New Monte Carlo schemes for simulating diffusions in discontinuous media

Antoine Lejay, Sylvain Maire

New Monte Carlo schemes for simulating diffusions in discontinuous media

Antoine Lejay[1,2,3,4,5] Sylvain Maire[6,7]

November 14, 2012

Abstract

We introduce new Monte Carlo simulation schemes for diffusions in a discontinuous media divided in subdomains with piecewise constant diffusivity. These schemes are higher order extensions of the usual schemes and take into account the two dimensional aspects of the diffusion at the interface between subdomains. This is achieved using either stochastic processes techniques or an approach based on finite differences. Numerical tests on elliptic, parabolic and eigenvalue problems involving an operator in divergence form show the efficiency of these new schemes.

Keywords: Monte Carlo method, Diffusion in discontinuous media, Skew Brownian motion, Finite differences

[1]Université de Lorraine, IECN, UMR 7502, Vandœuvre-lès-Nancy, F-54500, France
[2]CNRS, IECN, UMR 7502, Vandœuvre-lès-Nancy, F-54500, France
[3]Inria, Villers-lès-Nancy, F-54600, France
[4]Contact: IECN, BP 70238, F-54506 Vandœuvre-lès-Nancy CEDEX, France. Email: `Antoine.Lejay@iecn.u-nancy.fr`
[5]This work has been supported by the ANR SIMUDMRI (ANR-10-COSI-SIMUDMRI).
[6]LSIS (UMR CNRS 6168), Université de Toulon et du Var, France
[7]Contact: ISITV, Avenue G. Pompidou, BP 56, F-83262 La Valette du Var CEDEX, France. Email: `maire@univ-tnl.fr`

1 Introduction

Many diffusive models assume that the flow is proportional to the gradient of the concentration of a fluid. In the simplest situations, the media is homogeneous and additionally the incompressibility of the fluid is assumed. Hence, the concentration is the solution of an equation involving the Laplace operator.

In more realistic situations, the diffusivity coefficient a present some discontinuities and the involved operator takes the general form $\nabla(a\nabla\cdot)$. We can mention applications in geophysics when dealing with the Darcy law [12, 35, 37], magneto-electro-encephalography [27], population ecology [4], astrophysics [38], oceanography [11], ... The resulting numerical problems consist in solving partial differential equations in large domains presenting complex geometries and multi-scale features. Solving such problems is still a real issue while they have very important industrial and academic applications.

Several very efficient algorithms, either deterministic or probabilistic, exist to solve problems in homogeneous media. Among them are Monte Carlo algorithms which rely on the simulation of the Brownian motion using random walks on subdomains methods like the walk on spheres (WOS) or the walk on rectangles (WOR). When dealing with discontinuous media, the crucial question is the behavior of the random walk when hitting the interface between physical domains. Many algorithms have already been proposed mainly for one-dimensional media (See [6–8, 12, 21, 27–29, 34, 35, 37] for example), while a few others deal with locally isotropic media [17], generally in the steady state regime [3, 30].

The aim of this paper is twofold. First we remind the existing schemes and introduce some new ones to deal with interface conditions of locally isotropic media, which means that the coefficient a is scalar and piecewise constant. Second, we compare these new schemes and the old ones on elliptic, parabolic and eigenvalue problems on two-dimensional numerical examples.

The rest of the paper is organized as follows. In Section 2, we make a general description of the simulation algorithm and of the different kind of partial differential equations it can deal with. We also make an error analysis of the algorithm focusing especially on its bias. In Section 3, we remind the WOS and WOR methods and discuss how they should be used on our three kinds of problems. In Section 4, we describe the existing schemes in the one-dimensional case and especially the Skew Brownian motion from which they derive. Sections 5 and 6 are devoted to new schemes to deal with the discontinuity of the diffusivity coefficient in dimension two. The crucial point is to approximate efficiently the tangential component of the motion. In Section 5, this approximation is achieved by stochastic processes techniques based on occupation times or on a kinetic scheme. In Section 6, we use a local approximation of the divergence form operator by finite differences techniques. These schemes are higher order extensions of the standard ones developed

in [30]. In the final section, all the schemes are tested and compared numerically on elliptic, parabolic and eigenvalue problems in a simple domain constituted of two subdomains with different diffusivities.

2 The general simulation method and its error

Our main goal is to focus on the quality of the schemes that take discontinuities of the diffusivity into account. All these schemes are based on a particle moving in a domain D divided in subdomains in which the diffusivity is constant. For the sake of simplicity, we assume that D is divided in only two subdomains D_1 and D_2 with an interface Γ and that the boundary conditions on ∂D are of Dirichlet type. This means that the particle is killed when hitting the boundary. The schemes will be tested on three kinds of usual PDE problems that we now describe and remind of their probabilistic representations

Elliptic PDE. We consider the following elliptic PDE with Dirichlet boundary conditions in a domain $D = D_1 \cup D_2$ with a piecewise smooth boundary:

$$\frac{1}{2}\nabla(a(x)\nabla u(x)) = -f(x) \text{ on } D,\ u_{|\partial D}(x) = \varphi(x), \tag{1}$$

for a bounded, continuous function f in D and a continuous function φ on ∂D. The scalar coefficient a is constant on D_1 and D_2.

This equation has to be understood as a weak solution belonging to a Sobolev space. However, the solution u is α-Hölder continuous on D, smooth in D_1 and D_2, while the flux $a(x)\nabla u(x)$ is continuous on the interface Γ between D_1 and D_2 [13]. Let $(Z, (\mathcal{F}_t)_{t\geq 0}, (\mathbb{P}_x)_{x\in D})$ be the process generated by $\frac{1}{2}\nabla(a\nabla\cdot)$ and τ be its first exit time from D. This process enjoys the strong Markov property, has continuous paths and its transition density function may be compared to the Gaussian one [36].

Thanks to Itô formula and passing to the limit [36] in the approximation of the discontinuous coefficients by smooth ones, one gets the usual probabilistic representation of u as

$$u(x) = \mathbb{E}_x\left[\int_0^{\tau_D} f(Z_s)\,\mathrm{d}s\right] + \mathbb{E}_x[\varphi(Z_{\tau_D})]$$

where $\tau_D = \inf\{t > 0 \,|\, Z_t \notin D\}$.

Parabolic PDE. For f continuous and bounded on $[0,T]\times D$, ψ bounded and continuous on $\overline{D}$ and φ bounded and continuous on $[0,T)\times D$, we consider the

parabolic PDE

$$\begin{cases} \dfrac{\partial u(t,x)}{\partial t} - \dfrac{1}{2}\nabla(a(x)\nabla u(t,x)) - f(t,x) = 0 \text{ on } \mathbb{R}_+ \times D, \\ u(0,x) = \psi(x),\ x \in D, \\ u(t,x) = \varphi(t,x),\ x \in \partial D,\ t > 0. \end{cases} \tag{2}$$

The solution $u(t,x)$ is α-Hölder continuous in space, $\alpha/2$-Hölder continuous in time for some $\alpha \in [0,1]$ and locally smooth away from $\Gamma \cup \partial D$ [14]. Using the same tools than for the elliptic PDE, it can be shown that

$$u(t,x) = \mathbb{E}_x[\psi(Z_t)\mathbf{1}_{t\leq\tau_D}] + \mathbb{E}_x[\varphi(\tau_D, Z_{\tau_D})\mathbf{1}_{\tau_D<t}] + \mathbb{E}_x\left[\int_0^{t\wedge\tau_D} f(s,Z_s)\,\mathrm{d}s\right]$$

similarly to for non-divergence form operators [36].

Principal eigenvalue problem. Since $\frac{1}{2}\nabla(a\nabla\cdot)$ has a compact resolvent, it has a countable number of eigenvalues. The smallest eigenvalue λ_1 has a significant physical interpretation as it is related to the rate of convergence to the steady state regime. The eigenvalue problem writes

$$\begin{cases} \text{Find the smallest } \lambda > 0 \text{ such that for some } u \not\equiv 0, \\ -\dfrac{1}{2}\nabla(a\nabla u) = \lambda u \text{ on } D \text{ with } u_{|\partial D} = 0. \end{cases} \tag{3}$$

If τ_D is the first exit time from D and λ_1 the solution to (3), then for any $x \in D$, $\mathbb{P}_x[\tau_D > t] \approx ce^{-\lambda_1 t}$ for t large enough so that λ_1 may be estimated from the empirical distribution function of τ_D [19, 20].

2.1 General schemes

We fix a parameter $T > 0$, which is the time horizon in the parabolic problem (2) and verifies $T = +\infty$ for either the elliptic problem (1) or the eigenvalue problem (3). The numerical approximation of the solution $u(t,x)$ to (2), requires to simulate $\tau_D \wedge T$, $Z_{\tau_D\wedge T}$, $\sigma(T,x) = \int_0^{\tau_D\wedge T} f(s,Z_s)\,\mathrm{d}s$ knowing that $Z_0 = x$ and $\tau_D = \inf\{t > 0 | Z_t \notin D\}$. Knowing these quantities allows also one to solve the elliptic and eigenvalue problems (1) and (3) but they are not all necessary because for instance $f = 0$ in the eigenvalue problem.

We define for any $x \in \partial D$ and any $t \geq 0$ a function $\theta(t,x) \leq T - t$ which is either a $(\mathcal{F}_t)_{t\geq 0}$-stopping time or a deterministic time. This function is an approximation of an elapsed time or of a mean elapsed time of a trajectory starting at $x \in D$. For $x \in \partial D$, $t \geq 0$, we set $\theta(t,x) = 0$. We also set $t_0 = 0$ and

$t_{k+1} = t_k + \theta(t_k, Z_{t_k})$. A numerical scheme consists then in constructing a Markov chain $\{\nu_k, \xi_k\}_{k\geq 0}$ representing a couple (time,position) and a sequence $\{\sigma_k\}_{k\geq 0}$ of *local scores* such that

(i) $\xi_0 = x \in D$
(ii) $\nu_{k+1} - \nu_k$ is an approximation of $\theta(\nu_k, \xi_k)$
(iii) ξ_{k+1} is an approximation of $Z_{\theta(\nu_k,\xi_k)}$ when $Z_{\nu_k} = \xi_k$
(iv) σ_k is an approximation of $\int_{\nu_k}^{\nu_{k+1}} f(s, Z_s)\,\mathrm{d}s$ or of $\mathbb{E}\left[\int_{\nu_k}^{\nu_{k+1}} f(s, Z_s)\,\mathrm{d}s\right]$
(v) There exists an almost surely finite random variable n^* such that $\xi_{k+1} = \xi_k$ and $\nu_{k+1} = \nu_k$ when either $\xi_{n^*} \in \partial D$ or $\nu_{n^*} = T$ (the Markov chain hit the boundary and remains there, or no longer move after time T)
(vi) The total score of the walk is $S = \sum_{k=0}^{n^*-1} \sigma_k$.

Since a is constant in a given subdomain D_i, $i = 1, 2$, the process Z with $Z_0 = x$ is generated by $\frac{1}{2}a(x)\triangle$ and we have $\mathrm{d}Z_t = \sqrt{a(x)}\,\mathrm{d}W_t$ for $t < \tau_{D_i}$. The simulation of $(\tau_{D_i} \wedge T, Z_{\tau_{D_i}\wedge T})$ where τ_{D_i} is the first exit time from D_i of Z reduces to the one of a scaled Brownian case. For $x \in D_i$, $i = 1, 2$, in the parabolic (elliptic) case $\theta(t, x)$ is the (mean) exit time of a scaled Brownian motion from the subdomain D_i. We can hence use simulation techniques based on random walk on subdomains which are presented succinctly in Section 3. When f is not constant, $\int_0^{\tau_{D_i}\wedge T} f(Z_s)\,\mathrm{d}s$ or its mean value are also easily accurately computed by using for example the one random point method coupled or not with quantization tools [26].

The main difficulty remains the simulation of the particle motion when it starts from the interface. Using that the dynamic of the particle depends essentially on the local values of the diffusivity coefficients, one only has to focus on the case of a single discontinuity and locally constant coefficients around it. Many schemes have been proposed to deal with discontinuities of the diffusivity in a one-dimensional media: see [6–8, 12, 27–29, 34, 35, 37] for example. In Sections 4, 5 and 6, we present some existing schemes and introduce new ones with both better accuracy and better order. The update at the interface of the parameters of the Markov chain and of the local score will be described in these sections. The general method from which one may deduce easily the solutions to (1), (2) and (3) is given in Algorithm 1 below.

2.2 Error analysis for elliptic and parabolic PDE

For the sake of simplicity, we consider here only the elliptic problem (1) since the parabolic problem may be treated the same way. To any $x \in \overline{D}$, we associate a family of $(\mathcal{F}_t)_{t\geq 0}$-stopping times $\theta(t, x)$ such that $\theta(t, x) = 0$ for $x \in \partial D$, $t \geq 0$.

Data: A point $x \in D$ and a horizon $T > 0$ (possibly $T = +\infty$).
Result: A realization of an approximation of
$\left(\tau_D \wedge T, Z_{\tau_D \wedge T}, \mathbb{E}_x \left[\int_0^{\tau_D \wedge T} f(s, Z_s)\,\mathrm{d}s\right]\right)$ when $Z_0 = x$.
Set $\xi = x$, $S \leftarrow 0$ and $t \leftarrow 0$;
while $\xi \notin \partial D$ *or* $t < T$ **do**
  if $\xi \in \Gamma$ **then** `/* If the particle is at the interface */`
    Draw a new position ξ' after a time increment θ with $t + \theta \leq T$, as well as a score σ for the particle at the interface;
    if $\xi' \notin D$ **then** `/* The particle exits from` D `*/`
      `/* Perform a linear interpolation */`
      Determine the value $\lambda \in (0, 1)$ at which $\xi + \lambda\xi' \in \partial D$;
      Set $\theta \leftarrow \lambda\theta$;
      Set $\sigma \leftarrow \lambda\sigma$;
      Set $\xi' \leftarrow \xi + \lambda\xi'$;
    end
  else `/* If the particle is away from the interface */`
    Draw a realization (θ, ξ', σ) of an approximation of
    $\left(\tau_{D_i} \wedge T, Z_{\tau_{D_i} \wedge T}, \mathbb{E}_\xi \left[\int_t^{\tau_{D_i} \wedge T} f(s, Z_s)\,\mathrm{d}s\right]\right)$ where
    $\tau_{D_i} = \inf\{s > 0 | Z_s \in \Gamma \cup \partial D_i\}$ when $\xi \in D_i$ and $Z_t = \xi$;
  end
  Set $t \leftarrow t + \theta$; `/* Increment the time */`
  Set $\xi \leftarrow \xi'$; `/* Update the position */`
  Set $S \leftarrow S + \sigma$; `/* Update the total score */`
end
return (t, ξ, S)

Algorithm 1: The general simulation algorithm for a path.

Since $u_{|\partial D} = \varphi$, we have

$$u(x) = \mathbb{E}[u(Z_{\theta(0,x)}) \,|\, Z_0 = x] + S(x) \text{ with } S(x) = \mathbb{E}\left[\int_0^{\theta(0,x)} f(Z_s)\,\mathrm{d}s \,\middle|\, Z_0 = x\right]$$

and consequently

$$u(x) = Pu(x) + S(x) \text{ with } Pu(x) = \mathbb{E}[u(Z_{\theta(0,x)}) \,|\, Z_0 = x]. \tag{4}$$

Let $\{\xi_k\}_{k \geq 0}$ be the homogeneous Markov chain defined as above with $\xi_0 = x$ and such that ξ_{k+1} represents an approximation of $Z_{\theta(0,\xi_k)}$ (this means here that the future positions of the particle and the elapsed time depend only on the position of

the particle, not on the current time). Let $\{\sigma_k\}_{k\geq 0}$ be the corresponding sequence of scores which aims at approximating $S(\xi_k)$, such that σ_k depends only on ξ_k and ξ_{k+1} and verifying $\sigma_k = 0$ when $\xi_k \in \partial D$. We also recall that there exists an almost surely finite n^* such that for any $k \geq n^*$, $\xi_k = \xi_{n^*} \in \partial D$ and consequently $\sigma_k = 0$.

We define an approximation of $u(x)$ by

$$\overline{u}(x) = \mathbb{E}\left[\sum_{k=0}^{n^*-1} \sigma_k \middle| \xi_0 = x\right] + \mathbb{E}[\varphi(\xi_{n^*}) \,|\, \xi_0 = x]$$
$$= \mathbb{E}\left[\sum_{k\geq 0} \sigma_k \middle| \xi_0 = x\right] + \mathbb{E}[\varphi(\xi_\infty) \,|\, \xi_0 = x].$$

By conditioning with respect to ξ_1, we obtain

$$\overline{u}(x) = \mathbb{E}[\sigma_0 \,|\, \xi_0 = x] + \mathbb{E}[\overline{u}(\xi_1) \,|\, \xi_0 = x] = \overline{P}\overline{u}(x) + \overline{S}(x) \tag{5}$$

with $\overline{S}(x) = \mathbb{E}[\sigma_0 \,|\, \xi_0 = x]$. Letting $w(x) = u(x) - \overline{u}(x)$ and subtracting Eq. (5) to Eq. (4) leads to

$$w(x) = \overline{P}w(x) + (P - \overline{P})u(x) + S(x) - \overline{S}(x) \tag{6}$$

Since $\{\xi_k\}_{k\geq 0}$ is a Markov chain with transition matrix $\overline{P}$, for any bounded, measurable function f and any $k \geq 0$, $f(\xi_k) = \overline{P}^k f(x)$. This means that (6) may be written

$$w(x) = (1 - \overline{P})^{-1} r(x) \text{ or formally } w(x) = \sum_{k\geq 0} \overline{P}^k r(x)$$

with $r(x) = (P - \overline{P})u(x) + S(x) - \overline{S}(x)$. Hence,

$$u(x) = \overline{u}(x) + \sum_{k\geq 0} \mathbb{E}\big[(P - \overline{P})u(\xi_k) + S(\xi_k) - \overline{S}(\xi_k) \,\big|\, \xi_0 = x\big] \tag{7}$$

which gives an estimate of the induced bias notwithstanding the Monte Carlo error.

Remark 1. Our approach for studying the convergence of the scheme is very close to the one used in deterministic techniques involving consistency and stability. However, if $\overline{P}$ depends on a parameter h related to a timestep, $(I - \overline{P})^{-1}$ is not in general uniformly bounded in h (See Sections 2.3 and 7.1). The convergence of $\overline{u}(x)$ to $u(x)$ follows from the fact that the upper bound of $r(x)$ decreases faster to 0 than the growth of the upper bound of $(I - \overline{P})^{-1}$.

2.3 Analysis of the bias

Even though it seems difficult to perform a throughout analysis of the bias of the whole algorithm, Eq. (7) gives some clues to analyze the different schemes. As we have at our disposal exact or very accurate schemes to simulate $(\tau_{D_i}, Z_{\tau_{D_i}})$ or compute related functionals, we may assume that $\overline{P}u(x) = Pu(x)$ and $S(x) = \overline{S}(x)$ for $x \in D \setminus \Gamma$. We can hence consider that the source of bias is only due to the scheme to move the particle from the interface Γ into D.

The bias is roughly the product between the number of times $\xi_k \in \Gamma$ and the quantity $(P - \overline{P})u(\xi_k) + S(x) - \overline{S}(x)$. We should explain heuristically that for a simple domain with two layers, the number of times $\xi_k \in \Gamma$ is of order $\mathrm{O}(h^{-1})$, where h is the distance at which the particle is put away from Γ. For this, we consider the one-dimensional domain $D = [-L, L]$ with an interface at 0. If a Brownian particle is at position h on $[0, L]$, then it hits 0 before L with probability $1 - h/L$, whatever the diffusion coefficient is. Hence, the number of times the particle passes at the interface before reaching $-L$ or L follows a geometric distribution of parameter $1 - h/L$. The average number of steps is then L/h. In Section 7.1, this property will be confirmed numerically.

We now focus on the quantity $(P - \overline{P})u(x)$. The bias depends on how close $\overline{P}$ is to P. The kernel $\overline{P}$ is the density of a random variable ξ, while P is the density of $Z_{\theta(x)}$. A first way to construct a scheme is then to choose ξ as close as possible to $Z_{\theta(t,x)}$. Some schemes are presented in Section 5 relying on a fixed time step δt and with $\theta(t, x) = \delta t$. They are either justified by stochastic analysis or by PDE considerations. Since u is α-Hölder continuous on D for some $\alpha \in [0, 1]$ and is smooth in each of the D_i [13], this proves that $\mathbb{E}[u(\xi)]$ is close to $\mathbb{E}[u(Z_{\theta(t,x)})]$. In Section 4.4, we provide an upper bound on the bias when the particle is moved in the direction normal to the interface, by neglecting the tangential component.

A second idea consists in constructing $\overline{P}$ in a way such that $(\overline{P} - P)u(x)$ is as small as possible and that $\overline{P}$ may be interpreted as the density of a discrete or continuous random variable. In Section 6, we construct an operator $\overline{P}$ such that

$$u(x) = \overline{P}u(x) + \overline{S}(x) + r(x) \text{ for } x \in \Gamma,$$

where u is the solution to (1). Subtracting this equation to (4), we also obtain that

$$(P - \overline{P})u(x) + S(x) - \overline{S}(x) = r(x)$$

which shows that the terms $r(x)$ and $S(x) - \overline{S}(x)$ should be added to the total bias each time the particle lies at some points x on the interface. Constructing an approximation $\overline{P}$ of a differential operator P is a classical task in numerical analysis. The difficulty here is to construct an approximation $\overline{P}$ giving rise to a Markov chain.

3 Simulation in a media with constant diffusivity

We describe two methods to simulate a Brownian particle based on walk on subdomains, the walk on spheres method and the walk on rectangles method. These methods can provide unbiased or slightly biased simulations while being a lot faster than the Euler scheme.

3.1 Random walk on spheres

Proposed by E. M. Muller in 1956 [33], the random walk on spheres (WOS) gives a simple and efficient way to solve the Laplace equation with Dirichlet boundary conditions. The idea it to draw the next position of a Brownian motion at a point on the boundary of an arbitrary sphere centered on the particle and tangent to the domain. The algorithm stops when the particle is at a distance to the boundary of the domain smaller than a fixed parameter ε. Thanks to the invariance by rotation of the distribution of the Brownian motion, the position of the particle is uniformly distributed on the sphere. In dimension 2, this means that in radial coordinate, the angle is picked uniformly at random in $[0, 2\pi)$.

This method is very efficient to deal with elliptic equations of Laplace type. Indeed, the average number of steps until absorption is proportional to $|\ln(\varepsilon)|$. Its main drawback is a quite hard simulation of the first exit time from the unit sphere (see the discussion in [9]). Yet this exit time is a very important information for solving parabolic equations or computing the first eigenvalue by Monte Carlo methods as in [19, 20]. However, it is possible to solve this problem by using a tabulated version of the first exit time's distribution function of the unit ball (obtained for instance from a fine discretization of the Brownian motion with an Euler scheme) and scaling arguments.

3.2 Random walk on rectangles

The random walk on squares aims at superseding the random walk on spheres by simulating exactly (W_τ, τ), where τ is the first exit time from a square (or a hypersquare) centered on the starting point W_0 of a Brownian motion. In opposite to the case of spheres, W_τ and τ are not independent. Yet the couple (W_τ, τ) can be simulated rather efficiently. This method has been introduced independently by O. Faure [9] and by G.N. Milstein and M.V. Tretyakov [32].

The random walk on rectangles algorithm provides a variation of the random walk on squares. It allows one to compute the exit position and the first exit time (W_τ, τ) from any rectangle whatever the position of the starting point inside the rectangle is [5]. Note that both algorithms provide an efficient simulation of $(W_{\tau_D \wedge T}, \tau \wedge T)$ for any $T > 0$ fixed. This property will be especially useful in the

last example of Section 7.4 when using a splitting algorithm to compute the leading eigenelements of a divergence form operator. Sampling from $(W_{\tau_D \wedge T}, \tau_D \wedge T)$ using the WOS method is possible but it requires to store full discretized trajectories of a Brownian motion starting at the center of a the unit ball and killed at its boundary [25]. This introduces however supplementary errors not easy to quantify due to the discretization of the trajectories. This is precisely what we want to avoid here. We refer the readers to the articles [5, 32] and to the Ph.D. thesis [9] for a complete description of these algorithms.

4 Normal schemes for discontinuous media

We now propose some schemes for moving the particle away from the interface. For this, we place ourselves in the case where the interface can be seen locally as lying on a hyperplane. This approximation is usual for example when dealing with Euler schemes for SDE with reflection, corresponding to Dirichlet or Neumann boundary conditions (See e.g. [3, 10]).

As the diffusivity is isotropic, the behavior of the particle is invariant under rotation so that there is no difficulty in assuming that the interface lies locally in the hyperplane defined by $x = 0$.

Of course, considering a plane interface imposes some restrictions. In a general case, the time step shall be chosen so that the jumps of the particles are small compared to the local radius of the interface. Besides, we should take care of corners, as well as of situations with multiple interfaces which are close to each other.

More generally, it is important to recall that the displacement of the particle at the interface depends mainly on its local environment. This means that our schemes may be coupled with any scheme that can deal with a non constant diffusivity away from the interface.

Our first class of schemes are normal schemes where we focus only on the component of the process normal to the interface without moving the tangential component.

A *normal scheme* is a scheme where we focus only on the component of the process normal to the interface without moving the tangential component. This way, any of the already proposed one-dimensional schemes may be also used in any dimension. We review here three of them which are representative of the existing schemes. In Section 4.4, we study the bias induced by moving only the particle in the direction normal to the interface.

For the sake of simplicity, we consider that $d = 2$. As these schemes will be

used locally, we may consider up to some rotation that the scalar diffusivity is

$$a(x,y) = \begin{cases} a_1 & \text{if } x \geq 0, \\ a_2 & \text{if } x < 0, \end{cases}$$

where a_1 and a_2 are strictly positive coefficients, so that the interface is $\Gamma = \{(x,y) \,|\, x = 0\}$. Let $Z = (X,Y)$ be the 2-dimensional process generated by $L = \frac{1}{2}\nabla(a\nabla\cdot)$. Since $a(x,y)$ depends only on x, we write $a(x) = a(x,y)$. The normal component X is a one-dimensional stochastic process generated by $\frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}x}(a\frac{\mathrm{d}}{\mathrm{d}x}\cdot)$ where $a(x) = a_1$ if $x \geq 0$ and $a(x) = a_2$ if $x < 0$.

4.1 Skew scheme for an exact simulation of the x-component

As shown for example in [21], the x-component process X is solution to the SDE with local time

$$X_t = x + \int_0^t \sqrt{a(X_s)}\,\mathrm{d}W_s + \frac{a_1 - a_2}{a_1 + a_2} L_t^0(X)$$

where

$$L_t^0(X) = \lim_{\varepsilon \to 0} \frac{1}{2\varepsilon} \int_0^t \mathbf{1}_{X_s \in [-\varepsilon, \varepsilon]}\,\mathrm{d}\langle X \rangle_s$$

is the symmetric local time of X at 0 and W a Brownian motion. The quantity $\langle X \rangle_t = \int_0^t a(X_s)\,\mathrm{d}s$ is the bracket of the semi-martingale X. Letting $\Psi(x) = \int_0^x y/\sqrt{a(y)}\,\mathrm{d}y$, the process $\widehat{X} = \Psi(X)$ is solution to the SDE with local time

$$\widehat{X}_t = \Psi(x) + W_t + \theta L_t^0(\widehat{X}),\ \theta = \frac{\sqrt{a_1} - \sqrt{a_2}}{\sqrt{a_1} + \sqrt{a_2}}.$$

The process $\widehat{X}$ is called a *Skew Brownian motion* (SBM). It has been introduced by K. Itô and H. McKean as a generalization of the Brownian motion and may be constructed by several means (See [16] for a survey). In particular, when $x = 0$, $\widehat{X}_t = \kappa_t |B_t|$ for a Brownian motion B and a process κ independent from B which is constant on each excursion of the Brownian motion with

$$\mathbb{P}[\kappa_t = 1] = 1 - \mathbb{P}[\kappa_t = -1] = \frac{1+\theta}{2} = \frac{\sqrt{a_1}}{\sqrt{a_1} + \sqrt{a_2}}. \tag{8}$$

This justifies Algorithm 2, where the x-component is moved following the exact dynamics of the process.

The articles [15, 22] present alternative methods that could be used when $x \neq 0$. Thanks to the properties of the Brownian bridge, one can achieve a perfect simulation of the first time τ a Brownian motion reaches 0 (See [1, 22] for more details).

Data: A position $(X, Y) = (0, y)$ of the particle at time t.
Result: A position (X, Y) of the particle after a time step δt.
Compute $\alpha = \sqrt{a_1}/(\sqrt{a_2} + \sqrt{a_1})$;
Generate a random variate U uniform on $[0, 1)$;
Generate a random variate $\xi \sim \mathcal{N}(0, 1)$;
if $U \leq \alpha$ **then**
  Set $X \leftarrow \sqrt{a_1 \delta t}|\xi|$
else
  Set $X \leftarrow -\sqrt{a_2 \delta t}|\xi|$
end
Increment time by δt;
Y is not moved;

Algorithm 2: Skew scheme.

4.2 Hoteit scheme

Algorithm 3 was introduced by H. Hoteit *et al.* [12] and is justified by mass balance conditions. It could be also seen as an approximation of Algorithm 2, where the normal distribution is replaced by the uniform distribution. Note that as in Section 4.1, this scheme preserves the property $\mathbb{P}[X_{t+\delta t} > 0 | X_t = 0] = \sqrt{a_1}/(\sqrt{a_1} + \sqrt{a_2})$ which is crucial for a good approximation.

Data: A position $(X, Y) = (0, y)$ of the particle at time t.
Result: A position (X, Y) of the particle after a time step δt.
Generate a uniform random variate U on $[0, 1)$;
Move X to $\sqrt{3a_2 \delta t}(U - 1) + \sqrt{3a_1 \delta t} U$;
Increment time by δt;
Y is not moved;

Algorithm 3: Hoteit scheme.

4.3 Stochastic 1D Finite differences

This scheme was proposed in [30] and relies on a finite difference approximation of the divergence form operator that we remind in Section 6.1 below.

4.4 Bias induced by the skew scheme

Following our error analysis technique of Section 2.3, we give a bound on the bias for normal schemes. Let u be the solution to $\frac{1}{2}\nabla(a\nabla u) = -f$ with Dirichlet

Data: A position $(X, Y) = (0, y)$ of the particle at time t and a small parameter h.
Result: A position (X, Y) of the particle away from the interface.
Generate a uniform random variate U on $[0, 1)$;
if $U < a_1/(a_1 + a_2)$ **then**
  Move X to h
else
  Move X to $-h$
end
Y is not moved;
No time increment;

Algorithm 4: 1D finite differences scheme.

Boundary conditions $u_{|\partial D} = 0$, where f is continuous and bounded on D as well as its first and second derivative. Let $Z = (X, Y)$ be the stochastic process generated by $L = \frac{1}{2}\nabla(a\nabla)$, starting initially at point $(0, y)$ of the vertical interface Γ. When using the Skew scheme (Algorithm 2), after a time step of $\theta = \delta t$, the particle is replaced at (X_θ, y) and the time is incremented by θ. As the corresponding score is given by $\theta f(0, y)$ and because $S(0, y) = \mathbb{E}_{0,y}\left[\int_0^\theta f(X_s)\,\mathrm{d}s\right]$ and $\overline{S}(0, y) = \delta t f(0, y)$, we have

$$|S(x, y) - \overline{S}(0, y)| \leq 2\|f\|_\infty \theta = \mathrm{O}(h^2)$$

where the parameter $h = \sqrt{\theta}$ is the mean distance of replacement of the particle away from the interface. Let us consider

$$\eta(0, y) = \mathbb{E}_{(0,y)}[u(X_\theta, Y_\theta) - u(X_\theta, y)].$$

We assume that $(0, y)$ belongs to a subdomain D' such that

$$\mathbb{P}_{(0,y)}[(X_t, Y_t) \notin D' \text{ for some } t \in [0, \theta]] \text{ is negligible,} \tag{9}$$

which means that we are far enough from the intersection between the boundary ∂D and the interface Γ. The function u is globally continuous and of class $\mathcal{C}^1$ in the subdomains $D_i \cap D'$ [13].

Let $\vec{t}$ be the vector $(0, 1)$ tangential to the interface and set

$$v(x, y) = \vec{t} \cdot \nabla u(x, y) = \nabla_y u(x, y).$$

Since a depends only on the first component x, and f is differentiable, $v(x, y)$ is solution to

$$\frac{1}{2}\nabla(a(x)\nabla v(x, y)) = -\vec{t} \cdot \nabla f(x, y) \text{ on } D'.$$

This equation is simply obtained by differentiating $\frac{1}{2}\nabla(a\nabla u) = -f$ with respect to ∇_y. Again from the results of [13], v is itself bounded on D' with a bounded derivative inside $D' \cap D_i$, $i = 1, 2$. This could be rigorously shown by using a smooth approximation of the coefficient a (in which case u is a classical solution) and by passing to the limit. As in [3, Theorem 2.17], it is then possible to show that $v(x, y)$ and $\vec{t} \cdot \nabla v(x, y)$ are bounded on D' by applying the same argument twice.

Hence, $\vec{t} \cdot \nabla v(x, y) = \vec{t} \cdot \nabla(\vec{t} \cdot \nabla u(x, y))$ is regular with respect to y. We assume that $\theta \leq \tau_{D'}$ under (9), where $\tau_{D'}$ is the first exit time of (X, Y) from D'. By applying the Itô formula to the y-component between 0 and $\theta \wedge \tau_{D'}$,

$$\mathbb{E}_{(0,y)}[u(X_\theta, Y_\theta) - u(X_\theta, y) \mid (X_s)_{s\in[0,\theta]}] = \mathbb{E}_{(0,y)}\left[\int_0^\theta \vec{t} \cdot \nabla(\vec{t} \cdot \nabla u)(X_\theta, Y_s)a(X_s)\,\mathrm{d}s \middle| (X_s)_{s\in[0,\theta]}\right],$$

which leads to $|\eta(0, y)| \leq K\theta = \mathrm{O}(h^2)$, where K is the upper bound of $\vec{t} \cdot \nabla v(x, y)$ on D'.

Each time the particle reaches the interface, the global error is increased by at most $\mathrm{O}(h^2)$. Since in our test case, the number of times the particle passes through the interface is $\mathrm{O}(h^{-1})$ (see Section 7.1), the error is at most of order $\mathrm{O}(h)$. We will see in Figures 3 and 4 of the numerical part that the error is empirically of order $\mathrm{O}(h)$ for the normal schemes.

5 Multi-dimensional schemes for discontinuous media

We consider the same operator as in Section 4 which generates a stochastic process (X, Y). We consider now some schemes where X and Y are simultaneously moved.

5.1 Kinetic scheme

The kinetic approximation was introduced in [18] as a way to deal with the multi-dimensional case with locally isotropic coefficients. The idea is to use a neutron transport equation to approximate the diffusion equation around the interface, as one knows how to solve a transport equation even in presence of discontinuities.

Algorithm 5 presents the kinetic scheme in dimension $d = 2$. For $d = 3$, the parameter of the exponential time shall be multiplied by 3/2 and the direction chosen uniformly on the unit sphere. In this kinetic scheme, the particle is moved inside the domain D or to the boundary of the domain ∂D. The time is incremented by at most δt.

Data: The position $(X, Y) = (0, y)$ of the particle at time t and a parameter $\varepsilon > 0$.
Result: The new position of the particle (X, Y) at a random time $t + \eta$.
Generate a uniform random variate φ over $[-\pi, \pi)$;
if $\varphi \in (-\pi/2, \pi/2)$ **then**
  Generate an exponential random variate η with parameter $a_1\varepsilon^2$;
else
  Generate an exponential random variate η with parameter $a_2\varepsilon^2$;
end
Set $x' \leftarrow \eta\cos(\varphi)$ and $y' \leftarrow y + \eta\sin(\varphi)$;
`/* Check if the particle is still in the domain. */`
if $(x', y') \notin D$ **then**
  Find the time η_0 at which $(\varepsilon\eta_0\cos(\varphi), y + \varepsilon\eta_0\sin(\varphi)) \in \partial D$;
  Set $\eta \leftarrow \eta_0$;
end
`/* Check if the time is greater than` δt`. */`
if $\eta > \delta t$ **then**
  Set $\eta \leftarrow \delta t$;
end
Move the particle to $(\varepsilon\eta\cos(\varphi), y + \varepsilon\eta\sin(\varphi))$;
Increment the time by η;

Algorithm 5: Kinetic scheme.

5.2 Mixing scheme

In Section 4.1, we have seen how to simulate $X_{t+\delta t}$ when $X_t = 0$ by following the exact dynamics of this component. The y-component Y is solution to the SDE

$$Y_t = Y_0 + \int_0^t \sqrt{a(X_s)}\,\mathrm{d}W_s \tag{10}$$

for a Brownian motion W which is independent from X (See [15] for a proof).

The *mixing scheme* is a simple scheme to simulate $Y_{t+\delta t}$ from Y_t by using an approximation of (10). Since W and X are independent, the law of Y_t given $Y_0 = y$ and X follows a Gaussian distribution with mean y and variance

$$\mathrm{Var}(Y_t|Y_0, X) = \int_0^t a(X_s)\,\mathrm{d}s = a_1 A_+(0,t) + a_2 A_-(0,t)$$

where

$$A_+(0,t) = \int_0^t \mathbf{1}_{X_s \geq 0}\,\mathrm{d}s \text{ and } A_-(0,t) = \int_0^t \mathbf{1}_{X_s \leq 0}\,\mathrm{d}s$$

are the *occupation times* of $\mathbb{R}_+$ and $\mathbb{R}_-$ for X. Using the properties of the SBM (See Section 4.1), if $X_0 = 0$ one knows that

$$\mathbb{E}[A_+(0,t)] = t\alpha \text{ and } \mathbb{E}[A_-(0,t)] = t(1-\alpha) \text{ with } \alpha = \frac{\sqrt{a_1}}{\sqrt{a_1}+\sqrt{a_2}}.$$

The idea of the mixing scheme is then to move the y-component as a Gaussian random variable with variance equal to $\mathbb{E}[\mathrm{Var}(Y_t|Y_0, X)]$. This leads us to set

$$D^{\mathrm{mix}} = a_1\alpha + a_2(1-\alpha)$$

and $Y_t^{\mathrm{mix}} = y + \sqrt{tD^{\mathrm{mix}}}\xi$ where $\xi \sim \mathcal{N}(0,1)$. We can also remark that if $\mathbb{E}_{(x,y)}$ is the expectation of the process (X,Y) with initial data $(X_0,Y_0) = (x,y)$, then $\mathbb{E}[Y_t^{\mathrm{mix}}] = y = \mathbb{E}_{(0,y)}[Y_t]$ and

$$\mathbb{E}[(Y_t^{\mathrm{mix}} - y)^2] = tD^{\mathrm{mix}} = \mathbb{E}_{(0,y)}[(Y_t - y)^2].$$

The corresponding algorithm is presented in Algorithm 6.

Data: A position $(X,Y) = (0,y)$ of the particle at the interface.
Result: A position (X,Y) of the particle after a time step δt.
Compute $D^{\mathrm{mix}} = a_1\alpha + a_2(1-\alpha)$;
Move X according to one of the schemes of Section 4;
Move Y to $y + \sqrt{\delta t D^{\mathrm{mix}}}\xi$ for a random variate $\xi \sim \mathcal{N}(0,1)$;
Increment time by δt;

Algorithm 6: Mixing scheme.

5.3 A scheme relying on the occupation time

In [15], we have proposed a scheme to simulate $(X_t, A_+(0,t))$ by using again the Skew Brownian motion when $X_0 = 0$. This scheme relies on the excursion properties of the Skew Brownian motion. The last passage time G at zero of the SBM before t is Arc-Sine distributed on $[0,t]$, and then equal in law to $t\sin^2(\pi U_1/2)$, where $U_1 \sim \mathcal{U}(0,1)$. At time G, the SBM starts an excursion straddling t which is positive with probability $\alpha = \sqrt{a_1}/(\sqrt{a_1}+\sqrt{a_2})$ (See (8) in Section 4.1). Hence,

$$A_+(G,t) = \begin{cases} t-G & \text{if the sign of the excursion is positive} \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

Besides, at time t, since G is known,

$$|X_t| \overset{\mathrm{law}}{=} \sqrt{-2(t-G)\log(U_2)},\ U_2 \sim \mathcal{U}(0,1).$$

Of course, if the excursion straddling t is positive, then $X_t > 0$ and $X_t < 0$ otherwise. If $\alpha = 1/2$, then $A_+(0,G)/G$ is uniformly distributed over $(0,1)$. Otherwise, the density f of $A_+(0,G)/G$ is

$$f(x) = \alpha(1-\alpha)/2(\alpha^2(1-x) + (1-\alpha)^2 x)^{3/2},\ x \in (0,1)$$

which means that

$$A_+(0,G) \overset{\text{law}}{=} G\frac{\alpha^2}{1-2\alpha}\left(\frac{1}{\left(1-\frac{1-2\alpha}{1-\alpha}U_3\right)^2} - 1\right),\ U_3 \sim \mathcal{U}(0,1). \tag{12}$$

Finally, note that

$$A_+(0,t) = A_+(0,G) + A_+(G,t) \text{ and } A_-(0,t) = t - A_+(0,t) \text{ for any } t > 0.$$

Hence, when $X_t = 0$, the y-component is moved after a time step δt to

$$Y_{t+\delta t} = Y_t + \sqrt{\delta t D^{\text{equiv}}}\xi$$

where $\xi \sim \mathcal{N}(0,1)$ and

$$D^{\text{equiv}} = a_1 A_+(0,t) + a_2(t - A_+(0,t)). \tag{13}$$

The corresponding algorithm is presented in Algorithm 7.

We can also produce simpler schemes by replacing $A_+(0,G)$ by $\mathbb{E}[A_+(0,G)|G]$ or $A_+(G,t)$ by $\mathbb{E}[A_+(G,t)|G]$. In the former case, the averaging takes place before G. Instead of D^{equiv}, we define

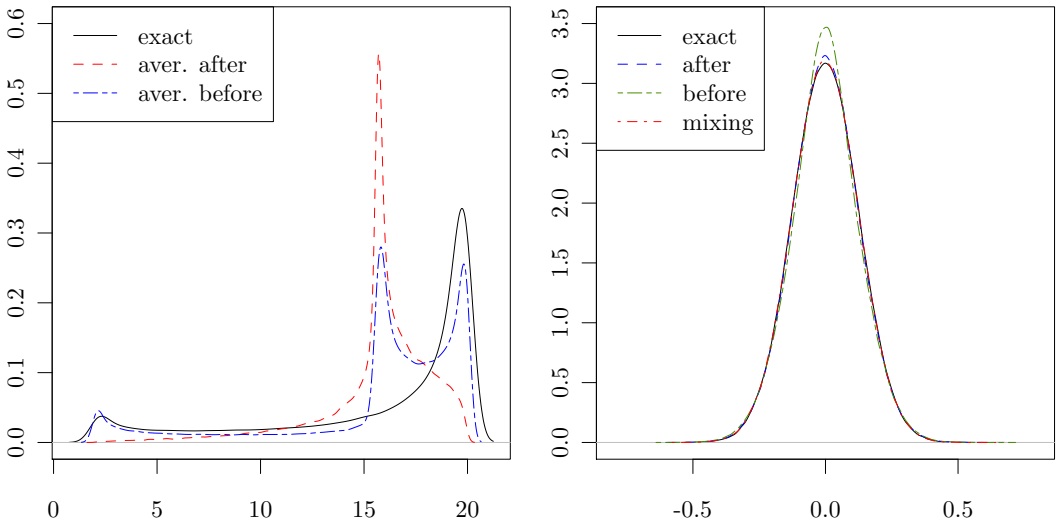
$$D^{\text{before}} = a_1(\alpha G + A_+(G,t)) + a_2((1-\alpha)G + (\delta_t - G - A_+(G,t))) \tag{14}$$

where $A_+(G,t)$ is computed using (11). In the latter case, the averaging takes place after G. Instead of D^{equiv}, we define

$$D^{\text{after}} = a_1 A_+(0,G) + a_2 A_-(0,G) + (t-G)(a_1\alpha + a_2(1-\alpha)) \tag{15}$$

where $A_+(0,G)$ is given by (12) and $A_-(0,G) = G - A_+(0,G)$. The resulting schemes can be seen as intermediate schemes between the mixing scheme of Section 5.2 and the one proposed in [15].

In Figure 1a, we see that the random variables D^{equiv}, D^{after} and D^{before} have different densities. In Table 1, we summarize the mean and variance of D^{equiv} with the coefficients we consider in our numerical tests of Section 7. However, we see in Figure 1b that the corresponding random variables $Y_{\delta t}$ have similar densities except that the one obtained using D^{before} is more concentrated around 0.



(a) Estimated densities of D^{equiv}, D^{after} and D^{before} with 1,000,000 samples. The expectation of D^{equiv} is $D^{\text{mix}} = 15.675$.

(b) Density of $Y_{\delta t}$ for four choices D^{equiv}, D^{after}, D^{before} and D^{mix} with 1,000,000 samples and $\delta t = 10^{-3}$.

Figure 1: Influence of the choice of a diffusivity for the y-component.

6 Stochastic 2D Finite difference schemes

6.1 1D Finite differences

For solving the Poisson-Boltzmann equation, M. Mascagni and N. Simonov have introduced a method based on finite differences to deal with the interface conditions [30]. We now sum up this method and assume without loss of generality that in all the following the interface Γ between the two subdomains D_1 and D_2 is vertical and located at $x = 0$. We refer for this to our discussion at the beginning of Section 4. The diffusion coefficient $a(x, y)$ is equal to a_1 in D_1 (when $x > 0$) and a_2 in D_2 (when $x < 0$).

Again, let us consider that u is the solution to (1). This solution is smooth on D_1 and D_2 and is continuous on D with some flux condition at the interface. The flux condition $a_1 \nabla u(0, y) = a_2 \nabla u(0, y)$ becomes

$$a_1 \frac{u(h, y) - u(0, y)}{h} = a_2 \frac{u(-h, 0) - u(0, y)}{h} + \mathrm{O}(h^2)$$

and then thanks to the continuity condition

$$u(x) = \frac{a_1}{a_1 + a_2} u(h, y) + \frac{a_2}{a_1 + a_2} u(-h, y) + \mathrm{O}(h^2). \tag{16}$$

Scheme	Mean	Variance
Exact D^{equiv}	$D^{\text{mix}} + 0.047$	29.5
Aver. after D^{after}	$D^{\text{mix}} + 0.003$	7.4
Aver. before D^{before}	$D^{\text{mix}} + 0.015$	22.1
$D^{\text{mix}} = a_1\alpha + a_2(1-\alpha) = 15.675$ for $a_1 = 20$, $a_2 = 2$		

Table 1: Mean and variance of D^{equiv} for the three possible schemes with $a_1 = 20$ and $a_2 = 2$.

Data: A position $(X, Y) = (0, y)$ of the particle at the interface.
Result: A position (X, Y) of the particle after a time step δt.
Draw a random variate $\gamma = \sin^2(\pi U_1/2)$ with $U_1 \sim \mathcal{U}(0,1)$;
Draw a Bernoulli random variate $\kappa \in \{-1, 1\}$ with $\mathbb{P}[\kappa = 1] = \alpha$ where $\alpha = \sqrt{a_1}/(\sqrt{a_2} + \sqrt{a_1})$;
Draw a random variate $\zeta = \sqrt{-2\delta t(1-\gamma)\log(U_2)}$ with $U_2 \sim \mathcal{U}(0,1)$;
Increment the time by δt;
Move X to $\kappa\zeta$;
Compute D^{equiv} with (13), (14) or (15);
Move Y to $y + \sqrt{D^{\text{equiv}}}\xi$ for a random variate $\xi \sim \mathcal{N}(0,1)$;

Algorithm 7: Occupation scheme.

Let ξ be a Bernoulli random variable such that

$$\mathbb{P}[\xi = h] = \frac{a_1}{a_1 + a_2} \text{ and } \mathbb{P}[\xi = -h] = \frac{a_2}{a_1 + a_2}.$$

Then (16) becomes

$$u(0, y) = \mathbb{E}[u(\xi, y)] + \mathrm{O}(h^2).$$

Comparing this expression with (5) justifies Algorithm 4 we have presented in Section 4.3. This method is of order h^2 and does not take into account the time spent during the replacement at position $-h$ or h. An asymmetric version has been proposed and studied in [3]. In dimension one, this way to move the particle may be justified by the approach of [7] in which the time is also incremented.

6.2 2D Finite differences

For Neumann or mixed boundary conditions, higher order methods have been introduced in [31] but they are only designed for walks on a fixed grid. In [25],

higher order methods not relying on a grid have been built for Neumann boundary conditions by writing both the approximations of the Laplace operator and of the boundary conditions at a point inside the domain. We shall now use this idea in the context of a divergence form operator with a discontinuous diffusivity.

The usual Finite differences approximation of the Laplace operator for a function u of class $\mathcal{C}^3$ is

$$\triangle^h u(x,y) = \frac{u(x+h,y)+u(x-h,y)+u(x,y+h)+u(x,y-h)-4u(x,y)}{h^2}$$

and we have

$$\triangle^h u(x,y) = \triangle u(x,y) + \mathrm{O}(h).$$

We can similarly write an approximation of the partial derivative by

$$\overline{\nabla}_x^h u(x,y) = \frac{4u(x+h,y)-3u(x,y)-u(x+2h,y)}{2h}$$

which is such that

$$\nabla_x u(x,y) = \overline{\nabla}_x^h u(x,y) + \mathrm{O}(h^2). \tag{17}$$

The key observation is that

$$\frac{h^2}{2}\triangle^h u(x+h,y) = -h\overline{\nabla}_x^h u(x,y) + u(x,y) - P^h u(x,y) \tag{18}$$

with

$$P^h u(x,y) = \frac{u(x+h,y+h)+u(x+h,y-h)}{2}.$$

Let u be the solution to

$$\frac{a_i}{2}\triangle u(x,y) = f(x,y) \text{ in } D_i \tag{19}$$

for $a_i > 0$ and D_i defined previously. We shall now use the previous approximations on both sides of the interface and take the transmission conditions into account. If u solves (19), then u is smooth on D_i, continuous on Γ and satisfies the transmission condition

$$a_1 \lim_{h\to 0} \frac{u(h,y)-u(0,y)}{h} = a_2 \lim_{h\to 0} \frac{u(-h,y)-u(0,y)}{-h},\ \forall (0,y)\in\Gamma. \tag{20}$$

We define two parameters $h_1, h_2 > 0$ both of order $\mathrm{O}(h)$ for the finite different approximations in each subdomain. Thanks to (17),

$$a_1\nabla_x u(0,y) = a_1\overline{\nabla}_x^{h_1} u(0,y) + \mathrm{O}(a_1 h_1^2)$$

and

$$a_2 \nabla_x u(0,y) = a_2 \overline{\nabla}_x^{-h_2} u(0,y) + \mathrm{O}(a_2 h_2^2).$$

The transmission condition (20) becomes

$$a_1 \overline{\nabla}_x^{h_1} u(0,y) = a_2 \overline{\nabla}_x^{-h_2} u(0,y) + \mathrm{O}(a_2 h_2^2 + a_1 h_1^2). \tag{21}$$

In D_1, we have

$$\frac{a_1 h_1^2}{2} \triangle^{h_1} u(x+h_1,y) = h_1^2 f(x+h_1,y) + \mathrm{O}(a_1 h_1^3) \tag{22a}$$

and in D_2, we have similarly

$$\frac{a_2 h_2^2}{2} \triangle^{-h_2} u(x-h_2,y) = h_2^2 f(x-h_2,y) + \mathrm{O}(a_2 h_2^3). \tag{22b}$$

With (18), (22a) becomes

$$a_1 u(x,y) - a_1 h_1 \overline{\nabla}^{h_1} u(x,y) - a_1 P^{h_1} u(x,y) = h_1^2 f(x+h_1,y) + \mathrm{O}(a_1 h_1^3), \tag{23a}$$

while (22b) becomes

$$a_2 u(x,y) + a_2 h_2 \overline{\nabla}^{-h_2} u(x,y) - a_2 P^{-h_2} u(x,y) = h_2^2 f(x-h_2,y) + \mathrm{O}(a_2 h_3^3). \tag{23b}$$

Multiplying (23a) by h_2 and (23b) by h_1 and summing, we obtain thanks to (21) at $x = 0$,

$$\begin{aligned} u(0,y) = {} & \frac{a_1 h_2}{a_1 h_2 + a_2 h_1} P^{h_1} u(0,y) + \frac{a_2 h_1}{a_1 h_2 + a_2 h_1} P^{-h_2} u(0,y) \\ & + \frac{h_1^2 h_2}{a_1 h_2 + a_2 h_1} f(h_1,y) + \frac{h_1 h_2^2}{a_1 h_2 + a_2 h_1} f(-h_2,y) + \mathrm{O}(a_1 h_1^2 h_2 + a_2 h_2^2 h_1). \end{aligned} \tag{24}$$

6.3 Choice of the parameters

There are now three natural choices for the parameters h_1 and h_2 with which one recovers modified versions of already proposed schemes. The first choice corresponds to the kinetic scheme, where the direction which was originally chosen uniformly in $(0, 2\pi)$ is replaced by a random variable taking only four values. In the second choice, the probabilities to go to one side or another of the interface are the same than in the one-dimensional approach of [30]. The third choice corresponds to the approach where the Skew Brownian motion is used. However the new schemes now rely on positions of replacements that are not normal to the interface.

We rewrite (24) in a generic form as

$$u(0,y) = pP^{h_1}u(0,y) + (1-p)P^{-h_2}(0,y) + R(f)(y) + \mathrm{O}(h^3) \tag{25}$$

with $p \in (0,1)$.

First choice: $h = a_1 h_2 = a_2 h_1$**.** Then

$$p = 1-p = \frac{1}{2} \text{ and } R(f)(y) = \frac{h^2}{2a_2^2 a_1} f(h_1, y) + \frac{h^2}{2a_1 a_2^2} f(-h_2, y).$$

Second choice: $h_1 = h_2 = h$**.** Then

$$p = \frac{a_1}{a_1+a_2},\ 1-p = \frac{a_2}{a_1+a_2} \text{ and } R(f)(y) = \frac{h^2}{a_1+a_2}(f(h,y) + f(-h,y)).$$

Third choice: $h_1 = h/\sqrt{a_2}$**,** $h_2 = h/\sqrt{a_1}$**.** Then

$$p = \frac{\sqrt{a_1}}{\sqrt{a_2}+\sqrt{a_1}},\ 1-p = \frac{\sqrt{a_2}}{\sqrt{a_2}+\sqrt{a_1}}$$
$$\text{and } R(f)(y) = \frac{h^2(a_2\sqrt{a_1})^{-1}}{\sqrt{a_1}+\sqrt{a_2}} f(h_1, y) + \frac{h^2(\sqrt{a_2}a_1)^{-1}}{\sqrt{a_1}+\sqrt{a_2}} f(-h_2, y).$$

6.4 The algorithm

Except when $h_1 = h_2$, the grids in D_1 and D_2 do not match on the interface. They could nevertheless be used as probabilistic schemes. Comparing (25) with (5), we define

$$\mathcal{G}(y) = \{(h_1, y-h_1), (h_1, y+h_1), (-h_2, y-h_2), (-h_2, y+h_2)\}$$

and a random variable ξ which takes its values in $\mathcal{G}(y)$ with

$$\mathbb{P}[\xi = (h_1, h_1)] = \mathbb{P}[\xi = (h_1, -h_1)] = \frac{p}{2}$$
$$\text{and } \mathbb{P}[\xi = (-h_2, h_1)] = \mathbb{P}[\xi = (-h_2, -h_1)] = \frac{1-p}{2}.$$

Furthermore, the local scores defined in Section 2.1 are given by $\sigma_k = R(f)(y)$.

For example, for the second choice of parameters,

$$R(f)(y) = \frac{h^2}{a_1+a_2}(f(h,y) + f(-h,y)),$$
$$\mathbb{P}[\xi = (h,h)] = \mathbb{P}[\xi = (h,-h)] = \frac{a_1}{2(a_1+a_2)}$$
$$\text{and } \mathbb{P}[\xi = (-h,h)] = \mathbb{P}[\xi = (-h,-h)] = \frac{a_2}{2(a_1+a_2)}.$$

The operator $\overline{P} = pP^{h_1} + (1-p)P^{-h_2}$ may be expressed as some expectation: $\overline{P}u(0,y) = \mathbb{E}[u(\xi)]$. Using ξ for replacing the particle away from the interface is then justified by (25) and the considerations of Section 2.3. Here, ξ serves as an approximation of $X_{\theta(x)}$ with $X_0 = (0,y)$ for some stopping time $\theta(x)$ which is not known. The score $\overline{S}(0,y)$ is equal to $R(f)(y)$. When $f = 1$, $\overline{S}(0,y) = R(1)(y)$ is an approximation of $\mathbb{E}[\theta(x)]$. In the parabolic case, this justifies to increment the time by $R(1)(y)$ (which in fact does not depend on y). The 2D finite differences algorithm is described in Algorithm 8.

Data: A position $(X,Y) = (0,y)$ of the particle at time t and a small parameter h.
Result: A position (X,Y) of the particle away from the interface at time $t + R(1)$.
Move the particle to one of the four points of $\mathcal{G}(y)$ using a realization of ξ based on one of the three choices of Section 6.3;
Increment the time by $R(1)$;

Algorithm 8: 2D Finite differences.

7 Numerical results

The aim of this numerical part is to compare the different schemes described in the previous sections on three different problems (elliptic, parabolic and eigenvalue problems) which require a finer and finer description of the stochastic process related to the divergence form operator. We really want to focus on the behaviour of the schemes near the interface of discontinuity of the diffusion coefficient and preferably not on the other parameters of these schemes. In order to do that, we consider for all three problems a very simple square domain $D = [-1,1]^2$ divided in two rectangular subdomains $D_1 = [-1,0] \times [-1,1]$ and $D_2 = [0,1] \times [-1,1]$ with respectively a_1 and a_2 for diffusion coefficients.

The elliptic problem only requires the computation of $\mathbb{E}_x[\tau_D]$ of the stochastic process X where τ_D is the exit time from D. For the parabolic problem, we need to approximate the law of the exit time τ_D via its distribution function $F(t) = \mathbb{P}_x[\tau_D \leq t]$ at a fixed time t. The eigenvalue problem requires the approximation of $F(t)$ for large values of t and more importantly the density of X_t given $t > \tau_D$.

The WOR method has no bias in such subdomains and for the WOS method we chose a very small absorption parameter $\varepsilon = 10^{-6}$ such that this parameter has a weak influence on the bias. The number of Monte Carlo simulations is large. Finally, for the first two problems, the starting point is taken close to the center of the domain in order to maximize the number of hits of the interface. This enables

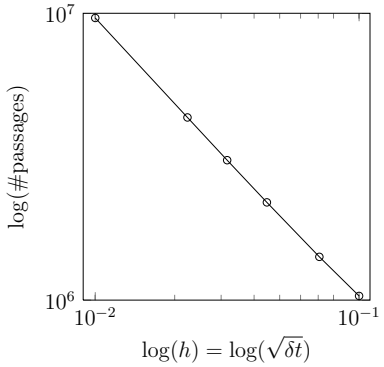


Figure 2: Number of passages at the interface in function of the parameter h.

to discriminate more efficiently the different schemes and reduces the impact of the other boundaries. Obviously these numerical tests are not designed for an optimization of the accuracy as a function of the CPU times as we focus mainly on the bias of the different schemes.

7.1 Number of passages at the interface

Following the discussion on the error analysis of Section 2.3, we show here that the number of passages at the interface is roughly of order $\mathrm{O}(h^{-1})$, where h is a parameter which characterizes the mean distance at which the particle is replaced away from the interface.

We have performed $N = 5 \cdot 10^5$ simulations using the mixing scheme, where the particle is placed away from the interface after a time step δt. As asserted by the Giorgi-Nash-Aronson estimate [36], the density of $X_{\delta t}$ is bounded — up to multiplicative constants — above and below by a Gaussian ones. Thus, we set $h = \sqrt{\delta t}$. In Figure 2, we represent in a log-log scale the number of passages N versus the value of $h = \sqrt{\delta t}$ for δt ranging from 10^{-4} to 10^{-2}. Using a linear regression, we have that $N = Ch^{-0.97}$, which shows that roughly, $N = \mathrm{O}(h^{-1})$ for this simple geometry.

7.2 Elliptic problem: mean exit time computation

We consider the Poisson type equation

$$\frac{1}{2}\nabla(a(x)\nabla u(x)) = -1$$

with $a_1 = 20$, $a_2 = 2$ and Dirichlet boundary conditions. The solution $u(x, y) = E_{(x,y)}[\tau_D]$ is computed at point $(x, y) \simeq (-0.1019, -0.00192)$ and we have $u(x, y) \simeq$

0.08765 thanks to a computation performed with the `Pdetool` package of `Matlab` with a very fine mesh. Inside each subdomain, the WOS method is used until reaching the absorption boundary layer $\varepsilon = 10^{-6}$. The contribution of a sphere of radius r in the subdomain D_i to the total score is simply $\frac{r^2}{2a_i}$. The number of Monte Carlo simulations is $N = 2 \cdot 10^6$. We divide the methods to deal with the interface conditions in three categories: normal schemes, 2D schemes and 2D finite differences methods. We first compare the different methods inside their own category. To give an indication of the Monte Carlo error, we have obtained for all methods $\frac{\sigma}{\sqrt{N}} \simeq 9 \cdot 10^{-5}$. We have made some other tests on either different starting points or different values of the diffusion coefficients. The conclusions were more or less the same than the ones we now present on this particular example.

7.2.1 Normal schemes

The schemes belonging to this category are the Skew (Algorithm 2), Hoteit (Algorithm 3) and the 1D finite differences (Algorithm 4). When the particle reaches the interface between the two subdomains, it is moved only in the direction normal to the interface. We observe that the three methods are accurate up to two or three digits for the parameters we consider. The skew scheme and the Hoteit scheme have a very similar accuracy. The Hoteit scheme is more competitive only because it takes more time to simulate a Gaussian random variable than a uniform one. The 1D finite difference scheme performs slightly better than the two other schemes. It certainly takes benefit that the other boundaries are very far from the interface. The values of the parameter δt of the Skew and Hoteit schemes and the parameter h of the finite Difference scheme have been chosen to provide comparable CPU times.

	Skew		Hoteit		Finite Diff.	
$(\delta t, h)$	Err	CPU	Err	CPU	Err	CPU
$(1 \cdot 10^{-2}, 0.5)$	$1.7 \cdot 10^{-2}$	24	$1.6 \cdot 10^{-2}$	21	$1.6 \cdot 10^{-2}$	17
$(5 \cdot 10^{-3}, 0.4)$	$1.2 \cdot 10^{-2}$	30	$1.1 \cdot 10^{-2}$	26	$1.2 \cdot 10^{-2}$	19
$(2 \cdot 10^{-3}, 0.3)$	$7.4 \cdot 10^{-3}$	41	$6.9 \cdot 10^{-3}$	36	$8.3 \cdot 10^{-3}$	23
$(1 \cdot 10^{-3}, 0.2)$	$5.3 \cdot 10^{-3}$	53	$4.9 \cdot 10^{-3}$	45	$5.2 \cdot 10^{-3}$	29
$(5 \cdot 10^{-4}, 0.1)$	$3.7 \cdot 10^{-3}$	69	$3.5 \cdot 10^{-3}$	59	$2.4 \cdot 10^{-3}$	47

7.2.2 2D schemes

The schemes belonging to this category are the kinetic scheme, the mixing scheme and the scheme based on occupation times. First, these 2D schemes are accurate

up to 4 digits for the parameters we consider. The accuracy of the mixing scheme and the occupation scheme is the same. The mixing scheme is more competitive because it requires the simulation of less random variables. These two new schemes are really more efficient than the kinetic scheme which was already an improvement of the 1D finite differences schemes as observed in [3, 18]. The values of the parameter ε of the kinetic time and the parameter δt of the Mixing and Occupation schemes have been chosen to provide comparable CPU times.

	Kinetic		Mixing		Occupation	
$(\varepsilon, \delta t)$	Err	CPU	Err	CPU	Err	CPU
$(5 \cdot 10^{-2}, 2 \cdot 10^{-2})$	$3.7 \cdot 10^{-2}$	20	$5.9 \cdot 10^{-3}$	17	$6.0 \cdot 10^{-3}$	24
$(3 \cdot 10^{-2}, 1 \cdot 10^{-2})$	$1.6 \cdot 10^{-2}$	27	$1.8 \cdot 10^{-3}$	21	$1.7 \cdot 10^{-3}$	30
$(2 \cdot 10^{-2}, 5 \cdot 10^{-3})$	$7.6 \cdot 10^{-3}$	34	$6.6 \cdot 10^{-4}$	28	$6.4 \cdot 10^{-4}$	40
$(1 \cdot 10^{-2}, 2 \cdot 10^{-3})$	$1.6 \cdot 10^{-3}$	58	$2.6 \cdot 10^{-4}$	39	$2.5 \cdot 10^{-4}$	57
$(7 \cdot 10^{-3}, 1 \cdot 10^{-3})$	$8.3 \cdot 10^{-4}$	77	$1.2 \cdot 10^{-4}$	51	$1.3 \cdot 10^{-4}$	76

7.2.3 2D finite differences methods

The 2D finite differences schemes tested are the three ones described in Section 6.3. The value of h_1 (resp. h_2 and h_3) in the table below is the parameter for the finite difference method corresponding to the first (resp. second and third) replacement method. We can see that all three schemes are accurate up to 4 digits for the parameters we consider. The method based on the second algorithm is the most accurate among the three 2D Finite differences schemes but also among all the methods considered.

	Finite Diff. 1		Finite Diff. 2		Finite Diff. 3	
(h_1, h_2, h_3)	Err	CPU	Err	CPU	Err	CPU
(0.7,0.5,0.6)	$2.2 \cdot 10^{-3}$	27	$3.8 \cdot 10^{-3}$	15	$3.1 \cdot 10^{-3}$	19
(0.6,0.4,0.5)	$1.7 \cdot 10^{-3}$	30	$2.2 \cdot 10^{-3}$	17	$2.2 \cdot 10^{-3}$	21
(0.5,0.3,0.4)	$1.2 \cdot 10^{-3}$	34	$1.1 \cdot 10^{-3}$	20	$1.3 \cdot 10^{-3}$	24
(0.4,0.2,0.3)	$7.5 \cdot 10^{-4}$	40	$4.5 \cdot 10^{-4}$	27	$8.5 \cdot 10^{-4}$	29
(0.3,0.1,0.2)	$4.8 \cdot 10^{-4}$	51	$1.1 \cdot 10^{-4}$	45	$3.5 \cdot 10^{-4}$	40

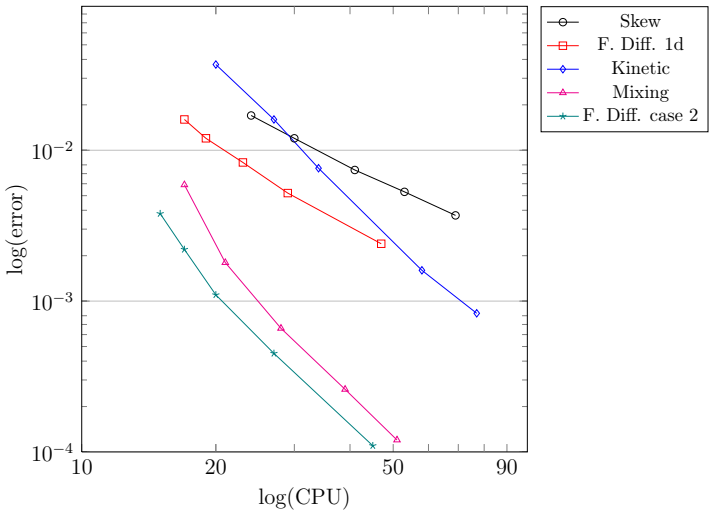


Figure 3: Comparison of error vs. CPU times for some of the schemes in the elliptic test case.

7.2.4 Global comparison

In Figure 3, we compare for 5 methods representative of the previous results the accuracy as a function of CPU times in a logarithmic scale. We can see on the slopes of the different curves that 2D schemes have a higher order of convergence than the 1D schemes. The 2D finite differences scheme is slightly more efficient than the mixing scheme. These two schemes are a lot more accurate than the kinetic, Skew and 1D finite differences scheme.

7.3 Parabolic problem: survival probability computation

We consider the parabolic problem

$$\frac{\partial u(t,x,y)}{\partial t} = \frac{1}{2}\nabla(a(x).\nabla u(t,x,y))$$

with $a_1 = 20$, $a_2 = 2$, Dirichlet boundary conditions and $u(0,x,y) \equiv 1$. The solution $u(t,x,y) = \mathbb{P}_{(x,y)}[\tau_D > t]$ is computed at point $(t,x,y) \simeq (0.1, 0.1006, 0)$ with $N = 10^6$ simulations. We have $u(t,x,y) \simeq 0.11638$ thanks again to a computation performed with the `Pdetool` package of `Matlab`. The contribution of a sphere of radius r in the subdomain D_i to the total time is now $\frac{r^2\tau_1}{a_i}$ where τ_1 is the exit time of the unit circle. In order to reduce the computational times, we precompute 10^7 values of this exit time using the inversion method. To sample from τ_1, we just pick

one value uniformly at random in the precomputed ones. To give an indication of the Monte Carlo error, we have obtained for all methods $\frac{\sigma}{\sqrt{N}} \simeq 3 \cdot 10^{-4}$.

7.3.1 Normal schemes

We observe roughly the same conclusions than in the elliptic case. This parabolic problem is harder to solve as we now have only two digits of accuracy instead of three. The performances of the skew and Hoteit schemes are even closer because more time is spent in the simulation of the exit time from the circles. The 1D Finite differences scheme is still slightly more efficient than the two other schemes.

	Skew		Hoteit		Finite Diff.	
$(\delta t, h)$	Err	CPU	Err	CPU	Err	CPU
$(1 \cdot 10^{-2}, 0.5)$	$8.4 \cdot 10^{-2}$	29	$6.9 \cdot 10^{-2}$	25	$4.8 \cdot 10^{-2}$	19
$(5 \cdot 10^{-3}, 0.4)$	$5.3 \cdot 10^{-2}$	33	$4.8 \cdot 10^{-2}$	31	$3.5 \cdot 10^{-2}$	22
$(2 \cdot 10^{-3}, 0.3)$	$3.2 \cdot 10^{-2}$	44	$2.9 \cdot 10^{-2}$	41	$2.4 \cdot 10^{-2}$	26
$(1 \cdot 10^{-3}, 0.2)$	$2.2 \cdot 10^{-2}$	56	$2.0 \cdot 10^{-2}$	52	$1.4 \cdot 10^{-2}$	32
$(5 \cdot 10^{-4}, 0.1)$	$1.5 \cdot 10^{-2}$	73	$1.4 \cdot 10^{-2}$	67	$6.1 \cdot 10^{-3}$	53

7.3.2 2D schemes

On this example, the kinetic scheme is clearly less efficient than the two others. In opposite to the elliptic case, the occupation scheme seems more efficient than the mixing scheme for two main reasons. First, the simulation of exit times from circles is the main cost of the simulation algorithms: both algorithms have now similar CPU times. Second, the occupation scheme gives a finer description of the stochastic process than the mixing scheme which is useful as we now need more than the mean exit time.

	Kinetic		Mixing		Occupation	
$(\varepsilon, \delta t)$	Err	CPU	Err	CPU	Err	CPU
$(4 \cdot 10^{-2}, 5 \cdot 10^{-2})$	$9.4 \cdot 10^{-2}$	25	$1.4 \cdot 10^{-2}$	18	$1.4 \cdot 10^{-2}$	19
$(3 \cdot 10^{-2}, 3 \cdot 10^{-2})$	$5.5 \cdot 10^{-2}$	30	$3.6 \cdot 10^{-3}$	23	$3.5 \cdot 10^{-3}$	24
$(2 \cdot 10^{-2}, 2 \cdot 10^{-2})$	$2.4 \cdot 10^{-2}$	38	$8.5 \cdot 10^{-4}$	29	$1.1 \cdot 10^{-3}$	30
$(1 \cdot 10^{-2}, 1 \cdot 10^{-2})$	$6.0 \cdot 10^{-3}$	63	$6.5 \cdot 10^{-4}$	41	$2.4 \cdot 10^{-4}$	42
$(8 \cdot 10^{-3}, 7 \cdot 10^{-3})$	$4.4 \cdot 10^{-3}$	74	$3.6 \cdot 10^{-4}$	53	$1.2 \cdot 10^{-4}$	55

7.3.3 2D finite differences methods

The second finite differences scheme is still the most efficient on this parabolic problem. All three algorithms are also less accurate than in the elliptic case and two of them hardly reach an accuracy of three digits.

	Finite Diff. 1		Finite Diff. 2		Finite Diff. 3	
(h_1, h_2, h_3)	Err	CPU	Err	CPU	Err	CPU
(0.7,0.5,0.6)	$1.2 \cdot 10^{-2}$	31	$1.5 \cdot 10^{-2}$	18	$1.2 \cdot 10^{-2}$	21
(0.6,0.4,0.5)	$9.0 \cdot 10^{-3}$	34	$1.0 \cdot 10^{-2}$	20	$9.7 \cdot 10^{-3}$	24
(0.5,0.3,0.4)	$6.6 \cdot 10^{-3}$	39	$6.3 \cdot 10^{-3}$	23	$6.7 \cdot 10^{-3}$	27
(0.4,0.2,0.3)	$4.1 \cdot 10^{-3}$	46	$2.3 \cdot 10^{-3}$	30	$4.0 \cdot 10^{-3}$	33
(0.3,0.1,0.2)	$2.2 \cdot 10^{-3}$	57	$2.8 \cdot 10^{-4}$	50	$1.8 \cdot 10^{-3}$	44

7.3.4 Global comparison

We compare the same schemes than in the elliptic case except the mixing scheme that has been replaced by the occupation scheme which is more efficient in the parabolic case. On this parabolic problem, the 2D finite differences scheme and the occupation scheme are even more efficient compared to the three other schemes. Nevertheless the occupation scheme is now preferable to the 2D finite differences maybe because it gives a finer approximation of the exit time.

7.4 Principal eigenelements computation

7.4.1 Description of the method

The method we propose to compute the principal eigenelements of the divergence form operator $L = \frac{1}{2}\nabla(a\nabla)$ in a bounded domain D is less standard than the resolution of the elliptic and parabolic problems studied in the previous subsections. This method was originally designed to compute the principal eigenvalue of neutron transport operators [23, 24] and was then applied to the Laplace operator in [19, 20]. The idea is to combine the spectral expansion of the solution of a Cauchy problem relative to the operator and the Monte Carlo approximation of its Feynman-Kac representation.

The operator $-L$ is self-adjoint, has a countable basis of eigenfunctions $\{\varphi_k\}_{k\in\mathbb{N}}$ corresponding to positive eigenvalues $\{\lambda_k\}_{k\in\mathbb{N}}$ and is hence very similar to the Laplace operator. The solution $u(t, x, y) = \mathbb{P}_{(x,y)}[\tau_D > t]$ of the parabolic problem

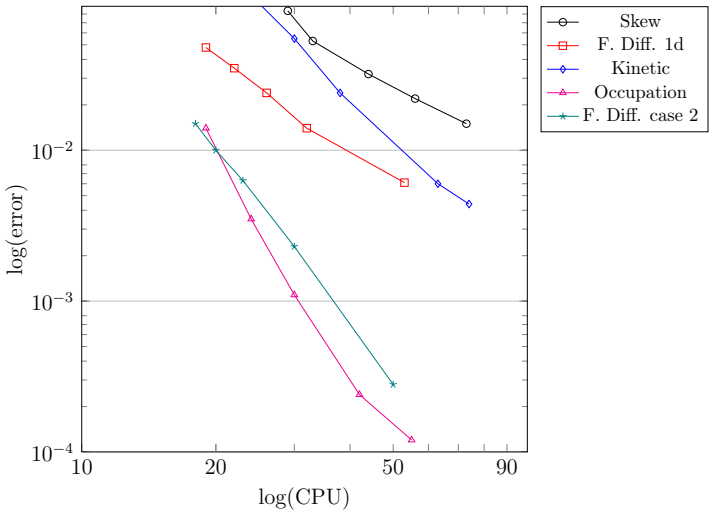


Figure 4: Comparison of error vs. CPU time for some of the schemes in the parabolic test case.

of Section 7.3 verifies

$$\mathbb{P}_{(x,y)}[\tau_D > t] = \sum_{k=1}^{\infty} (u_0, \varphi_k) \exp(-\lambda_k t) \varphi_k(x,y) \simeq \beta \exp(-\lambda_1 t) + \mathrm{O}(\exp(-\lambda_2 t))$$

which provides an estimator $\lambda_1(t_1, t_2)$ of λ_1 defined by

$$\lambda_1(t_1, t_2) = -\frac{\ln(\mathbb{P}_{(x,y)}[\tau_D > t_2]) - \ln(\mathbb{P}_{(x,y)}[\tau_D > t_1])}{t_2 - t_1}$$

using two discretization times t_1 and t_2 chosen large enough so that the term $\mathrm{O}(\exp(-\lambda_2 t))$ is negligible. Some more sophisticated estimators based on regression have also been introduced in [20].

The survival probability $\mathbb{P}_{(x,y)}[\tau_D > T]$ is very small for T large and thus we have introduced in [19] a splitting method to compute more accurately this survival probability than with a crude Monte Carlo method. We write for $t < T$

$$\mathbb{P}_{\mu}[\tau_D > T] = \mathbb{P}_{\pi_t}[\tau_D > T - t] \mathbb{P}_{\mu}[\tau_D > t]$$

where μ is the law of Z_0 and π_t the law of Z_t conditioned by the event $\{\tau_D > t\}$. The conditional law π_t is approximated by the empirical law of the particles still alive at time t. Using the spectral expansion, we can prove that this conditional law converges to φ_1 up to a multiplicative constant when T goes to infinity. The

$t_i = i\Delta t$	$\mathbb{P}_{\pi_{i-1}}[\tau > t_i]$	$\mathbb{P}_{\mu}[\tau > t_i]$
0.1	0.203	0.2034
0.2	0.320	0.0651
0.3	0.328	0.0213
0.4	0.330	0.0071
0.5	0.328	0.0023
0.6	0.329	0.0007

Table 2: Survival probabilities for the mixing scheme for $N = 2 \cdot 10^5$ particle when $\Delta t = 0.1$ and the timestep is $\delta t = 10^{-3}$.

survival probability is now approximated by a product of the probabilities of two events that are less rare and hence more accurately approximated by means of Monte Carlo procedures.

In order to chose the times t_1 and t_2 online, we use the splitting method iteratively until convergence of the empirical approximation of π_t to φ_1. We first choose a step Δt and an initial distribution μ such that $P_\mu[\tau_D > \Delta t]$ is sufficiently large. We compute iteratively

$$\mathbb{P}_{\pi_{n\Delta t}}[\tau_D > 2\Delta t] = \mathbb{P}_{\pi_{(n+1)\Delta t}}[\tau_D > \Delta t]\mathbb{P}_{\pi_{n\Delta t}}[\tau_D > \Delta t]$$

until this probability converges to a fix value. This means that $\pi_{n\Delta t}$ has converged to φ_1. Then, we set $t_1 = n\Delta t, t_2 = (n+1)\Delta t$ and compute $\lambda_1(t_1, t_2)$ thanks to the transition probabilities from a slice to the next one.

As observed in [19, 20], the bias of the method is very sensitive to the approximation of the stochastic process Z. It is hence a very good test for the different methods of simulation of Z. at the interface between subdomains we have introduced and studied in the previous sections. For the simulation in the subdomains, the walk on rectangles method will be used for the reasons explained in Section 3.2.

7.4.2 Numerical results

We test our method in the previous domain with initial distribution $\mu = \pi_0$ the uniform law in $D_{\text{init}} = [-0.8, 0.8]^2$ and with a step $\Delta t = 0.1$. The approximate value of λ_1 using `Matlab` is 11.095.

Table 2 shows the survival probabilities for the mixing scheme. We see that $\mathbb{P}_{\pi_{t_{i-1}}}[t > \tau_i]$ fluctuates around 0.329 as soon as we have reached the third slice at time $t_3 = 0.3$. We can hence assume that we are in the quasi-stationary regime for larger times. Similar results were obtained for the other schemes we have tested.

Table 3 shows the eigenvalue approximation for three schemes: mixing, skew and Finite difference 2 using $\lambda(t_1, t_2)$ with $t_1 = 0.3$ and $t_2 = 0.6$. We denote by

	Mixing		Skew		Finite Diff. 2		
δt	Error	CPU	Error	CPU	h	Error	CPU
10^{-2}	$-4.1 \cdot 10^{-2}$	77	$5.4 \cdot 10^{-1}$	86	0.34	$-1.8 \cdot 10^{-1}$	74
10^{-3}	$-5.3 \cdot 10^{-2}$	152	$1.3 \cdot 10^{-1}$	161	0.104	$-5.8 \cdot 10^{-2}$	134
10^{-4}	$-2.7 \cdot 10^{-2}$	358	$2.6 \cdot 10^{-2}$	355	0.034	$-4.8 \cdot 10^{-2}$	307
10^{-5}	$-4.4 \cdot 10^{-2}$	1,004	$-2.2 \cdot 10^{-2}$	994	0.011	$-5.7 \cdot 10^{-2}$	852

Table 3: Absolute error $\lambda(t_1, t_2) - \lambda_1$ of the eigenvalue estimator for the three schemes with $t_1 = 0.3$ and $t_2 = 0.6$ for $\Delta t = 0.1$ and $N = 2 \cdot 10^5$ particles.

Error the absolute error $\lambda(t_1, t_2) - \lambda_1$. We observe that the results are very good for small values of h for the finite difference scheme and a small time step δt for the normal scheme. For the mixing scheme, the error fluctuates around $4 \cdot 10^{-2}$, which means that the first eigenvalue λ_1 is accurately estimated even with the largest values of δt. In any case, we recover the same relative error we have obtained in our previous paper [19] for the computation of the principal eigenvalue of the Laplace operator. This confirms the efficiency of our new schemes and especially of the mixing scheme for problems of parabolic type.

8 Conclusion

We have introduced new schemes for the simulation of diffusions in discontinuous, locally isotropic media. They take into account the two-dimensional aspects of the diffusion by means of stochastic process techniques based on occupation times or by using finite differences approximation. These new schemes have both better accuracy and better order than the schemes based on normal schemes. In fact, we have shown theoretically and numerically on various examples that the new schemes reach an order $\mathrm{O}(h^2)$ instead of an order $\mathrm{O}(h)$ for the normal schemes. The kinetic scheme provided in [18] has also a an order close to a $\mathrm{O}(h^2)$ but is clearly less accurate than the new schemes.

In our numerical test cases, we have chosen to use the random walk on spheres or on rectangles methods because they are both very fast and almost unbiased. This enables to focus on the error induced by the schemes dealing with the replacements at the interface. Yet there is no problem in using any of the proposed schemes locally close to a discontinuity, and other simulation schemes such as the Euler scheme, in regions where the diffusivity is smooth enough, but not necessarily constant. For example, we could simulate the process until it reaches the boundary using the Euler scheme for killed diffusion process (see *e.g.* [10]) and then restart the particle at the interface with one of the proposed schemes. This implies that

the diffusivity coefficients vary slowly around the interface so that they could be approximated locally by constant ones, as done for example in [8, 27–29]. However, at the best of our knowledge, quantifying the error done by mixing two schemes used in different regions remains a largely open and difficult problem which requires some precise estimates of the time spend by the particle in each region (See for example [2] for a related discussion in the one-dimensional case).

Acknowledgment The authors wish to thank the referees whose remarks improved this article.

References

[1] P. Baldi, *Exact asymptotics for the probability of exit from a domain and applications to simulation*, Ann. Probab. **23** (1995), no. 4, 1644–1670.

[2] F. Bernardin, M. Bossy, M. Martinez, and D. Talay, *On mean numbers of passage times in small balls of discretized Itô processes*, Electron. Commun. Probab. **14** (2009), 302–316, DOI 10.1214/ECP.v14-1479.

[3] M. Bossy, N. Champagnat, S. Maire, and D. Talay, *Probabilistic interpretation and random walk on spheres algorithms for the Poisson-Boltzmann equation in Molecular Dynamics*, ESAIM M2AN **44** (2010), no. 5, 997–1048.

[4] R.S. Cantrell and C. Cosner, *Diffusion Models for Population Dynamics Incorporating Individual Behavior at Boundaries: Applications to Refuge Design*, Theoretical Population Biology **55** (1999), no. 2, 189–207, DOI 10.1006/tpbi.1998.1397.

[5] M. Deaconu and A. Lejay, *A random walk on rectangles algorithm*, Methodol. Comput. Appl. Probab. **8** (2006), no. 1, 135–151, DOI 10.1007/s11009-006-7292-3.

[6] P. Étoré, *On random walk simulation of one-dimensional diffusion processes with discontinuous coefficients*, Electron. J. Probab. **11** (2006), no. 9, 249–275.

[7] P. Étoré and A. Lejay, *A Donsker theorem to simulate one-dimensional processes with measurable coefficients*, ESAIM Probab. Stat. **11** (2007), 301–326, DOI 10.1051/ps:2007021.

[8] P. Étoré and M. Martinez, *Exact simulation of one-dimensional stochastic differential equations involving the local time at zero of the unknown process* (2011), available at `arXiv:1102.2565`. Preprint.

[9] O. Faure, *Simulation du mouvement brownien et des diffusions*, Ph.D. thesis, École Nationale des Ponts et Chaussées, 1992.

[10] Emmanuel Gobet, *Euler schemes and half-space approximation for the simulation of diffusion in a domain*, ESAIM Probab. Statist. **5** (2001), 261–297, DOI 10.1051/ps:2001112.

[11] U. Gräwe, E. Deleersnijder, S.H.A.M. Shah, and A.W. Heemink, *Why the Euler scheme in particle tracking is not enough: the shallow-sea pycnocline test case*, Ocean Dynamics **62** (2012), no. 4, 501–514.

[12] H. Hoteit, R. Mose, A. Younes, F. Lehmann, and Ph. Ackerer, *Three-dimensional modeling of mass transfer in porous media using the mixed hybrid finite elements and the random-walk methods*, Math. Geology **34** (2002), no. 4, 435–456.

[13] O. A. Ladyženskaja, V. Ja. Rivkind, and N. N. Ural′ceva, *Solvability of diffraction problems in the classical sense*, Trudy Mat. Inst. Steklov. **92** (1966), 116–146.

[14] O. A. Ladyženskaja, V. A. Solonnikov, and N. N. Ural′ceva, *Linear and quasilinear equations of parabolic type*, Translations of Mathematical Monographs, Vol. 23, American Mathematical Society, Providence, R.I., 1967.

[15] A. Lejay, *Simulation of a stochastic process in a discontinuous layered medium*, Electron. Comm. Probab. **16** (2011), 764–774.

[16] A. Lejay, *On the constructions of the Skew Brownian motion*, Probab. Surv. **3** (2006), 413–466.

[17] A. Lejay, *Simulating a diffusion on a graph. Application to reservoir engineering*, Monte Carlo Methods Appl. **9** (2003), no. 3, 241–256.

[18] A. Lejay and S. Maire, *Simulating diffusions with piecewise constant coefficients using a kinetic approximation*, Comput. Methods Appl. Mech. Engrg. **199** (2010), no. 29-32, 2014-2023, DOI 10.1016/j.cma.2010.03.002.

[19] A. Lejay and S. Maire, *Computing the first eigenelements of some linear operators using a branching Monte Carlo method*, J. Comput. Phys. **227** (2008), no. 23, 9794–9806, DOI 10.1016/j.jcp.2008.07.018.

[20] A. Lejay and S. Maire, *Computing the principal eigenvalue of the Laplace operator by a stochastic method*, Math. Comput. Simulation **73** (2007), no. 3, 351–363, DOI 10.1016/j.matcom.2006.06.011.

[21] A. Lejay and M. Martinez, *A scheme for simulating one-dimensional diffusion processes with discontinuous coefficients*, Ann. Appl. Probab. **16** (2006), no. 1, 107–139, DOI 10.1214/105051605000000656.

[22] A. Lejay and G. Pichot, *Simulating diffusion processes in discontinuous media: a numerical scheme with constant time steps*, J. Comput. Phys. **231** (2012), no. 21, 7299–7314, DOI 10.1016/j.jcp.2012.07.011.

[23] S. Maire, *Réduction de variance pour l'intégration numérique et pour le calcul critique en transport neutronique*, Ph.D. thesis, Université de Toulon et du Var, 2001.

[24] S. Maire and D. Talay, *On a Monte Carlo method for neutron transport criticality computations*, IMA J. Numer. Anal. **26** (2006), no. 4, 657–685.

[25] S. Maire and É. Tanré, *Monte Carlo approximations of the Neumann problem* (2012). Preprint.

[26] S. Maire and E. Tanré, *Some new simulation schemes for the evaluation of Feynman-Kac representations*, Monte Carlo methods Appl. **14** (2008), no. 1, 29–51.

[27] M. Martinez, *Interprétations probabilistes d'opérateurs sous forme divergence et analyse de méthodes numériques associées*, Ph.D. thesis, Université de Provence / INRIA Sophia-Antipolis, 2004.

[28] M. Martinez and D. Talay, *Discrétisation d'équations différentielles stochastiques unidimensionnelles à générateur sous forme divergence avec coefficient discontinu*, C. R. Math. Acad. Sci. Paris **342** (2006), no. 1, 51–56, DOI 10.1016/j.crma.2005.10.025.

[29] M. Martinez and D. Talay, *One-Dimensional parabolic diffraction equations: Pointwise estimates and discretization of related stochastic differential equations with weighted local times*, Electron. J. Probab. **17** (2012), no. 27, 30, DOI 10.1214/EJP.v17-1905.

[30] M. Mascagni and N. A. Simonov, *Monte Carlo methods for calculating some physical properties of large molecules*, SIAM J. Sci. Comput. **26** (2004), no. 1, 339–357, DOI 10.1137/S1064827503422221.

[31] G.A. Mikhaĭlov and R.N. Makarov, *Solution of boundary value problems of the second and third kind by the Monte Carlo methods*, Siberian Math. J. **38** (1997), no. 3, 518–527.

[32] G. N. Milstein and M. V. Tretyakov, *Simulation of a space-time bounded diffusion*, Ann. Appl. Probab. **9** (1999), no. 3, 732–779, DOI 10.1214/aoap/1029962812.

[33] M. E. Muller, *Some continuous Monte Carlo methods for the Dirichlet problem*, Ann. Math. Statist. **27** (1956), 569–589.

[34] J. Ramirez, E. Thomann, E. Waymire, J. Chastenet, and B. Wood, *A Note on the Theoretical Foundations of Particle Tracking Methods in Heterogeneous Porous Media*, Water Resour. Res. **44** (2007), W01501, DOI 10.1029/2007WR005914.

[35] P. Salamon, D. Fernàndez-Garcia, and J. J. Gómez-Hernández, *A review and numerical assessment of the random walk particle tracking method*, J. Contaminant Hydrology **87** (2006), no. 3-4, 277–305, DOI 10.1016/j.jconhyd.2006.05.005.

[36] D. W. Stroock, *Diffusion semigroups corresponding to uniformly elliptic divergence form operators*, Séminaire de Probabilités, XXII, Lecture Notes in Math., vol. 1321, Springer, Berlin, 1988, pp. 316–347.

[37] G.J.M. Uffink, *A random walk method for the simulation of macrodispersion in a stratified aquifer*, Relation of Groundwater Quantity and Quality, IAHS Publication (IAHS Publication, ed.), IAHS Press, Walingford, UK, 1985, pp. 103—114.

[38] M. Zhang, *Calculation of diffusive shock acceleration of charged particles by skew Brownian motion*, Astrophys. J. **541** (2000), 428–435.