Int. J. Appl. Engng Ed. Vol. 7, No. 4, pp. 316–320, 1991
Printed in Great Britain.

0742–0269/91 $3.00+0.00

A Monte Carlo Method Without Employing Random Numbers*

M. N. O. SADIKU†
Department of Electrical Engineering, Temple University, Philadelphia, PA 19122, U.S.A.

This paper presents a Monte Carlo method which does not employ random numbers. The solution from the method is therefore independent of the random number generator of the computing facility. This procedure, known as the Exodus method, is generally faster and more accurate than the conventional Monte Carlo methods (fixed and floating random walks). Typical potential problems are solved to illustrate the method.‡

INTRODUCTION

THE practical value in teaching Monte Carlo techniques to undergraduate engineers has been shown [1, 2]. It is recommended that such a development be included in an introductory electromagnetic course for a number of reasons. First, the method presents potential theory from a stochastic viewpoint rather from a deterministic model using the energy equation. Second, the Monte Carlo methods (MCMs) in potential theory help student visualize the physical significance of some mathematical techniques commonly used in solving potential problems. For example, the use of Green's function has direct equivalent in Monte Carlo solutions. Lastly, incorporation of MCMs in undergraduate electrostatics can serve as a good introduction to this widely stochastic approach.

The Monte Carlo technique is essentially a means of estimating expected values, and hence is a form of numerical quadrature [3, 4]. Although the technique can be applied to simple processes and estimating multidimensional integrals, the technique has been suggested for solving potential problems [1, 5–9].

The most popular versions of the probabilistic or Monte Carlo solution of differential equations are the *fixed random walk* and the *floating random walk*. These MCMs employ random numbers which are usually machine dependent. Consequently, the results are dependent on the random number generator.

The objective of this paper is to present a Monte Carlo method which does not employ random numbers. This technique, known as the *Exodus method*, is generally faster and more accurate. Although, the Exodus method is discussed briefly in [1, 11], a complete development of the method and its implementation is presented in this paper. Typical potential problems are solved using the Exodus method. The results are compared with exact solutions or solutions obtained from other numerical methods.

EXODUS METHOD

Consider the solution to solve Laplace's equation

$$\nabla^2 V = 0 \text{ in region } R \tag{1a}$$

subject to Dirichlet boundary condition

$$V = V_p \text{ on boundary } B. \tag{1b}$$

To obtain a Monte Carlo solution to (1), we first divide the solution region R into mesh and derive the finite difference equivalent of (1). Assuming a mesh with $\Delta x = \Delta y = \Delta$ as illustrated in Fig. 1, the finite difference equivalent of (1) is [1]

$$\begin{aligned} V(x, y) = p_{x+} V(x+\Delta, y) + p_{x-} V(x-\Delta, y) \\ + p_{y+} V(x, y+\Delta) \\ + p_{y-} V(x, y-\Delta) \end{aligned} \tag{2}$$

where

$$p_{x+} = p_{x-} = p_{y+} = p_{y-} = 1/4. \tag{3}$$

A probabilistic interpretation to (2) is that a random walking particle at an arbitrary point (x, y) in R has probabilities p_{x+}, p_{x-}, p_{y+}, and p_{y-} of moving from (x, y) to the neighboring points $(x+\Delta, y)$, $(x-\Delta, y)$, $(x, y+\Delta)$, and $(x, y-\Delta)$ respectively.

We define the transition probability p_k as the probability that a random walk starting at point (x_0, y_0) in R ends at a boundary point (x_k, y_k) with prescribed potential $V_p(k)$, i.e.

$$p_k = \text{Prob}(x_0, y_0 \rightarrow x_k, y_k). \tag{4}$$

If there are M boundary nodes (excluding the corner points since a random walk never termin-

* Paper accepted 7 December 1990.
† Associate Professor.
‡ The treatment is applied in an introductory course on electromagnetics and takes two hours of class together with an introductory lecture described in Reference (1).

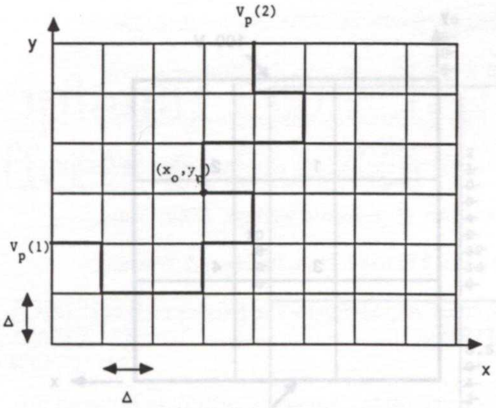

Fig. 1. Configuration for random walk.

ates at those points), the potential at the starting point (x_0, y_0) of the random walks is

$$V(x_0, y_0) = \sum_{k=1}^{M} p_k V_p(k). \qquad (5)$$

If m is the number of sides R has ($m = 4$ in Fig. 1) and $V_p(k)$ is the potential of side k, (5) can be simplified to

$$V(x_0, y_0) = \sum_{k=1}^{m} p_k V_p(k), \qquad (6)$$

where p_k^* in this case is the probability that a random walk terminates at side k. Since $V_p(k)$ is specified, our problem is reduced to finding p_k. We find p_k using the Exodus method in a manner similar to iterative process applied in [10]. The *Exodus method*, first suggested in [11], does not employ random numbers and is generally faster and more accurate than the fixed random walk. It basically consists of dispatching numerous walkers (say 10^6) simultaneously in directions controlled by the probabilities of going from one node to its neighbors. As these walkers arrive at new nodes, they are dispatched according to the probabilities until a set number (say 99.999%) have reached the boundaries. The advantage of Exodus method is its independence on the random number generator.

Let $P(i, j)$ be the number of particles at point (i, j) in R. We begin the application of the Exodus method by initially setting $P(i, j) = 0$ at all points (both fixed and free) except at point (x_0, y_0) where $P(i, j)$ assumes a large number N (say, $N = 10^6$ or more). We scan the mesh by dispatching the particles at each free node to its neighboring nodes according to the probabilities p_{x+}, p_{x-}, p_{y+}, and p_{y-} as illustrated in Fig. 2. Note that in Fig. 2(b), new $P(i, j) = 0$ at that node, while old $P(i, j)$ is shared among the neighboring nodes. When all the free nodes in R are scanned, we record the number of particles that have reached the boundary (i.e. at the fixed nodes). We keep scanning the mesh until a set number of particles (say 99.99% of N) have reached the boundary. If N_k is the number of particles that reached side k, we calculate

$$p_k = \frac{N_k}{N} \qquad (7)$$

Hence (6) can be written as

$$V(x_0, y_0) = \frac{\sum_{k=1}^{m} N_k V_p(k)}{N} \qquad (8)$$

so that the problem is reduced to just finding N_k, given N and $V_p(k)$.

TYPICAL EXAMPLE

We now illustrate the Exodus method by means of three examples. The first two examples have analytic solutions so that the accuracy and validity of the method can be checked. The calculations in the first two examples will be done by hand for pedagogic reasons. Of course, the results of these calculations can be improved by using computer. The third example do not have exact solution. Its computer solution is compared with results obtained from other numerical techniques.

Example 1. Given the one-dimensional differential equation

$$\frac{d^2\Phi}{dx^2} = 0, \qquad 0 \leqslant x \leqslant 1$$

subject to $\Phi(0) = 0$, $\Phi(1) = 10$, use the Exodus method to find $\Phi(0.25)$ by injecting 256 particles at $x = 0.25$

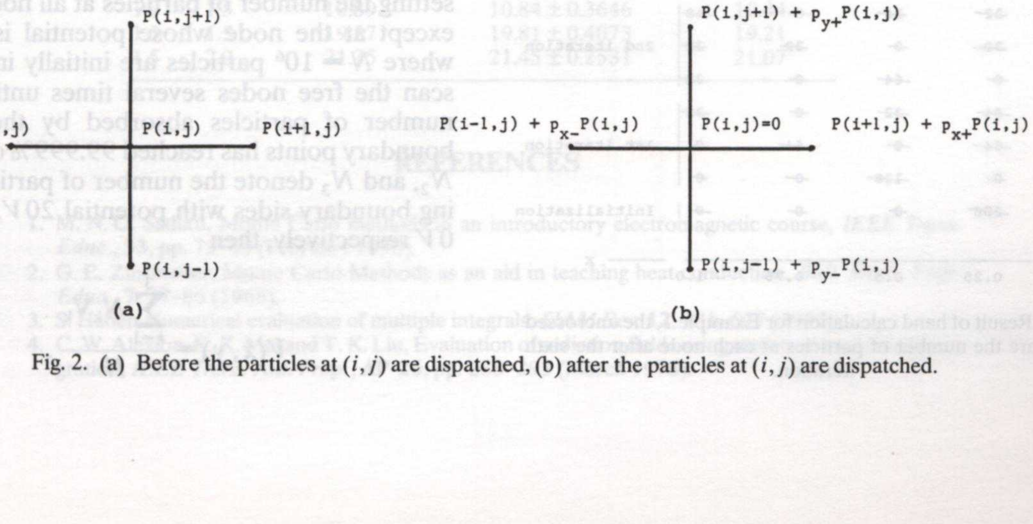

Fig. 2. (a) Before the particles at (i, j) are dispatched, (b) after the particles at (i, j) are dispatched.

Solution

The exact solution to this problem is $\Phi = 10x$ so that $\Phi(0.25) = 2.5$. To apply the Exodus method, we first obtain the finite difference equivalent of one-dimensional ordinary differential equation as

$$\Phi(x) = p_+\Phi(x+\Delta) + p_-\Phi(x-\Delta) \quad (9)$$

where $p_+ = p_- = \frac{1}{2}$. Since we are interested in the value of Φ at $x = 0.25$ and the calculations are to be done by hand, we select $\Delta = 0.25$ so that there are only 3 free nodes ($x = 0.25, 0.5,$ and 0.75) and 2 fixed nodes ($x = 0, 10$) to work with. As illustrated in Fig. 3, we initialize with 256 particles at $x = 0.25$ while setting the number of particles at other nodes equal to zero. Starting with the free node at $x = 0.25$ and proceeding to the free nodes at $x = 0.5$ and $x = 0.75$ in that order, we dispatch particles according to probabilities p_+ and p_-. The first iteration is completed when nodes at $x = 0.25$, 0.5 and 0.75 are scanned. We repeat the scanning or dispatching process for several iterations. To avoid confusion, each time a free node is scanned, we cross out the old number of particles at all nodes. (Note that at each scanning or iteration, the total number of particles remain 256.) After the sixth iteration, we obtain

$$\Phi(0.25) \simeq \frac{190}{256}.0 + \frac{63}{256}.10 = 2.461$$

which is only 1.56% off the exact value of 2.5.

Example 2. Use the Exodus method to find the potential at node 4 in Fig. 4. Inject 256 particles at node 4 and scan nodes in the order 1, 2, 3, 4.

0	0.25	0.5	0.75	1.0	
190	2	1	0	63	
~~190~~	~~2~~	~~0~~	~~2~~	~~62~~	6th iteration
~~190~~	~~0~~	~~4~~	~~0~~	~~62~~	
~~188~~	~~4~~	~~2~~	~~0~~	~~62~~	
~~188~~	~~4~~	~~0~~	~~4~~	~~60~~	5th iteration
~~188~~	~~0~~	~~8~~	~~0~~	~~60~~	
~~184~~	~~8~~	~~4~~	~~0~~	~~60~~	
~~184~~	~~8~~	~~0~~	~~8~~	~~56~~	4th iteration
~~184~~	~~0~~	~~16~~	~~0~~	~~56~~	
~~176~~	~~16~~	~~8~~	~~0~~	~~56~~	
~~176~~	~~16~~	~~0~~	~~16~~	~~48~~	3rd iteration
~~176~~	~~0~~	~~32~~	~~0~~	~~48~~	
~~160~~	~~32~~	~~16~~	~~0~~	~~48~~	
~~160~~	~~32~~	~~0~~	~~32~~	~~32~~	2nd iteration
~~160~~	~~0~~	~~64~~	~~0~~	~~32~~	
~~128~~	~~64~~	~~32~~	~~0~~	~~32~~	
~~128~~	~~64~~	~~0~~	~~64~~	~~0~~	1st iteration
~~128~~	~~0~~	~~128~~	~~0~~	~~0~~	
~~0~~	~~256~~	~~0~~	~~0~~	~~0~~	Initialization
0	0.25	0.5	0.75	1.0	x

Fig. 3. Result of hand calculation for Example 1, the uncrossed values are the number of particles at each node after the sixth iteration.

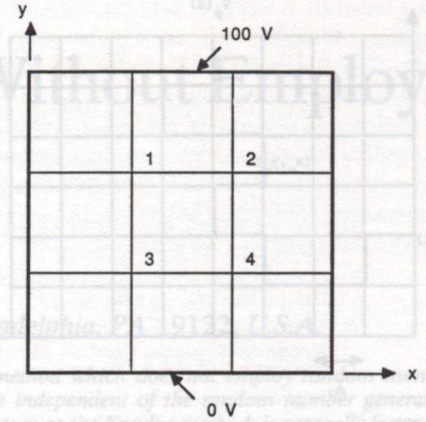

Fig. 4. For Example 2.

Solution

The exact solution obtained by the method of separation of variables in [10]

$$V(x, y) = \frac{400}{\pi}\sum_{n=0}^{\infty}\frac{\sin k\pi x \sinh k\pi y}{k \sinh k\pi}, \quad k = 2n+1$$

so that at node 4, $V_4 = V(\frac{2}{3}, \frac{1}{3}) = 11.928$.

To apply the Exodus method, we initialize by setting the number of particles at all nodes equal to zero except at node 4 where 256 particles are injected. We scan the free nodes in the order 1, 2, 3, 4 and dispatch particles as explained in Fig. 2 and displayed in Fig. 5. After the fourth iteration, we obtain

$$N_1 = \text{no. of particles reaching } 100\ V \text{ side} = 10 + 21 = 31.$$

Hence

$$V_4 = \sum_{k=1}^{4} p_k V_k = \frac{31}{256}.100 = 12.11$$

which is just 1.5% off the exact value of 11.928.

Example 3. Use the Exodus method to determine the potential at points (1.5, 0.5), (1.0,1.5), (1.5, 1.5), and (1.5, 2.0) in the two-dimensional potential system in Fig. 6.

Solution

Unlike the first two examples, this example is solved by developing a Fortran code based on the Exodus Method. Using $\Delta = 0.05$ and initially setting the number of particles at all nodes to zero except at the node whose potential is required, where $N = 10^6$ particles are initially injected, we scan the free nodes several times until the total number of particles absorbed by the fixed or boundary points has reached 99.999% of N. If N_1, N_2, and N_3 denote the number of particles reaching boundary sides with potential $20\ V$, $30\ V$, and $0\ V$ respectively, then

$$V(x, y) = \frac{\sum_{k=1}^{3} N_k V_k}{N}.$$

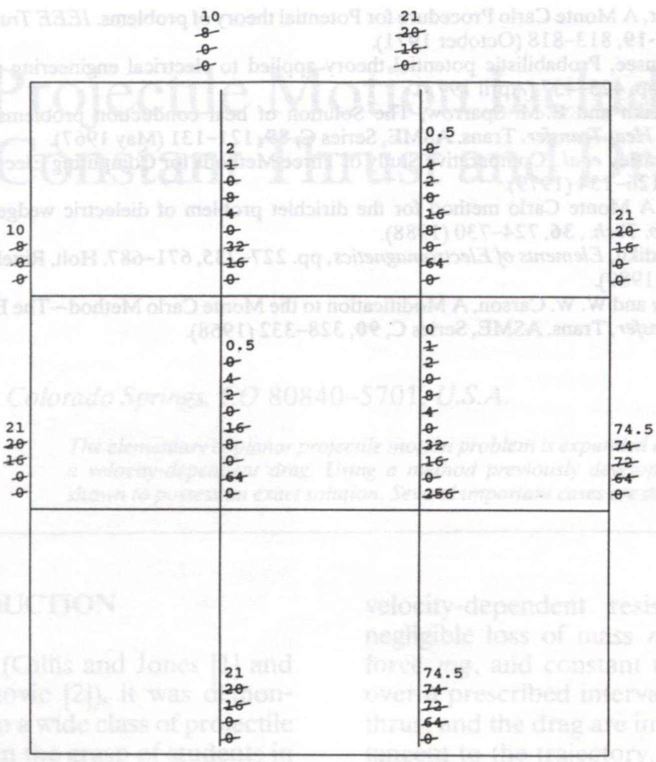

Fig. 5. Result of hand calculation for Example 2, the uncrossed values are the number of particles at each node after the fourth iteration (not drawn to scale).

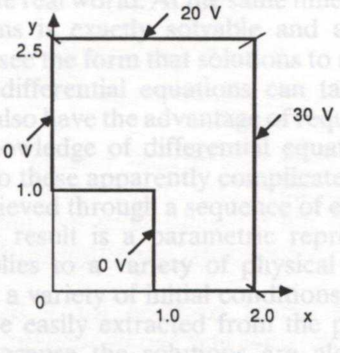

Fig. 6. For Example 3.

By running the program for different points, the result in Table 1 was obtained. Since the problem has no exact solution, we compare the result with those obtained using the floating random walk Monte Carlo method with $\Delta = 0.05$ and 2000 walks and finite different method with $\Delta = 0.05$ and 500 iterations [1, 10]. As evident from Table 1, the Exodus method provides a more accurate solution in less amount of time compared with the floating random walk.

Table 1. Results of Example 3

x	y	Exodus method V	Floating Random Walk ($V \pm \delta$)	Finite difference solution (V)
1.5	0.5	11.53	11.55 ± 0.5956	11.44
1.0	1.5	10.89	10.84 ± 0.3646	10.44
1.5	1.5	19.57	19.81 ± 0.4073	19.21
1.5	2.0	21.35	21.45 ± 0.2531	21.07

REFERENCES

1. M. N. O. Sadiku, Monte Carlo methods in an introductory electromagnetic course, *IEEE Trans. Educ.*, **33**, pp. 73–80 (February 1990).
2. G. E. Zinsmeiter, Monte Carlo Methods as an aid in teaching heat conduction, *Bull. Mech. Engr. Educ.*, 7, 77–86 (1968).
3. S. Haber, Numerical evaluation of multiple integrals, *SIAM Rev.* **12**, 481–527 (1970).
4. C. W. Alexion, K. K. Mei and T. K. Liu, Evaluation of radiation fields using statistical method of integration, *IEEE Trans. Ant. Prop.*, **AP-26**, pp. 288–293 (March 1978).

5. G. M. Royer, A Monte Carlo Procedure for Potential theory of problems. *IEEE Trans. Micro. Theo. Tech.* **MTT-19**, 813–818 (October 1971).
6. R. M. Bevensee, Probabilistic potential theory applied to electrical engineering problems, *Proc. IEEE*, **61**, pp. 423–437 (April 1973).
7. A. Haji-Sheikh and E. M. Sparrow, The Solution of heat conduction problems by probability methods, *J. Heat Transfer*. Trans. ASME, Series C, **89**, 121–131 (May 1967).
8. M. D. R. Beasley *et al*., Comparative Study of Three Methods for Computing Electric Fields, *Proc. IEE*, **126**, 126–134 (1979).
9. R. Schlott, A Monte Carlo method for the dirichlet problem of dielectric wedges, *IEEE Trans. Micro. Theo. Tech.*, **36**, 724–730 (1988).
10. M. N. O. Sadiku, *Elements of Electromagnetics*, pp. 227–235, 671–687. Holt, Rinehart & Winston, New York (1989).
11. A. F. Emery and W. W. Carson, A Modification to the Monte Carlo Method—The Exodus Method, *J. Heat Transfer*, Trans. ASME, Series C, **90**, 328–332 (1968).