



HAL
open science

Approximation of solutions of some heat problems with third boundary conditions by means of functional integrals

Jean-Paul Morillon

► **To cite this version:**

Jean-Paul Morillon. Approximation of solutions of some heat problems with third boundary conditions by means of functional integrals. Third International Congress on Industrial and Applied Mathematics, Jul 1995, Hambourg, Germany. hal-00999788

HAL Id: hal-00999788

<https://hal.science/hal-00999788>

Submitted on 4 Jun 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**Approximation of solutions of some heat problems
with third boundary conditions
by means of functional integrals**

J.-P. MORILLON

Université de La Réunion

Laboratoire PIMENT

Email : jean-paul.morillon@univ-reunion.fr

ICIAM '95 — Hamburg

Tuesday, July 4, 1995.

1 Introduction

Numerical solution of boundary value problems can be performed by the usual finite difference or finite element methods and adequate space mesh. Monte Carlo methods can also be relevant; they lead, after equations discretization, to specific treatments related to domain local geometry (see, for instance, References 1–4).

This work is also investigating another method which provides integral representations of the solution to stationary deterministic linear boundary value problems. In particular, the Dirichlet problem is known to have an integral representation⁵ which is used to obtain an algorithm which computes the solution by implementing a random walk simulation.⁶

The ensuing numerical methods do not require the storage of a grid discretization in computer memory. Programming is short, easy to check step by step; moreover, the implementation to a higher dimension requires but a few additional lines.

More specifically, a stationary deterministic linear boundary value problem is to be solved with boundary condition of Fourier or mixed types. We show, in particular, using examples, that the boundary condition treatments can be reduced to the counting of absorptions and reflections on the boundaries.

Section 2 is devoted to the stochastic representations of solutions. In Section 3, we present the approaches of these representations by the realizations of random processes, and we establish the corresponding computational algorithms. In Section 4, numerical simulations of random walks provide results associated with two- or three-dimensional geometries. These methods are discussed in the last section.

2 Representation of solutions

This section is devoted to the stochastic representations of solutions of linear boundary value problems by means of stochastic integrals. The Fourier, and mixed boundary conditions are presented.

Let us consider an open bounded region G in \mathbb{R}^d (the dimension d being fixed). Let us denote $x \in \mathbb{R}^d$ the space variable and n the unit inward normal defined on the boundary ∂G .

We will show how, from stochastic differential equations (SDE) (for these SDE, see for instance References 7–10, and references therein), we can represent solutions of some boundary value problems. We will use the expectation of integrals, i.e. functionals of trajectories and random functions which are solutions of SDE. The application of Itô's formula associated with SDE leads to a representation of solutions.

From a functional point of view, the obtained representations have only been established according to regularity hypotheses on the geometry of domain G , and on the boundary data. Nevertheless, we are using here the stochastic representations even when data are not necessarily smooth : for instance, the boundary ∂G of domain can show corners or edges (this procedure previously proved to be effective in References 1, 2 and 4 for the discretized equations and in Reference 6 for the Dirichlet problem).

2.1 Problem with Fourier boundary condition

Let us consider the following problem with Fourier boundary condition :

$$\left\{ \begin{array}{l} -\frac{1}{2} \Delta u = f \quad \text{in } G, \\ \frac{\partial u}{\partial n} - \mu u = g \quad \text{on } \partial G, \end{array} \right. \quad (1)$$

where the function u , defined from G into \mathbb{R} , is to be determined, and the data are the functions :

$$f : G \longrightarrow \mathbb{R}, \quad \text{and} \quad g : \partial G \longrightarrow \mathbb{R},$$

and μ is a strictly positive constant.

Let us introduce the Markov process defined in G by the Laplace operator within the domain, or by reflection on the boundary in the direction of the inward normal :

$$X_t^x = x + W_t + \int_0^t \mathbb{1}_{\partial G} (X_s^x) n (X_s^x) d\xi_s^x, \quad t \geq 0, \quad (2)$$

where W_t is the standard Wiener process with values in \mathbb{R}^d so that : $X_0^x = x$, $\mathbb{1}_{\partial G}$ is the indicator of ∂G , and ξ_t^x is a non-decreasing process which increases only when t hits the set $\{s > 0 \mid X_s^x \in \partial G\}$, so that : $\xi_0^x = 0$.

Some functional results regarding SDE (2) with reflection on ∂G , are given in References 5 and 11.

The application of Itô's formula to :

$$u (X_t^x) \exp \left(-\mu \int_0^t \mathbb{1}_{\partial G} (X_s^x) d\xi_s^x \right)$$

leads to the representation of the solution of (1) in the form :

$$u(x) = E [Y], \quad x \in G, \quad (3)$$

with the random variable :

$$\begin{aligned} Y &= \int_0^{+\infty} f (X_t^x) \exp \left(-\mu \int_0^t \mathbb{1}_{\partial G} (X_s^x) d\xi_s^x \right) dt \\ &- \int_0^{+\infty} g (X_t^x) \exp \left(-\mu \int_0^t \mathbb{1}_{\partial G} (X_s^x) d\xi_s^x \right) \mathbb{1}_{\partial G} (X_t^x) d\xi_t^x. \end{aligned} \quad (4)$$

2.2 Problem with mixed boundary conditions

The partition $\partial G = \Gamma_D \cup \Gamma_F$ is considered, where the parts Γ_D , and Γ_F correspond respectively to the Dirichlet, and Fourier boundary conditions. Let us now consider the following problem with mixed boundary conditions, and written down with the above notation :

$$\begin{cases} -\frac{1}{2} \Delta u = f & (G) \\ u = g_1 & (\Gamma_D) \\ \frac{\partial u}{\partial n} - \mu u = g_2 & (\Gamma_F) \end{cases} \quad (5)$$

where μ is a strictly positive constant.

Let us introduce the Markov process : $X_t^x = x + W_t + \int_0^t \mathbb{1}_{\Gamma_F}(X_s^x) n(X_s^x) d\xi_s^x$, $t \geq 0$, where $\mathbb{1}_{\Gamma_F}$ is the indicator of the set Γ_F , and where ξ_t^x is a non-decreasing process which only increases when t hits the set $\{s > 0 \mid X_s^x \in \Gamma_F\}$, so that $\xi_0^x = 0$.

The hitting time τ of Γ_D is defined by : $\tau = \inf \{t > 0 \mid X_t^x \in \Gamma_D\}$.

The application of Itô's formula to :

$$u(X_t^x) \exp\left(-\mu \int_0^t \mathbb{1}_{\Gamma_F}(X_s^x) d\xi_s^x\right)$$

leads to the representation of the solution of (5) in the form :

$$u(x) = E [Y] \quad (6)$$

with the random variable :

$$\begin{aligned} Y &= \int_0^\tau f(X_t^x) \exp\left(-\mu \int_0^t \mathbb{1}_{\Gamma_F}(X_s^x) d\xi_s^x\right) dt + g_1(X_\tau^x) \exp\left(-\mu \int_0^\tau \mathbb{1}_{\Gamma_F}(X_s^x) d\xi_s^x\right) \mathbb{1}_{\Gamma_D}(X_\tau^x) \\ &\quad - \int_0^\tau g_2(X_t^x) \exp\left(-\mu \int_0^t \mathbb{1}_{\Gamma_F}(X_s^x) d\xi_s^x\right) \mathbb{1}_{\Gamma_F}(X_t^x) d\xi_t^x \end{aligned} \quad (7)$$

3 Resolution algorithms

The main purpose of this section is to give an approach of the representations of Section 2 by means of realization of random processes, in order to compute the solutions of boundary value problems.

The solution u to problem (1), respectively (6), is represented as the mean of a random variable Y involving functions of a reflected random walk, respectively. of a reflected then absorbed random walk.

The representation (3–5), respectively (6–7), shows that it is sufficient to compute NT approximated values of Y , i.e. Y_1, \dots, Y_{NT} , in order to obtain an estimation of $u(x)$ as the mean :

$$u(x) = \frac{1}{NT} \sum_{n=0}^{NT} Y_n .$$

Each value Y_n results from a simulation of a random walk reflected by the domain boundary and limited by a fixed number NR of reflections, respectively reflected by the boundary part Γ_F , and lastly absorbed by Γ_D . Each simulated random walk is instantaneously reflected by the boundary. Since the simulated process happens to leave the domain, its trajectory resumes to the interior last position ; this procedure goes on until the number of steps reaches the value NR , respectively until the absorption by Γ_D . Each random walk is then finite.

The random processes X_t^x and ξ_t^x are discretized in time by the Euler method ; in particular, if h is the fixed step size of a simulated reflected random walk, then the space, time and reflection increments are defined by :

$$\Delta X_t^x = h, \quad \Delta t = h^2/d \quad \text{and} \quad \Delta \xi_t^x = h .$$

The space step $h > 0$ being fixed, the process X_t^x is simulated by the sequence of random vectors X_0, X_1, \dots defined by means of the recurrent formulae :

$$X_0 = x \in G \text{ (being the initialization)}, \quad \text{and} \quad X_{k+1} = X_k + h D_k, \quad k \in \mathbb{N},$$

where D_k is a random vector so that, if $e_i, i = 1, \dots, d$, is the canonical basis of \mathbb{R}^d , D_k verifies :

$$\text{Prob}(D_k = e_i) = \text{Prob}(D_k = -e_i) = \frac{1}{2d}, \quad i = 1, \dots, d.$$

In the two-dimensional case, for each time interval, the fixed space step h is stepping along one of the two axes of an orthonormal basis. This random axis is chosen with probability 1/2, either in the positive or negative direction, being also chosen with probability 1/2 (see the following basic step algorithm, written in pseudo-Pascal, where the function `random` gives a real number in $[0, 1[$).

Algorithm of random basic step

```

U1 := random;          U2 := random;
if  U1 < 0.5 then if  U2 < 0.5 then X2 := X2 + H
                        else X2 := X2 - H
                        else if  U2 < 0.5 then X1 := X1 + H
                        else X1 := X1 - H;

```

In the two-dimensional case, the boundary reflections is reflecting along one of the two axes of an orthonormal basis (see the following reflection algorithm).

Reflection algorithm

```

Procedure Reflection(X1,X2)
{(X1,X2) is the position hitting by the random walk }
{(D1,D2) is a shift generating reflection }
  if  D1 <> 0 then X1 := X1 - D1
  if  D2 <> 0 then X2 := X2 - D2

```


Each simulation gives a realization X_k , $0 \leq k \leq NR$, respectively. $0 \leq k \leq N$, of a random walk in G so that :

$$X_0 = x \in G, \dots, X_{NR-1} \in G, \quad \text{and} \quad X_{NR} \in G,$$

respectively :

$$X_0 = x \in G, \dots, X_{N-1} \in G, \quad \text{and} \quad X_N \in \Gamma_D,$$

which trajectory is the polygonal line connecting successively the points X_k and X_{k+1} , where NR is a fixed number, respectively. N depends on each random walk.

Then this procedure is iterated NT times in order to obtain a sample of Y , and an estimation of its mean, i.e. $u(x)$. The integer NT corresponds to the number of simulated random walks. This integer NT is the maximum index of the main loop of the associated program. The second (and latter) loop follows, step by step, each random walk by adding up successively the values of the source function f , then those of the reflection effect, respectively. and lastly the absorption effect. Finally the score of each corresponding Y_n is added up.

3.1 Problem with Fourier boundary conditions

We fix the maximum number of reflections NR . Let NP be the number of steps of a NR reflections random walk (NP depends on NR). For each simulated reflected random walk (X_k ; $0 \leq k \leq NP$), the following score is computed :

$$\begin{aligned} Y_n = & \Delta t \cdot \sum_{k=1}^{NP} f(X_k) \cdot \exp \left(-\Delta \xi \cdot \mu \cdot \sum_{l=1}^k \mathbb{1}_{\partial G}(X_l) \right) \\ & - \Delta \xi \cdot \sum_{k=1}^{NP} g(X_k) \cdot \exp \left(-\Delta \xi \cdot \mu \cdot \sum_{l=1}^k \mathbb{1}_{\partial G}(X_l) \right) \mathbb{1}_{\partial G}(X_k). \end{aligned}$$

For each reflection, a counter `COUNT` is incremented, and the values of the effects of the Fourier boundary condition and the source function are stored.

The algorithm requires two embedded loops checked at each step. The main loop index increases up to the number NT of random walks, the latter loop index up to the number NR of reflections. Let us consider a domain G in \mathbb{R}^2 and a point $(X1D, X2D)$ given in G . Then a possible algorithm A1 of the Fourier problem can be written in the form :

```

YM := 0.0;
for i:=1 to NT do
  begin
    X1:= X1D;    X2:= X2D;    YA:= 0.0;    COUNT:= 0;
    while COUNT < NR do
      begin
        YA := YA + F(X1,X2) * EXP(-MU*COUNT*H); { Source effect inside }
        X1:= X1 + random basic step;           X2:= X2 + random basic step;
        INTER( X1, X2, IMMO );
        if IMMO = true then
          begin
            Procedure Reflection(X1,X2);    COUNT := COUNT + 1;
            YAF := YAF + G(X1,X2) * EXP(-MU*COUNT*H); { Fourier condition }
            YA := YA + 2 * F(X1,X2) * EXP(-MU*COUNT*H); { Source effect }
          end;
        end;
      end;
    YM := YM + H * H * YA / 2 - H * YAF;
  end;
U:= YM / NT; { Solution as the arithmetic mean }

```

where MU is a strictly positive constant, and with the procedure `INTER` checking the reflection of the random walk in progress. The logical variable `IMMO` takes the value *true* when the random walk hits the boundary. The function `F` defines the effect of the source for each reached point of the walk inside G , and `G` defines the reflection effect for a hitting point.

3.2 Problem with mixed boundary conditions

The integer NT corresponds to the number of simulated random walks, first reflected by the boundary part Γ_F , then absorbed by Γ_D , and consequently to the number of absorptions by Γ_D . The number of steps of such random walks is denoted N . For each simulated random walk $(X_k; 0 \leq k \leq N)$, we compute the following score :

$$\begin{aligned}
Y_n = & \Delta t \cdot \sum_{k=1}^N f(X_k) \cdot \exp\left(-\Delta\xi \cdot \mu \cdot \sum_{l=1}^k \mathbb{1}_{\Gamma_F}(X_l)\right) + g_1(X_N) \cdot \exp\left(-\Delta\xi \cdot \mu \cdot \sum_{l=1}^N \mathbb{1}_{\Gamma_F}(X_l)\right) \mathbb{1}_{\Gamma_D}(X_N) \\
& - \Delta\xi \cdot \sum_{k=1}^{N-1} g_2(X_k) \cdot \exp\left(-\Delta\xi \cdot \mu \cdot \sum_{l=1}^k \mathbb{1}_{\Gamma_F}(X_l)\right) \mathbb{1}_{\Gamma_F}(X_k) .
\end{aligned}$$

The extension of algorithm `A1` to the mixed case is easily implemented : the above loop has only to be replaced by another loop which follows the trajectory of a random walk, step by step, until absorption. A possible algorithm A2 to the problem with mixed boundary conditions admits the following interior loop :

```

while INSIDE = true do begin
  YA := YA + F(X1,X2) * EXP(-MU*COUNT*H);
  X1:= X1 + random basic step;      X2:= X2 + random basic step;
  INTER( X1, X2, INSIDE, IMMO_N, IMMO_F );
  if IMMO_F = true then begin
    Procedure Reflection(X1,X2); COUNT := COUNT + 1;
    YAF := YAF + G2(X1,X2) * EXP(-MU*COUNT*H);
    YA := YA + 2 * F(X1,X2) * EXP(-MU*COUNT*H);
  end; { if }
end; { while }
YM := YM + H * H * YA / D + G1(X1,X2) * EXP(-MU*COUNT*H) - H * YAF;

```

The computer program includes the functions F , G_1 , and G_2 given by problem (5), the procedures `Basic step` and `Reflection` previously described, as well as the modified procedure `INTER` :

```

Procedure INTER( X1, X2, INSIDE, IMMO_N, IMMO_F );
  INSIDE := true;          IMMO_F := false;

  if (X1,X2 ∈  $\Gamma_D$ ) then INSIDE := false;   if (X1,X2 ∈  $\Gamma_F$ ) then IMMO_F := true;

```

As for the previous algorithm, algorithm *A2* includes two loops :

- the main loop index increases up to the number of random walks until the fixed maximum NT ;
- the second loop generates a random walk reflected by Γ_F , and then absorbed by Γ_D .

When the trajectory hits Γ_F , one reflection occurs; if it hits Γ_D , counter `COUNT` increases. When an absorption point is hit, a position test stops the simulated random walk.

4 Numerical experiments

The validity and suitability of the present approach, i.e. the stochastic representations in Section 2 and their approximate solutions in Section 3, are investigated here by considering and evaluating a set of boundary value problems. The results are compared with analytical solutions found in the literature.

The algorithms described above have been implemented. The programs, written in Fortran or Pascal, have been run on Sun Spark work stations or compatible PC. We have used some generators of pseudo-random numbers : `random` of Turbo-Pascal, and `ran`, `r_addran()` of Fortran 77.

4.1 Problems with Fourier boundary conditions

The following example is a problem with Fourier condition on the boundary of a square. The second example is an extension to a cube.

Two-dimensional case. Let us consider problem (1), originating from Reference 12, on the square $G =]0, 1[\times]0, 1[$, with the data :

$$\begin{aligned} \forall (x, y) \in G, \quad f(x, y) &= x(1-x) + y(1-y), \\ \forall (x, y) \in \partial G, \quad g(x, y) &= \begin{cases} x(1-x) & \text{if } (0 \leq x \leq 1, y = 0 \text{ or } y = 1), \\ y(1-y) & \text{if } (x = 0 \text{ or } x = 1, 0 \leq y \leq 1), \end{cases} \\ \mu &\equiv 1. \end{aligned}$$

This problem admits a unique solution written in the form : $u(x, y) = x(1-x)y(1-y)$. Using Algorithm A1, Tables I–II show the results at the interior point (0.5; 0.5), and boundary point (1.0; 0.5) respectively.

- Interior point : $(0.5; 0.5) \in G$ — Exact value : $u = 0.0625$ — Step size $h = 0.025$ — Sample size $NT = 10^5$.

Table I. Problem with Fourier boundary conditions — Approximate solution $u_c(0.5; 0.5)$

Number of reflections NR	Mean value u_c	Relative error $ (u_c - u)/u $
200	0.062713	$3.4 \cdot 10^{-3}$
1000	0.062525	$0.4 \cdot 10^{-3}$

- Boundary point : $(1.0; 0.5) \in \partial G$ — Exact value : $u = 0$ — Step size : $h = 0.005$.

Table II. Problem with Fourier boundary conditions — Approximate solution $u_c(1.0; 0.5)$

Sample size : $NT = 10^5$		Number of reflections : $NR = 5\,000$	
Number of reflections NR	Mean value u_c	Sample size NT	Mean value u_c
200	$-5.24 \cdot 10^{-4}$	10^4	$-2.77 \cdot 10^{-4}$
1000	$-2.27 \cdot 10^{-4}$	10^5	$-1.58 \cdot 10^{-4}$
5000	$-1.58 \cdot 10^{-4}$	$2 \cdot 10^5$	$0.89 \cdot 10^{-4}$

Three-dimensional case. Let us consider problem (1), in the cube $G =]0; 1[\times]0; 1[\times]0; 1[$, with the following data :

$$\begin{aligned} \forall (x, y, z) \in G, \quad f(x, y) &= x(1-x) + y(1-y) + z(1-z), \\ \forall (x, y, z) \in \partial G, \quad g(x, y) &= \begin{cases} x(1-x)y(1-y) & \text{if } z = 0 \text{ or } z = 1, \\ y(1-y)z(1-z) & \text{if } x = 0 \text{ or } x = 1, \\ z(1-z)x(1-x) & \text{if } y = 0 \text{ or } y = 1, \end{cases} \\ \mu &\equiv 1. \end{aligned}$$

This problem admits the unique solution : $u(x, y, z) = x(1-x)y(1-y)z(1-z)$. Using Algorithm A1, the results at point $(1.0; 0.5; 0.5)$ belonging to ∂G , are shown in Table III, with the following data :

Step size : $h = 0.005$, Sample size : $NT = 10^4$, Exact value : $u = 0$.

Table III. Problem with Fourier boundary conditions — Approximate solution $u_c(1.0; 0.5; 0.5)$

Number of reflections NR	Mean value u_c
200	$-1.42 \cdot 10^{-4}$
1000	$-0.31 \cdot 10^{-4}$

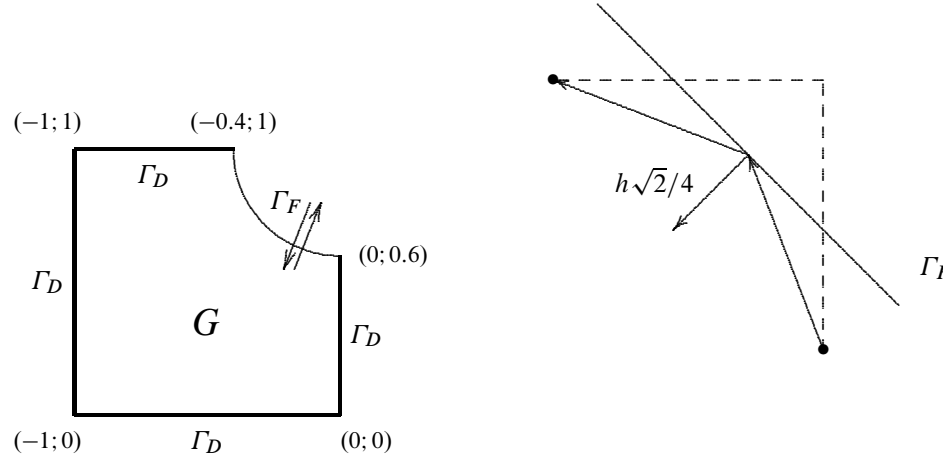


FIG. 1 – Non convex domain G for the problem with mixed boundary conditions — Mirror-reflection on Γ_F

4.2 Problems with mixed boundary conditions

Let us consider problem (5), originating from Reference 13, in the non convex domain $G =]-1; 0[\times]0; 1[\setminus \{(x, y) \in \mathbb{R}^2 \mid x^2 + (y - 1)^2 \leq 0.16\}$ (cf. Figure 1), with the following data :

$$\begin{aligned}
 f &\equiv -2 && \text{in } G, \\
 g_1(x, y) &= x^2 + (y - 1)^2 && \text{on } \Gamma_D = \partial G \setminus \Gamma_F, \\
 g_2 &\equiv 0.32 && \text{on } \Gamma_F = \{(x, y) \in \mathbb{R}^2 \mid x^2 + (y - 1)^2 = 0.16, x \leq 0, y \leq 1\}, \\
 \mu &= 3 && \text{on } \Gamma_F.
 \end{aligned}$$

This problem admits the unique solution : $u(x, y) = x^2 + (y - 1)^2$.

Let us consider the following modified reflection procedure (see also Figure 1) :

```

Procedure REBOUND(X1,X2);
if D1 <> 0 then X1 := X1 - D2;
if D2 <> 0 then X2 := X2 - D1;

```


The space, time and reflection increments are defined by :

$$\delta X_t^x = |X_{k+1} - X_k| = h, \quad \delta t = |t_{k+1} - t_k| = h^2/2 \quad \text{and} \quad \delta \xi_t^x = |\xi_{k+1} - \xi_k| = h\sqrt{2}/4.$$

Using Algorithm A2 with such reflection, the results at point $(-0.5; 0.5)$ belonging to ∂G , are shown in Table IV. The empirical variance :

$$\sigma_c^2 = \frac{1}{NT} \sum_{n=1}^{NT} (Y_n - u_c)^2$$

where Y_n is the score obtained from the n -th random walk, and u_c the mean value of the NT scores, has been computed : $\sigma_c = 0,570$.

Table IV. Problem with mixed boundary conditions — Approximate solution $u_c(-0.5; 0.5)$

Step size h	Sample size NT	Mean value u_c	Relative error $ u_c - u /u$
0,005	2×10^4	4,962	$7,6 \times 10^{-3}$
	3×10^4	4,955	$9,0 \times 10^{-3}$

5 Concluding remarks

We have presented stochastic methods for the numerical approximation of some linear boundary value problems (see also Reference 14). The representations are deduced from the application of the Itô's formula for the problems with Fourier or mixed conditions with non smooth boundary. These

representations have given some easily implementable algorithms. The programs are short, easy to write and check step by step.

The extension to an upper dimension problem, in particular the three-dimensional case, only requires a minor modification of the basic step (two lines are only to be added), and a few additional lines in the procedure `INTER` to locate the boundary. We do not have to store in memory any mesh of the region G , nor deal with the corresponding numbering-lists.

As for the classical Monte Carlo schemes, this stochastic method admits an expected rate of convergence of about $1/\sqrt{NT}$. The essential properties of Monte Carlo methods are maintained; on the one hand, this stochastic method of the solution approximation does not depend on the choice of the point, on the other hand, its implementation on a parallel computer is intrinsically possible.

Lastly, the random walks are simulated simply by means of repeated Bernoulli's elementary events.

Acknowledgements The support of the Conseil Général du Maine-et-Loire is gratefully acknowledged.

References

1. A. Haji-Sheikh, *Monte Carlo methods*, in W.J. Minkowycz et al. (eds.), *Handbook of Numerical Heat Transfer*, Chapter 16, John Wiley & Sons, Inc., New York, 1988.
2. M. Cessenat, R. Dautray, G. Ledanois, P.-L. Lions, É. Pardoux, and R. Sentis, *Méthodes probabilistes pour les équations de la physique*, Éditions Eyrolles (Série Synthèses), Paris, 1989.
3. G. Marshall, 'Monte Carlo methods for the solution of nonlinear partial differential equations', *Computer Physics Communications*, **56**, 51–61 (1989).
4. H. J. Kushner and P. G. Dupuis, *Numerical Methods for Stochastic Control Problems in Continuous Time*, Springer-Verlag, New York, 1992.

5. M. I. Freidlin, *Functional Integration and Partial Differential Equations*, vol. 109 of *Annals of Mathematics Studies*, Princeton University Press, Princeton, New Jersey, 1985.
6. J. E. Souza de Cursi, 'Numerical methods for linear boundary value problems based on Feynman-Kac representations', *Mathematics and Computers in Simulation*, **36**, 1–16 (1994).
7. É. Pardoux and D. Talay, 'Discretization and simulation of stochastic differential equations', *Acta Applicandae Mathematicae*, **3**, 23–47 (1985).
8. P. E. Kloeden and E. Platen, *Numerical Solution of Stochastic Differential Equations*, vol. 23 of *Applications of Mathematics*, Springer-Verlag, New York, 1992.
9. A. Bensoussan and J.-L. Lions, *Contrôle impulsionnel et inéquations quasi-variationnelles*, vol. 11 of *Méthodes Mathématiques de l'informatique*, Dunod, Paris, 1982.
10. I. Karatzas and S. E. Shreve, *Brownian Motion and Stochastic Calculus*, Springer-Verlag, New York, 1991.
11. P. L. Lions and A. S. Sznitman, 'Stochastic differential equations with reflecting boundary conditions', *Comm. Pure Appl. Math.*, **37**, 511–537 (1984).
12. P. L. George and K. Leeuwijn, *Bibliothèque thermique MODULEF*, Fiches techniques “Éléments finis” 100, INRIA, Rocquencourt, 1989.
13. B. Muller, D. Steer, and M. Vidrascu, *MODULEF : Quelques exemples de tests de thermique*, Fiches techniques “Éléments finis” 87, INRIA, Rocquencourt, 1988.
14. J.-P. Morillon, *Numerical solutions of linear mixed boundary value problems using stochastic representations*, soumis à *International Journal for Numerical Methods in Engineering* (1995).