Computing the principal eigenelements of some linear operators using a branching Monte Carlo method

Antoine Lejay [a,*], Sylvain Maire [b]

[a] Projet TOSCA, IECN (Nancy-Université, CNRS, INRIA), Campus Scientifique, BP 239, 54506 Vandoeuvre-lès-Nancy CEDEX, France
[b] ISITV, Université de Toulon et du Var, Avenue G. Pompidou, BP 56, 83262 La Valette du Var CEDEX, France

ARTICLE INFO

ABSTRACT

In earlier work, we developed a Monte Carlo method to compute the principal eigenvalue of linear operators, which was based on the simulation of exit times. In this paper, we generalize this approach by showing how to use a branching method to improve the efficacy of simulating large exit times for the purpose of computing eigenvalues. Furthermore, we show that this new method provides a natural estimation of the first eigenfunction of the adjoint operator. Numerical examples of this method are given for the Laplace operator and an homogeneous neutron transport operator.

1. Introduction

The value of the principal eigenvalue of the neutron transport operator or the diffusion operator in a bounded domain gives relevant physical information regarding the large time behavior of the solution of the associated Cauchy problems. For diffusion operators (e.g. Laplace), this eigenvalue controls the speed of convergence toward the steady-state; probabilistically this is the rate of absorption of Brownian particles on the boundary. In addition, the principal eigenvalue also appears in problems related to stochastic analysis; for example see [13] or [19, Chap. VI, §8]. In the case of the neutron transport operator, the sign of this first eigenvalue determines if the system is sub-critical or super-critical [9]. This kind of problem also appear in quantum physics, for example for the calculation of the ground state for a Hamiltonian.

The numerical computation of the principal eigenvalue and eigenfunction using a deterministic method requires the handling of very large matrices obtained from sufficiently refined discretizations of the operator. The power method [27] is then

used to compute this principal eigenvalue and the corresponding eigenvector. There exists Monte Carlo versions of this power method for stochastic matrices [29] or for a wide class of many-body problems and also for neutron transport criticality computations [21,22,17,18]. The idea is to approximate the eigenfunction using particles which are updated using a branching mechanism from a generation to another until convergence. Most of the time in relation to the Schrödinger equation, this has given rise to a large class of methods known under various names – Quantum Monte Carlo, Diffusion Monto Carlo, Green function Monte Carlo,... – that are used to compute the ground state [1,23,36,14,4]. Many of these approaches rely on the choice of a good approximation – a trial function – of the ground state deduced from the problem to perform an importance sampling with a correcting drift, to project the density, or to obtain an approximation of the density of the particles. This trial function can be deduced from the use of pseudo-potentials [31], from considerations on nodal lines (the fixed-node approximation [14,34]), from parameter optimization [38]. On some particular problems, these methods could lead to very accurate results.

In our context, one cannot necessarily deduce a good trial function from the shape of the domain. One of the authors [32] has shown how to estimate the principal eigenvalue of the neutron transport operator without computing the eigenfunction by combining, the probabilistic, Feynman-Kac representation of the solution of the related Cauchy problem and the spectral expansion of the solution. When considering homogeneous neutron transport operators, this method reduces to the computation of the first time, τ, of when a properly defined particle exits the domain. The idea is to estimate $F(t) = \mathbb{P}_x[\tau < t]$, when all the particles start from a single point x, and to use the approximation $F(t) \sim C \exp(\lambda_1 t)$ for t large, where $\lambda_1 < 0$ is the desired principal eigenvalue. The eigenvalue, λ_1, is then evaluated using linear regression. This method was adapted to the Laplace operator in [25] and could be suitable for more general diffusion operators. In contrast to the neutron transport problem, for which exact simulation schemes exist, the choice of a simulation scheme is crucial in this case. In a previous paper [25], we have also promoted the *random walk on squares* [30,6] and the *random walk on rectangles* [10] methods as the best ones for use on a polygonal domain, D.

We consider the Cauchy problem for the following evolution equation

$$\frac{\partial u(t,x)}{\partial t} = Au(t,x) + c(x)u(t,x) \text{ with } u(0,\cdot) = u_0, \tag{1}$$

in a bounded domain D, where A is a linear operator with absorption on ∂D, and c is a gain or loss factor (depending on its sign). In both cases, the solution u admits a Feynman-Kac representation

$$u(t,x) = \mathbb{E}_x\left[u_0(X_t)\exp\left(\int_0^t c(X_s)\mathrm{d}s\right)\mathbf{1}_{\tau>t}\right], \tag{2}$$

where $(X, \mathbb{P}_x)$ is the Markov process associated to the operator A and $\tau = \inf\{t \geqslant 0 | X_t \notin D\}$ is the first exit time of this process from D. For instance, if A is the Laplace operator, i.e. $A = \frac{1}{2}\triangle$, the process, X, is ordinary Brownian motion. The processes related to transport operators are described in Section 4. We consider only homogeneous neutron transport problems, that is c is constant, and we choose $u_0 = 1$. The solution to this special case of the transport equation is

$$u(t,x) = \exp(ct)\mathbb{P}_x[t < \tau], \tag{3}$$

so that the value of $u(t,x)$ can be deduced from the distribution function of the first exit time τ. In the case of the Laplace operator, we also let $u_0 = 1$ and we set $c = 0$, so that the solution is just $u(t,x) = \mathbb{P}_x[t < \tau]$.

From an analytical point of view, the operator A generates a semi-group which has a transition density $p(t,x,y)$ with respect to Lebesgue measure, and the solution, u, to (1) in the case that $c \neq 0$ may be written as

$$u(t,x) = \exp(ct)\int_D p(t,x,y)u_0(y)\mathrm{d}y.$$

In both cases, an indirect application of the Kreïn-Rutman theorem ([8, Appendix of Chap. VIII], [33]) implies that there exists an eigenvalue, λ_1, of the operator, A, such that any element, λ, of A's spectrum has a real part smaller than λ_1. In addition, this eigenvalue, λ_1, has multiplicity one and its associated eigenfunction does not vanish on the open domain D.

Throughout this paper, we assume that when c is constant, the solution $u(t,x)$ may be expanded as

$$u(t,x) = \langle\varphi_1^*, u_0\rangle\varphi_1(x)\exp((c + \lambda_1)t) + R(t,x) \tag{4}$$

with $R(t,x) = \mathrm{o}(\exp(\lambda_1 t))$, $\langle f, g\rangle = \int_D f(x)g(x)\mathrm{d}x$, φ_1^* is the first eigenfunction of the adjoint operator, A^*, of A such that $\varphi_1^* > 0$, $\langle\varphi_1^*, \varphi_1\rangle = 1$, and the initial condition, u_0, is in a reasonable space of functions that contain the constant functions on D. The expansion (4) holds for most of the neutron transport operators used in practice, for self-adjoint operators with a compact resolvent, such as the Laplace operator on a bounded domain, and more generally, for a large class of diffusion operators.

Using (3) and (4), the distribution function $F(t) = \mathbb{P}_x[\tau < t]$ of the first exit time τ of D can be then expanded as

$$F(t) = 1 - \mathbb{P}_x[t < \tau] = 1 - \exp(-ct)(\langle 1, \varphi_1^*\rangle\exp(\lambda_1 t)\varphi_1(x) + R(t,x)).$$

The idea now is to approximate $F(t)$ by an empirical distribution function obtained by simulating first exit times, τ. Therefore, we have developed some statistical methods to approximate λ_1 from this empirical distribution function.

The problems we have in estimating λ_1 in this way are the following. First, we need to estimate $F(t)$ for $t > T$ for T large enough that the approximation $1 - F(t) \simeq C \exp((\lambda_1 - c)t)$ is valid. On the other hand, as we use an empirical distribution function, $F_N(t)$, with N samples to estimate $F(t)$, the variance of $\log(1 - F_N(t))$ explodes as $t \to \infty$. A first idea is to estimate $F_N(t)$ at two times t_0 and t_1, and use the difference of the values at these times to estimate λ_1 [28,32]. Another possible approach, developed in [25], is to find a time window $[t_0, t_1]$, in which $F_N(t)$ is a good approximation of $F(t)$. A last approach is to note that for $t > t_0$ and $c = 0$, the exit time, τ, from D is distributed like an exponential random variable with parameter $-\lambda_1$. Standard estimators, like the maximum likelihood estimator, can then be used.

The purpose of this article is to improve the results of [32] and [25]. To achieve this improvement, we propose a variance reduction scheme for the empirical approximation of $F(t)$ which is very easy to implement. As a byproduct of this method, we can also estimate the first eigenfunction, φ_1^*, of the adjoint, A^*, of A. As the Laplace operator is self-adjoint, this is also the first eigenfunction of $\frac{1}{2}\triangle$. For the neutron transport operator, the adjoint is also a neutron transport operator, so that the first eigenvalue of A may be computed using our method on A^*.

This new method is based on a branching mechanism which has been used in many fields (see for example [11,12,5,24,26]). As we are interested in estimating the asymptotic behavior of $F(t)$ when t is large, we may restrict ourselves to estimating

$$\frac{F(t) - F(T)}{1 - F(T)} = \mathbb{P}_x[\tau < t | \tau \geqslant T],$$

for a fixed $T > 0$ and $t \geqslant T$. We assume that we know the distribution π_T of the stochastic process X_T starting at x. By the Markov property, for $t > T$,

$$\mathbb{P}_x[\tau < t | \tau \geqslant T] = \mathbb{P}_{\pi_T}[\tau < t] = \int_D \mathbb{P}_y[\tau < t] \mathrm{d}\pi_T(y).$$

The algorithm we use is the following. We fix $T > 0$ and we get compute the estimator $\widehat{\pi}_T$ of π_T using a Monte Carlo method, then we simulate the first exit time, τ, from D for the process X with $\widehat{\pi}_T$ as initial distribution and we compute λ_1 using the methods previously discussed in [32,25].

The number of particles we use to estimate the empirical distribution function of τ given $\{t > T\}$ is the same as the number of particles we use to estimate $\widehat{\pi}_T$. This approach compensates for the absorption of particles at the boundary. We may need also to estimate $\widehat{\pi}_{T_1}, \ldots, \widehat{\pi}_{T_N}$ at some times $T_1 < \ldots < T_N$, using a branching mechanism at each of these times in order to get a good approximation of $\widehat{\pi}_T$. Not only does this provides a much better approximation of $F(t)$ – up to multiplicative and additive constants – when t is large, but $\widehat{\pi}_T$ approximates the first eigenfunction of A^* when T is large enough.

This branching mechanism is similar to those used in quantum Monte Carlo methods with at least two main differences. First the number of branching times used in our method is very small (no more than 3 in the numerical experiments) and the branching mechanism is very simple. Then, we do not try to converge toward the principal eigenfunction of the operator but we obtain a direct approximation of the one of the adjoint operator.

With our approach, we do not need to have *a priori* information on the principal eigenfunction nor on the Green functions on the whole domain. Yet we need an accurate simulation of the dynamic of the particles especially when close to the boundary of the domain. To sum up, our branching mechanism is a numerical tool relying on the Markov property to handle rare events simulations for operators in bounded domains with no potential.

2. Estimating the principal eigenvalue and its associated eigenfunction

The idea of the algorithm is to launch N particles started at a given point x, following them up to a fixed time T_1, and to recording the positions of the particles that have not yet been absorbed at the boundary. Then, we again simulate N particles, this time using the empirical distribution of X_{T_1} given $\{\tau > T_1\}$ as the initial distribution. From the Markov property, the particles have a distribution close to $\mathbb{P}_x[\cdot | \tau > T_1]$. We can then use several time slices $T_2, \ldots, T_k$ to obtain a good approximation of the behavior of the particle given $\{\tau > T_k\}$.

2.1. The algorithm

Our algorithm is the following

- Fix some times $T_0 = 0 < T_1 < T_2 < \ldots < T_k$, a number, N, of samples and a point, $x \in D$. Set $\widehat{\pi}_{T_0} = \delta_x$.
- For i from 0 to $k - 1$ do
  - Using $\widehat{\pi}_{T_i}$ as the initial distribution, simulate N independent realizations $\{X^{(j)}\}_{j=1,\ldots,N}$ of $X_{(T_{i+1} - T_i) \wedge \tau}$, where τ is the first exit time from D.
  - Let $N(i)$ be the subset of $\{1, \ldots, N\}$ of random variables such that $X^{(j)}$ belongs to D. Set $\widehat{\pi}_{T_{i+1}} = \frac{1}{|N(i)|} \sum_{j \in N(i)} \delta_{X^{(j)}}$.

- Using $\widehat{\pi}_{T_k}$ as the initial distribution, simulate N realizations $\{t^{(j)}\}_{j=1,\ldots,N}$ of the first exit time τ from D.
- Estimate λ_1 from $\{t^{(j)}\}_{j=1,\ldots,N}$ and estimate φ_1^* from the the realizations $\{X^{(j)}\}_{j=1,\ldots,N(i)}$ of the position of $X_{T_{i+1}}$.

Of course, the quality of the result is sensitive to the choices of $T_1, \ldots, T_k, N$, and in a smaller way, to the starting point x. Yet, as we have shown in [25], the quality of the method used to simulate τ and X_t is one of the main concerns for the accuracy of the estimators.

2.2. How to choose the final time slice?

To get a good estimate of the principal eigenvalue, we should choose the times $T_1, \ldots, T_k$ appropriately, and T_k should be chosen to be large enough. As was already noted, the distribution of $(X_{t+T})_{t \geqslant 0}$ for T sufficiently large is essentially given by the principal eigenvalue and its associated eigenfunction. In particular, the density of X_{T_i} given $\{T_i < \tau\}$ and $X_{T_{i+1}}$ given $\{T_{i+1} < \tau\}$ tends to converge to the principal eigenfunction φ_1^* (normalized to be a probability density). One can then test the L^2-difference between two successive densities. A simpler criterion is obtained by setting

$$p_i = \mathbb{P}[\tau > T_{i+1} | \tau \geqslant T_i] \simeq \varphi_1(x)\langle 1, \varphi_1^* \rangle \exp(\lambda_1(T_{i+1} - T_i)),$$

when T_i is large, and choosing the first i such that p_i is close to p_{i+1}.

2.3. Estimating the principal eigenvalue from the empirical distribution function

We now present a few possible estimators for λ_1 that can be obtained from the simulated values of the first exit time τ.

(a) Interpolation method. This method is very simple. It was originally introduced by one of the author in his thesis [28,32] (see also [25]). Given two times t_0 and $t_1 > t_0$, we estimate $F(t_0)$ and $F(t_1)$ from a Monte Carlo simulation, which gives $\widehat{F}(t_0)$ and $\widehat{F}(t_1)$. If t_0 and t_1 are large enough, then

$$\lambda_{\mathrm{LI}}(t_0, t_1) = \frac{1}{t_1 - t_0} \log\left(\frac{\widehat{F}(t_1)}{\widehat{F}(t_0)}\right)$$

is an estimator for λ_1. One can also compute a confidence interval for λ_1 [25].

(b) Least square estimators. We construct the empirical distribution function $\widehat{F}(t)$ of $F(t)$ for t sufficiently large, and then estimate $\log(1 - \widehat{F}(t)) \simeq K + \lambda_1 t$ using a least square method.

The error between $\log(1 - \widehat{F}(t))$ and $\log(1 - F(t))$ is approximatively given by

$$\log(1 - \widehat{F}(t)) \simeq \log(1 - F(t)) + \frac{\eta \circ F^{-1}(t)}{\sqrt{N}(1 - F(t))},$$

where $(\eta(t))_{t \in [0,1]}$ is a Brownian bridge process. A consequence of this computation is that we take t in some interval $[t_0, t_1]$ with t_0 large enough so that the principal eigenvalue dominates the approximation of $F(t)$, and $t_1 > t_0$ not too large so that the variance of the last term is kept small.

Hence, we pick m time points $\{\theta_i\}_{i=1,\ldots,m}$ in $[t_0, t_1]$ and then we use $(\theta_i, \log(1 - \widehat{F}(\theta_i)))$ as the points to perform linear regression. Of course, this estimator depends on the choice of the θ_i. In [25], we discussed how to choose the best estimator when relatively few points are used (with respect to the number of bins of the histogram used to construct $\widehat{F}$). Another possibility consists of using a linear interpolation of the discrete values of the function $\widehat{F}$ and then using many points $\{\theta_i\}_{i=1,\ldots,m}$. If we pick m random points $\{\theta_i\}_{i=1,\ldots,m}$ in $[t_0, t_1]$, then the least squares estimator is very stable with respect to the choice of the points, when m is large (in our numerical example, we construct our histograms with 1000 bins and $m = 10,000$). Note that the variability in the estimator as a function of the choice of $[t_0, t_1]$ is greater than the variability given by the confidence interval for the slope of the curve in the linear regression. The quality of the estimator may be deduced from the quantity $1 - R^2$, where R^2 is the coefficient of determination.

(c) Maximum likelihood. For T sufficiently large, $\mathbb{P}_x[\tau > t | \tau > T] \simeq C \exp(\lambda_1 t)$. Thus τ is an exponential random variable with parameter $-\lambda_1$ (see [2]). The probability density of τ given $\{\tau > T\}$ is $p(t, \lambda) = -\lambda \exp(-\lambda(t - T))$ with $\lambda = \lambda_1$. Hence, it is possible to use the standard estimators of the parameter of an exponential distribution. A natural estimator of λ_1 is the maximum likelihood estimator, i.e., the value λ_{ML} which maximizes $\lambda \mapsto \prod_{i=1}^{M} p(\tau^{(i)}, \lambda)$, where $\{\tau^{(i)}\}_{i=1,\ldots,M}$ are the values of τ greater than T, that is

$$\lambda_{\mathrm{ML}} = -\frac{M}{\sum_{i=1}^{M}(\tau^{(i)} - T)}.$$

It is also a classical result that such estimators are asymptotically normal. In addition, the variance of this estimator is known to be related to the Fisher information $I(\lambda_1)$ of the exponential distribution. This means that $\sqrt{M}(\lambda_{\mathrm{ML}} - \lambda_1)$ converges to a normal distribution of mean 0 and variance $1/I(\lambda_1)$ with

$$I(\lambda) = \int_{T}^{+\infty} \frac{(\partial_\lambda p(t, \lambda))^2}{p(t, \lambda)} \, \mathrm{d}t = \frac{1}{\lambda^2}.$$

On this topic, see for example [39].

(d) Other possible estimators. Other estimators have been proposed to estimate the parameter of an exponential distribution: see for example [35,15]. In the previously cited articles, the proposed estimators are robust and thus less sensitive to the presence of outliers than the maximum likelihood estimator.

2.4. Estimating the principal eigenfunction

When $c = 0$, comparing (2) and (4), respectively with a general bounded, measurable function u_0 and with the function $u_0 = 1$ leads to

$$\mathbb{E}_x[u_0(X_t)|t < \tau] = \frac{\mathbb{E}_x[u_0(X_t); t < \tau]}{\mathbb{E}_x[1; t < \tau]} = \frac{\langle u_0, \varphi_1^* \rangle \varphi_1(x) \exp(\lambda_1 t) + o(\exp(\lambda_1 t))}{\langle 1, \varphi_1^* \rangle \varphi_1(x) \exp(\lambda_1 t) + o(\exp(\lambda_1 t))} \simeq \frac{\langle u_0, \varphi_1^* \rangle}{\langle 1, \varphi_1^* \rangle}$$

when the time t is sufficiently large. Thus, the density of the position, X_t, given $\{t < \tau\}$ is $\varphi_1^*/\langle 1, \varphi_1^* \rangle$ for large t, assuming that $\varphi_1^* > 0$ is in D (one knows that φ_1^* is of one sign on D).

The simplest way to estimate the principal eigenvalue φ_1^* of the adjoint, A^*, of A is to construct an histogram of the positions X_t at a given sufficiently large time, t, for a sample of the surviving particles at this time. A less trivial way is to construct φ_1^* as a superposition of a distribution density — a kernel — around each simulated point (see [37] for example). This gives one a more regular density. In the numerical examples of Sections 3 and 4, we show that we can obtain good approximations of this eigenfunction using each of the two methods.

With λ_1, φ_1 and φ_1^*, we have a complete description of the solution of the Cauchy problem at large times for every initial functions and any point, as can be seen in the expansion (4). If the function, φ_1, is not completely known, we at least know the value of φ_1 at the point where the simulation starts. We can then estimate the solution to the Cauchy problem (4) at this point for any initial condition at large time. Our algorithm allows us to estimate φ_1^*. In some cases, if A is self-adjoint or if the domain and the operator exhibit some symmetries, one can deduce φ_1 directly from φ_1^*. More generally, for a wide class of operators, it is possible to apply our algorithm to the adjoint of A so we estimate φ_1.

2.4.1. On the adjoint of the homogeneous neutron transport operator

The neutron transport operator is a member of a particular class of transport operators that includes particle collision, more specifically they are of the form

$$Au(x, v) = \sum_{i=1}^{d} v_i \frac{\partial u}{\partial x_i}(x, v) + v \int_{\mathbf{V}} \pi(x, v, v')(u(x, v') - u(x, v)) \mathrm{d}v',$$

where $(x, v) \in D = \mathbf{S} \times \mathbf{V} \subset \mathbb{R}^d \times \mathbb{R}^d$, $v \in \mathbb{R}_+^*$ and $\pi(x, v, \cdot)$ is a distribution function on $\mathbf{V}$ for any $(s, v) \in D$. We can interpret this via the simulation of a particle with position X_t and velocity V_t. When a collision occurs at time t, the new velocity V_{t+} of the particle is chosen randomly using the density $\pi(X_t, V_t, \cdot)$. The particle then moves with constant velocity until the next collision that at an exponentially time with mean v.

Formally, the adjoint A^* of A is

$$A^*u(x, v) = -\sum_{i=1}^{d} v_i \frac{\partial u}{\partial x_i}(x, v) + v \int_{\mathbf{V}} \pi(x, v', v)(u(x, v') - u(x, v)) \mathrm{d}v'.$$

If $\pi(x, \cdot, v)$ is a probability density distribution, then one easily obtains that the solution to

$$\frac{\partial u(t, x, v)}{\partial t} = A^*u(t, x, v) \text{ on } D \text{ with } u(0, x, v) = u_0(x, v)$$

is equal to $\widehat{u}(t, x, -v) = u(t, x, v)$, where $\widehat{u}$ is solution to

$$\frac{\partial \widehat{u}(t, x, v)}{\partial t} = \widehat{A}\widehat{u}(t, x, v) \text{ on } \widehat{D} \text{ with } \widehat{u}(0, x, v) = u(x, -v).$$

Here $\widehat{D} = \mathbf{S} \times (-\mathbf{V})$ and

$$\widehat{A}u(x, v) = \sum_{i=1}^{d} v_i \frac{\partial u}{\partial x_i}(x, v) + v \int_{-\mathbf{V}} \widehat{\pi}(x, v, v')(u(x, v') - u(x, v)) \mathrm{d}v', \widehat{\pi}(x, v, v') = \pi(x, -v', -v), \ (v, v') \in (-\mathbf{V})^2.$$

Hence, the principal eigenfunction φ_1 of A is also the principal eigenfunction of the adjoint, which may then be deduced from the principal eigenfunction $\widehat{\varphi}_1$ of $\widehat{A}$ by $\varphi_1(x, v) = \widehat{\varphi}_1(x, -v)$.

Thus, under the assumption that $\pi(x, v, \cdot)$ and $\pi(x, \cdot, v)$ are probability densities – this hypothesis is really practical –, one can use our algorithm on $\widehat{A}$ to get the principal eigenfunction, φ_1, of A, as well as its associated eigenvalue. As we will see in examples, there are realistic cases where one can deduce the principal eigenfunction of A from that of A^* using symmetry arguments.

The principal eigenvalue of $\widehat{A}$ is equal to the principal eigenvalue of A, so that taking the average of the two estimators for these principal eigenvalues gives a slightly better approximation of these quantities.

2.4.2. On diffusion processes

Of course, if A is the Laplace operator, $\frac{1}{2}\triangle$, then $\varphi_1^* = \varphi_1$ since A is self-adjoint. Thus, our algorithm directly gives us the principal eigenfunction of A.

We also claimed in [25], that our approach can be used for a more general diffusion process whose infinitesimal generator is

$$A = \frac{1}{2} \sum_{i,j=1}^d a_{ij} \frac{\partial^2}{\partial x_i \partial x_j} + \sum_{i=1}^d b_i \frac{\partial}{\partial x_i}.$$

Although A is not generally self-adjoint, under mild regularity assumptions on the coefficients and the domain, it has a discrete spectrum and there exists a real eigenvalue λ_1 such that any other (possibly complex) eigenvalue has a smaller real part; see [33] for example. In order to compute the principal eigenvalue of A, an appropriate simulation scheme will be used. The problem of obtaining a good approximation of the first exit time from a domain has given rise to a large literature; see [30,16,20,3] for example.

The adjoint A^* of A is given by

$$A^* = \frac{1}{2} \sum_{i,j=1}^d \frac{\partial^2}{\partial x_i \partial x_j} (a_{ij} \cdot) - \sum_{i=1}^d \frac{\partial}{\partial x_i} (b_i \cdot),$$

and our algorithm directly computes the principal eigenfunction, φ_1^*, of A^*. The principal eigenvalue of A^* is also λ_1. If one wishes to compute the principal eigenfunction, φ_1 of A, one can use that $A^{**} = A$ and thus the principal eigenfunction of the adjoint of A^* is φ_1. If the coefficients, a and b, are smooth enough, this operator A^* may be written as

$$A^* = \frac{1}{2} \sum_{i,j=1}^d a_{ij} \frac{\partial^2}{\partial x_i \partial x_j} + \sum_{i=1}^d \beta_i \frac{\partial}{\partial x_i} + \gamma,$$

with

$$\beta_i = \frac{1}{2} \sum_{j=1}^d \frac{\partial a_{ij}}{\partial x_i} - b_i \text{ and } \gamma = \frac{1}{2} \sum_{i,j=1}^d \frac{\partial^2 a_{ij}}{\partial x_i \partial x_j} - \sum_{i=1}^d \frac{\partial b_i}{\partial x_i}.$$

Note: if the coefficients are constant, then $\gamma = 0$, and so φ_1 may be computed by simulating the process associated with A^*.

We now deal with the case of non-constant γ. The approach presented here can also be used to deal with a non-homogeneous creation/destruction rate, c. Let L be the differential operator

$$L = \frac{1}{2} \sum_{i,j=1}^d a_{ij} \frac{\partial^2}{\partial x_i \partial x_j} + \sum_{i=1}^d \beta_i \frac{\partial}{\partial x_i},$$

so that $A^* = L + \gamma$. Let us assume that γ is bounded by a constant, α, and let v be the solution to

$$\frac{\partial v}{\partial t} = Lv + \gamma v - \alpha v \text{ with } v(0, \cdot) = v_0,$$

with a Dirichlet boundary condition on the boundary of the cylinder, $\mathbb{R}_+ \times \partial D$. The solution, v, may be represented by the Feynman-Kac formula

$$v(t,x) = \mathbb{E}_x \left[v_0(Y_t) \exp \left(\int_0^t \gamma(Y_s) \mathrm{d}s - \alpha t \right); t < \tau \right]$$

where Y is the process generated by L. Let ζ be an exponential random variable with parameter 1, then $v(t,x)$ may be written also

$$v(t,x) = \mathbb{E}_x \left[v_0(Y_t); \int_0^t \gamma(Y_s) \mathrm{d}s - \alpha t > \zeta \text{ and } t < \tau \right]. \tag{5}$$

On the other hand

$$v(t,x) \simeq \exp(\lambda_1^* t) \langle v_0, \psi_1^* \rangle \psi_1(x) \text{ when } t \text{ islarge}, \tag{6}$$

where λ_1^* is the principal eigenvalue of $A^* - \gamma = L^* - \gamma$ and ψ_1 is its associated eigenfunction with $\langle \psi_1, \psi_1^* \rangle = 1$. We note that $\psi_1 = \varphi_1^*, \psi_1^* = \varphi_1$ and $\lambda_1^* = \lambda_1 - \alpha$. As above, we obtain from (5) and (6) that

$$\frac{\langle v_0, \varphi_1 \rangle}{\langle 1, \varphi_1 \rangle} \simeq \mathbb{P}_x \left[v_0(Y_t) \int_0^t \gamma(Y_s) \mathrm{d}s - \alpha t > \zeta \text{ and } t < \tau \right]$$

when t is large. Hence, a branching Monte Carlo method may still be used. But here, unless $\gamma = 0$, one also needs to compute the integral $\int_0^{t \wedge \tau} \gamma(Y_s) \mathrm{d}s$ along the simulated Y paths.

Computing numerically the principal eigenvalue λ_1^*, while estimating φ_1 also helps us to improve slightly the estimation of λ_1.

2.5. Estimating the second eigenvalue of the Laplace operator?

As the Laplace operator with Dirichlet boundary conditions is self-adjoint, the spectrum, $\{\lambda_k\}_{k \geqslant 1}$, of $\frac{1}{2}\triangle$ is countable, real and negative. In addition,

$$p(t, x, y) = \sum_{k \geqslant 1} \exp(\lambda_k t)\varphi_k(x)\varphi_k(y), \ t > 0, \ x, y \in D, \tag{7}$$

where we use the convention $\dots b \leqslant \lambda_3 \leqslant \lambda_2 < \lambda_1 < 0$, and the φ_k are the normalized eigenfunctions associated with λ_k. The distribution function, $F(t) = \mathbb{P}_x[\tau < t]$, is given by the relation

$$1 - F(t) = \mathbb{E}_x[1; t < \tau] = \int_D p(t, x, y)\mathrm{d}y = \sum_{k \geqslant 1} \mathrm{e}^{-\lambda_k t}\varphi_k(x)\int_D \varphi_k(y)\mathrm{d}y.$$

One may wonder whether or not it is possible to estimate – at least roughly – the second eigenvalue with this algorithm, as the density may be written

$$p(t, x, y) = \mathrm{e}^{\lambda_1 t}\varphi_1(x)\varphi_1(y) + \mathrm{e}^{\lambda_2 t}\varphi_2(x)\varphi_2(y) + r(t, x, y),$$

where $\mathrm{e}^{-\lambda_2 t}r(t, x, y)$ decreases to 0, and both λ_1 and φ_1 are estimated using the previous algorithm. Two methods appear to be natural.

(a) Subtract from $F_N(t)$ the estimate of the quantity $1 - \exp(\lambda_1 t)\varphi_1(x)\langle\varphi_1, 1\rangle$ and estimate $\varphi_2\langle\varphi_2, 1\rangle \exp(\lambda_2 t)$ in the same way the λ_1 was estimated.

Instead of starting from the point x, one may also look for a probability measure, μ, on D such that $\int_D \varphi_1(x)\mathrm{d}\mu(x)$ is as small as possible. This is justified by

$$\mathbb{P}_\mu[T < \tau] = \int_D \int_D p(t, x, y)\mathrm{d}y\mathrm{d}\mu(x) \simeq \left(\int_D \varphi_1\mathrm{d}\mu\right)\langle\varphi_1, 1\rangle\exp(\lambda_1 t) + \left(\int_D \varphi_2\mathrm{d}\mu\right)\langle\varphi_2, 1\rangle\exp(\lambda_2 t).$$

The effect is to increase the relative importance of the second eigenvalue when approximating $\mathbb{P}_\mu[T < \tau]$.

(b) Create a function, $\varphi_1^\perp$, that is orthogonal with respect to the $L^2(D)$ scalar product. If $\|\varphi_1\|_{L^2(D)} = 1$, then $\varphi_1^\perp = 1 - (\int_D \varphi_1(y)\mathrm{d}y)\varphi_1$ is such a function. Then evaluate the quantities

$$v_i = \mathbb{E}_x[\varphi_1^\perp(X_{T_i}); T_i < \tau] \simeq \exp(\lambda_2 T_i)\varphi_2(x)\langle\varphi_2, \varphi_1^\perp\rangle$$

for $i = 1, 2$, where T_1 and T_2 are two times that are not too large, then set $\lambda_2 = (T_1 - T_2)^{-1}\log(v_1/v_2)$.

On the test cases of Section 3, we have performed numerical experiments for both these methods. Unfortunately, none of these methods has provided a stable enough estimator of λ_2.

3. Numerical examples for the Laplace operator

In this Section, we give two numerical examples related to Laplace operators. The first test case is has D as just a rectangle, where the eigenvalues and the eigenfunctions are explicitly known. This case gives us the inherent limit of the implementation of the Monte Carlo method. One cannot expect to get a better precision for a general case than for this case with the same number of samples. The second one has a slightly more complicated geometry, and has been already studied in [25].

In the sequel, we call λ_{ML} the maximum likelihood estimator. The width of the 90% confidence interval is $2\lambda_1\sqrt{1.64}/\sqrt{M}$, where M is the number of samples used to compute the maximum likelihood estimator. We use $\lambda_{\mathrm{LS}}(t_0, t_1)$ for the least square estimator on the time interval $[t_0, t_1]$, and a large set of points ($m = 10,000$).

3.1. Case of a rectangle

The eigenvalues and eigenfunctions of $\frac{1}{2}\triangle$ are explicitly known when D is the rectangle $[-L, L] \times [-\ell, \ell]$. We have, for any integers $n, m \geqslant 1$,

$$\lambda_{n,m} = -\frac{1}{2}\left(\left(\frac{n\pi}{2L}\right)^2 + \left(\frac{m\pi}{2\ell}\right)^2\right) \ \text{and} \ \varphi_{n,m}(x, y) = \sin\left(\frac{n\pi}{2L}(x + L)\right)\sin\left(\frac{n\pi}{2\ell}(y + \ell)\right).$$

We consider the rectangle $D = [-2, 2] \times [-3/2, 3/2]$, for which the principal eigenvalue $\lambda_{1,1}$ is -0.856735. In order to study how to optimize the estimation of $\lambda_{1,1}$, we perform five scenarios with N particles each. The results are summarized in Table 1. Except in Case (b), the particles start from the center of the rectangle in order to have a large number of particles that survive to time T_1.

We use several slices at times T_i, and we compute the empirical distribution function, ψ_i, of X_{T_i} obtained from the samples. To construct these functions, we use an histogram of cells of size 0.01×0.01. The functions ψ_i are normalized to have a

Table 1
Estimation of the principal eigenvalue of the rectangle using a sample size of N

Estimator	Value	$1 - R^2$
Exact value	-0.856735	
Simulation (a) $N = 10^6, \theta = 1$ units		
λ_{ML} $(\tau \geqslant 2)$	$-0.8525 \pm 2.5 \times 10^{-3}$	
λ_{LS} $(2, 4)$	-0.8527	2×10^{-6}
λ_{LS} $(2, 6)$	-0.8518	3×10^{-6}
λ_{LS} $(2, 8)$	-0.8521	2×10^{-5}
Simulation (a) $N = 10^7, \theta = 10$ units		
λ_{ML} $(\tau \geqslant 2)$	$-0.8554 \pm 8 \times 10^{-4}$	
λ_{LS} $(2, 4)$	-0.8553	5×10^{-7}
λ_{LS} $(2, 6)$	-0.8569	8×10^{-6}
λ_{LS} $(2, 8)$	-0.8580	6×10^{-6}
Simulation (b) $N = 10^6, \theta = 3.5$ units		
λ_{ML} $(\tau \geqslant 0)$	$-0.8570 \pm 1.1 \times 10^{-3}$	
λ_{LS} $(0, 2)$	-0.8571	2×10^{-6}
λ_{LS} $(0, 4)$	-0.8588	4×10^{-6}
λ_{LS} $(0, 6)$	-0.8564	7×10^{-6}
λ_{LS} $(0, 8)$	-0.8556	15×10^{-6}
Simulation (c) $N = 10^6, \theta = 3$ units		
λ_{ML} $(\tau \geqslant 2)$	-0.8564 ± 10^{-3}	
λ_{LS} $(2, 4)$	-0.8568	6×10^{-6}
λ_{LS} $(2, 6)$	-0.8571	7×10^{-7}
λ_{LS} $(2, 8)$	-0.8615	6×10^{-5}
Simulation (d) $N = 10^6, \theta = 4.5$ units		
λ_{ML} $(\tau \geqslant 4)$	-0.8577 ± 10^{-3}	
λ_{LS} $(4, 6)$	-0.8567	1.2×10^{-6}
λ_{LS} $(4, 8)$	-0.8577	1.4×10^{-6}
λ_{LS} $(4, 10)$	-0.8549	7.0×10^{-6}
Simulation (e) $N = 10^6, \theta = 6$ units		
λ_{ML} $(\tau \geqslant 6)$	-0.8564 ± 10^{-3}	
λ_{LS} $(6, 8)$	-0.8573	4×10^{-7}
λ_{LS} $(6, 10)$	-0.8565	9×10^{-7}
λ_{LS} $(6, 12)$	-0.8569	2.6×10^{-6}

The quantity θ gives the relative execution time.

L^2-norm equal to 1. In all cases, the L^2-norm of the difference between ψ_i and the adequately normalized eigenfunction $\varphi_{1,1}$ is around 0.5.

   Simulation (a). Here we do not use the branching algorithm, so we only keep the first exit times that are larger than 2.

   Simulation (b). In order to have a comparison between the use of the empirical distribution function for X_T for T sufficiently large, instead of the density $\varphi_{1,1}/\int_D \varphi_{1,1}$, we randomly pick the starting point according to the probability distribution, μ, with density $\widehat{\varphi} = \varphi_{1,1}/\int_D \varphi_{1,1}$. For μ, since the eigenfunctions are orthogonal, we note that $\mathbb{P}_\mu[\tau > t] = \exp(\lambda_1 t)$, so that τ is an exponential random variable with parameter $-\lambda_1$.

   Simulation (c). Here we use only one time slice at $T_1 = 2$. The probability that the particle survived to time T_1 is $p_1 = 19.2\%$.

   Simulation (d). We use the time slices $T_1 = 2$ and $T_2 = 4$. The probabilities of survival, p_i, at times T_i are $p_1 = 19.2\%$, and $p_2 = 5.2\%$.
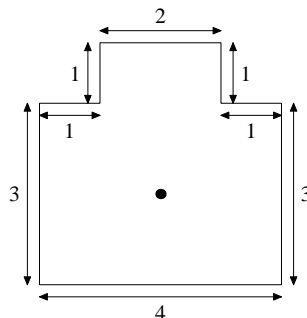

Fig. 1. A 2-dimensional domain. The dot represents the starting point.

Table 2
Estimation of the principal eigenvalue of the 2d-test case using a sample size, N

Estimator	Value	$1 - R^2$
Reference value	-0.73952	
Simulation (a) $N = 10^6, \theta = 1$ *units*		
λ_{ML} $(\tau \geqslant 2)$	$-0.7373 \pm 1.5 \times 10^{-3}$	
λ_{LS} $(2,4)$	-0.7403	2.0×10^{-5}
λ_{LS} $(2,6)$	-0.7381	1.1×10^{-5}
λ_{LS} $(2,8)$	-0.7378	1.5×10^{-5}
Simulation (a) $N = 10^7, \theta = 10$ *units*		
λ_{ML} $(\tau \geqslant 2)$	$-0.7374 \pm 0.5 \times 10^{-3}$	
λ_{LS} $(2,4)$	-0.7397	2.7×10^{-6}
λ_{LS} $(2,6)$	-0.7387	0.8×10^{-6}
λ_{LS} $(2,8)$	-0.7367	5.5×10^{-6}
Simulation (b) $N = 10^6, \theta = 2.5$ *units*		
λ_{ML} $(\tau \geqslant 2)$	$-0.7378 \pm 0.9 \times 10^{-3}$	
λ_{ML} $(\tau \geqslant 3)$	$-0.7373 \pm 1.3 \times 10^{-3}$	
λ_{LS} $(2,4)$	-0.7366	1.0×10^{-6}
λ_{LS} $(2,6)$	-0.7378	7×10^{-7}
λ_{LS} $(3.5,5.5)$	-0.7390	8×10^{-7}
λ_{LS} $(2,8)$	-0.7392	1.2×10^{-6}
Simulation (c) $N = 10^6, \theta = 4.2$ *units*		
λ_{ML} $(\tau \geqslant 4)$	$-0.7389 \pm 0.9 \times 10^{-3}$	
λ_{ML} $(\tau \geqslant 5)$	$-0.7389 \pm 1.3 \times 10^{-3}$	
λ_{LS} $(4,6)$	-0.7396	1.9×10^{-6}
λ_{LS} $(4,8)$	-0.7391	1.5×10^{-6}
λ_{LS} $(4,10)$	-0.7375	3.5×10^{-6}

The quantity θ gives the relative execution time.

Simulation (e). We use the time slices $T_1 = 2, T_2 = 4$ and $T_3 = 6$. The probabilities of survival are $p_1 = 19.2\%$, $p_2 = 5.2\%$, and $p_3 = 0.9\%$.

3.2. A 2-dimensional test case

Our estimation algorithm was presented in [25] for a 2-dimensional test case, which is the domain presented in Fig. 1. The results are then compared with the ones obtained with the pdetool package from Matlab. Using a very fine mesh, this deterministic solver gives the value $\lambda_1 = 0.73952$. Our numerical results are given in Table 2.

Simulation (a). Here, there is no branching. For the maximum likelihood estimator, we keep only the values of τ that are greater than 2, which means that we use only 36% of the particles.

Simulation (b). We use only a single time slice at $T = 2$. Regarding the principal eigenvalue, we compute φ_1 at time T, with an histogram with square cell 0.05×0.05, which we compare with the eigenfunction given by Matlab. The L^2-norm of the difference between these two functions is 0.1.

Simulation (c). We use two time slices at $T_1 = 2$ and $T_2 = 4$. The proportion of particles remaining at the first slice is $p_1 = 36\%$, and at the second slice it is $p_2 = 8\%$. The L^2-norm of the difference between the eigenfunction given by the Monte Carlo Method at time $T = 4$ and the one given by the finite-element method is 0.1.

Remark 1. We note that with (7), if one knows λ_1 and ψ_1, then one can approximate the density, $p(t, x, y)$, of the Laplace operator as $p(t, x, y) = \exp(\lambda_1 t)\varphi_1(x)\varphi_1(y)$ for t sufficiently large. This gives us a large time approximation of the solution of the Cauchy problems

$$\frac{\partial u(t, x)}{\partial t} = \frac{1}{2} \triangle u(t, x) \text{ with } u(0, x) = u_0(x)$$

for any function u_0, since $u(t, x) = \int_D p(t, x, y) u_0(y) \mathrm{d}y$.

4. Numerical examples with neutron transport operators

4.1. The Lehner-Wing model

4.1.1. Description and stochastic representation

We study the Cauchy problem

$$\frac{\partial u}{\partial t} = -v \frac{\partial u}{\partial x} - u(t, x, v) + \frac{c}{2} \int_V u(t, x, v') \mathrm{d}v',$$

with initial conditions $u(x, v, 0) = 1$, and absorbing boundary conditions. The spatial domain is $\mathbf{S} = (0, d)$, the velocity domain is $\mathbf{V} = (-1, 1)$, and c is a positive constant. This model is homogeneous and isotropic and we rewrite it as

$$\frac{\partial u}{\partial t} = Au(t, x, v) + (c - 1)u(t, x, v), \tag{8}$$

with

$$Au = -v\frac{\partial u}{\partial x} + c\left\{\frac{1}{2}\int_V u(t, x, v')\mathrm{d}v' - u(t, x, v)\right\},$$

to obtain the stochastic representation of its solution. Let us consider the velocity $(V_t)_{t\geqslant 0}$, of a particle with collisions that occur at random times. After a collision, the velocity is chosen from a uniform distribution on $\mathbf{V}$. The cumulative distribution of the time between two collisions is

$$1 - \exp\left[-\int_0^t c\mathrm{d}s\right] = 1 - \exp(-ct).$$

The process we consider now is a solution $(X_t, V_t)_{t\geqslant 0}$ of the differential equation $\frac{\mathrm{d}X_t}{\mathrm{d}t} = -V_t$ with initial conditions $X_0 = x$ and $V_0 = v$. The infinitesimal generator of this process $(X_t, V_t)_{t\geqslant 0}$, is A. The solution of (8) may thus be written as

$$u(t, x, v) = \exp((c - 1)t)\mathbb{P}_{x,v}[\tau > t],$$

where τ is the exit time from $D = \mathbf{S} \times \mathbf{V}$, for the process $(X_t, V_t)_{t\geqslant 0}$, with $X_0 = x$ and $V_0 = v$ under $\mathbb{P}_{x,v}$.

This model is known as the *Lehner-Wing model* [9, Chap. 21, p. 1164], and is also called *multiplying slabs* [7].

There are two kinds of eigenvalue problems in neutron transport; the first is the criticality computation. In our problem, it consists in finding the value of the parameter, c, such that the principal eigenvalue of the operator $Bu = Au + (c - 1)u$ is equal to 0. This means that the principal eigenvalue λ_1 of A is equal to $1 - c$, since the principal eigenvalue of B is $\lambda_1 + c - 1$. The second kind of problem is the computation of this eigenvalue for a given value of c.

We first assume that we have a very good approximation of the parameter, c, corresponding to the critical value. In this situation, we have to check that the principal eigenvalue is fairly close to 0, using our estimator. Second, we take two values of c below and above the critical value and compute the corresponding principal eigenvalues; we then compute the critical parameter using the secant method.

4.1.2. In the critical case

As our next numerical example, we simulate $(X_t, V_t)_{t\geqslant 0}$ when the spatial domain $\mathbf{S}$ is $(0, 8)$. The value of the critical parameter here is $c = 1.03639014$ (see [7]).

In our branching algorithms, we use slices at times $T_1 = 40$, $T_2 = 80$, and $T_3 = 120$. In order to estimate the principal eigenvalue of A, we use the interpolation estimator $\lambda_{\mathrm{LI}}(t_0, t_1)$ with or without branching using the values of $F(t_0) = \mathbb{P}_{x,v}[\tau < t_0]$, and $F(t_1) = \mathbb{P}_{x,v}[\tau < t_1]$ (see Section 2.3). The probabilities $p_i = 1 - F(T_i)$ tell us that the particles are still alive at time T_i with probabilities $p_1 = 30.7\%, p_2 = 7.2\%$ and $p_3 = 1.7\%$. Note that the ratios $p_2/p_1 = 0.2333$ and $p_3/p_2 = 0.2334$ are very close. This confirms that the system has reached the steady-state after T_1. When the successive ratios p_{i+1}/p_i become stationary, the behavior of the system is dominated by the principal eigenvalue and eigenfunction. In Table 3, we do not report the principal eigenvalue λ_1 of A, but the principal eigenvalue of $\alpha = c - 1 + \lambda_1$ of B, which will be close to 0.

4.1.3. Computation of the criticality factor

We will use the secant method to compute the criticality factor. In Table 4, we compute the values of $\alpha(c) = c - 1 + \lambda_1(c)$ for $c = c_{\min} = 1.036$, and $c = c_{\max} = 1.037$ using the previous method, assuming for the sake of simplicity that we are already near criticality. The estimator we use for λ_1 is still the one given by linear interpolation, and we set $\alpha_{\mathrm{LI}}(t_0, t_1) = c - 1 + \lambda_{\mathrm{LI}}(t_0, t_1)$.

The approximation of the criticality factor is then

$$\widehat{c} = c_{\min} - \alpha_{\min}\frac{c_{\max} - c_{\min}}{\alpha_{\max} - \alpha_{\min}} = 1.036406,$$

Table 3
Estimation of the principal eigenvalue α of B for the critical value of c with N particles

N	With branching		Without branching	
	10^6	10^7	10^6	10^7
$\alpha_{\mathrm{LI}}(T_1, T_2)$	2.2×10^{-5}	1.0×10^{-5}	6.5×10^{-5}	0.2×10^{-5}
$\alpha_{\mathrm{LI}}(T_2, T_3)$	-3.3×10^{-5}	1.4×10^{-5}	1.6×10^{-4}	9.0×10^{-5}
$\alpha_{\mathrm{LI}}(T_1, T_3)$	-5.4×10^{-6}	1.2×10^{-5}	1.1×10^{-4}	5.1×10^{-5}
Error max	3.3×10^{-5}	1.1×10^{-5}	1.6×10^{-4}	9×10^{-5}
Relative Time	2	20	1	10

The starting point is $(x, v) = (4, -0.2)$.

Table 4
Estimation of the principal eigenvalue α of B for c close to the critical value with N particles

| N | $c = c_{\min} = 1.036$ | | $c = c_{\max} = 1.037$ | |
	10^6	10^7	10^6	10^7
α_{LI} (T_1, T_2)	−0.000435	−0.000415	0.000633	0.000610
α_{LI} (T_2, T_3)	−0.00423	−0.000430	0.000564	0.00062
α_{LI} (T_1, T_3)	−0.000429	−0.00042	0.000598	0.000615

The starting point is $(x, v) = (4, -0.2)$.

where $\alpha_{\min} = -0.000421$ and $\alpha_{\max} = 0.00615$ are obtained by averaging the 3 estimators of Table 4 using $N = 10^7$ particles. The value of $\widehat{c}$ is close to $1.5 \cdot 10^{-5}$ of the true one. In a previous paper [32], this level of accuracy required $N = 10^9$ simulations.

4.1.4. Estimation of the solution of the Cauchy problem at large time

We now show how the estimation of the principal eigenfunction provides a complete description of the solution to (8) at large time. Table 5

Using the spectral expansion of the solution $u(t, x, v)$ to (8) with the initial condition $u(0, x, v) = 1$ we get

$$u(t, x, v) = \exp((c-1)t)(\beta(x, v)\exp(\lambda_1 t) + o(\exp(\lambda_1 t))), \tag{9}$$

where λ_1 is the principal eigenvalue of the neutron transport operator, A, and $\beta(x, v) = \langle 1, \varphi_1^* \rangle \varphi_1(x, v)$. As $u(t, x, v) = \exp((c-1)t)\mathbb{P}_{x,v}[t < \tau]$, λ_1 is related to the rate of absorption of the particles by the boundary. The function, $\beta(x, v)$, is equal to $\langle 1, \varphi_1^* \rangle \varphi_1(x, v)$, where φ_1 and φ_1^* denote the principal eigenfunctions of A and A^* with $\langle \varphi_1^*, \varphi_1 \rangle = 1$ (see for example Chap. XXI, §3 in [9]). The interpolation method, or the least squares method, also give the value of $\beta_0 = \beta(x_0, v_0)$ at the starting point (x_0, v_0) of the particles, which can also be deduced from $\beta_0 \exp(\lambda_1 t) = \mathbb{P}_{x_0, v_0}[t < \tau]$. With the branching method, we can approximate

$$\psi_1^*(x, v) = \frac{\varphi_1^*(x, v)}{\langle 1, \varphi_1^* \rangle}$$

with the density, (X_T, V_T), for T sufficiently large. We set $\psi_1(x, v) = \varphi_1(x, v)\langle 1, \varphi_1^* \rangle$. Using this expression for $\beta(x, v)$, the solution to (9) with $u(0, \cdot, \cdot) = u_0$ becomes

$$u(t, x, v) = \langle u_0, \psi_1^* \rangle \exp((c-1-\lambda_1)t)\psi_1(v, x) + o(\exp((c-1-\lambda_1)t)),$$

where $\psi_1^*(x, v)$ can be estimated from the simulations. To estimate $\psi_1^*(x, v)$ and β_0, one can instead estimate $\psi_1(x_0, v_0)$ at the starting point (x_0, v_0). Using the symmetry properties of the coefficients of the neutron transport operator and the symmetries of the domain, we get that $\psi_1(x, v) = \psi_1^*(d - x, -v)$. We give a numerical illustration of this in the critical case above with 10^7 particles. For $\beta_0 = \beta(x_0, v_0)$ with $(x_0, v_0) = (4, -0.2)$, using the times T_2 and T_3, we obtain $\beta_0 \simeq 1.316$. We now estimate ψ_1 – the density of (X_T, V_T) at time $T = 80$ – from the positions $\left(X_T^{(i)}, V_T^{(i)}\right)$ of the J_0 particles remaining at this time. We then use a convolution kernel so that

$$\psi_1(x, v) = \frac{1}{2\pi J_0 h^2} \sum_{i=1}^{J_0} \exp\left(-\frac{(x - X^{(i)})^2}{2h^2}\right) \exp\left(-\frac{(v - V^{(i)})^2}{2h^2}\right).$$

With $h = 0.1$, we compute $\psi_1(4, -0.2)$ and obtain 0.0883, which leads to an approximation of $K = \langle 1, \psi_1^* \rangle = 14.9$. Thus, in the critical case,

$$u(t, x, v) = K\psi_1(x, v) + o(\exp((c-1+\lambda_1)t)).$$

We have also computed $\psi_1(4, 0.2) = 0.0892$, thanks to the same kernel approximation, which should be equal, by a symmetry argument, to $\psi_1(4, -0.2)$. The difference between $\psi_1(x_0, v)$ and $\psi_1(x_0, -v)$ with $x_0 = 4$ in the middle of **S** provides us with a test for checking for a possible error in the algorithm. This also indicates the highest accuracy one can expect for ψ_1. Thus, with simulations starting from a single point, we obtain a complete description of $u(t, x, v)$ for any $(x, v) \in D$ and t large

Table 5
Estimation of the principal eigenvalue α of B for the multiplying spheres models with $N = 10^7$ particles

c	1.138	0.1384602	1.139
α_{LI} (T_1, T_2)	−0.000419	7.1×10^{-5}	0.00067
α_{LI} (T_2, T_3)	−0.000540	1.8×10^{-5}	0.00056
α_{LI} (T_1, T_3)	−0.000480	4.7×10^{-5}	0.00061
Mean	−0.00052	4.5×10^{-5}	0.000613

enough. We can use this description for some rare event estimations. For example, if one needs to compute $\mathbb{P}_{x,v}[\tau > T]$ for a larger value of T than the one used in this simulation, one can use the approximation

$$\mathbb{P}_{x,v}[\tau > T] \simeq \exp(\lambda_1 T) K \psi_1(x, v).$$

Thanks to the Markov property, for any measurable event, Γ, that this depends only on what happens after time T sufficiently large,

$$\mathbb{P}_{x,v}[(X_t, V_t)_{t \geqslant T} \in \Gamma] \simeq K \psi_1(x, v) \exp(\lambda_1 T) \mathbb{P}_{\psi_1^*}[(X_t, V_t)_{t \geqslant 0} \in \theta_T^{-1} \Gamma],$$

where $(\theta_t)_{t \geqslant 0}$ is the shift operator of the Markov process. Thus, one has only to perform a Monte Carlo estimation of $\mathbb{P}_{\psi_1^*}[(X_t, V_t)_{t \geqslant 0} \in \theta_T^{-1} \Gamma]$ to get an estimate of $\mathbb{P}_{x,v}[(X_t, V_t)_{t \geqslant T} \in \Gamma]$.

4.2. Multiplying spheres

We now consider a similar problem, where the positions of the particles take their values in a ball, and the velocities take their values on the unit sphere. The numerical resolution of such a problem requires the discretization of 5 variables by means of deterministic methods.

4.2.1. The physical model
We now consider the Cauchy problem

$$\frac{\partial u(t, x, v)}{\partial t} = -v \nabla_x u(t, x, v) + (c - 1) u(t, x, v) + c \left(\frac{1}{4\pi} \int_{\mathbb{S}_2} u(t, x, v') dv' - u(t, x, v) \right),$$

with an initial condition of $u(x, v, 0) = 1$, and absorbing boundary conditions. The velocity domain is the unit sphere $\mathbb{S}_2$, the spatial domain is the unit ball of radius d, and c plays the same role as in the previous model. The solution of this equation is

$$u(t, x, v) = \exp((c - 1)t) \mathbb{P}_{x,v}[\tau > t],$$

where the transport process $(X_t, V_t)_{t \geqslant 0}$ is a solution of the differential equation $\frac{dX_t}{dt} = -V_t$, with initial conditions $X_0 = x$ and $V_0 = v$. The velocity after a collision has a uniform law on $\mathbb{S}_2$. The cumulative distribution of the time between two collisions is

$$1 - \exp\left(-\int_0^t c ds \right) = 1 - \exp(-ct).$$

The simulation of $(X_t, V_t)_{t \geqslant 0}$ is explained in [28,32,25].

4.2.2. Numerical results
We compute an approximation of the criticality factor when $d = 4$ using the least squares method in the approximation of the principal eigenvalues. We perform the simulation using time slices at $T_1 = 20$, $T_2 = 40$ and $T_3 = 60$, using $N = 10^7$ particles for values of c close to the critical one. The criticality factor is about 1.1384602 [7]. The principal eigenvalues of B relative to $c = 1.138$ and $c = 1.139$ are respectively about -5×10^{-4} and 6×10^{-4}. The approximation of the criticality factor, given by the secant method, is $c \simeq 1.138459$ which corresponds to an error of about 10^{-6}. In a previous paper [32], this level of accuracy required $N = 2 \times 10^9$ simulations.

5. Conclusion

In the estimation of the principal eigenvalue with a Monte Carlo method, the branching algorithm is a very satisfactory way to improve the quality of the simulation proposed in [25]. In all the numerical tests presented here, the branching algorithm provided higher accuracy than our previous method for comparable simulation times. Indeed, with the previous method, the Monte Carlo error was roughly proportional to $1/\sqrt{pN}$, where p was the fraction of particles we keep to estimate the principal eigenvalue. As long time estimates are needed, the value of p was rather small. With the branching algorithm, the Monte Carlo error is roughly on the order of $1/\sqrt{N}$. Using the empirical distribution of the positions of the particles at a given time instead of the exact distribution has a low impact on the quality of the estimation.

In addition, this method gives us a way to estimate the principal eigenfunction of the adjoint operator using the empirical distribution density of the remaining particles. This could be important in some applications, especially in the neutron transport criticality problem. In addition, it may help to improve and accelerate the estimation of the probability of some events occurring at large times. Similar Monte Carlo techniques can also be used to simulate the principal eigenfunction of the operator when the latter is not self-adjoint, nor the eigenfunctions of the operator and its adjoint are related by symmetry relations. From this approximation of the eigenfunctions, we can also express an approximation of the solution to the Cauchy problem for large times and at any point, while the simulation only requires a single starting point.

The only drawback of this method is that it requires one to store a large amount of data. However, the amount of data increases only linearly with the dimension. On the other hand, as is typical of Monte Carlo methods, the computational cost

does not really depend on dimension. In addition, the branching algorithm is easy to implement, and hence may be used for high-dimensional problems.

Acknowledgments

References

[1] J.B. Anderson, A random-walk simulation of the Shrödinger equation: H_3^+, J. Chem. Phys. 63 (1975) 499.
[2] N. Balakrishnan, A.P. Basu, The Exponential Distribution: Theory, Methods, and Applications, Gordon and Breach, 1996.
[3] F.M. Buchmann, W.P. Petersen, An exit probability approach to solving high-dimensional Dirichlet problems, SIAM J. Sci. Comput. 28 (3) (2006) 1153–1166.
[4] D.M. Ceperley. Solving quantum many-body problems with random walks, in: H.J. Gardner, C.M. Savage (Eds.), Computational Physics, Proceedings of Ninth Physics Summer School, Australian National University, World Scientific Publishing Company, 1997.
[5] F. Cérou, P. Del Moral, F. LeGland, P. Lezaud, Genetic genealogical model in rare event analysis, ALEA 1 (2006) 181–203.
[6] F. Campillo, A. Lejay, A Monte Carlo method without grid for a fractured porous domain model, Monte Carlo Methods Appl. 8 (2) (2002) 129–148.
[7] E.B. Dahl, N.G. Sjostrand, Eigenvalue spectrum of multiplying slabs and spheres for monoenergetic neutrons with anisotropic scattering, Nulc. Sci. Eng. 69 (1979) 114–125.
[8] R. Dautray, J.-L. Lions, Spectre des opérateurs vol. 5 of Analyse Mathématique et Calcul Numérique pour les Sciences et Techniques, Masson, 1987.
[9] R. Dautray, J.-L. Lions, Évolution: numérique transport vol. 9 of Analyse Mathématique et Calcul Numérique pour les Sciences et Techniques, Masson, 1987.
[10] M. Deaconu, A. Lejay, A random walk on rectangles algorithm, Methodol. Comput. Appl. Probab. 8 (1) (2006) 135–151.
[11] P. Del Moral, Feynman-Kac formulae, Probability and its Applications, Springer-Verlag, New York, 2004. Genealogical and interacting particle systems with applications.
[12] P. Del Moral, J. Garnier, Genealogical particle analysis of rare events, Ann. Appl. Probab. 15 (4) (2005) 2496–2534.
[13] M.D. Donsker, S.R.S. Varadhan, On the principal eigenvalue of second-order elliptic differential operators, Comm. Pure Appl. Math. 29 (6) (1976) 595–621.
[14] W.M.C. Foulkes, L. Mitas, R.J. Needs, G. Rajagopal, Quantum Monte Carlo simulations of solids, Rev. Mod. Phys. 73 (1) (2001) 33–83.
[15] U. Gather, V. Schultze, Robust estimation of scale of an exponential distribution, Statist. Neerlandica 53 (3) (1999) 327–341.
[16] E. Gobet, Weak approximation of killed diffusion using Euler schemes, Stochastic Process. Appl. 87 (2) (2000) 167–197.
[17] B.L. Hammond, W.A. Lester, P.J. Reynolds, Monte Carlo Methods in Ab Initio Quantum Chemistry, World Scientific Publishing Company Inc., 1994.
[18] Y. Iba, Population Monte Carlo algorithms, Trans. Jpn. Soc. Artif. Intell. 16 (2) (2001) 279–286.
[19] N. Ikeda, S. Watanabe, Stochastic differential equations and diffusion processes, North-Holland Mathematical Library, vol. 24, North-Holland Publishing Co., Amsterdam, 1989. Kodansha, Ltd., Tokyo.
[20] K.M. Jansons, G.D. Lythe, Multidimensional exponential timestepping with boundary test, SIAM J. Sci. Comput. 27 (3) (2005) 793–808.
[21] M.H. Kalos, Monte Carlo calculations of the groud state of three- and four-body nuclei, Phys. Rev. 128 (4) (1962) 1791–1795.
[22] M.H. Kalos, Équation du transport. École CEA-EDF-INRIA sur les méthodes de Monte-Carlo, INRIA reports, 1981.
[23] I. Kosztin, B. Faber, K. Schulten, Introduction to the Diffusion Monte Carlo Method, Am. J. Phys. 64 (1996) 633.
[24] A. Lagnoux, Rare event simulation, Probab. Eng. Inform. Sci. 20 (1) (2006).
[25] A. Lejay, S. Maire, Computing the principal eigenvalue of the laplace operator by a stochastic method, Math. Comput. Simulation 73 (3) (2007) 351–363.
[26] A. Lagnoux Renaudie, Analyse des modèles de branchement avec duplication des trajectoires pour l'étude des événements rares. PhD thesis, Université Toulouse 3, 2006.
[27] P. Lascaux, R. Théodor, Analyse numérique matricielle appliquée á l'art de l'ingénieur. Tome 2. Méthodes itératives, second ed., Masson, Paris, 1994.
[28] S. Maire. Réduction de variance pour l'intégration numérique et pour le calcul critique en transport neutronique. PhD thesis, Université de Toulon et de Var, 2001.
[29] M. Mascagni, A. Karaivanova, A Monte Carlo approach for finding more than one eigenpair, Numerical methods and applications, Lecture Notes in Comput. Sci., vol. 2542, Springer, Berlin, 2003. pp. 123–131.
[30] G.N. Milstein, M.V. Tretyakov, Simulation of a space-time bounded diffusion, Ann. Appl. Probab. 9 (3) (1999) 732–779.
[31] L. Mitas, E.L. Shirley, D.M. Ceperley, Nonlocal Pseudopotentials and Diffusion Monte Carlo, J. Chem. Phys. 95 (1991) 3467.
[32] S. Maire, D. Talay, On a Monte Carlo method for neutron transport criticality computations, IMA J. Numer. Anal. 26 (4) (2006) 657–685.
[33] R.G. Pinsky, Positive Harmonic Functions and Diffusion, Cambridge University Press, 1996.
[34] P.J. Reynolds, D.M. Ceperley, B.J. Adler, W.A. Lester, Fixed-Node Quantum Monte Carlo for Molecules, J. Chem. Phys. 77 (1982) 5593.
[35] P.J. Rousseeuw, C. Croux, Alternatives to the median absolute deviation, J. Am. Statist. Assoc. 88 (1993) 1273–1283.
[36] K.E. Schmidt, P. Niyaz, A. Vaught, M.A. Lee, Green's function Monte Carlo method with exact imaginary-time propagation, Phys. Rev. E. 71 (1995) 016707.
[37] B.W. Silverman, Density estimation for statistics and data analysis, Monographs on Statistics and Applied Probability, Chapman and Hall, London, 1986.
[38] C.J. Umrigar, K.G. Wilson, J.W. Wilkins, Optimized trial wave functions for quantum Monte Carlo calculations, Phys. Rev. Lett. 60 (1988) 1719.
[39] D. Williams, Weighing the odds, Cambridge University Press, Cambridge, 2001.