Direct Monte Carlo Simulation of Time-Dependent Problems

by

Matthew. N. O. Sadiku[1], Cajetan M. Akujuobi[1], Sarhan M. Musa[1], and Sudarshan R. Nelatury[2]

[1]Center of Excellence for Communication Systems Technology Research (CECSTR)
College of Engineering
Prairie View A&M University, Prairie View, TX 77446
Email: mnsadiku, cmakujuobi, smmusa @pvamu.edu

[2]School of Engineering and Engineering Technology
Pennsylvania State University
Erie, PA 16563-1701
Email: srn3@psu.edu

Abstract: *Monte Carlo method is well known for solving static problems such as Laplace's or Poisson's equation. In this paper, we extend the applicability of the conventional Monte Carlo method to solve time-dependent (heat) problems. We illustrate this with some examples and present results in one-dimension (1-D) and two-dimension (2-D) that agree with the exact solutions.*

I. Introduction

Monte Carlo methods are nondeterministic modeling approaches for solving physical and engineering problems. They have been applied successfully for solving differential and integral equations, for finding eigenvalues, for inverting matrices, and for evaluating multiple integrals [1-5]. Monte Carlo methods are well known for solving static problems such as Laplace's or Poisson's equation. They are hardly applied in solving parabolic and hyperbolic partial differential equations. The so-called "Monte Carlo simulation of Maxwell's equation" [6-9] gives the impression that Monte Carlo method is being applied to time-dependent problems. This is not a direct or explicit solution of Maxwell equations like the finite-difference time-domain (FDTD) scheme [10-12].

In this paper, we extend the applicability of the conventional Monte Carlo method to solve directly time-dependent (heat) problems. We deal with the case of rectangular solution regions. We compare Monte Carlo solutions with the finite difference and exact solutions. Our results for one-dimension (1-D) and two-dimension (2-D) problems agree with the exact solutions. The Monte Carlo treatment is so straightforward that it can be presented to undergraduate students without difficulties.

II. Diffusion Equation

Consider the skin effect on a solid cylindrical conductor. The current density distribution within a good conducting wire $(\sigma / \omega \in >> 1)$ obeys the diffusion equation

$$\nabla^2 \boldsymbol{J} = \mu\sigma \frac{\partial \mathbf{J}}{\partial t} \tag{1}$$

We may derive the diffusion equation directly from Maxwell's equations. We recall that

$$\nabla \times \mathbf{H} = \mathbf{J} + \mathbf{J}_d \tag{2}$$

where $\mathbf{J} = \sigma \mathbf{E}$ is the conduction current density and $\mathbf{J}_d = \frac{\partial \mathbf{D}}{\partial t}$ is the displacement current density. For $\sigma / \omega \in >> 1, \mathbf{J}_d$ is negligibly small compared with **J**. Hence

$$\nabla \times \mathbf{H} \simeq \mathbf{J} \tag{3}$$

Also,

$$\begin{aligned} \nabla \times \mathbf{E} &= -\mu \frac{\partial \mathbf{H}}{\partial t} \\ \nabla \times \nabla \times \mathbf{E} = \nabla \nabla \cdot \mathbf{E} - \nabla^2 \mathbf{E} &= -\mu \frac{\partial}{\partial t} \nabla \times \mathbf{H} \end{aligned} \tag{4}$$

Since $\nabla \cdot \mathbf{E} = 0,$ introducing eq. (3), we obtain

$$\nabla^2 \mathbf{E} = \mu \frac{\partial \mathbf{J}}{\partial t} \tag{5}$$

Replacing **E** with **J**/σ, eq. (5) becomes

$$\nabla^2 \boldsymbol{J} = \mu\sigma \frac{\partial \mathbf{J}}{\partial t}$$

which is the diffusion (or heat) equation.

We now consider the Monte Carlo solution of the diffusion (or heat) equation in one-dimensional (1-D) and two-dimensional (2-D) forms in rectangular coordinate system.

III. One-Dimensional Heat Equation

To be concrete, consider the one-dimensional heat equation:

$$U_{xx} = U \quad ,0 < x < 1, \quad t > 0 \qquad (6)$$

Boundary conditions:

$$U(0,t) = 0 = U(1,t) \ , \quad t > 0 \qquad (7a)$$

Initial condition:

$$U(x,0) = 100 \ \ ,0 < x < 1 \qquad (7b)$$

In eq. (6), U_{xx} indicates second partial derivative with respect to x, while U_t indicates partial derivative with respect to t. The problem models temperature distribution in a rod or eddy current in a conducting medium [13]. In order to solve this problem using the Monte Carlo method, we first need to obtain the finite difference equivalent of the partial differential equation in eq.(6). Using the central-space and backward-time scheme, we obtain

$$\frac{U(i+1,n) - 2(U(i,n) + U(i-1,n)}{(\Delta x)^2} = \frac{U(i,n) - U(i,n-1)}{\Delta t} \qquad (8)$$

where $x = i\Delta x$ and $t = n\Delta t$. If we let

$$\alpha = \frac{(\Delta x)^2}{\Delta t} \qquad (9)$$

eq.(8) becomes

$$U(i,n) = P_{x+}U(i+1,n) + P_{x-}U(i-1,n) + P_{t-}U(i,n-1) \qquad (10)$$

where

$$P_{x+} = P_{x-} = \frac{1}{2+\alpha} \ , \ P_{t-} = \frac{\alpha}{2+\alpha} \qquad (11)$$

Notice that $P_{x+} + P_{x-} + P_{t-} = 1$. Equation (10) can be given a probabilistic interpretation. If a random-walking particle is instantaneously at the point (x,y), it has probabilities P_{x+}, P_{x-}, and P_{t-} of moving from (x,t) to $(x+\Delta x,t)$, $(x-\Delta x,t)$, and $(x,t-\Delta t)$ respectively. The particle can only move toward the past, but never toward the future. A means of determining which way the particle should move is to generate a random number r, $0 < r < 1$, and instruct the particle to walk as follows:

$$\begin{aligned} (x,t) &\to (x+\Delta x,t) \ \text{ if } (0 < r < 0.25) \\ (x,t) &\to (x-\Delta x,t) \ \text{ if } (0.25 < r < 0.5) \\ (x,t) &\to (x,t-\Delta t) \ \text{ if } ((0.5 < r < 1)) \end{aligned} \qquad (12)$$

where it is assumed that $\alpha = 2$. Most modern software such as MATLAB have a random number generator to obtain r.

To calculate U at point (x_o,t_o), we follow the following random walk algorithm:

1. Begin a random walk at $(x,t) = (x_o,t_o)$.

2. Generate a random number $0 < r < 1$, and move to the next point using eq. (12).
3(a). If the next point is not on the boundary, repeat step 2.
3(b). If the random walk hits the boundary, terminate the random walk. Record U_b at the boundary and go to step 1 and begin another random walk.
4. After N random walks, determine

$$U(x_o, t_o) = \frac{1}{N}\sum_{K=1}^{N} U_b(K) \qquad (13)$$

where N, the number of random walks is assumed large. A typical random walk is illustrated in Fig. 1.

Example 1 As a numerical example, consider the solution of the problem in eqs. (6) and (7). We select $\alpha = 2$, $\Delta x = 0.1$, so that $\Delta t = 0.005$ and $P_{x+} = P_{x-} = \frac{1}{4}$, $P_{t-} = \frac{1}{2}$.
We calculate U at $x_0 = 0.4$, $t = 0.01$, 0.02, 0.03, 0.04, 0.10. The MATLAB code for the problem is shown in Fig. 2. As evident in the program, $N = 1000$. As shown in Table 1, we compare the results with the finite different solution and exact solution [14]:

$$U(x,t) = \frac{400}{\pi}\sum_{K=0}^{\infty}\frac{1}{n}\sin(n\pi x)e^{(-n^2\pi^2 t)}, \; n = 2K+1 \qquad (14)$$

For the exact solution in eq. (14), the infinite series was truncated at $K = 10$.

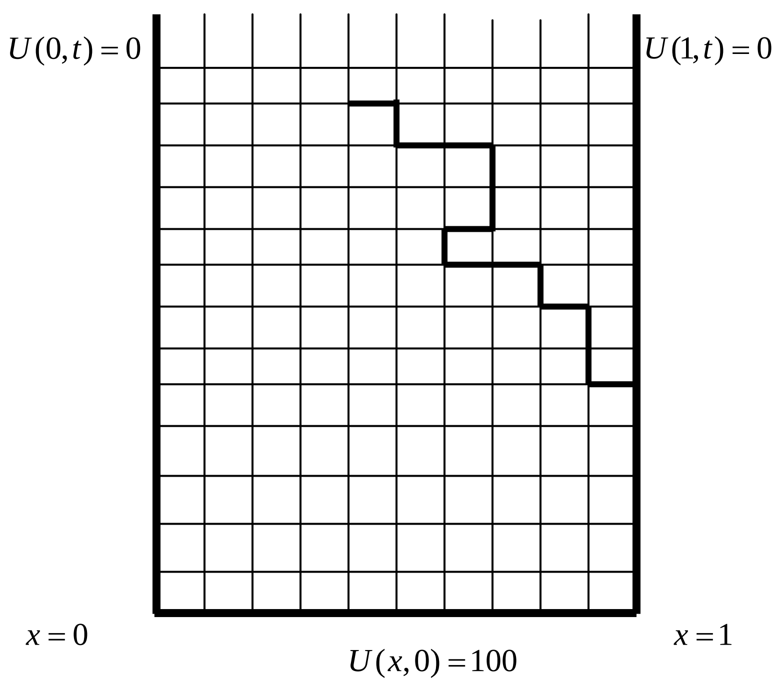

Fig. 1 A typical random walk in rectangular domain.

Table 1 Comparing Monte Carlo (MCM) solution with finite difference (FD) and exact solution $(x_o = 0.4)$

t	Exact	MCM	FD
0.01	99.53	94.4	100
0.02	95.18	93.96	96.87
0.03	83.2	87.62	89.84
0.04	80.88	81.54	82.03
0.10	45.13	46.36	45.18

```
% This program solves one-dimensional diffusion (or heat)
equation
% using Monte Carlo method

nrun = 1000;
delta = 0.1;
% deltat=2*delta^2;
deltat = 0.005;
A=1.0;
xo=0.4;
to=0.1;
io=xo/delta;
jo=to/deltat;
no=to/deltat;
imax=A/delta;
sum=0;
for k=1:nrun
   i=io;
   n=no;
   while i<=imax & n<=no
   r=rand; %random number between 0 and 1
      if (r >= 0.0 & r <= 0.25)
      i=i+1;
      end
      if (r >= 0.25 & r <= 0.5)
      i=i-1;
      end
      if (r  >= 0.5 & r <= 1.0)
      n=n-1;
      end
      if (n < 0)
      break;
      end
     % check if (i,n) is on the boundary
      if(i == 0.0)
```

```
            sum=sum+ 0.0;
             break;
         end
         if(i == imax)
            sum=sum+ 0.0;
            break;
         end
         if(n == 0.0)
            sum=sum+ 100;
          break;
         end
     end % while

 end
u=sum/nrun
```

Fig. 2 MATLAB program for Example 1.

IV. Two-Dimensional Heat Equation

Suppose we are interested in the solution of the two-dimensional heat equation:

$$U_{xx} + U_{yy} = U_t \ , \ 0 < x < 1, \ 0 < y < 1, \ t > 0 \qquad (15)$$

Boundary conditions:

$$U(0, y, t) = 0 = U(1, y, t), \ 0 < y < 1, \ t > 0 \qquad (16a)$$

$$U(x, 0, t) = 0 = U(x, 1, t), \ 0 < x < 1, \ t > 0 \qquad (16b)$$

Initial condition:

$$U(x, y, 0) = 10xy, \ 0 < x < 1, \ 0 < y < 1 \qquad (16c)$$

Using the central-space and backward-time scheme, we obtain the finite difference equivalent as

$$\frac{(i+1, j, n) - 2U(i, j, n) + U(i-1, j, n)}{(\Delta x)^2} + \frac{(i, j+1, n) - 2U(i, j, n) + U(i, j-1, n)}{(\Delta y)^2} = \frac{(i, j, n) - U(i, j, n-1)}{(\Delta t)^2} \qquad (17)$$

Let $\Delta x = \Delta y = \Delta$ and

$$\alpha = \frac{\Delta^2}{\Delta t} \qquad (18)$$

eq. (17) becomes

$$U(i, j, n) = P_{x+}U(i+1, j, n) + P_{x-}U(i-1, j, n) + P_{y+}U(i, j+1, n) + P_{y-}U(i, j-1, n) + P_{t-}U(i, j, n-1) \qquad (19)$$

where

$$P_{x+} = P_{x-} = P_{y+} = P_{y-} = \frac{1}{4+\alpha} \qquad (20a)$$

$$P_{t-} = \frac{\alpha}{4+\alpha} \qquad (20b)$$

Note that $P_{x+} + P_{x-} + P_{y+} + P_{y-} + P_{t-} = 1$ so that a probabilistic interpretation can be given to eq. (19). A random walking particle at point (x, y, t) moves to $(x+\Delta, y, t)$, $(x-\Delta, y, t)$, $(x, y+\Delta, t)$, $(x, y-\Delta, t)$, $(x, y, t-\Delta t)$ with probabilities, P_{x+}, P_{x-}, P_{y+}, P_{y-}, and P_{t-} respectively. By generating a random number $0 < r < 1$, we instruct the particle to move as follows:

$$\begin{aligned}
(x,y,t) &\to (x+\Delta, y, t) \text{ if } (0 < r < 0.2) \\
(x,y,t) &\to (x-\Delta, y, t) \text{ if } (0.2 < r < 0.4) \\
(x,y,t) &\to (x, y+\Delta, t) \text{ if } (0.4 < r < 0.6) \\
(x,y,t) &\to (x, y-\Delta, t) \text{ if } (0.6 < r < 0.8) \\
(x,y,t) &\to (x, y, t-\Delta t) \text{ if } ((0.8 < r < 1))
\end{aligned} \qquad (21)$$

assuming that $\alpha = 1$. Therefore, we take the following steps to calculate U at point (x_o, y_o, t_o):

1. Begin each random walk at $(x, y, t) = (x_o, y_o, t_o)$.
2. Generate a random number $0 < r < 1$, and move the next point according to eq. (21).

3(a). If the next point is not on the boundary, repeat step 2.
3(b). If the random walk hits the boundary, terminate the random walk. Record U_b at the boundary and go to step 1 and begin another random walk.
3. After N random walks, determine

$$U(x_o, y_o, t_o) = \frac{1}{N}\sum_{K=1}^{N} U_b(K) \qquad (22)$$

The only difference between 1-D and 2-D is that there are three kinds of displacements in 1-D while there are five displacements (four spatial ones and one temporal one) in 2-D.

Example 2 As a numerical example, consider the solution of the problem in eqs. (15) and (16). We select $\alpha = 1$, $\Delta = 0.1$, so that $\Delta t = 0.01$ and we calculate U at $x = 0.5$, $y = 0.5$, $t = 0.05$, 0.1, 0.15, 0.2, 0.25, 0.3. In Monte Carlo simulations, we used $N = 1000$. As shown in Table 2, we compare the results from the Monte Carlo method (MCM) with the finite difference (FD) solution and exact solution [15]:

$$U(x, y, t) = \frac{40}{\pi^2}\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\frac{\cos(m\pi)\cos(n\pi)}{mn}\sin(m\pi x)\times\sin(n\pi y)e^{(-\lambda_{mn}^2 t)}, \qquad (23)$$

where $\lambda_{mn}^2 = (m\pi)^2 + (n\pi)^2$. In the exact solution in eq. (23), the infinite series was truncated at $m = 10$ and $n = 10$. Due to the randomness of the Monte Carlo solution, each MCM result in Tables 1 and 2 was obtained by running the simulation five times and taking the average.

Table 2 Comparing Monte Carlo solution with finite difference and exact solution

t	Exact	MCM	FD
0.05	1.491	1.534	1.518
0.10	0.563	0.6627	0.5627
0.15	0.216	0.267	0.2063
0.20	0.078	0.106	0.0756
0.25	0.029	0.0419	0.0277
0.30	0.015	0.019	0.0102

V. Conclusion

In this paper, we have demonstrated how the conventional Monte Carlo method (the fixed random walk) can be applied to time-dependent problems such as the heat equation in both rectangular and cylindrical coordinates. For 1-D and 2-D cases, we notice that the Monte Carlo solutions agree well with the finite difference solution and the exact analytical solutions and it is easier to understand and program than the finite difference method. The method does not require the need for solving large matrices and is trivially easy to program so that even undergraduates can understand it. The randomness of the MCM results can be eliminated if we apply the Exodus method, another Monte Carlo technique [16, 17]. The idea can be extended to other time-dependent problems such as Maxwell's equations or the wave equation.

VI. References

[1] T. E. Price and D. P. Story, "Monte Carlo simulation of numerical integration," *J. Statistical Computation and Simulation*, vol. 23, 1985, pp. 97-112.

[2] J. Padovan, "Solution of anisotropic heat conduction problems by Monte Carlo procedures," *Trans. ASME*, August 1974, pp. 428-430.

[3] K. K. Sabelfeld, *Monte Carlo Methods in Boundary Value Problems*. Berlin/New York: Springer-Verlag, 1991, p.2.

[4] N. Metropolis and S. Ulam, "The Monte Carlo Method," *Jour. Amer. Stat. Assoc.* vol. 44, no. 247, 1949, pp. 335-341.

[5] I. M. Sobol, *A Primer for the Monte Carlo Method*. Boca Raton: CRC Press, 1994.

[6] J. Guo et al., "Frequency dependence of scattering by dense media of small particles based on Monte Carlo simulation of Maxwell equations," *IEEE Trans. Geoscience & Remote Sensing*, vol. 40, no.1, Jan. 2002, pp. 153-161.

[7] L. Tsang, C. E. Mandt, and K. H. Ding, "Monte Carlo simulations of the extinction rate of dense media with randomly distributed dielectric spheres based on solution of Maxwell's equations," *Optical Letters*, vol. 17, no. 5, 1992, pp. 314-316.

[8] R. L. Wagner et al., "Monte Carlo simulation of electromagnetic scattering from two-dimensional random rough surfaces," *IEEE Trans. Antennas and Propagation*, vol. 45, no. 2, Feb. 1997, pp.235-245.

[9] K. W. Lam et al., "On the analysis of statistical distributions of UWB signal scattering by random rough surfaces based on Monte Carlo simulations of Maxwell equations," *IEEE Trans. Antennas and Propagation*, vol. 52, no. 12, Dec. 2004, pp.235-245.

[10] K. S. Yee,"Numerical solution of initial boundary-value problems involving Maxwell's equations in isotropic media," *IEEE Trans. Antennas and Propagation*, vol. 14, May. 1966, pp.302-307.

[11] A. Taflove and M. E. Brodwin, "Numerical solution of steady-state electromagnetic scattering problems using the time-dependent Maxwell's equations," *IEEE Microwave Theory & Techniques*, vol. 23, no.8, Aug. 1975, pp. 623-630.

[12] M. Okoniewski, "Vector wave equation 2D-FDTD method for guided wave equation," *IEEE Microwave Guided Wave Letters*, vol. 3, no. 9, Sept. 1993, pp. 307-309.

[13] D. Netter, J. Levenque, P. Masson and A. Rezzoug, "Monte Carlo method for transient eddy-current calculations," IEEE *Trans. Magnetics*, vol.40, no.5, Sept. 2004, pp.3450-3456.

[14] M. N. O. Sadiku, *Numerical Techniques in Electromagnetics.* Boca Raton, FL: CRC Press, 2nd edition, 2001, pp. 125-130.

[15] D. L. Powers, *Boundary Value Problems.* New York: Academic Press, 1972, pp. 135-139.

[16] M. N. O. Sadiku. and D. Hunt, "Solution of Dirichlet problems by the Exodus Method," *IEEE Transactions of Microwave Theory & Techniques*, vol. 40, no. 1, Jan. 1992, pp. 89-95.

[17] M. N. O. Sadiku, S. O. Ajose, and F. Zhibao, "Applying the Exodus Method to solve Poisson's Equation," *IEEE Transactions of Microwave Theory & Techniques*, V. 42, No. 4, April 1994, pp. 661-666.