypothesis that the costs of the transition are asymptotically constant.

This type of result can also be deduced for the approximate Ornstein–Uhlenbeck process, but the analysis is much more complicated.

5.2. Numerical application

We suppose that the unknown drift is constant and equal to μ_i on disjoint intervals taken over each interval i :

$$\mu_i \in [\mu_i^1, \mu_i^2].$$

We arbitrarily determine a ‘mean’ drift constant on disjoint intervals taken over each interval i , $\mu_i^0 = (\mu_i^1 + \mu_i^2)/2$. We then determine the thresholds L_i such that the transition probabilities are equal to p and the optimal sampling numbers for the mean drift. Now we sample ρ_N processes according to

$$dX_t = -\mu(X_t)X_t dt + \sigma dW_t, \tag{5.2}$$

which we discretize as

$$X_{n+h} = -(\mu h - 1)X_n + \sigma G,$$

where h is the discretization step and G has an $\mathcal{N}(0, h)$ distribution. The sampling numbers used during this learning phase are the optimal sampling numbers for the mean drift determined previously. In this way we obtain estimates of the transition probabilities that allow us to estimate the optimal sampling numbers. We then sample $N - \rho_N$ processes according to discretization (5.2) with the estimated sampling numbers.

Remark 5.2. (*Simulation procedure.*) Owing to the discretization of the process and since the drift is not constant, a bias is introduced at each change of regime of the drift. We propose here a procedure to reduce this bias. Let $(m_i)_i$ denote the levels of drift change, and let $(X_n)_n$ be the studied process. Suppose that X_n is such that $m_{i-1} < X_n < m_i$. Then X_n evolves according to (5.2) with drift μ_i and

$$X_{n+1} = X_n - \mu_i h X_n + \sigma \sqrt{h} Z,$$

where Z is a realization of the $\mathcal{N}(0, 1)$ distribution.

If $X_{n+1} < m_i$, there is no problem: the process evolves from X_n according to (5.2) with drift μ_i and the next point X_{n+1} is still in the same regime of drift. The simulation is simply continued. On the other hand, if $X_{n+1} > m_i$, we should have changed the drift between X_n and X_{n+1} , which introduces the bias mentioned at the beginning of the remark. Thus, we solve the second-degree equation in X_{n+1} :

$$X_{n+1} = X_n - \frac{\mu_i(m_i - X_n) + \mu_{i+1}(X_{n+1} - m_i)}{X_{n+1} - X_n}hX_n + \sigma\sqrt{h}Z.$$

Finally, X_{n+1} is the obtained solution.

5.2.1. *Application.* We aim to estimate the probability that the process governed by (5.2) (with $\sigma = 0.3$) and starting from $x = 0.1$ reaches level $L = 4$ before going to $L_0 = 0$. The values of the different parameters corresponding to the target probability

$$P_x(X_{T_{L_0,L}} = L) = 1.676\,79 \times 10^{-08}$$

are reported in Table 1. The thresholds for the mean drift and $p = 0.2$ are given in Table 2.

TABLE 1.

$[m_i, m_{i+1}]$	μ_i	μ_i^1	μ_i^2	μ_i^0
$[0, \frac{1}{2}]$	0.05	0.05	0.06	0.055
$[\frac{1}{2}, 1]$	0.06	0.06	0.07	0.065
$[1, \frac{3}{2}]$	0.07	0.07	0.08	0.075
$[\frac{3}{2}, 2]$	0.09	0.08	0.09	0.085
$[2, \frac{5}{2}]$	0.09	0.09	0.10	0.095
$[\frac{5}{2}, 3]$	0.11	0.10	0.11	0.105
$[3, \frac{7}{2}]$	0.11	0.11	0.12	0.115
$[\frac{7}{2}, 4]$	0.12	0.12	0.13	0.125

TABLE 2.

Threshold	Value
L_1	0.477 833 928
L_2	1.406 139 441
L_3	2.073 610 760
L_4	2.480 952 409
L_5	2.785 321 189
L_6	3.039 774 162
L_7	3.257 152 634
L_8	3.453 693 339
L_9	3.632 850 394
L_{10}	3.795 331 329
L_{11}	3.949 233 228
L_{12}	4

TABLE 3.

Method	Parameter	Estimation	Error	Length CI 95%	Simulation cost	
ITSBSM	$N = 10^4$	—	—	—	—	
	$N = 10^5$	1.69×10^{-08}	1.21×10^{-10}	2.12×10^{-09}	1.19×10^{09}	
	$N = 10^6$	1.68×10^{-08}	1.21×10^{-12}	1.21×10^{-09}	1.19×10^{10}	
TSBSM	$N = 10^5$	1.75×10^{-08}	7.68×10^{-10}	1.72×10^{-09}	1.21×10^{09}	
IOSBSM	$N = 10^5$	1.66×10^{-08}	1.32×10^{-10}	4.30×10^{-09}	1.30×10^{10}	
SALO	$H = 10^2$	1.89×10^{-08}	2.16×10^{-09}	4.08×10^{-09}	9.02×10^{07}	
	$H = 10^3$	1.75×10^{-08}	7.61×10^{-10}	1.72×10^{-09}	9.18×10^{08}	
	$H = 10^4$	1.67×10^{-08}	1.27×10^{-10}	4.07×10^{-10}	6.45×10^{09}	
AACG ($p = \frac{1}{4}$)	$N = 10^5$	1.05×10^{-08}	6.23×10^{-09}	2.06×10^{-08}	9.30×10^{08}	
	($p = \frac{1}{3}$)	$N = 10^5$	1.14×10^{-08}	5.41×10^{-09}	1.66×10^{-08}	1.60×10^{09}
	($p = \frac{1}{2}$)	$N = 10^5$	1.22×10^{-08}	4.61×10^{-09}	2.56×10^{-08}	1.90×10^{09}
	($p = \frac{2}{3}$)	$N = 10^5$	1.98×10^{-08}	3.04×10^{-09}	2.97×10^{-08}	2.20×10^{09}
	($p = \frac{3}{4}$)	$N = 10^5$	1.86×10^{-08}	1.86×10^{-09}	2.86×10^{-08}	2.30×10^{09}

In Table 3 we present the results obtained with different methods, $N = 10^4, 10^5,$ or 10^6 processes generated at the start, and 50 iterations. The abbreviations TSBSM, ITSBSM, IOSBSM, SALO, and AACG stand for the two-step branching splitting model, the improved two-step branching splitting model (see Remark 5.2), the improved one-step branching splitting model (see Remark 5.2), the sequential algorithm of LeGland and Oudjane [14] (in this technique we run trajectories between two successive thresholds (which are given at the beginning of the simulation) until some number of successes, $H,$ is reached), and the adaptive algorithm of Cérou and Guyader [4], respectively. (In the AACG technique we run N trajectories until 0 is reached. The thresholds are then defined during the simulation in such a way that exactly a proportion p of these trajectories have succeeded to reach the level. To have a constant population, the trajectories that have failed are resampled on the successful ones.)

6. Conclusion

In this paper we proposed a two-step algorithm based on the branching splitting model. A precise analysis showed that we asymptotically dedicate $\mu_s C^{2/3}$ particles to the learning phase and $C/C_{\text{opt}} - \mu_s C^{2/3}$ particles to the second phase, where C_{opt} is a constant and μ_s is derived from the optimization of the algorithm, i.e. assuming that the number of particles generated during the learning phase behaved like $\mu C^{1-\alpha}$, we let $\alpha = \frac{1}{3}$.

This result followed directly from the particular choice of the splitting numbers, the R_i s, which are solutions of an optimization problem. Otherwise, if the R_i s are not taken in such a way, the optimal α would simply be equal to $\frac{1}{2}$, as expected a priori. This emphasized the importance of choosing the transition probabilities and the splitting numbers as close as possible to the optimals, and of having good estimates for the parameters chosen a priori. It also emphasized the need to have an adaptive algorithm in multiple phases that would be more efficient: evaluating the parameters at each step and continuing the simulation according to

these estimates. However, in such an algorithm, the calculation using martingale techniques rapidly becomes even more complex than those of this paper (see, e.g. Appendices A and B). Moreover, to show that the adaptive algorithm works better than the algorithm presented here would be difficult, and the gain resulting from such an approach would be very hard to quantify.

In terms of an efficient algorithm, the best thing to do would be to estimate and to increment not only the splitting numbers but also the transition probabilities, and then to move, before each phase, the thresholds according to these evaluations. But the precise analysis becomes more and more complex. Moreover, such an algorithm is based on the hypothesis that we can move the thresholds, which in practical settings is generally not the case.

Finally, it would also be worth analyzing non-Markovian models, multidimensional models as well as unidimensional models. In multidimensional models, the probability of reaching some level L_{i+1} starting from level L_i depends on the entry point in L_i . Such studies would be interesting and worth analyzing, but once again they are difficult to solve.

Appendix A. Proof of Lemma 3.1

In this appendix we prove Lemma 3.1. The first part of the lemma is clearly well known and follows directly from Chernoff’s bounding method and optimization. Then, first of all,

$$E(f(\tilde{P})) = E(f(\hat{P})\mathbf{1}_{\hat{P} \in [a,b]}) + E(f(\tilde{P})\mathbf{1}_{\hat{P} \in [a,b]^c}).$$

But, since $\tilde{P} \in [a, b]$ and f is bounded on $[a, b]$ by some constant M_f ,

$$E(f(\tilde{P})\mathbf{1}_{\hat{P} \in [a,b]^c}) \leq M_f P(\hat{P} \notin [a, b]) \leq 2M_f \exp\{-Nh(a, b)\}$$

by the first part of Lemma 3.1 and with $h(a, b) = \min\{l(a), l(b)\}$.

Then, we can note that

$$\begin{aligned} E(\hat{P}) &= P, & \text{var}(\hat{P}) &= \frac{P(1-P)}{N}, & E((\hat{P} - P)^3) &= \frac{P(1-P)}{N^2}(1-2P), \\ \text{and } E((\hat{P} - P)^4) &= \frac{P(1-P)}{N^2} \left((P^3 + (1-P)^3) \frac{1}{N} + \left(1 - \frac{1}{N}\right) P(1-P) \right), \end{aligned}$$

and since f is \mathcal{C}^2 on $[a, b]$, by a Taylor expansion,

$$f(\hat{P})\mathbf{1}_{\hat{P} \in [a,b]} = \left(f(P) + (\hat{P} - P)f'(P) + \frac{(\hat{P} - P)^2}{2} f''(P) + o((\hat{P} - P)^2) \right) \mathbf{1}_{\hat{P} \in [a,b]},$$

and so

$$\begin{aligned} E(f(\hat{P})\mathbf{1}_{\hat{P} \in [a,b]}) &= E \left(f(P) + (\hat{P} - P)f'(P) + \frac{(\hat{P} - P)^2}{2} f''(P) + o((\hat{P} - P)^2) \right) \\ &\quad - E \left(\left(f(P) + (\hat{P} - P)f'(P) + \frac{(\hat{P} - P)^2}{2} f''(P) + o((\hat{P} - P)^2) \right) \mathbf{1}_{\hat{P} \notin [a,b]} \right). \end{aligned}$$

The first term on the right-hand side is simply

$$\begin{aligned} f(P) + E(\hat{P} - P)f'(P) + \text{var}(\hat{P})\frac{f''(P)}{2} + E(o(\hat{P} - P)^2) \\ = f(P) + \frac{P(1-P)}{2N} f''(P) + o\left(\frac{1}{N}\right). \end{aligned}$$

Since $P \in [a, b]$, and f and its two first derivatives are bounded on $[a, b]$, the second term on the right-hand side is bounded by some constant times $P(\hat{P} \notin [a, b])$, itself bounded by $\exp\{-Nh(a, b)\}$, allowing us to conclude.

Appendix B. Cost asymptotic expression

In this appendix we prove Theorem 4.1. The asymptotic expression of the cost is given by

$$C = N \left(C_{\text{opt}} + \frac{B_1}{\lambda N^{1-\alpha}} + B_2 \frac{\lambda}{N^\alpha} + o\left(\frac{1}{N^\alpha}\right) + o\left(\frac{1}{N^{1-\alpha}}\right) \right),$$

where B_1, B_2 , and C_{opt} are constants given in Theorem 4.1.

Since C is given by (4.1), we need to evaluate $E(\tilde{p}_i^{(1)})$ (which is obviously p_i) and $E(\tilde{r}_i \tilde{p}_i^{(2)})$. Now define $\varphi(x) = \sqrt{1/x - 1}$ and $\psi(x) = \sqrt{x(1-x)}$. Then $\tilde{R}_i = \varphi(\tilde{P}_{i+1}^{(1)})/\psi(\tilde{P}_i^{(1)})$. To derive the formula, we proceed by induction, successive conditionings, and several uses of Lemma 3.1. for $k = i$ (and similarly for $k = i - 1, \dots, 2$), we have the following.

- Condition on \mathcal{F}_{2k} to isolate what happens from L_k and then apply Lemma 3.1 near P_{k+1} to the function ϕ when $k = i$ and to the function φ/ψ when $k = i - 1 \dots 2$ and

$$\tilde{P}_{k+1}^{(1)} = P_{k+1} + \frac{1}{\lambda N^{1-\alpha} r_k^0 \tilde{P}_k^{(1)}} \sum_{j=1}^{\lambda N^{1-\alpha} r_k^0 \tilde{P}_k^{(1)}} \text{Ber}'_j,$$

where $\text{Ber}'_j \stackrel{\mathcal{L}}{=} \text{Ber}(P_{k+1}) - P_{k+1}$.

- Condition on \mathcal{F}_{2k-1} to isolate what happens from L_{k-1} after the first phase and then apply Lemma 3.1 near P_k to the function Id and

$$\tilde{P}_k^{(2)} = P_k + \frac{1}{(N - \lambda N^{1-\alpha}) \tilde{r}_k \tilde{P}_{k-1}^{(2)}} \sum_{j=1}^{(N - \lambda N^{1-\alpha}) \tilde{r}_k \tilde{P}_{k-1}^{(2)}} \text{Ber}'_j,$$

where $\text{Ber}'_j \stackrel{\mathcal{L}}{=} \text{Ber}(P_k) - P_k$.

At step $k = i$, we obtain

$$\begin{aligned} E(\tilde{r}_i \tilde{p}_i^{(2)}) &= E\left(\frac{\tilde{r}_{i-1}}{\psi(\tilde{P}_i^{(1)})} \tilde{p}_i^{(2)} E_{2i}(\varphi(\tilde{P}_{i+1}^{(1)}))\right) \\ &= \varphi(P_{i+1}) E\left(\frac{\tilde{r}_{i-1}}{\psi(\tilde{P}_i^{(1)})} \tilde{p}_i^{(2)}\right) \\ &\quad + \frac{P_{i+1}(1 - P_{i+1})}{2\lambda N^{1-\alpha} r_i^0} \varphi''(P_{i+1}) E\left(\frac{\tilde{r}_{i-1}}{\psi(\tilde{P}_i^{(1)})} \frac{\tilde{p}_i^{(2)}}{\tilde{P}_i^{(1)}}\right) + o\left(\frac{1}{N^{1-\alpha}}\right), \end{aligned}$$

where

$$E\left(\frac{\tilde{r}_{i-1}}{\psi(\tilde{P}_i^{(1)})} \tilde{p}_i^{(2)}\right) = E\left(\frac{\tilde{r}_{i-1}}{\psi(\tilde{P}_i^{(1)})} \tilde{p}_{i-1}^{(2)} E_{2i-1}(\tilde{P}_i^{(2)})\right) = P_i E\left(\frac{\tilde{r}_{i-1}}{\psi(\tilde{P}_i^{(1)})} \tilde{p}_{i-1}^{(2)}\right) + o\left(\frac{1}{N^{1-\alpha}}\right)$$

and

$$E\left(\frac{\tilde{r}_{i-1}}{\psi(\tilde{P}_i^{(1)})} \frac{\tilde{P}_i^{(2)}}{\tilde{P}_i^{(1)}}\right) = \frac{r_{i-1}}{\psi(P_i)} + o\left(\frac{1}{N^{1-\alpha}}\right).$$

Then proceeding for k from $i - 1$ to 2 leads to

$$\begin{aligned} E(\tilde{r}_i \tilde{P}_i^{(2)}) &= \frac{r_i}{R_1} \frac{P_i}{P_1} \varphi(P_2) E\left(\frac{\tilde{P}_1^{(2)}}{\psi(\tilde{P}_1^{(1)})}\right) \\ &+ \frac{r_i P_i}{2\lambda N^{1-\alpha}} \sum_{k=1}^{i-1} \frac{P_{k+1}(1 - P_{k+1})}{r_k^0 R_k R_{k+1}} \frac{\varphi(P_{k+2})}{P_k \psi(P_k)} \left(\frac{\varphi}{\psi}\right)''(P_{k+1}) \\ &+ \frac{P_{i+1}(1 - P_{i+1})}{2\lambda N^{1-\alpha} r_i^0} \varphi''(P_{i+1}) \frac{r_{i-1}}{\psi(P_i)} + o\left(\frac{1}{N^{1-\alpha}}\right). \end{aligned}$$

To derive the required expression, it remains to apply Lemma 3.1 to $(x, y) \mapsto y/\psi(x)$.

Appendix C. Variance asymptotic expression

In this appendix we prove Theorem 4.2. The asymptotic expression of the variance is given by

$$\text{var}(\tilde{P}) = \frac{P(L)^2}{N} \left(V_{\text{opt}} + \frac{1}{\lambda N^{1-\alpha}} A_1 + \frac{\lambda}{N^\alpha} A_2 + \frac{\lambda^2}{N^{2\alpha}} A_3 + o\left(\frac{1}{N^{1-\alpha}}\right) + o\left(\frac{1}{N^{2\alpha}}\right) \right),$$

where A_1, A_2, A_3 , and C_{opt} are constants given in Theorem 4.2.

To prove the result, we establish a recursive relation between $\text{var}(\tilde{P}) = \text{var}(\tilde{p}_{M+1})$ and $\text{var}(\tilde{p}_M)$, and we conclude by iterating this relation. During the proof, in order to simplify the exposition, we use ‘ \approx ’ to denote any asymptotic expansion at the order $1/N^{(1+\alpha)\wedge(2-\alpha)}$. Now, for $k = 1, \dots, M$, let Z_k be the ratio between the numbers of particles having reached level L_k in phases 1 and 2:

$$Z_k = \frac{\tilde{P}_k^{(1)} r_k^0}{\tilde{P}_k^{(2)} \tilde{r}_k} =: \frac{\bar{Z}_k}{\tilde{R}_k} = \bar{Z}_k \frac{\psi(\tilde{P}_k^{(1)})}{\varphi(\tilde{P}_{k+1}^{(1)})}.$$

Then

$$\tilde{P}_k = \tilde{P}_k^{(2)} \frac{1 + (\lambda/N^\alpha)(Z_{k-1} \tilde{P}_k^{(1)}/\tilde{P}_k^{(2)} - 1)}{1 + (\lambda/N^\alpha)(Z_{k-1} - 1)}.$$

We proceed as in Appendix A by successive conditionings and several applications of Lemma 3.1. Since the estimate $\tilde{P} = \tilde{p}_{M+1}$ of $P(L)$ is expressed as a product $\tilde{P}_1 \cdots \tilde{P}_{M+1}$, conditionings appear to be particularly efficient to deduce the required recursive relation.

A first conditioning by \mathcal{F}_{2M+1} leads to

$$\begin{aligned} \text{var}(\tilde{P}) &= \text{var}(\tilde{p}_{M+1}) \\ &= \text{var}\left(\frac{\tilde{p}_M}{\tilde{p}_M^{(2)} \tilde{r}_M \{1 + (\lambda/N^\alpha)(Z_M - 1)\}}\right. \\ &\quad \times \left. \left\{ \frac{\lambda}{N^\alpha} \tilde{p}_{M+1}^0 r_M^0 + \left(1 - \frac{\lambda}{N^\alpha}\right) \tilde{p}_M^{(2)} \tilde{r}_M \mathbb{E}_{2M+1}(\tilde{P}_{M+1}^{(2)}) \right\}\right) \\ &\quad + \mathbb{E}\left(\left(\frac{\tilde{p}_M}{\tilde{p}_M^{(2)} \tilde{r}_M \{1 + (\lambda/N^\alpha)(Z_M - 1)\}}\right)^2 \left(\left(1 - \frac{\lambda}{N^\alpha}\right) \tilde{p}_M^{(2)} \tilde{r}_M\right)^2 \text{var}_{2M+1}(\tilde{P}_{M+1}^{(2)})\right) \\ &\approx P_{M+1}^2 \text{var}\left(\tilde{p}_M \left\{1 + \frac{\lambda}{N^\alpha} Z_M \left(\frac{\tilde{P}_{M+1}^{(1)}}{\tilde{p}_{M+1}^{(1)}} - 1\right) \left(1 - \frac{\lambda}{N^\alpha} (Z_M - 1)\right)\right\}\right) \\ &\quad + \frac{P_{M+1}(1 - P_{M+1})}{N} \left(1 - \frac{\lambda}{N^\alpha}\right) \\ &\quad \times \mathbb{E}\left(\frac{\tilde{p}_M^2}{\tilde{p}_M^{(2)} \tilde{r}_M} \left(1 - 2\frac{\lambda}{N^\alpha} (Z_M - 1) + 3\frac{\lambda^2}{N^{2\alpha}} (Z_M - 1)^2\right)\right). \end{aligned}$$

Using $\mathbb{E}_{2M+1}(\tilde{P}_{M+1}^{(2)}) = P_{M+1} + o(1/N)$,

$$\text{var}_{2M+1}(\tilde{P}_{M+1}^{(2)}) = \frac{P_{M+1}(1 - P_{M+1})}{N(1 - \lambda/N^\alpha) \tilde{p}_M^{(2)} \tilde{r}_M} + o\left(\frac{1}{N}\right) \quad (\text{by Lemma 3.1}),$$

and $(1 + x)^\alpha \approx (1 + \alpha x + \alpha(\alpha + 1)/2x^2)$ as $\alpha \rightarrow 0$, the right-hand side of the above equation is asymptotically equal to

$$\begin{aligned} &P_{M+1}^2 \text{var}\left(\tilde{p}_M \left(1 + \frac{\lambda}{N^\alpha} \bar{Z}_M \psi(\tilde{P}_M^{(1)}) \mathbb{E}_{2M}(f(\tilde{P}_{M+1}^{(1)}))\right)\right) \\ &+ P_{M+1}^2 \mathbb{E}\left(\tilde{p}_M^2 \frac{\lambda^2}{N^{2\alpha}} (\bar{Z}_M \psi(\tilde{P}_M^{(1)}))^2 \text{var}_{2M}(f(\tilde{P}_{M+1}^{(1)}))\right) \\ &+ \frac{P_{M+1}(1 - P_{M+1})}{N} \left(1 - \frac{\lambda}{N^\alpha}\right) \mathbb{E}\left(\frac{\tilde{p}_M^2}{\tilde{p}_M^{(2)} \tilde{r}_{M-1}} \psi(\tilde{P}_M^{(1)}) \mathbb{E}_{2M}(g(\tilde{P}_{M+1}^{(1)}))\right) \\ &=: T_1 + T_2 + T_3, \end{aligned}$$

where we have used two conditionings by \mathcal{F}_{2M} :

$$f(x) = \frac{1}{\varphi(x)} \left(\frac{x}{P_{M+1}} - 1\right) \left(1 - \frac{\lambda}{N^\alpha} \left(\bar{Z}_M \frac{\psi(\tilde{P}_M^{(1)})}{\varphi(x)} - 1\right)\right),$$

and

$$g(x) = \frac{1}{\varphi(x)} \left(1 - 2\frac{\lambda}{N^\alpha} \left(\bar{Z}_M \frac{\psi(\tilde{P}_M^{(1)})}{\varphi(x)} - 1\right) + 3\frac{\lambda^2}{N^{2\alpha}} \left(\bar{Z}_M \frac{\psi(\tilde{P}_M^{(1)})}{\varphi(x)} - 1\right)^2\right).$$

Let us first consider the calculation of T_1 . Applying Lemma 3.1 to f , we obtain

$$\begin{aligned} T_1 &\approx P_{M+1}^2 \operatorname{var} \left(\tilde{p}_M \left[1 + \frac{1}{N} \frac{\psi(\tilde{P}_M^{(1)})\varphi(P_{M+1})^{-1}}{\tilde{p}_M^{(2)}\tilde{r}_{M-1}P_{M+1}} \left\{ 1 - \frac{\lambda}{N^\alpha} \left(2\bar{Z}_M \frac{\psi(\tilde{P}_M^{(1)})}{\varphi(P_{M+1})} - 1 \right) \right\} \right] \right) \\ &\approx P_{M+1}^2 \operatorname{var}(\tilde{p}_M) \\ &\quad + \frac{1}{4N^2\varphi(P_{M+1})^2} \operatorname{var} \left(\frac{\tilde{p}_M\psi(\tilde{P}_M^{(1)})}{\tilde{p}_M^{(2)}\tilde{r}_{M-1}} \left\{ 1 - \frac{\lambda}{N^\alpha} \left(2\bar{Z}_M \frac{\psi(\tilde{P}_M^{(1)})}{\varphi(P_{M+1})} - 1 \right) \right\} \right) \\ &\quad + \frac{P_{M+1}}{N\varphi(P_{M+1})} \operatorname{cov} \left(\tilde{p}_M, \frac{\tilde{p}_M\psi(\tilde{P}_M^{(1)})}{\tilde{p}_M^{(2)}\tilde{r}_{M-1}} \left\{ 1 - \frac{\lambda}{N^\alpha} \left(2\bar{Z}_M \frac{\psi(\tilde{P}_M^{(1)})}{\varphi(P_{M+1})} - 1 \right) \right\} \right). \end{aligned}$$

As the second term on the right-hand side (which is in $1/N^2$) is negligible, we study only the term involving the covariance, which is equal to

$$\begin{aligned} &E \left(\tilde{p}_M^2 \frac{\psi(\tilde{P}_M^{(1)})}{\tilde{p}_M^{(2)}\tilde{r}_{M-1}} \left\{ 1 - \frac{\lambda}{N^\alpha} \left(2\bar{Z}_M \frac{\psi(\tilde{P}_M^{(1)})}{\varphi(P_{M+1})} - 1 \right) \right\} \right) \\ &\quad - E(\tilde{p}_M) E \left(\tilde{p}_M \frac{\psi(\tilde{P}_M^{(1)})}{\tilde{p}_M^{(2)}\tilde{r}_{M-1}} \left\{ 1 - \frac{\lambda}{N^\alpha} \left(2\bar{Z}_M \frac{\psi(\tilde{P}_M^{(1)})}{\varphi(P_{M+1})} - 1 \right) \right\} \right) \\ &=: T_{11} - T_{12}T_{13}, \end{aligned}$$

and we show that it is null. As the integration procedure is the same for all the expectations introduced in this proof, we detail only the calculation of $E(\tilde{p}_M)$ in order to simplify the exposition:

$$\begin{aligned} T_{12} &= E(\tilde{p}_M) \\ &= E \left(\tilde{p}_{M-1} \tilde{P}_M^{(2)} \left\{ \frac{1 + (\lambda/N^\alpha)(Z_{M-1}\tilde{P}_M^{(1)}/\tilde{P}_M^{(2)} - 1)}{1 + (\lambda/N^\alpha)(Z_{M-1} - 1)} \right\} \right) \\ &= E \left(\tilde{p}_{M-1} \left(1 + \frac{\lambda}{N^\alpha} (Z_{M-1} - 1) \right)^{-1} E_{2M-1}(u_M(\tilde{P}_M^{(2)})) \right), \end{aligned}$$

conditioning on \mathcal{F}_{2M-1} and noting that

$$u_M(x) = x \left(1 + \frac{\lambda}{N^\alpha} \left(Z_{M-1} \frac{\tilde{P}_M^{(1)}}{x} - 1 \right) \right).$$

However, by Lemma 3.1 (the second term being negligible),

$$\begin{aligned} E_{2M-1}(u_M(\tilde{P}_M^{(2)})) &= u_M(P_M) + \frac{P_M(1 - P_M)}{2N(1 - \lambda/N^\alpha)} u_M''(P_M) + o\left(\frac{1}{N}\right) \\ &= P_M \left(1 + \frac{\lambda}{N^\alpha} \left(Z_{M-1} \frac{\tilde{P}_M^{(1)}}{P_M} - 1 \right) \right) + o\left(\frac{1}{N^\alpha}\right). \end{aligned}$$

Now let

$$\begin{aligned} v_M(x) &= \left(1 - \frac{\lambda}{N^\alpha} \left(\bar{Z}_{M-1} \frac{\psi(\tilde{P}_{M-1}^{(1)})}{\varphi(x)} - 1 \right) + \frac{\lambda^2}{2N^{2\alpha}} \left(\bar{Z}_{M-1} \frac{\psi(\tilde{P}_{M-1}^{(1)})}{\varphi(x)} - 1 \right)^2 \right) \\ &\quad \times \left(1 + \frac{\lambda}{N^\alpha} \left(\bar{Z}_{M-1} \frac{\psi(\tilde{P}_{M-1}^{(1)})}{\varphi(x)} \frac{x}{P_M} - 1 \right) \right), \end{aligned}$$

and apply Lemma 3.1 to v_M to obtain

$$E_{2(M-1)}(v_M(\tilde{P}_M^{(1)})) = v_M(P_M) + \frac{P_M(1 - P_M)}{2\lambda N^{1-\alpha}} v_M''(P_M) + o\left(\frac{1}{N^{1-\alpha}}\right) = 1 + o\left(\frac{1}{N}\right).$$

Then, by conditioning on $\mathcal{F}_{2(M-1)}$ we can deduce that

$$T_{12} \approx P_M E(\tilde{p}_{M-1} E_{2(M-1)}(v_M(\tilde{P}_M^{(1)}))) \approx P_M E(\tilde{p}_{M-1}) \approx \dots \approx P_M.$$

Making successive conditionings and applications of Lemma 3.1, T_{11} becomes

$$\begin{aligned} T_{11} &\approx \frac{P_M \cdots P_2}{R_{M-1} \cdots R_2} \frac{\psi(P_M)}{\varphi(P_2)} E\left(\tilde{P}_1^2 \frac{\psi(\tilde{P}_1^{(1)})}{\tilde{P}_1^{(2)}} \left\{ 1 - \frac{\lambda}{N^\alpha} \left(2 \frac{\tilde{P}_1^{(1)} r_M^0}{\tilde{P}_1^{(2)}} \frac{\psi(\tilde{P}_1^{(1)})}{\varphi(P_2)} - 1 \right) \right\}\right) \\ &= \frac{P_M \psi(P_M)}{r_{M-1}} \left(\left\{ 1 - \frac{\lambda}{N^\alpha} \left(2 \frac{r_M^0}{r_M} - 1 \right) \right\} - \frac{1/8}{\lambda N^{1-\alpha}} \frac{1}{P_1(1 - P_1)} \right) + o\left(\frac{1}{N^{1-\alpha}}\right). \end{aligned}$$

By an application of Lemma 3.1 to Φ , defined by

$$\Phi(x, y) = y\psi(x) \left(1 + 2 \frac{\lambda}{N^\alpha} \left(\frac{x}{y} - 1 \right) + \frac{\lambda^2}{N^{2\alpha}} \left(\frac{x}{y} - 1 \right)^2 \right) \left(1 - \frac{\lambda}{N^\alpha} \left(\frac{2r_M^0}{\varphi(P_2)} \frac{x\psi(x)}{y} - 1 \right) \right),$$

we obtain

$$\begin{aligned} E(\Phi(\tilde{P}_1^{(1)}, \tilde{P}_1^{(2)})) &= \Phi(P_1, P_1) + P_1(1 - P_1) \left(\frac{\Phi_x''(P_1, P_1)}{2\lambda N^{1-\alpha}} + \frac{\Phi_y''(P_1, P_1)}{2N(1 - \lambda/N^\alpha)} \right) \\ &\quad + o\left(\frac{1}{N^{1-\alpha}}\right) \\ &= P_1\psi(P_1) \left(1 - \frac{\lambda}{N^\alpha} \left(2 \frac{r_M^0}{r_M} - 1 \right) \right) + P_1(1 - P_1) \frac{\Phi_x''(P_1, P_1)}{2\lambda N^{1-\alpha}} \\ &\quad + o\left(\frac{1}{N^{1-\alpha}}\right). \end{aligned}$$

In the same way we obtain

$$T_{13} = \frac{\psi(P_M)}{r_{M-1}} \left(\left\{ 1 - \frac{\lambda}{N^\alpha} \left(2 \frac{r_M^0}{r_M} - 1 \right) \right\} - \frac{1/8}{\lambda N^{1-\alpha}} \frac{1}{P_1(1 - P_1)} \right) + o\left(\frac{1}{N^{1-\alpha}}\right),$$

from which we deduce the nullity of the covariance term. Finally, $T_1 \approx P_{M+1}^2 \text{var}(\tilde{p}_M)$.

Let us now consider the calculation of T_2 . Making successive conditionings and applications of Lemma 3.1, it follows that

$$\begin{aligned} T_2 &\approx \frac{\lambda}{N^{1+\alpha}} (1 - P_{M+1}) P_{M+1} \cdots P_2 \frac{r_M^0}{(R_M \cdots R_2)^2 \varphi(P_2)^2} \\ &\quad \times E\left(\tilde{P}_1^2 \frac{\tilde{P}_1^{(1)} \psi(\tilde{P}_1^{(1)})^2}{\tilde{P}_1^{(2)}} \left\{ 1 - 2 \frac{\lambda}{N^\alpha} \left(\frac{r_M^0}{R_M \cdots R_2} \frac{\psi(\tilde{P}_1^{(1)})}{\varphi(P_2)} \frac{\tilde{P}_1^{(1)}}{\tilde{P}_1^{(2)}} - 1 \right) \right\}\right) \\ &= \frac{\lambda}{N^{1+\alpha}} (1 - P_{M+1}) P_{M+1} \frac{r_M^0}{r_M^2} \left(1 - 2 \frac{\lambda}{N^\alpha} \left(\frac{r_M^0}{r_M} - 1 \right) \right) + o\left(\frac{1}{N^{1+\alpha}}\right), \end{aligned}$$

where the last step follows from an application of Lemma 3.1 to Ψ , defined by

$$\Psi(x, y) = x\psi(x)^2 \left(1 + \frac{\lambda}{N^\alpha} \left(\frac{x}{y} - 1\right)\right) \left(1 - 2 \frac{\lambda}{N^\alpha} \left(\frac{r_M^0}{R_M \cdots R_2} \frac{\psi(x)}{\varphi(P_2)} \frac{x}{y} - 1\right)\right),$$

just keeping the first term.

Let us now consider the calculation of T_3 . Applying Lemma 3.1 to g ,

$$\begin{aligned} T_3 \approx & \frac{1}{2\lambda N^{2-\alpha}} \frac{P_{M+1} - 1/4}{r_M r_M^0} + \frac{P_{M+1}(1 - P_{M+1})}{N} \left(1 - \frac{\lambda}{N^\alpha}\right) \\ & \times \mathbb{E} \left(\frac{\tilde{p}_M^2}{\tilde{p}_M' \tilde{r}_{M-1}} \frac{\psi(\tilde{P}_M^{(1)})}{\varphi(P_{M+1})} \left\{ 1 - 2 \frac{\lambda}{N^\alpha} \left(\frac{r_M^0}{R_M \cdots R_2} \frac{\psi(\tilde{P}_1^{(1)})}{\varphi(P_2)} \frac{\tilde{P}_1^{(1)}}{\tilde{P}_1^{(2)}} - 1 \right) \right. \right. \\ & \left. \left. + 3 \frac{\lambda^2}{N^{2\alpha}} \left(\frac{r_M^0}{R_M \cdots R_2} \frac{\psi(\tilde{P}_1^{(1)})}{\varphi(P_2)} \frac{\tilde{P}_1^{(1)}}{\tilde{P}_1^{(2)}} - 1 \right)^2 \right\} \right). \end{aligned}$$

But the second term on the right-hand side, after successive conditionings and applications of Lemma 3.1, becomes

$$\begin{aligned} & \frac{1 - P_{M+1}}{N} \left(1 - \frac{\lambda}{N^\alpha}\right) \frac{P_{M+1} \cdots P_2}{R_M \cdots R_2 \varphi(P_2)} \\ & \times \mathbb{E} \left(\tilde{P}_1^2 \frac{\psi(\tilde{P}_1^{(1)})}{\tilde{P}_1'} \left\{ 1 - 2 \frac{\lambda}{N^\alpha} \left(\frac{r_M^0}{R_M \cdots R_2} \frac{\psi(\tilde{P}_1^{(1)})}{\varphi(P_2)} \frac{\tilde{P}_1^{(1)}}{\tilde{P}_1^{(2)}} - 1 \right) \right. \right. \\ & \left. \left. + 3 \frac{\lambda^2}{N^{2\alpha}} \left(\frac{r_M^0}{R_M \cdots R_2} \frac{\psi(\tilde{P}_1^{(1)})}{\varphi(P_2)} \frac{\tilde{P}_1^{(1)}}{\tilde{P}_1^{(2)}} - 1 \right)^2 \right\} \right) \\ & \approx \frac{P_{M+1}(1 - P_{M+1})}{N r_M} \left(1 - \frac{\lambda}{N^\alpha}\right) \left(1 - 2 \frac{\lambda}{N^\alpha} \left(\frac{r_M^0}{r_M} - 1\right) + 3 \frac{\lambda^2}{N^{2\alpha}} \left(\frac{r_M^0}{r_M} - 1\right)^2\right) \\ & \quad - \frac{1 - P_{M+1}}{8\lambda N^{2-\alpha}} \frac{P_{M+1}}{r_M P_1 (1 - P_1)}, \end{aligned}$$

where the last step follows from an application of Lemma 3.1 to Γ , defined by

$$\begin{aligned} \Gamma(x, y) = & y\psi(x) \left(1 + 2 \frac{\lambda}{N^\alpha} \left(\frac{x}{y} - 1\right) + \frac{\lambda^2}{N^{2\alpha}} \left(\frac{x}{y} - 1\right)^2\right) \\ & \times \left(1 - 2 \frac{\lambda}{N^\alpha} \left(\frac{r_M^0}{R_M \cdots R_2} \frac{\psi(x)}{\varphi(P_2)} \frac{x}{y} - 1\right) + 3 \frac{\lambda^2}{N^{2\alpha}} \left(\frac{r_M^0}{R_M \cdots R_2} \frac{\psi(x)}{\varphi(P_2)} \frac{x}{y} - 1\right)^2\right), \end{aligned}$$

just keeping the first two terms.

From all these results, we deduce the recursive relation between $\text{var}(\tilde{p}_{M+1}) (= \text{var}(\tilde{P}))$ and $\text{var}(\tilde{p}_M)$:

$$\begin{aligned} \text{var}(\tilde{P}) \approx & P_{M+1}^2 \text{var}(\tilde{p}_M) + \frac{\lambda}{N^{1+\alpha}} (1 - P_{M+1}) P_{M+1} \frac{r_M^0}{r_M^2} \left(1 - 2 \frac{\lambda}{N^\alpha} \left(\frac{r_M^0}{r_M} - 1\right)\right) \\ & + \frac{P_{M+1}(1 - P_{M+1})}{N r_M} \left(1 - \frac{\lambda}{N^\alpha}\right) \left(1 - 2 \frac{\lambda}{N^\alpha} \left(\frac{r_M^0}{r_M} - 1\right) + 3 \frac{\lambda^2}{N^{2\alpha}} \left(\frac{r_M^0}{r_M} - 1\right)^2\right) \\ & + \frac{1}{2\lambda N^{2-\alpha}} \frac{1}{r_M} \left(\frac{P_{M+1} - 1/4}{r_M^0} - \frac{P_{M+1}}{4} \frac{1 - P_{M+1}}{P_1(1 - P_1)}\right). \end{aligned}$$

Now it just remains to evaluate $\text{var}(\tilde{p}_1)(= \text{var}(\tilde{P}_1))$ to initialize the induction: applying Lemma 3.1,

$$\begin{aligned} \text{var}(\tilde{P}_1) &= \text{var}\left(\tilde{P}_1^{(2)}\left\{1 + \frac{\lambda}{N^\alpha}\left(\frac{\tilde{P}_1^{(1)}}{\tilde{P}_1^{(2)}} - 1\right)\right\}\right) + o\left(\frac{1}{N^{1-\alpha}}\right) \\ &= P_1(1 - P_1)\left(\frac{\Delta_x^2(P_1, P_1)}{\lambda N^{1-\alpha}} + \frac{\Delta_y^2(P_1, P_1)}{N(1 - \lambda/N^\alpha)}\right) + o\left(\frac{1}{N^{1-\alpha}}\right) \\ &= P_1(1 - P_1)\left(\frac{1}{\lambda N^{1-\alpha}} \frac{\lambda^2}{N^{2\alpha}} + \frac{(1 - \lambda/N^\alpha)^2}{N(1 - \lambda/N^\alpha)}\right) + o\left(\frac{1}{N^{1-\alpha}}\right) \\ &= \frac{P_1(1 - P_1)}{N} + o\left(\frac{1}{N^{1-\alpha}}\right), \end{aligned}$$

where $\Delta(x, y) = y(1 + (\lambda/N^\alpha)(x/y - 1))$, and we conclude by an induction.

Appendix D. Proof of Lemma 4.1

In this appendix we prove Lemma 4.1. We detail only the case in which $\alpha = \frac{1}{3}$ (the other results can be easily deduced using the same technique). First, since N is given by $G = G_f$ and by the asymptotic expression of G , N satisfies

$$N - N^{2/3} f_1(\lambda) + N^{1/3} f_4(\lambda) \sim G_f, \tag{D.1}$$

and so we can consider

$$N \sim G_f + \mu G_f^{2/3} + \nu G_f^{1/3} = G_f \left(1 + \frac{\mu}{G_f^{1/3}} + \frac{\nu}{G_f^{2/3}}\right)$$

for some functions μ and ν to be determined. Substituting this value into (D.1), we finally obtain

$$G_f \sim G_f + (\mu - f_1(\lambda))G_f^{2/3} + \left(\nu + f_4(\lambda) - \frac{2}{3}f_1(\lambda)\mu\right)G_f^{1/3},$$

from which we deduce that $\mu = f_1(\lambda)$, $\nu = \frac{2}{3}f_1(\lambda)^2 - f_4(\lambda)$, and

$$N \sim G_f \left(1 + \frac{f_1(\lambda)}{G_f^{1/3}} + \frac{(2/3)f_1(\lambda)^2 - f_4(\lambda)}{G_f^{2/3}}\right).$$

Now substituting this N into the asymptotic expression of F in Lemma 4.1, we obtain

$$\begin{aligned} F(\lambda, N) &\sim \frac{1}{G_f} \left(1 - \frac{\mu}{G_f^{1/3}} - \frac{\nu}{G_f^{2/3}} + \mu^2 \frac{1}{G_f^{2/3}}\right) + \frac{f_1(\lambda)}{G_f^{4/3}} \left(1 - \frac{4}{3} \frac{\mu}{G_f^{1/3}}\right) \\ &\quad + \frac{(f_2 + f_3)(\lambda)}{G_f^{5/3}} \\ &\sim \frac{1}{G_f} + \frac{1}{G_f^{4/3}}(f_1(\lambda) - \mu) + \frac{1}{G_f^{5/3}} \left((f_2 + f_3)(\lambda) - \frac{4}{3}\mu f_1(\lambda) - \nu + \mu^2\right) \\ &\sim \frac{1}{G_f} + \frac{1}{G_f^{5/3}}((f_2 + f_3)(\lambda) + f_4(\lambda) - f_1(\lambda)^2), \end{aligned}$$

from which we deduce the required result for $\alpha = \frac{1}{3}$.

Acknowledgements

I gratefully acknowledge my supervisors Dominique Bakry and Pascal Lezaud for their helpful suggestions and comments. I also thank the reviewer for a critical and attentive reading of the manuscript and constructive detailed comments.

References

- [1] ALDOUS, D. (1989). *Probability Approximations via the Poisson Clumping Heuristic* (Appl. Math. Sci. **77**). Springer, New York.
- [2] ALDOUS, D. AND VAZIRANI, U. V. (1994). ‘Go with the winners’ algorithms. In *Proc. 35th IEEE Symp. Foundations Comput. Sci.*, IEEE Computer Society Press, Silver Spring, MD, pp. 492–501.
- [3] ATHREYA, K. B. AND NEY, P. E. (1972). *Branching Processes*. Springer, New York.
- [4] CÉROU, F. AND GUYADER, A. (2005). Adaptive multilevel splitting for rare event analysis. Tech. Rep. 5710, INRIA.
- [5] DE BOER, P. T. (2000). Analysis and efficient simulation of queueing models of telecommunication systems. Doctoral Thesis, University of Twente.
- [6] DEL MORAL, P. (2004). *Feynman–Kac Formulae*. Springer, New York.
- [7] DIACONIS, P. AND HOLMES, S. (1995). Three examples of Monte-Carlo Markov chains: at the interface between statistical computing, computer science, and statistical mechanics. In *Discrete Probability and Algorithms* (Minneapolis, MN, 1993; IMA Vol. Math. Appl. **72**), Springer, New York, pp. 43–56.
- [8] DOUCET, A., DE FREITAS, N. AND GORDON, N. (2001). An introduction to sequential Monte Carlo methods. In *Sequential Monte Carlo Methods in Practice*, Springer, New York, pp. 3–14.
- [9] HARRIS, T. E. (2002). *The Theory of Branching Processes*. Corrected republication of the 1963 edition. Dover, New York.
- [10] HEIDELBERGER, P. (1995). Fast simulation of rare events in queueing and reliability models. *ACM Trans. Model. Comput. Simul.* **5**, 43–85.
- [11] JERRUM, M. AND SINCLAIR, A. (1997). The Markov chain Monte Carlo method: an approach to approximate counting and integration. In *Approximation Algorithms for NP-hard Problems*. PWS Publishing, Boston, MA, pp. 482–520.
- [12] LAGNOUX, A. (2006). Rare event simulation. *Prob. Eng. Inf. Sci.* **20**, 45–66.
- [13] LAGNOUX-RENAUDIE, A. (2008). Effective branching splitting method under cost constraint. *Stoch. Process. Appl.* **118**, 1820–1851.
- [14] LEGLAND F. AND OUDJANE, N. (2006). A sequential particle algorithm that keeps the particle system alive. In *Stochastic Hybrid Systems* (Lecture Notes Control Inf. Sci. **337**), Springer, Berlin, pp. 351–389.
- [15] LYONS, R. AND PERES, Y. (2009). *Probability on Trees and Networks*. Cambridge University Press.
- [16] SADOWSKY, J. S. (1996). On Monte Carlo estimation of large deviations probabilities. *Ann. Appl. Prob.* **6**, 399–422.
- [17] VILLÉN-ALTAMIRANO, M. AND VILLÉN-ALTAMIRANO, J. (1991). Restart: a method for accelerating rare event simulations. In *Queueing Performance and Control in ATM*, Elsevier Science Publishers, North-Holland, pp. 71–76.
- [18] VILLÉN-ALTAMIRANO, M. AND VILLÉN-ALTAMIRANO, J. (1997). Restart: an efficient and general method for fast simulation of rare event. Tech. Rep. 7, Universidad Politécnica de Madrid.