Chapter 8

Monte Carlo Methods

"Chance favors the prepared mind." Anonymous

8.1 Introduction

Unlike the deterministic numerical methods covered in the foregoing chapters, Monte Carlo methods are nondeterministic (probabilistic or stochastic) numerical methods employed in solving mathematical and physical problems. The Monte Carlo method (MCM), also known as the *method of statistical trials,* is the marriage of two major branches of theoretical physics: the probabilistic theory of random process dealing with Brownian motion or random-walk experiments and potential theory, which studies the equilibrium states of a homogeneous medium [1]. It is a method of approximately solving problems using sequences of random numbers. It is a means of treating mathematical problems by finding a probabilistic analog and then obtaining approximate answers to this analog by some experimental sampling procedure. The solution of a problem by this method is closer in spirit to physical experiments than to classical numerical techniques.

It is generally accepted that the development of Monte Carlo techniques as we presently use them dates from about 1944, although there are a number of undeveloped instances on much earlier occasions. Credit for the development of MCM goes to a group of scientists, particularly von Neumann and Ulam, at Los Alamos during the early work on nuclear weapons. The groundwork of the Los Alamos group stimulated a vast outpouring of literature on the subject and encouraged the use of MCM for a variety of problems [2]–[4]. The name "Monte Carlo" comes from the city in Monaco, famous for its gambling casinos.

Monte Carlo methods are applied in two ways: simulation and sampling. Simulation refers to methods of providing mathematical imitation of real random phenomena. A typical example is the simulation of a neutron's motion into a reactor wall, its zigzag path being imitated by a random walk. Sampling refers to methods of deducing properties of a large set of elements by studying only a small, random subset. For example, the average value of $f(x)$ over $a < x < b$ can be estimated from its average over a finite number of points selected randomly in the interval. This amounts to a Monte Carlo method of numerical integration. MCMs have been applied successfully for solving differential and integral equations, for finding eigenvalues, for inverting matrices, and particularly for evaluating multiple integrals.

The simulation of any process or system in which there are inherently random components requires a method of generating or obtaining numbers that are random. Examples of such simulation occur in random collisions of neutrons, in statistics, in queueing models, in games of strategy, and in other competitive enterprises. Monte Carlo calculations require having available sequences of numbers which appear to be drawn at random from particular probability distributions.

8.2 Generation of Random Numbers and Variables

Various techniques for generating random numbers are discussed fully in [5]–[10]. The almost universally used method of generating random numbers is to select a function $g(x)$ that maps integers into random numbers. Select x_0 somehow, and generate the next random number as $x_{k+1} = g(x_k)$. The commonest function $g(x)$ takes the form

$$g(x) = (ax + c) \bmod m \tag{8.1}$$

where

$$\begin{aligned} x_0 &= \text{starting value or a seed } (x_0 > 0)\ , \\ a &= \text{multiplier } (a \geq 0)\ , \\ c &= \text{increment } (c \geq 0)\ , \\ m &= \text{the modulus} \end{aligned}$$

The modulus m is usually 2^t for t-digit binary integers. For a 31-bit computer machine, for example, m may be 2^{31-1}. Here x_0, a, and c are integers in the same range as $m > a, m > c, m > x_0$. The desired sequence of random numbers $\{x_n\}$ is obtained from

$$\boxed{x_{n+1} = (ax_n + c) \bmod m} \tag{8.2}$$

This is called a *linear congruential sequence.* For example, if $x_0 = a = c = 7$ and $m = 10$, the sequence is

$$7, 6, 9, 0, 7, 6, 9, 0, \ldots \tag{8.3}$$

It is evident that congruential sequences always get into a loop; i.e., there is ultimately a cycle of numbers that is repeated endlessly. The sequence in Eq. (8.3) has a period of length 4. A useful sequence will of course have a relatively long period. The terms *multiplicative congruential method* and *mixed congruential method* are used by many authors to denote linear congruential methods with $c = 0$ and $c \neq 0$, respectively. Rules for selecting x_0, a, c, and m can be found in [6, 10].

Here we are interested in generating random numbers from the uniform distribution in the interval (0,1). These numbers will be designated by the letter U and are obtained from Eq. (8.2) as

$$U = \frac{x_{n+1}}{m} \tag{8.4}$$

Thus U can only assume values from the set $\{0, 1/m, 2/m, \ldots, (m-1)/m\}$. (For random numbers in the interval (0,1), a quick test of the randomness is that the mean is 0.5. Other tests can be found in [3, 6].) For generating random numbers X uniformly distributed in the interval (a, b), we use

$$\boxed{X = a + (b-a)\,U} \tag{8.5}$$

Random numbers produced by a computer code (using Eqs. (8.2) and (8.4)) are not truly random; in fact, given the seed of the sequence, all numbers U of the sequence are completely predictable. Some authors emphasize this point by calling such computer-generated sequences *pseudorandom numbers*. However, with a good choice of a, c, and m, the sequences of U appear to be sufficiently random in that they pass a series of statistical tests of randomness. They have the advantage over truly random numbers of being generated in a fast way and of being reproducible, when desired, especially for program debugging.

It is usually necessary in a Monte Carlo procedure to generate random variable X from a given probability distribution $F(x)$. This can be accomplished using several techniques [6], [13]–[15] including the *direct method* and *rejection method.*

The direct method, otherwise known as inversion or transform method, entails inverting the cumulative probability function $F(x) = \text{Prob}(X \leq x)$ associated with the random variable X. The fact that $0 \leq F(x) \leq 1$ intuitively suggests that by generating random number U uniformly distributed over (0,1), we can produce a random sample X from the distribution of $F(x)$ by inversion. Thus to generate random X with probability distribution $F(x)$, we set $U = F(x)$ and obtain

$$X = F^{-1}(U) \tag{8.6}$$

where X has the distribution function $F(x)$. For example, if X is a random variable that is exponentially distributed with mean μ, then

$$F(x) = 1 - e^{-x/\mu}, \quad 0 < x < \infty \tag{8.7}$$

Solving for X in $U = F(X)$ gives

$$X = -\mu \,\ln(1-U) \tag{8.8}$$

Since $(1-U)$ is itself a random number in the interval (0,1), we simply write

$$X = -\mu \ln U \tag{8.9}$$

Sometimes the inverse $F^{-1}(x)$ required in Eq. (8.6) does not exist or is difficult to obtain. This situation can be handled using the rejection method. Let $f(x) = \dfrac{dF(x)}{dx}$ be the probability density function of the random variable X. Let $f(x) = 0$ for $a > x > b$, and $f(x)$ is bounded by M (i.e., $f(x) \leq M$) as shown in Fig. 8.1. We generate two random numbers (U_1, U_2) in the interval (0,1). Then

$$X_1 = a + (b-a)U_1 \quad \text{and} \quad f_1 = U_2 M \tag{8.10}$$

are two random numbers with uniform distributions in (a, b) and $(0, M)$, respectively. If

$$f_1 \leq f(X_1) \tag{8.11}$$

then X_1 is accepted as choice of X, otherwise X_1 is rejected and a new pair (U_1, U_2) is tried again. Thus in the rejection technique all points falling above $f(x)$ are rejected, while those points falling on or below $f(x)$ are utilized to generate X_1 through $X_1 = a + (b-a)U_1$.

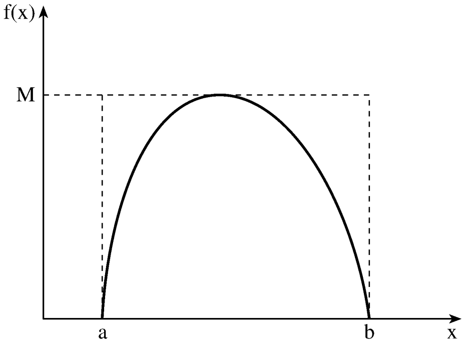


Figure 8.1

The rejection method of generating a random variable with probability density function $f(x)$.

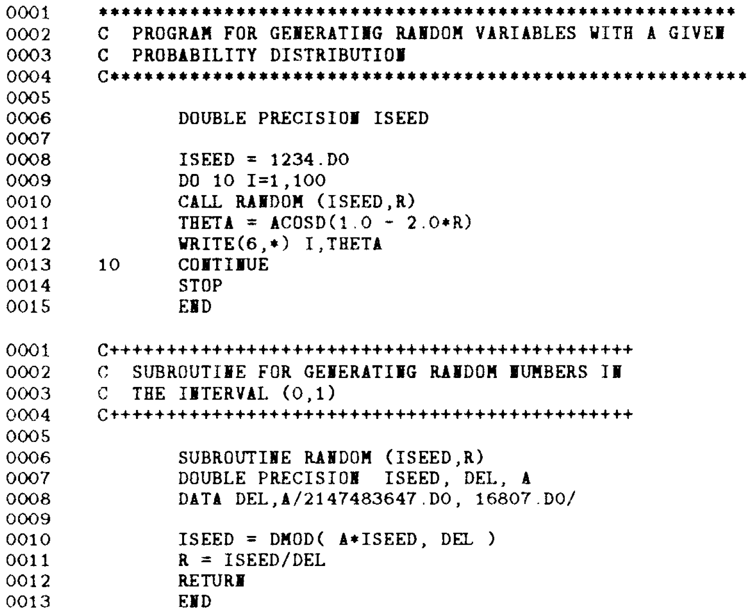
Example 8.1

Develop a subroutine for generating random number U uniformly distributed between 0 and 1. Using this subroutine, generate random variable Θ with probability distribution given by

$$T(\theta) = \frac{1}{2}(1-\cos\theta), \quad 0 < \theta < \pi \qquad \square$$

Solution

The subroutine for generating U is shown in Fig. 8.2. In this subroutine, $m = 2^{21} - 1 = 2147483647$, $c = 0$, and $a = 7^5 = 16807$. By supplying a seed (e.g., 1234), the subroutine provides one random number U per call in the main program. The seed is selected as any integer between 1 and m.



```
********************************************************
C  PROGRAM FOR GENERATING RANDOM VARIABLES WITH A GIVEN
C  PROBABILITY DISTRIBUTION
C********************************************************

       DOUBLE PRECISION ISEED

       ISEED = 1234.D0
       DO 10 I=1,100
       CALL RANDOM (ISEED,R)
       THETA = ACOSD(1.0 - 2.0*R)
       WRITE(6,*) I,THETA
10     CONTINUE
       STOP
       END
```

```
C+++++++++++++++++++++++++++++++++++++++++++++++
C  SUBROUTINE FOR GENERATING RANDOM NUMBERS IN
C  THE INTERVAL (0,1)
C+++++++++++++++++++++++++++++++++++++++++++++++

       SUBROUTINE RANDOM (ISEED,R)
       DOUBLE PRECISION  ISEED, DEL, A
       DATA DEL,A/2147483647.D0, 16807.D0/

       ISEED = DMOD( A*ISEED, DEL )
       R = ISEED/DEL
       RETURN
       END
```

Figure 8.2
Random number generator; for Example 8.1.

The subroutine in Fig. 8.2 is meant to illustrate the concepts explained in this section. Most computers have routines for generating random numbers.

To generate the random variable Θ, set

$$U = T(\Theta) = \frac{1}{2}(1 - \cos\Theta) ,$$

then

$$\Theta = T^{-1}(U) = \cos^{-1}(1 - 2U)$$

Using this, a sequence of random numbers Θ with the given distribution is generated in the main program of Fig. 8.2. ▮

8.3 Evaluation of Error

Monte Carlo procedures give solutions which are averages over a number of tests. For this reason, the solutions contain fluctuations about a mean value, and it is impossible to ascribe a 100% confidence in the results. To evaluate the statistical uncertainty in Monte Carlo calculations, we must resort to various statistical techniques associated with random variables. We briefly introduce the concepts of expected value and variance, and utilize the central limit theorem to arrive at an error estimate [13, 16].

Suppose that X is a random variable. The *expected value* or *mean value* $\bar{x}$ of X is defined as

$$\bar{x} = \int_{-\infty}^{\infty} x f(x)\, dx \tag{8.12}$$

where $f(x)$ is the probability density distribution of X. If we draw random and independent samples, $x_1, x_2, \ldots, x_N$ from $f(x)$, our estimate of x would take the form of the mean of N samples, namely,

$$\hat{x} = \frac{1}{N} \sum_{n=1}^{N} x_n \tag{8.13}$$

While $\bar{x}$ is the true mean value of X, $\hat{x}$ is the unbiased estimator of $\bar{x}$, an unbiased estimator being one with the correct expectation value. Although the expected value of $\hat{x}$ is equal to $\bar{x}$, $\hat{x} \neq \bar{x}$. Therefore, we need a measure of the spread in the values of $\hat{x}$ about $\bar{x}$.

To estimate the spread of values of X, and eventually of $\hat{x}$ about $\bar{x}$, we introduce the *variance* of X defined as the expected value of the square of the deviation of X from $\bar{x}$, i.e.,

$$\operatorname{Var}(x) = \sigma^2 = \overline{(x-\bar{x})^2} = \int_{-\infty}^{\infty} (x-\bar{x})^2 f(x)\, dx \tag{8.14}$$

But $(x-\bar{x})^2 = x^2 - 2x\bar{x} + \bar{x}^2$. Hence

$$\sigma^2(x) = \int_{-\infty}^{\infty} x^2 f(x)\, dx - 2\bar{x} \int_{-\infty}^{\infty} x f(x)\, dx + \bar{x}^2 \int_{-\infty}^{\infty} f(x)\, dx \tag{8.15}$$

or

$$\sigma^2(x) = \overline{x^2} - \bar{x}^2 \tag{8.16}$$

The square root of the variance is called the *standard deviation,* i.e.,

$$\sigma(x) = \left(\overline{x^2} - \bar{x}^2\right)^{1/2} \tag{8.17}$$

The standard deviation provides a measure of the spread of x about the mean value $\bar{x}$; it yields the order of magnitude of the error. The relationship between the variance of $\hat{x}$ and the variance of x is

$$\sigma\left(\hat{x}\right) = \frac{\sigma(x)}{\sqrt{N}} \tag{8.18}$$

This shows that if we use $\hat{x}$ constructed from N values of x_n according to Eq. (8.13) to estimate $\bar{x}$, then the spread in our results of $\hat{x}$ about $\bar{x}$ is proportional to $\sigma(x)$ and falls off as the number of N of samples increases.

In order to estimate the spread in $\hat{x}$, we define the *sample variance*

$$S^2 = \frac{1}{N-1} \sum_{n=1}^{N} \left(x_n - \hat{x}\right)^2 \tag{8.19}$$

Again, it can be shown that the expected value of S^2 is equal to $\sigma^2(x)$. Therefore the sample variance is an unbiased estimator of $\sigma^2(x)$. Multiplying out the square term in Eq. (8.19), it is readily shown that the *sample standard deviation* is

$$S = \left(\frac{N}{N-1}\right)^{1/2} \left[\frac{1}{N} \sum_{n=1}^{N} x_n^2 - \hat{x}^2\right]^{1/2} \tag{8.20}$$

For large N, the factor $N/(N-1)$ is set equal to one.

As a way of arriving at the *central limit theorem,* a fundamental result in probability theory, consider the binomial function

$$B(M) = \frac{N!}{M!(N-M)!} p^M q^{N-M} \tag{8.21}$$

which is the probability of M successes in N independent trials. In Eq. (8.21), p is the probability of success in a trial, and $q = 1 - p$. If M and $N - M$ are large, we may use *Stirling's formula*

$$n! \sim n^n e^{-n} \sqrt{2\pi n} \tag{8.22}$$

so that Eq. (8.21) is approximated [17] as the normal distribution:

$$B(M) \simeq f\left(\hat{x}\right) = \frac{1}{\sigma\left(\hat{x}\right)\sqrt{2\pi}} \exp\left[-\frac{\left(\hat{x} - \bar{x}\right)^2}{2\sigma^2\left(\hat{x}\right)}\right] \tag{8.23}$$

where $\bar{x} = N_p$ and $\sigma(\hat{x}) = \sqrt{Npq}$. Thus as $N \to \infty$, the central limit theorem states that the probability density function which describes the distribution of $\hat{x}$ that results from N Monte Carlo calculations is the normal distribution $f(\hat{x})$ in Eq. (8.23). In other words, the sum of a large number of random variables tends to be normally distributed. Inserting Eq. (8.18) into Eq. (8.23) gives

$$f\left(\hat{x}\right) = \sqrt{\frac{N}{2\pi}} \frac{1}{\sigma(x)} \exp\left[-\frac{N\left(\hat{x} - \bar{x}\right)^2}{2\sigma^2(x)}\right] \tag{8.24}$$

The normal (or Gaussian) distribution is very useful in various problems in engineering, physics, and statistics. The remarkable versatility of the Gaussian model stems from the central limit theorem. For this reason, the Gaussian model often applies to situations in which the quantity of interest results from the summation of many irregular and fluctuating components. In Example 8.2, we present an algorithm based on central limit theorem for generating Gaussian random variables.

Since the number of samples N is finite, absolute certainty in Monte Carlo calculations is unattainable. We try to estimate some limit or interval around $\bar{x}$ such that we can predict with some confidence that $\hat{x}$ falls within that limit. Suppose we want the probability that $\hat{x}$ lies between $\bar{x}-\delta$ and $\bar{x}+\delta$. By definition,

$$\text{Prob}\left\{\bar{x}-\delta < \hat{x} < \bar{x}+\delta\right\} = \int_{\bar{x}-\delta}^{\bar{x}+\delta} f\left(\hat{x}\right) d\hat{x} \tag{8.25}$$

By letting $\lambda = \dfrac{(\hat{x}-\bar{x})}{\sqrt{2/N}\sigma(x)}$,

$$\begin{aligned}\text{Prob}\left\{\bar{x}-\delta < \hat{x} < \bar{x}+\delta\right\} &= \frac{2}{\sqrt{\pi}} \int_0^{(\sqrt{N/2})(\delta/\sigma)} e^{-\lambda^2} d\lambda \\ &= \text{erf}\left(\sqrt{N/2}\frac{\delta}{\sigma(x)}\right) \qquad (8.26\text{a})\end{aligned}$$

or

$$\text{Prob}\left\{\bar{x} - z_{\alpha/2}\frac{\sigma}{\sqrt{N}} \le \hat{x} \le \bar{x} + z_{\alpha/2}\frac{\sigma}{\sqrt{N}}\right\} = 1-\alpha \tag{8.26b}$$

where $\text{erf}(x)$ is the error function and $z_{\alpha/2}$ is the upper $\alpha/2 \times 100$ percentile of the standard normal deviation. Equation (8.26) may be interpreted as follows: if the Monte Carlo procedure of taking random and independent observations and constructing the associated random interval $\bar{x} \pm \delta$ is repeated for large N, approximately $\text{erf}\left(\sqrt{\frac{N}{2}}\frac{\delta}{\sigma(x)}\right) \times 100$ percent of these random intervals will contain $\hat{x}$. The random interval $\hat{x} \pm \delta$ is called a *confidence interval* and $\text{erf}\left(\sqrt{\frac{N}{2}}\frac{\delta}{\sigma(x)}\right)$ is the *confidence level.* Most Monte Carlo calculations use error $\delta = \sigma(x)/\sqrt{N}$, which implies that $\hat{x}$ is within one standard deviation of $\bar{x}$, the true mean. From Eq. (8.26), it means that the probability that the sample mean $\hat{x}$ lies within the interval $\hat{x} \pm \sigma(x)/\sqrt{N}$ is 0.6826 or 68.3%. If higher confidence levels are desired, two or three standard deviations may be used. For example,

$$\text{Prob}\left(\bar{x} - M\frac{\sigma(x)}{\sqrt{N}} < \hat{x} < \bar{x} + M\frac{\sigma(x)}{\sqrt{N}}\right) = \begin{cases} 0.6826, & M=1 \\ 0.954, & M=2 \\ 0.997, & M=3 \end{cases} \tag{8.27}$$

where M is the number of standard deviations.

In Eqs. (8.26) and (8.27), it is assumed that the population standard deviation σ is known. Since this is rarely the case, σ must be estimated by the sample standard deviation S calculated from Eq. (8.20) so that the normal distribution is replaced by the student's t-distribution. It is well known that the t-distribution approaches the normal distribution as N becomes large, say $N > 30$. Equation (8.26) is equivalent to

$$\text{Prob}\left\{\bar{x} - \frac{St_{\alpha/2;N-1}}{\sqrt{N}} \leq \hat{x} \leq \bar{x} + \frac{St_{\alpha/2;N-1}}{\sqrt{N}}\right\} = 1 - \alpha \tag{8.28}$$

where $t_{\alpha/2;N-1}$ is the upper $100 \times \alpha/2$ percentage point of the student's t-distribution with $(N - 1)$ degrees of freedom; and its values are listed in any standard statistics text. Thus the upper and lower limits of a confidence interval are given by

$$\boxed{\begin{aligned} \text{upper limit} &= \bar{x} + \frac{St_{\alpha/2;N-1}}{\sqrt{N}} \qquad (8.29) \\ \text{lower limit} &= \bar{x} - \frac{St_{\alpha/2;N-1}}{\sqrt{N}} \qquad (8.30) \end{aligned}}$$

For further discussion on error estimates in Monte Carlo computations, consult [18, 19].

Example 8.2

A random variable X with Gaussian (or normal) distribution is generated using the central limit theorem. According to the central limit theorem, the sum of a large number of independent random variables about a mean value approaches a Gaussian distribution regardless of the distribution of the individual variables. In other words, for any random numbers, Y_i, $i = 1, 2, \ldots, N$ with mean $\overline{Y}$ and variance $\text{Var}(Y)$,

$$Z = \frac{\sum_{i=1}^{N} Y_i - N\overline{Y}}{\sqrt{N}\,\text{Var}(Y)} \tag{8.31}$$

converges asymptotically with N to a normal distribution with zero mean and a standard deviation of unity. If Y_i are uniformly distributed variables (i.e., $Y_i = U_i$), then $\overline{Y} = 1/2$, $\text{Var}(Y) = 1/\sqrt{12}$, and

$$Z = \frac{\sum_{i=1}^{N} U_i - N/2}{\sqrt{N/12}} \tag{8.32}$$

and the variable

$$X = \sigma Z + \mu \tag{8.33}$$

approximates the normal variable with mean μ and variance σ^2. A value of N as low as 3 provides a close approximation to the familiar bell-shaped Gaussian distribution. To ease computation, it is a common practice to set $N = 12$ since this choice eliminates the square root term in Eq. (8.32). However, this value of N truncates the distribution at $\pm 6\sigma$ limits and is unable to generate values beyond 3σ. For simulations in which the tail of the distribution is important, other schemes for generating Gaussian distribution must be used [20]–[22].

Thus, to generate a Gaussian variable X with mean μ and standard deviation σ, we follow these steps:

(1) Generate 12 uniformly distributed random numbers $U_1, U_2, \ldots, U_{12}$.

(2) Obtain $Z = \sum_{i=1}^{12} U_i - 6$.

(3) Set $X = \sigma Z + \mu$. □

8.4 Numerical Integration

For one-dimensional integration, several quadrature formulas, such as presented in Section 3.10, exist. The numbers of such formulas are relatively few for multidimensional integration. It is for such multidimensional integrals that a Monte Carlo technique becomes valuable for at least two reasons. The quadrature formulas become very complex for multiple integrals, while the MCM remains almost unchanged. The convergence of Monte Carlo integration is independent of dimensionality, which is not true for quadrature formulas. The statistical method of integration has been found to be an efficient way to evaluate two- or three-dimensional integrals in antenna problems, particularly those involving very large structures [23]. Two types of Monte Carlo integration procedures, the crude MCM and the MCM with antithetic variates, will be discussed. For other types, such as hit-or-miss and control variates, see [24]–[26]. Application of MCM to improper integrals will be covered briefly.

8.4.1 Crude Monte Carlo Integration

Suppose we wish to evaluate the integral

$$I = \int_R f \tag{8.34}$$

where R is an n-dimensional space. Let $\mathbf{X} = (X^1, X^2, \ldots, X^n)$ be a random variable that is uniformly distributed in R. Then $f(\mathbf{X})$ is a random variable whose mean value is given by [27, 28]

$$\overline{f(\mathbf{X})} = \frac{1}{|R|}\int_R f = \frac{I}{|R|} \tag{8.35}$$

and the variance by

$$\mathrm{Var}(f(\mathbf{X})) = \frac{1}{|R|}\int_R f^2 - \left(\frac{1}{|R|}\int_R f\right)^2 \tag{8.36}$$

where

$$|R| = \int_R d\mathbf{X} \tag{8.37}$$

If we take N independent samples of $\mathbf{X}$, i.e., $\mathbf{X}_1, \mathbf{X}_2, \ldots \mathbf{X}_N$, all having the same distribution as $\mathbf{X}$ and form the average

$$\frac{f(\mathbf{X}_1) + f(\mathbf{X}_2) + \cdots + f(\mathbf{X}_N)}{N} = \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{X}_i) \tag{8.38}$$

we might expect this average to be close to the mean of $f(\mathbf{X})$. Thus, from Eqs. (8.35) and (8.38),

$$\boxed{I = \frac{|R|}{N}\sum_{i=1}^{N} f(\mathbf{X}_i)} \tag{8.39}$$

This Monte Carlo formula applies to any integration over a finite region R. For the purpose of illustration, we now apply Eq. (8.39) to one- and two-dimensional integrals.

For a one-dimensional integral, suppose

$$I = \int_a^b f(x)\,dx \tag{8.40}$$

Applying Eq. (8.39) yields

$$\boxed{I = \frac{b-a}{N}\sum_{i=1}^{N} f(X_i)} \tag{8.41}$$

where X_i is a random number in the interval (a, b), i.e.,

$$X_i = a + (b-a)\,U, \quad 0 < U < 1 \tag{8.42}$$

For a two-dimensional integral

$$I = \int_a^b \int_c^d f\left(X^1, X^2\right)\,dX^1\,dX^2\,, \tag{8.43}$$

the corresponding Monte Carlo formula is

$$\boxed{I = \frac{(b-a)(d-c)}{N} \sum_{i=1}^{N} f\left(X_i^1, X_i^2\right)} \tag{8.44}$$

where

$$\begin{aligned} X_i^1 &= a + (b-a)\,U^1, \quad 0 < U^1 < 1 \\ X_i^2 &= c + (d-c)\,U^2, \quad 0 < U^2 < 1 \end{aligned} \tag{8.45}$$

The convergence behavior of the unbiased estimator I in Eq. (8.39) is slow since the variance of the estimator is of the order $1/N$. Accuracy and convergence is increased by reducing the variance of the estimator using an improved method, the method of antithetic variates.

8.4.2 Monte Carlo Integration with Antithetic Variates

The term *antithetic variates* [29, 30] is used to describe any set of estimators which mutually compensate each other's variations. For convenience, we assume that the integral is over the interval (0,1). Suppose we want an estimator for the single integral

$$I = \int_0^1 g(U)\,dU \tag{8.46}$$

We expect the quantity $\frac{1}{2}[g(U) + g(1-U)]$ to have smaller variance than $g(U)$. If $g(U)$ is too small, then $g(1-U)$ will have a good chance of being too large and conversely. Therefore, we define the estimator

$$I = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{2} \left[g\left(U_i\right) + g\left(1 - U_i\right)\right] \tag{8.47}$$

where U_i are random numbers between 0 and 1. The variance of the estimator is of the order $\frac{1}{N^4}$, a tremendous improvement over Eq. (8.39). For two-dimensional integral,

$$I = \int_0^1 \int_0^1 g\left(U^1, U^2\right)\,dU^1\,dU^2\,, \tag{8.48}$$

and the corresponding estimator is

$$\begin{aligned} I = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{4} \Big[& g\left(U_i^1, U_i^2\right) + g\left(U_i^1, 1 - U_i^2\right) \\ & + g\left(1 - U_i^1, U_i^2\right) + g\left(1 - U_i^1, 1 - U_i^2\right) \Big] \end{aligned} \tag{8.49}$$

Following similar lines, the idea can be extended to higher order integrals. For intervals other than (0,1), transformations such as in Eqs. (8.41) to (8.45) should be applied. For example,

$$\begin{aligned}\int_a^b f(x)\,dx &= (b-a)\int_0^1 g(U)\,dU \\ &\simeq \frac{b-a}{N}\sum_{i=1}^{N}\frac{1}{2}\left[g\left(U_i\right)+g\left(1-U_i\right)\right] \qquad (8.50)\end{aligned}$$

where $g(U) = f(X)$ and $X = a+(b-a)U$. It is observed from Eqs. (8.47) and (8.49) that as the number of dimensions increases, the minimum number of antithetic variates per dimension required to obtain an increase in efficiency over crude Monte Carlo also increases. Thus the crude Monte Carlo method becomes preferable in many dimensions.

8.4.3 Improper Integrals

The integral

$$I = \int_0^\infty g(x)\,dx \tag{8.51}$$

may be evaluated using Monte Carlo simulations [31]. For a random variable X having probability density function $f(x)$, where $f(x)$ integrates to 1 on interval $(0, \infty)$,

$$\int_0^\infty \frac{g(x)}{f(x)}\,dx = \int_0^\infty g(x)\,dx \tag{8.52}$$

Hence, to compute I in Eq. (8.51), we generate N independent random variables distributed according to a probability density function $f(x)$ integrating to 1 on the interval $(0, \infty)$. The sample mean

$$\overline{g(x)} = \frac{1}{N}\sum_{i=1}^{N}\frac{g\left(x_i\right)}{f\left(x_i\right)} \tag{8.53}$$

gives an estimate for I.

Example 8.3

Evaluate the integral

$$I = \int_0^1 \int_0^{2\pi} e^{j\alpha\rho\cos\phi}\rho\,d\rho\,d\phi$$

using the Monte Carlo method. □

Solution

This integral represents radiation from a circular aperture-antenna with a constant amplitude and phase distribution. It is selected because it forms at least part of every radiation integral. The solution is available in the closed form, which can be used to assess the accuracy of the Monte Carlo results. In closed form,

$$I(\alpha) = \frac{2\pi J_1(\alpha)}{\alpha}$$

where $J_1(\alpha)$ is Bessel function of the first order.

A simple program for evaluating the integral employing Eqs. (8.44) and (8.45), where $a = 0, b = 1, c = 0$, and $d = 2\pi$, is shown in Fig. 8.3. The program calls the routine RANDU in Vax 11/780 to generate random numbers U^1 and U^2. For different values of N, both the crude and antithetic variate Monte Carlo methods are used in evaluating the radiation integral, and the results are compared with the exact value in Table 8.1 for $\alpha = 5$. In applying Eq. (8.49), the following correspondences are used:

$$U^1 \equiv X^1, U^2 \equiv X^2, 1 - U^1 \equiv b - X^1 = (b-a)\left(1 - U^1\right) ,$$

$$1 - U^2 \equiv d - X^2 = (d-c)\left(1 - U^2\right) \quad \blacksquare$$

Table 8.1 Results of Example 8.3 on Monte Carlo Integration of Radiation Integral

N	Crude MCM	Antihetic variates MCM
500	$-0.2892 - j0.0742$	$-0.2887 - j0.0585$
1000	$-0.5737 + j0.0808$	$-0.4982 - j0.0080$
2000	$-0.4922 - j0.0040$	$-0.4682 - j0.0082$
4000	$-0.3999 - j0.0345$	$-0.4216 - j0.0323$
6000	$-0.3608 - j0.0270$	$-0.3787 - j0.0440$
8000	$-0.4327 - j0.0378$	$-0.4139 - j0.0241$
10,000	$-0.4229 - j0.0237$	$-0.4121 - j0.0240$
Exact: $-0.4116 + j0$		

8.5 Solution of Potential Problems

The connection between potential theory and Brownian motion (or random walk) was first shown in 1944 by Kakutani [32]. Since then the resulting so-called probabilistic potential theory has been applied to problems in many disciplines such as heat conduction [33]–[38], electrostatics [39]–[46], and electrical power engineering [47, 48]. An underlying concept of the probabilistic or Monte Carlo solution of

```
C**********************************************
C  INTEGRATION USING CRUDE MONTE CARLO
C  AND ANTITHETIC METHODS
C
C ONLY FEW LINES NEED BE CHANGED TO USE THIS
C PROGRAM FOR ANY MULTI-DIMENSIONAL INTEGRATION
C**********************************************

       DATA  IS1,IS2,IS3,IS4/1234,5678,9012,3456/
       DATA  A,B,C/0.0,1.0,0.0/
                    ! LIMITS OF INTEGRATION
       COMPLEX F,SUM1, SUM2, J, AREA1, AREA2

C
C   SPECIFY THE INTEGRAND
C
      F(RHO,PHI) = RHO*CEXP(J*ALPHA*RHO*COS(PHI))

      J = (0.0,1.0)
      ALPHA = 5.0
      PIE = 3.1415927
      D = 2.0*PIE
      DO 30 NRUN = 500,10000,500  ! NO. OF RUNS
      SUM1 = (0.0,0.0)
      SUM2 = (0.0,0.0)
      DO 10 I=1,NRUN
      CALL RANDU(IS1,IS2,U1)
      CALL RANDU(IS3,IS4,U2)
      X1 = A + (B - A)*U1
      X2 = C + (D - C)*U2
      X3 = (B - A)*(1.0 - U1)
      X4 = (D - C)*(1.0 - U2)
      SUM1 = SUM1 + F(X1,X2)
      SUM2 = SUM2 + F(X1,X2) + F(X1,X4) + F(X3,X2)
     1           + F(X3,X4)
10    CONTINUE
      AREA1 = (B-A)*(D-C)*SUM1/FLOAT(NRUN)
      AREA2 = (B-A)*(D-C)*SUM2/(4.0*FLOAT(NRUN))
      PRINT *,NRUN, AREA1, AREA2
      WRITE(6,*) NRUN, AREA1,AREA2
      WRITE(6,20) NRUN,AREA1,AREA2
20    FORMAT(2X,'NRUN =',I5,3X,'AREA1 = ',F12.6,3X,F12.6,'AREA2 = ',
     1          F12.6,3X,F12.6,/)
30    CONTINUE
      STOP
      END
```

Figure 8.3

Program for Monte Carlo evaluation of two-dimensional integral; for Example 8.3.

differential equations is the random walk. Different types of random walk lead to different Monte Carlo methods. The most popular types are the *fixed-random walk* and *floating random walk*. Other types that are less popular include the *Exodus method, shrinking boundary method, inscribed figure method,* and the *surface density method.*

8.5.1 Fixed Random Walk

Suppose, for concreteness, that the MCM with fixed random walk is to be applied to solve Laplace's equation

$$\nabla^2 V = 0 \quad \text{in region } R \tag{8.54a}$$

subject to Dirichlet boundary condition

$$V = V_p \text{ on boundary } B \tag{8.54b}$$

We begin by dividing R into mesh and replacing ∇^2 by its finite difference equivalent. The finite difference representation of Eq. (8.54a) in two-dimensional R is given by Eq. (3.26), namely,

$$\boxed{\begin{aligned} V(x,y) = p_{x+}V(x+\Delta,y) + p_{x-}V(x-\Delta,y) \\ +p_{y+}V(x,y+\Delta) + p_{y-}V(x,y-\Delta) \end{aligned}} \tag{8.55a}$$

where

$$\boxed{p_{x+} = p_{x-} = p_{y+} = p_{y-} = \frac{1}{4}} \tag{8.55b}$$

In Eq. (8.55), a square grid of mesh size Δ, such as in Fig. 8.4, is assumed. The equation may be given a probabilistic interpretation. If a random walking particle is instantaneously at the point (x, y), it has probabilities p_{x+}, p_{x-}, p_{y+}, and p_{y-} of moving from (x, y) to $(x+\Delta, y)$, $(x-\Delta, y)$, $(x, y+\Delta)$, and $(x, y-\Delta)$, respectively. A means of determining which way the particle should move is to generate a random number U, $0 < U < 1$ and instruct the particle to walk as follows:

$$\boxed{\begin{array}{ll} (x,y) \rightarrow (x+\Delta,y) & \text{if } 0 < U < 0.25 \\ (x,y) \rightarrow (x-\Delta,y) & \text{if } 0.25 < U < 0.5 \\ (x,y) \rightarrow (x,y+\Delta) & \text{if } 0.5 < U < 0.75 \\ (x,y) \rightarrow (x,y-\Delta) & \text{if } 0.75 < U < 1 \end{array}} \tag{8.56}$$

If a rectangular grid rather than a square grid is employed, then $p_{x+} = p_{x-}$ and $p_{y+} = p_{y-}$, but $p_x \neq p_y$. Also for a three-dimensional problem in which cubical cells are used, $p_{x+} = p_{x-} = p_{y+} = p_{y-} = p_{z+} = p_{z-} = \frac{1}{6}$. In both cases, the interval $0 < U < 1$ is subdivided according to the probabilities.

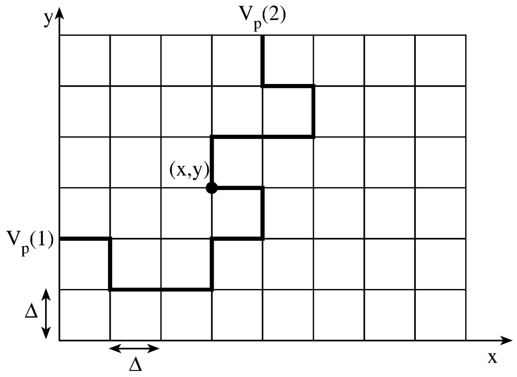


Figure 8.4
Configuration for fixed random walks.

To calculate the potential at (x, y), a random-walking particle is instructed to start at that point. The particle proceeds to wander from node to node in the grid until it reaches the boundary. When it does, the walk is terminated and the prescribed potential V_p at that boundary point is recorded. Let the value of V_p at the end of the first walk be denoted by $V_p(1)$, as illustrated in Fig. 8.4. Then a second particle is released from (x, y) and allowed to wander until it reaches a boundary point, where the walk is terminated and the corresponding value of V_p is recorded as $V_p(2)$. This procedure is repeated for the third, fourth, ..., and Nth particle released from (x, y), and the corresponding prescribed potential $V_p(3)$, $V_p(4)$, ..., $V_p(N)$ are noted. According to Kakutani [32], the expected value of $V_p(1)$, $V_P(2)$, ..., $V_p(N)$ is the solution of the Dirichlet problem at (x, y), i.e.,

$$\boxed{V(x, y) = \frac{1}{N}\sum_{i=1}^{N} V_p(i)} \tag{8.57}$$

where N, the total number of walks, is large. The rate of convergence varies as $\sqrt{N}$ so that many random walks are required to ensure accurate results.

If it is desired to solve Poisson's equation

$$\nabla^2 V = -g(x, y) \quad \text{in } R \tag{8.58a}$$

subject to

$$V = V_p \quad \text{on } B\,, \tag{8.58b}$$

then the finite difference representation is in Eq. (8.25), namely,

$$\begin{aligned} V(x, y) = {} & p_{x+}V(x+\Delta, y) + p_{x-}V(x-\Delta, y) \\ & + p_{y+}V(x, y+\Delta) + p_{y-}V(x, y-\Delta) + \frac{\Delta^2 g}{4} \end{aligned} \tag{8.59}$$

where the probabilities remain as stated in Eq. (8.55b). The probabilistic interpretation of Eq. (8.59) is similar to that for Eq. (8.55). However, the term $\Delta^2 g/4$ in Eq. (8.59) must be recorded at each step of the random walk. If m_i steps are required for the ith random walk originating at (x, y) to reach the boundary, then one records

$$V_p(i) + \frac{\Delta^2}{4} \sum_{j=1}^{m_i-1} g\left(x_j, y_j\right) \tag{8.60}$$

Thus the Monte Carlo result for $V(x, y)$ is

$$\boxed{V(x, y) = \frac{1}{N} \sum_{i=1}^{N} V_p(i) + \frac{\Delta^2}{4N} \sum_{i=1}^{N} \left[\sum_{j=1}^{m_i-1} g\left(x_j, y_j\right) \right]} \tag{8.61}$$

An interesting analogy to the MCM just described is the walking drunk problem [15, 35]. We regard the random-walking particle as a "drunk," the squares of the mesh as the "blocks in a city," the nodes as "crossroads," the boundary B as the "city limits," and the terminus on B as the "policeman." Though the drunk is trying to walk home, he is so intoxicated that he wanders randomly throughout the city. The job of the policeman is to seize the drunk in his first appearance at the city limits and ask him to pay a fine V_p. What is the expected fine the drunk will receive? The answer to this problem is in Eq. (8.57).

On the dielectric boundary, the boundary condition $D_{1n} = D_{2n}$ is imposed. Consider the interface along y = constant plane as shown in Fig. 8.5. According to Eq. 3.46, the finite difference equivalent of the boundary condition at the interface is

$$V_o = p_{x+} V_1 + p_{x-} V_2 + p_{y+} V_3 + p_{y-} V_4 \tag{8.62a}$$

where

$$p_{x+} = p_{x-} = \frac{1}{4}, \qquad p_{y+} = \frac{\epsilon_1}{2\left(\epsilon_1 + \epsilon_2\right)}, \qquad p_{y-} = \frac{\epsilon_2}{2\left(\epsilon_1 + \epsilon_2\right)} \tag{8.62b}$$

An interface along x = constant plane can be treated in a similar manner.

On a line of symmetry, the condition $\frac{\partial V}{\partial n} = 0$ must be imposed. If the line of symmetry is along the y-axis as in Fig. 8.6(a), according to Eq. 3.48.

$$V_o = p_{x+} V_1 + p_{y+} V_3 + p_{y-} V_4 \tag{8.63a}$$

where

$$p_{x+} = \frac{1}{2} \quad p_{y+} = p_{y-} = \frac{1}{4} \tag{8.63b}$$

The line of symmetry along the x-axis, shown in Fig. 8.6(b), is treated similarly following Eq. 3.49.

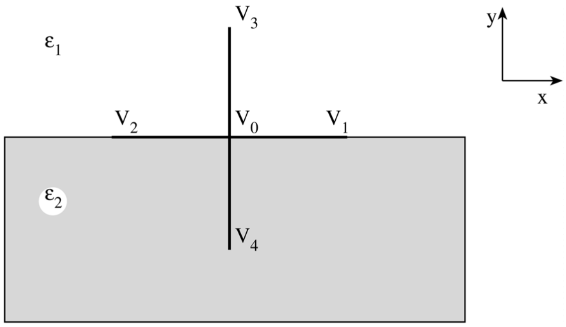
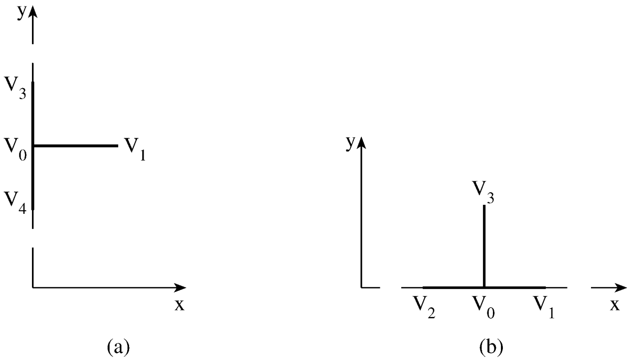


Figure 8.5
Interface between media of dielectric permittivities ϵ_1 and ϵ_2.



(a) (b)

Figure 8.6
Satisfying symmetry conditions: (a) $\partial V/\partial x = 0$, (b) $\partial V/\partial y = 0$.

For an axisymmetric solution region such as shown in Fig. 8.7, $V = V(\rho, z)$. The finite difference equivalent of Eq. (8.54a) for $\rho \neq 0$ is obtained in Section 3.10 as

$$\begin{aligned} V(\rho, z) = {} & p_{\rho+} V(\rho + \Delta, z) + p_{\rho-} V(\rho - \Delta, z) \\ & + p_{z+} V(\rho, z + \Delta) + p_{z-} V(\rho, z - \Delta) \end{aligned} \tag{8.64}$$

where $\Delta\rho = \Delta z = \Delta$ and the random walk probabilities are given by

$$\begin{aligned} p_{z+} &= p_{z-} = \frac{1}{4} \\ p_{\rho+} &= \frac{1}{4} + \frac{\Delta}{8\rho} \\ p_{\rho-} &= \frac{1}{4} - \frac{\Delta}{8\rho} \end{aligned} \tag{8.65}$$

For $\rho = 0$, the finite difference equivalent of Eq. (8.54a) is Eq. 3.120, namely

$$V(0, z) = p_{\rho+}V(\Delta, z) + p_{z+}V(0, z + \Delta) + p_{z-}V(0, z - \Delta) \tag{8.66}$$

so that

$$p_{\rho+} = \frac{4}{6}, \quad p_{\rho-} = 0, \quad p_{z+} = \frac{1}{6} = p_{z-} \tag{8.67}$$

The random-walking particle is instructed to begin walk at (ρ_o, z_o). It wanders through the mesh according to the probabilities in Eqs. (8.65) and (8.67) until it reaches the boundary where it is absorbed and the prescribed potential $V_p(1)$ is recorded. By sending out N particles from (ρ_o, z_o) and recording the potential at the end of each walk, we obtain the potential at (ρ_o, z_o) as [49]

$$V(\rho_o, z_o) = \frac{1}{N}\sum_{i=1}^{N} V_p(i) \tag{8.68}$$

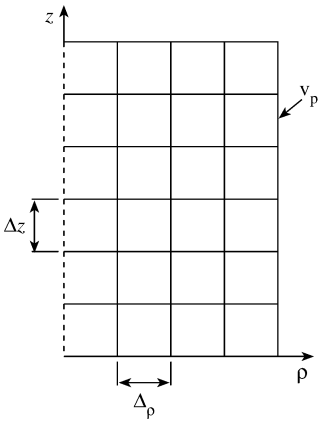


Figure 8.7
Typical axisymmetric solution region.

This MCM is called *fixed random walk* type since the step size Δ is fixed and the steps of the walks are constrained to lie parallel to the coordinate axes. Unlike in the finite difference method (FDM), where the potential at all mesh points are determined simultaneously, MCM is able to solve for the potential at one point at a time. One disadvantage of this MCM is that it is slow if potential at many points is required and is therefore recommended for solving problems for which only a few potentials are required. It shares a common difficulty with FDM in connection with irregularly shaped bodies having Neumann boundary conditions. This drawback is fully removed by employing MCM with floating random walk.

8.5.2 Floating Random Walk

The mathematical basis of the floating random walk method is the mean value theorem of potential theory. If S is a sphere of radius r, centered at (x, y, z), which lies wholly within region R, then

$$V(x, y, z) = \frac{1}{4\pi a^2} \int_S V\left(r'\right) dS' \tag{8.69}$$

That is, the potential at the center of any sphere within R is equal to the average value of the potential taken over its surface. When the potential varies in two dimensions, $V(x, y)$ is given by

$$V(x, y) = \frac{1}{2\pi\rho} \oint_L V\left(\rho'\right) dl' \tag{8.70}$$

where the integration is around a circle of radius ρ centered at (x, y). It can be shown that Eqs. (8.69) and (8.70) follow from Laplace's equation. Also, Eqs. (8.69) and (8.70) can be written as

$$V(x, y, z) = \int_0^1 \int_0^1 V(a, \theta, \phi)\, dF\, dT \tag{8.71}$$

$$V(x, y) = \int_0^1 V(a, \phi)\, dF \tag{8.72}$$

where

$$F = \frac{\phi}{2\pi}, \quad T = \frac{1}{2}(1 - \cos\theta) \tag{8.73}$$

and θ and ϕ are regular spherical coordinate variables. The functions F and T may be interpreted as the probability distributions corresponding to ϕ and θ. While $dF/d\phi =$ constant, $dT/d\theta = \frac{1}{2}\sin\theta$; i.e., all angles ϕ are equally probable, but the same is not true for θ.

The floating random walk MCM depends on the application of Eqs. (8.69) and (8.70) in a statistical sense. For a two-dimensional problem, suppose that a random-walking particle is at some point (x_j, y_j) after j steps in the ith walk. The next $(j + 1)$th step is taken as follows. First, a circle is constructed with center at (x_j, y_j) and radius ρ_j, which is equal to the shortest distance between (x_j, y_j) and the boundary. The ϕ coordinate is generated as a random variable uniformly distributed over $(0, 2\pi)$, i.e., $\phi = 2\pi U$, where $0 < U < 1$. Thus the location of the random-walking particle after the $(j + 1)$th step is illustrated in Fig. 8.8 and given as

$$\boxed{\begin{aligned} x_{j+1} &= x_j + \rho_j \cos\phi_j \\ y_{j+1} &= y_j + \rho_j \sin\phi_j \end{aligned}} \qquad \begin{aligned} &(8.74a) \\ &(8.74b) \end{aligned}$$

The next random walk is executed by constructing a circle centered at (x_{j+1}, y_{j+1}) and of radius ρ_{j+1}, which is the shortest distance between (x_{j+1}, y_{j+1}) and the

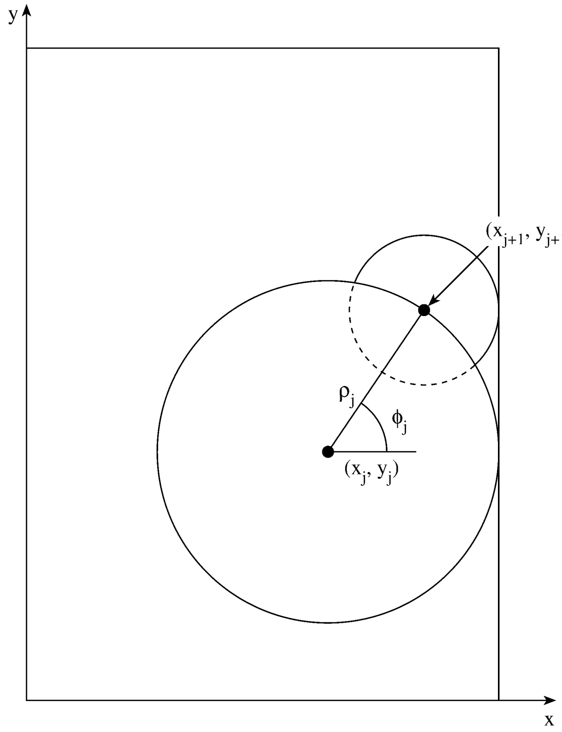


Figure 8.8
Configuration for floating random walks.

boundary. This procedure is repeated several times, and the walk is terminated when the walk approaches some prescribed small distance τ of the boundary. The potential $V_p(i)$ at the end of this ith walk is recorded as in fixed random walk MCM and the potential at (x, y) is eventually determined after N walks using Eq. (8.57).

The floating random walk MCM can be applied to a three-dimensional Laplace problem by proceeding along lines similar to those outlined above. A random-walking particle at (x_j, y_j, z_j) will step to a new location on the surface of a sphere whose radius r_j is equal to the shortest distance between point (x_j, y_j, z_j) and the boundary. The ϕ coordinate is selected as a random number U between 0 and 1, multiplied by 2π. The coordinate θ is determined by selecting another random number U between 0 and 1, and solving for $\theta = \cos^{-1}(1 - 2U)$ as in Example 8.1. Thus the location of the particle after its $(j + 1)$th step in the ith walk is

$$\boxed{\begin{aligned} x_{j+1} &= x_j + r_j \cos\phi_j \sin\theta_j && (8.75a) \\ y_{j+1} &= y_j + r_j \sin\phi_j \sin\theta_j && (8.75b) \\ z_{j+1} &= z_j + r_j \cos\theta_j && (8.75c) \end{aligned}}$$

Finally, we apply Eq. (8.57).

Solving Poisson's equation (8.58) for a two-dimensional problem requires only a slight modification. For a three-dimensional problem, $V(a, \theta, \phi)$ in Eq. (8.71) is replaced by $[V(a, \theta, \phi) + r^2 g/6]$. This requires that the term $gr_j^2/6$ at every jth step of the ith random walk be recorded.

An approach for handling a discretely inhomogeneous medium is presented in [39, 43, 44, 50].

It is evident that in the floating random walk MCM, neither the step sizes nor the directions of the walk are fixed in advance. The quantities may be regarded as "floating" and hence the designation *floating random walk.* A floating random walk bypasses many intermediate steps of a fixed random walk in favor of a long jump. Fewer steps are needed to reach the boundary, and so computation is much more rapid than in fixed random walk.

8.5.3 Exodus Method

The *Exodus method,* first suggested in [51] and developed for electromagnetics in [52, 53], 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 random walk 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 the Exodus method is its independence of the random number generator.

To implement the Exodus method, we first divide the solution region R into mesh, such as in Fig. 8.4. Suppose p_k is the probability that a random walk starting from point (x, y) ends at node k on the boundary with prescribed potential $V_p(k)$. For M boundary nodes (excluding the corner points since a random walk never terminates at those points), the potential at the starting point (x, y) of the random walks is

$$\boxed{V(x, y) = \sum_{k=1}^{M} p_k V_p(k)} \tag{8.76}$$

If m is the number of different boundary potentials ($m = 4$ in Fig. 8.4), Eq. (8.76) can be simplified to

$$V(x, y) = \sum_{k=1}^{m} p_k V_p(k) \tag{8.77}$$

where p_k in this case is the probability that a random walk terminates on boundary 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 the iterative process applied in Section 3.5.

Let $P(i, j)$ be the number of particles at point (i, j) in R. We begin by setting $P(i, j) = 0$ at all points (both fixed and free) except at point (x, y), where $P(i, j)$ assumes a large number N (say, $N = 10^6$ or more). By a scanning process, we dispatch 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. 8.9. Note that in Fig. 8.9(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 as illustrated in Fig. 8.9, we record the number of particles that have reached the boundary (i.e., the fixed nodes). We keep scanning the mesh until a set number of particles (say 99.99% of N) have reached the boundary, where the particles are absorbed. If N_k is the number of particles that reached side k, we calculate

$$p_k = \frac{N_k}{N} \tag{8.78}$$

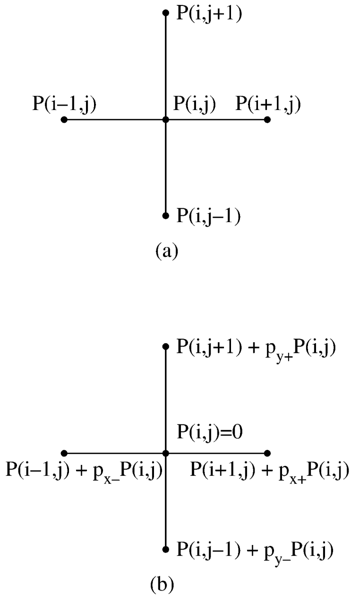


Figure 8.9
(a) Before the particles at (i, j) are dispatched, (b) after the particles at (i, j) are dispatched.

Hence Eq. (8.77) can be written as

$$\boxed{V(x, y) = \frac{\sum_{k=1}^{M} N_k V_p(k)}{N}} \tag{8.79}$$

Thus the problem is reduced to just finding N_k using the Exodus method, given N and $V_p(k)$. We notice that if $N \to \infty$, $\Delta \to 0$, and all the particles were allowed to reach the boundary points, the values of p_k and consequently $V(x, y)$ would be exact. It is easier to approach this exact solution using the Exodus method than any other MCMs or any other numerical techniques such as difference and finite element methods.

We now apply the Exodus method to Poisson's equation. To compute the solution of the problem defined in Eq. (8.58), for example, at a specific point (x_o, y_o), we need the *transition probability* p_k and the *transient probability* q_ℓ. The transition probability p_k is already defined as the probability that a random walk starting at the point of interest (x_o, y_o) in R ends at a boundary point (x_k, y_k), where potential $V_p(k)$ is prescribed, i.e.,

$$p_k = \text{Prob}\,(x_o, y_o \to x_k, y_k) \tag{8.80}$$

The transient probability q_k is the probability that a random walk starting at point (x_o, y_o) passes through point (x_ℓ, y_ℓ) on the way to the boundary, i.e.,

$$p_\ell = \text{Prob}\left(x_o, y_o \xrightarrow{x_\ell, y_\ell} \text{boundary } B\right) \tag{8.81}$$

If there are m boundary (or fixed) nodes (excluding the corner points since a random walk never terminates at those points) and M_f free nodes in the mesh, the potential at the starting point (x_o, y_o) of the random walks is

$$V\,(x_o, y_o) = \sum_{k=1}^{m} p_k V_p(k) + \sum_{\ell=1}^{M_f} q_\ell G_\ell\ , \tag{8.82}$$

where

$$G_\ell = \Delta^2 g\,(x_\ell, y_\ell)\,/4$$

If M_b is the number of different boundary potentials, the first term in Eq. (8.82) can be simplified so that

$$V\,(x_o, y_o) = \sum_{k=1}^{M_b} p_k V_p(k) + \sum_{\ell=1}^{M_f} q_\ell G_\ell\ , \tag{8.83}$$

where p_k in this case is the probability that a random walk terminates on boundary k. Since $V_p(k)$ is specified and the source term G_ℓ is known, our problem is reduced to finding the probabilities p_k and q_ℓ. We notice from Eq. (8.83) that the value of $V(x_o, y_o)$ would be "exact" if the transition probabilities p_k and the transient probabilities q_ℓ were known exactly. These probabilities can be obtained in one of two ways: either analytically or numerically. The analytical approach involves using an expansion technique described in [54]. But this approach is limited to homogeneous rectangular solution regions. For inhomogeneous or non-rectangular regions, we must resort to some numerical simulation. The Exodus method offers a numerical means of finding p_k and q_ℓ. The fixed random walk can also be used to compute the transient and transition probabilities.

To apply the Exodus method, let $P(i, j)$ be the number of particles at point (i, j) in R, while $Q(i, j)$ is the number of particles passing through the same point. We begin the application of the Exodus method by setting $P(i, j) = 0 = Q(i, j)$ at all nodes (both fixed and free) except at free node $(x_o y_o)$ where both $P(i, j)$ and $Q(i, j)$ are set equal to a large number N_p (say, $N_p = 10^6$ or more). In other words, we inject a large number of particles at (x_o, y_o) to start with. By scanning the mesh iteratively as is usually done in finite difference analysis, we dispatch the particles at each free node to its neighboring nodes according to the random walk probabilities p_{x+}, p_{x-}, p_{y+}, and p_{y-} as illustrated in Fig. 8.9. Note that in Fig. 8.9(b), new $P(i, j) = 0$ at that node, while old $P(i, j)$ is shared among the neighboring nodes. As shown in Fig. 8.10, the value of $Q(i, j)$ does not change at that node, while Q at the neighboring nodes is increased by the old $P(i, j)$ that is shared by those nodes. While $P(i, j)$ keeps records of the number of particles at point (i, j) during each iteration, $Q(i, j)$ tallies the number of particles passing through that point.

At the end of each iteration (i.e., scanning of the free nodes in R as illustrated in Figs. 8.9 and 8.10), we record the number of particles that have reached the boundary (i.e., the fixed nodes) where the particles are absorbed. We keep scanning the mesh in a manner similar to the iterative process applied in finite difference solution until a set number of particles (say 99.99% of N_p) have reached the boundary. If N_k is the number of particles that reached boundary k, we calculate

$$p_k = \frac{N_k}{N_p} \tag{8.84}$$

Also, at each free node, we calculate

$$q_\ell = \frac{Q_\ell}{N_p}\ , \tag{8.85}$$

where $Q_\ell = Q(i, j)$ is now the total number of particles that have passed through that node on their way to the boundary. Hence (8.83) can be written as

$$\boxed{V\left(x_o, y_o\right) = \frac{\sum_{k=1}^{M_b} N_k V_p(k)}{N_p} + \frac{\sum_{\ell=1}^{M_f} Q_\ell G_\ell}{N_p}} \tag{8.86}$$

Thus the problem is reduced to just finding N_k and Q_ℓ using the Exodus method, given N_p, $V_p(k)$, and G_ℓ. If $N_p \to \infty$, $\Delta \to 0$, and all the particles were allowed to reach the boundary points, the values of p_k and q_ℓ and consequently $V(x_o, y_o)$ would be exact. It is interesting to note that the accuracy of the Exodus method does not really depend on the number of particle N_p. The accuracy depends on the step size Δ and the number of iteration or the tolerance, the number of particles (say 0.001% of N_p), which are yet to reach the boundary before the iteration is terminated. However, a large value of N_p reduces the truncation error in the computation.

Example 8.4

Give a probabilistic interpretation using the finite difference form of the energy

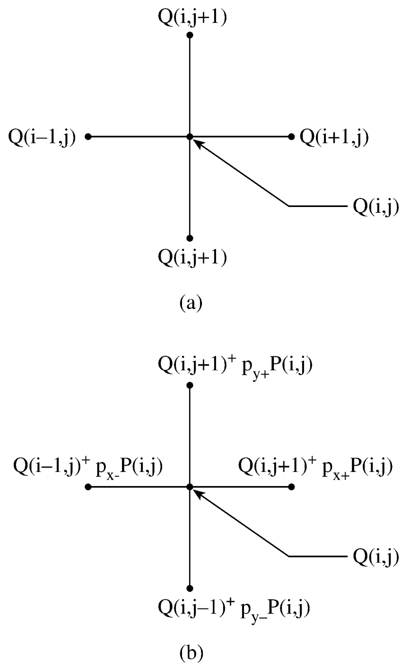


Figure 8.10

Number of particles passing through node (i, j) and its neighboring nodes: (a) before the particles at the node are dispatched, (b) after the particles at the node are dispatched.

equation

$$u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y} = \alpha\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)$$

Assume a square grid of size Δ. □

Solution

Applying a backward difference to the left-hand side and a central difference to the right-hand side, we obtain

$$\begin{aligned} u\frac{T(x,y) - T(x-\Delta,y)}{\Delta} &+ v\frac{T(x,y) - T(x,y-\Delta)}{\Delta} \\ &= \alpha\frac{T(x+\Delta,y) - 2T(x,y) + T(x-\Delta,y)}{\Delta^2} \\ &\quad + \alpha\frac{T(x,y+\Delta) - 2T(x,y) + T(x,y-\Delta)}{\Delta^2} \end{aligned} \tag{8.87}$$

Rearranging terms leads to

$$\begin{aligned} T(x, y) = {} & p_{x+} T(x+\Delta, y) + p_{x-} T(x-\Delta, y) \\ & + p_{y+} T(x, y+\Delta) + p_{y-} T(x, y-\Delta) \end{aligned} \tag{8.88}$$

where

$$p_{x+} = p_{y+} = \frac{1}{\frac{u\Delta}{\alpha} + \frac{v\Delta}{\alpha} + 4} \tag{8.89a}$$

$$p_{x-} = \frac{\left(1 + \frac{\Delta u}{\alpha}\right)}{\frac{u\Delta}{\alpha} + \frac{v\Delta}{\alpha} + 4} \tag{8.89b}$$

$$p_{y-} = \frac{\left(1 + \frac{\Delta v}{\alpha}\right)}{\frac{u\Delta}{\alpha} + \frac{v\Delta}{\alpha} + 4} \tag{8.89c}$$

Equation (8.88) is given probabilistic interpretation as follows: a walker at point (x, y) has probabilities p_{x+}, p_{x-}, p_{y+}, and p_{y-} of moving to point $(x+\Delta, y)$, $(x-\Delta, y)$, $(x, y+\Delta)$, and $(x, y-\Delta)$, respectively. With this interpretation, Eq. (8.88) can be used to solve the differential equation with fixed random MCM. ■

Example 8.5

Consider a conducting trough of infinite length with square cross section shown in Fig. 8.11. The trough wall at $y = 1$ is connected to 100 V, while the other walls are grounded as shown. We intend to find the potential within the trough using the fixed random walk MCM. □

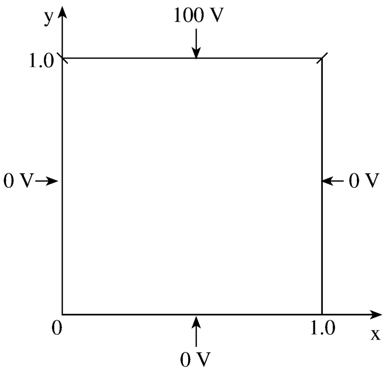


Figure 8.11
For Example 8.5.

Solution

The problem is solving Laplace's equation subject to

$$V(0, y) = V(1, y) = V(x, 0) = 0, \ V(x, 1) = 100 \tag{8.90}$$

The exact solution obtained by the method of separation of variables is given in Eq (2.31), namely,

$$V(x, y) = \frac{400}{\pi} \sum_{n=0}^{\infty} \frac{\sin k\pi x \sinh k\pi y}{k \sinh k\pi}, \quad k = 2n + 1 \tag{8.91}$$

Applying the fixed random MCM, the flowchart in Fig. 8.12 was developed. Based on the flowchart, the program of Fig. 8.13 was developed. A built-in standard subroutine RANDU in VAX 750 (also in VAX 780) was used to generate random numbers U uniformly distributed between 0 and 1. The step size Δ was selected as 0.05. The results of the potential computation are listed in Table 8.2 for three different locations. The average number of random steps $\overline{m}$ taken to reach the boundary is also shown. It is observed from Table 8.2 that it takes a large number of random steps for small step size and that the error in MCM results can be less than 1%.

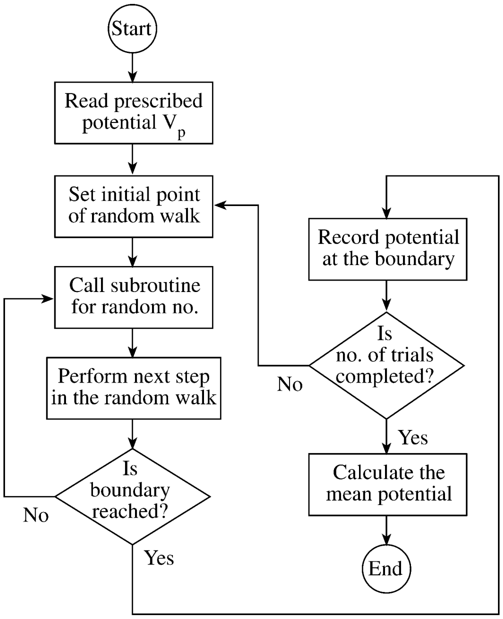

Start
Read prescribed potential V_p
Set initial point of random walk
Call subroutine for random no.
Perform next step in the random walk
Is boundary reached?
No
Yes
Record potential at the boundary
Is no. of trials completed?
No
Yes
Calculate the mean potential
End

Figure 8.12
Flowchart for random walk of Example 8.5.

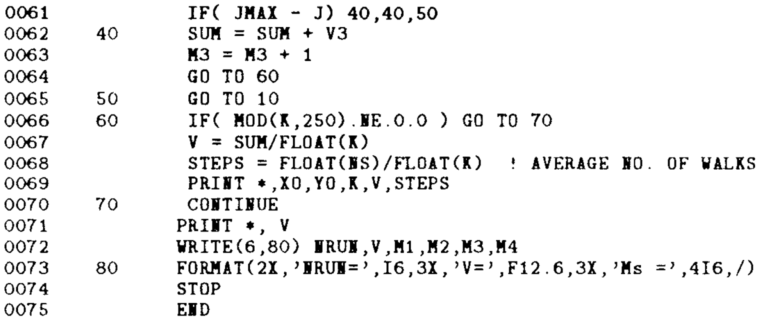
Rather than using Eq. (8.57), an alternative approach of determining $V(x, y)$ is to calculate the probability of a random walk terminating at a grid point located on the

```
C****************************************************
C  MONTE CARLO SOLUTION OF POTENTIAL PROBLEM
C  INVOLVING LAPLACE'S EQUATION
C***************************************************
C
C   SPECIFY INPUT PARAMETERS
C
       DATA  V1,V2,V3,V4/0.0,0.0,100.0,0.0/
       DATA  IS1,IS2/1234,5678/
       DATA  P1,P2,P3/0.25,0.5,0.75/
       DATA A,B/1.0,1.0/

       NRUN = 1000  ! NO. OF RUNS
       DELTA = 0.05   ! STEP SIZE
C
C  INITIALIZE VARIABLES
C
       X0 = 0.25
       Y0 = 0.75
       I0 = X0/DELTA
       J0 = Y0/DELTA
       IMAX = A/DELTA
       JMAX = B/DELTA
       SUM = 0.0
       NS = 0  ! NO. OF WALKS BEFORE REACHING BOUNDARY
       M1 = 0  ! NO. OF WALKS TERMINATING AT V1
       M2 = 0
       M3 = 0
       M4 = 0  ! NO. OF WALKS TERMINATING AT V4
C
C   START RUNNING MONTE CARLO SIMULATION
C
       DO 70 K=1,NRUN
       I=I0
       J=J0
10     CALL RANDU(IS1,IS2,R)
       NS = NS + 1
       IF( R.GE.0.0.AND. R.LT.P1)  I = I + 1
       IF( R.GE.P1. AND. R.LT.P2)  J = J + 1
       IF( R.GE.P2. AND. R.LT.P3)  I = I - 1
       IF( R.GE.P3)                J = J - 1
C
C   CHECK IF (I,J) IS ON THE BOUNDARY
C
        IF( I.EQ.0) THEN
        SUM = SUM + V4
        M4 = M4 +1
        GO TO 60
        ELSE
        ENDIF
        IF( IMAX - I) 20,20,30
20       SUM = SUM +  V2
         M2 = M2 +1
          GO TO 60
30      IF( J.EQ.0) THEN
        SUM = SUM + V1
        M1 = M1 + 1
        GO TO 60
        ELSE
        ENDIF
```

Figure 8.13

Program for Example 8.5 *(Continued)*.



```
           IF( JMAX - J) 40,40,50
   40      SUM = SUM + V3
           M3 = M3 + 1
           GO TO 60
   50      GO TO 10
   60      IF( MOD(K,250).NE.0.0 ) GO TO 70
           V = SUM/FLOAT(K)
           STEPS = FLOAT(NS)/FLOAT(K)   ! AVERAGE NO. OF WALKS
           PRINT *,X0,Y0,K,V,STEPS
   70      CONTINUE
          PRINT *, V
          WRITE(6,80) NRUN,V,M1,M2,M3,M4
   80     FORMAT(2X,'NRUN=',I6,3X,'V=',F12.6,3X,'Ms =',4I6,/)
          STOP
          END
```

Figure 8.13

(Cont.) **Program for Example 8.5.**

boundary. The information is easily extracted from the program used for obtaining the results in Table 8.2. To illustrate the validity of this approach, the potential at (0.25, 0.75) was calculated. For $N = 1000$ random walks, the number of walks terminating at $x = 0$, $x = 1$, $y = 0$ and $y = 1$ are 461, 62, 66, and 411, respectively. Hence, according to Eq. (8.79)

$$V(x, y) = \frac{461}{1000}(0) + \frac{62}{1000}(0) + \frac{66}{1000}(0) + \frac{411}{1000}(100) = 41.1 \tag{8.92}$$

The statistical error in the simulation can be found. In this case, the potential on the boundary takes values 0 or $V_o = 100$ so that $V(x, y)$ has a binomial distribution with mean $V(x, y)$ and variance

$$\sigma^2 = \frac{V(x, y)\,[V_o - V(x, y)]}{N} \tag{8.93}$$

At point (0.5, 0.5), for example, $N = 1000$ gives $\sigma = 1.384$ so that at 68% confidence interval, the error is $\delta = \sigma/\sqrt{N} = 0.04375$. ∎

Example 8.6

Use the floating random walk MCM to determine the potential at points (1.5, 0.5), (1.0, 1.5), and (1.5, 2.0) in the two-dimensional potential system in Fig. 8.14. □

Solution

To apply the floating random walk, we use the flowchart in Fig. 8.12 except that we apply Eq. (8.74) instead of Eq. (8.56) at every step in the random walk. A program based on the modified flowchart was developed. The shortest distance ρ from (x, y) to the boundary was found by dividing the solution region in Fig. 8.14 into three

Table 8.2 Results of Example 8.5

x	y	N	$\overline{m}$	Monte Carlo solution	Exact solution
0.25	0.75	250	66.20	42.80	43.20
		500	69.65	41.80	
		750	73.19	41.60	
		1000	73.95	41.10	
		1250	73.67	42.48	
		1500	73.39	42.48	
		1750	74.08	42.67	
		2000	74.54	43.35	
0.5	0.5	250	118.62	21.60	25.00
		500	120.00	23.60	
		750	120.27	25.89	
		1000	120.92	25.80	
		1250	120.92	25.92	
		1500	120.78	25.27	
		1750	121.50	25.26	
		2000	121.74	25.10	
0.75	0.25	250	64.82	7.60	6.797
		500	68.52	6.60	
		750	68.56	6.93	
		1000	70.17	7.50	
		1250	72.12	8.00	
		1500	71.78	7.60	
		1750	72.40	7.43	
		2000	72.40	7.30	

rectangles and checking

$$\text{if } \{(x, y) : 1 < x < 2, 0 < y < 1\}, \rho = \text{minimum}\{x - 1, 2 - x, y\}$$

$$\text{if } \{(x, y) : 0 < x < 1, 1 < y < 2.5\}, \rho = \text{minimum}\{x, y - 1, 2.5 - y\}$$

$$\text{if } \{(x, y) : 1 < x < 2, 1 < y < 2.5\} ,$$

$$\rho = \text{minimum}\left\{2 - x, 2.5 - y, \sqrt{(x - 1)^2 + (y - 1)^2}\right\}$$

A prescribed tolerance $\tau = 0.05$ was selected so that if the distance between a new point in the random walk and the boundary is less than τ, it is assumed that the boundary is reached and the potential at the closest boundary point is recorded.

Table 8.3 presents the Monte Carlo result with the average number of random steps $\overline{m}$. It should be observed that it takes fewer walks to reach the boundary in floating random walk than in fixed random walk. Since no analytic solution exists, we compare Monte Carlo results with those obtained using finite difference with $\Delta = 0.05$ and 500 iterations. As evident in Table 8.3, the Monte Carlo results agree well with the finite difference results even with 1000 walks. Also, by dividing the solution region

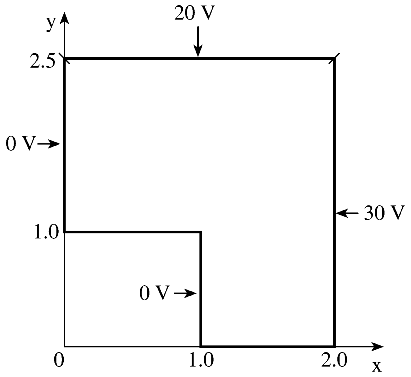


Figure 8.14
For Example 8.6.

into 32 elements, the finite element results [58] at points (1.5, 0.5), (1.0, 1.5), and (1.5, 2.0) are 11.265, 9.788, and 21.05 V, respectively.

Unlike the program in Fig. 8.13, where the error estimates are not provided for the sake of simplicity, the program in Fig. 8.15 incorporates evaluation of error estimates in the Monte Carlo calculations. Using Eq. (8.29), the error is calculated as

$$\delta = \frac{S t_{\alpha/2;n-1}}{\sqrt{n}}$$

In the program in Fig. 8.15, the number of trials n (the same of N in Section 8.3), with different seed values, is taken as 5 so that $t_{\alpha/2;n-1} = 2.776$. The sample variance S is calculated using Eq. (8.19). The values of δ are also listed in Table 8.3. Notice that unlike in Table 8.2, where $\overline{m}$ and V are the mean values after N walks, $\overline{m}$ and V in Table 8.3 are the mean values of n trials, each of which involves N walks, i.e., the “mean of the mean” values. Hence the results in Table 8.3 should be regarded as more accurate than those in Table 8.2. ∎

Example 8.7

Apply the Exodus method to solve the potential problem shown in Fig. 8.16. The potentials at $x = 0, x = a$, and $y = 0$ sides are zero while the potential at $y = b$ sides is V_o. Typically, let

$$V_o = 100, \quad \epsilon_1 = \epsilon_o, \quad \epsilon_2 = 2.25\epsilon_o, \quad a = 3.0,\ b = 2.0,\ c = 1.0 \qquad \square$$

Solution

The analytic solution to this problem using series expansion technique discussed in Section 2.7 is:

```
****************************************************
C  MONTE CARLO (FLOATING RANDOM WALK) SOLUTION OF
C  POTENTIAL PROBLEM  INVOLVING LAPLACE'S EQUATION
C  ERROR ESTIMATES ARE ALSO EVALUATED
C***************************************************

       DATA   PIE/3.1415/
       DIMENSION  T(10),V(10),STEPS(10)
       DATA ( T(I), I=2,10 )/12.706, 4.303, 3.182, 2.776,
     1              2.571,   2.447, 2.365, 2.306, 2.262/
               ! T(I) ARE THE  T-DISTRIBUTION PARAMETERS

C
C  INITIALIZE VARIABLES
C
       NTRIALS = 5 ! NO. OF TRIALS
       XO = 1.5  !  STARTING POINT, WHERE POTENTIAL IS
       YO = 0.5  !                      REQUIRED
       TOL = 0.005  ! TOLERANCE
C
C   START RUNNING MONTE CARLO SIMULATION
C
       DO 120 NRUN=250,2000,250   ! NO. OF RUNS
       DO 80 J=1,NTRIALS
       IS1 = 1000*FLOAT(J)
       IS2 = 2000*FLOAT(J)
       SUM = 0.0
       M = 0 ! NO. OF STEPS TAKEN TO REACH BOUNDARY
       DO 60 I=1,NRUN
       X = XO
       Y = YO
10     CALL RANDU(IS1,IS2,RN)
       PHI = 2.0*PIE*RN
C      FIND THE SHORTEST DISTANCE
       RC = SQRT( (X-1.0)**2  +  (Y-1.0)**2 )
       IF(Y.GT.1.0)  GO TO 20
       R = X - 1.0
       IF( R.GT.(2.0-X) ) R = 2.0 - X
       IF(R.GT.Y) R = Y
       GO TO 40
20     IF(X.GT.1.0) GO TO 30
       R = X
       IF( R.GT.(Y-1.0) ) R = Y - 1.0
       IF( R.GT.(2.5-Y) ) R = 2.5 - Y
       GO TO 40
30     R = 2.0 - X
       IF(R.GT.RC) R = RC
       IF( R.GT.(2.5-Y) ) R = 2.5 - Y
40     X = X + R*COS(PHI)
       Y = Y + R*SIN(PHI)
       M = M + 1
C
C   CHECK IF (X,Y) IS ON THE BOUNDARY
C
       IF( (X.LT.(1.0+TOL)).AND.(Y.LT.(1.0+TOL)) ) GO TO
                     ! FOR THE CORNER POINT
       IF( X.GE.(2.0-TOL) ) THEN
        SUM = SUM + 30.0
        GO TO 50
```

Figure 8.15

Applying floating random walk to solve the problem in Fig. 8.14; for Example 8.6 *(Continued)*.

```
        ELSE
       ENDIF
       IF( Y.GE.(2.5-TOL) ) THEN
        SUM = SUM + 20.0
        GO TO 50
       ELSE
       ENDIF
       IF(Y.GT.1.0. AND. X.LT.TOL) GO TO 50
       IF(Y.LT.1.0.AND.X.LT.(1.0-TOL)) GO TO 50
       IF(Y.LE.TOL.AND.X.GE.1.0) GO TO 50
       IF(Y.LE.(1.0+TOL).AND.X.LT.1.0) GO TO 50
       GO TO 10
50     CONTINUE
60     CONTINUE
        V(J) = SUM/FLOAT(NRUN)
        STEPS(J) = FLOAT(M)/FLOAT(NRUN)  ! AVERAGE NO. OF WALKS
         PRINT *,X0,Y0,V(J),STEPS(J)
         WRITE(6,*) X0,Y0,V(J),STEPS(J)
        WRITE(6,70) X0,Y0,NRUN,V(J),STEPS(J)
70     FORMAT(2X,'X = ',F5.2,3X,'Y=',F5.2,3X,
     1       'NRUN=',I6,3X,'V=',F12.6,3X,'STEPS=',F10.3,/)
80     CONTINUE
C
C   FIND THE MEAN VALUE OF V AND MEAN NO. OF STEP
C
       SUM = 0.0
       SUM1 = 0.0
       DO 90 I=1,NTRIALS
       SUM = SUM + V(I)
       SUM1 = SUM1 + STEPS(I)
90     CONTINUE
       VMEAN = SUM/FLOAT(NTRIALS)
       STEPM = SUM1/FLOAT(NTRIALS)
C
C  CALCULATE ERROR
C
       SUM = 0.0
       DO 100 I=1,NTRIALS
       SUM = SUM + ( V(I) - VMEAN )**2
100    CONTINUE
       STD = SQRT( SUM/FLOAT(NTRIALS-1) )
       ERROR = STD*T(NTRIALS)/SQRT( FLOAT(NTRIALS) )
       PRINT *,NTRIALS,VMEAN,STEPM,ERROR
       WRITE(6,110) NTRIALS,VMEAN,STEPM,ERROR
110    FORMAT(2X,'NO. OF TRIALS',I6,3X,'MEAN V =',F12.6,3X,
     1       'MEAN M = ',F12.6,3X,'ERROR=',F12.6,//)
120    CONTINUE
       STOP
       END
```

Figure 8.15

***(Cont.)* Applying floating random walk to solve the problem in Fig. 8.13; for Example 8.6.**

Table 8.3 Results of Example 8.6

x	y	N	$\overline{m}$	Monte Carlo solution ($V \pm \delta$)	Finite Difference solution (V)
1.5	0.5	250	6.738	11.52 ± 0.8973	11.44
		500	6.668	11.80 ± 0.9378	
		750	6.535	11.83 ± 0.4092	
		1000	6.476	11.82 ± 0.6205	
		1250	6.483	11.85 ± 0.6683	
		1500	6.465	11.72 ± 0.7973	
		1750	6.468	11.70 ± 0.6894	
		2000	6.460	11.55 ± 0.5956	
1.0	1.5	250	8.902	10.74 ± 0.8365	10.44
		500	8.984	10.82 ± 0.3709	
		750	8.937	10.75 ± 0.5032	
		1000	8.928	10.90 ± 0.7231	
		1250	8.836	10.84 ± 0.7255	
		1500	8.791	10.93 ± 0.5983	
		1750	8.788	10.87 ± 0.4803	
		2000	8.811	10.84 ± 0.3646	
1.5	2.0	250	7.242	21.66 ± 0.7509	21.07
		500	7.293	21.57 ± 0.5162	
		750	7.278	21.53 ± 0.3505	
		1000	7.316	21.53 ± 0.2601	
		1250	7.322	21.53 ± 0.3298	
		1500	7.348	21.51 ± 0.3083	
		1750	7.372	21.55 ± 0.2592	
		2000	7.371	21.45 ± 0.2521	

$$V = \begin{cases} \displaystyle\sum_{k=1}^{\infty} \sin\beta x \left[a_n \sinh\beta y + b_n \cosh\beta y\right], & 0 \leq y \leq c \\ \displaystyle\sum_{k=1}^{\infty} c_n \sin\beta x \sinh\beta y, & c \leq y \leq b \end{cases} \tag{24}$$

where

$$\beta = \frac{n\pi}{a}, \quad n = 2k - 1$$

$$\begin{aligned} a_n &= 4V_o \left[\epsilon_1 \tanh\beta c - \epsilon_2 \coth\beta c\right]/d_n\,, \\ b_n &= 4V_o \left(\epsilon_2 - \epsilon_1\right)/d_n\,, \\ c_n &= 4V_o \left[\epsilon_1 \tanh\beta c - \epsilon_2 \coth\beta c + \left(\epsilon_2 - \epsilon_1\right)\coth\beta c\right]/d_n\,, \\ d_n &= n\pi \sinh\beta b \left[\epsilon_1 \tanh\beta c - \epsilon_2 \coth\beta c + \left(\epsilon_2 - \epsilon_1\right)\coth\beta b\right] \end{aligned} \tag{25}$$

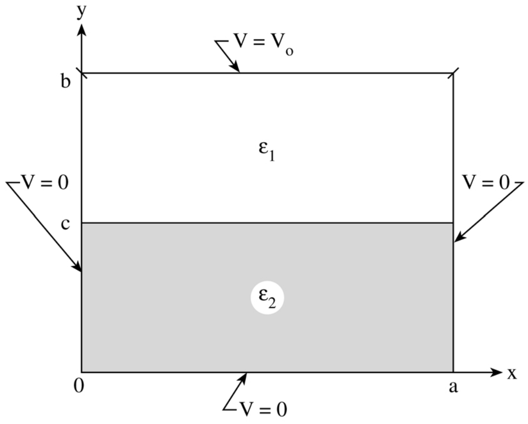


Figure 8.16
Potential system for Example 8.7.

The potentials were calculated at five typical points using the Exodus method, the fixed random walk Monte Carlo method, and the analytic solution. The number of particles, N, was taken as 10^7 for the Exodus method and the step size $\Delta = 0.05$ was used. For the fixed random walk method, $\Delta = 0.05$ and 2000 walks were used. It was noted that 2000 walks were sufficient for the random walk solutions to converge. The results are displayed in Table 8.4. In the table, δ is the error estimate, which is obtained by repeating each calculation five times and using statistical formulas provided in [13]. It should be noted from the table that the results of the Exodus method agree to four significant places with the exact solution. Thus the Exodus method is more accurate than the random walk technique. It should also be noted that the Exodus method does not require the use of a random number routine and also the need of calculating the error estimate. The Exodus method, therefore, takes less computation time than the random walk method. ▮

Table 8.4 Results of Example 8.7

x	y	Exodus Method V	Fixed Random Walk ($V \pm \delta$)	Finite Difference V	Exact Solution V
0.5	1.0	13.41	13.40 ± 1.113	13.16	13.41
1.0	1.0	21.13	20.85 ± 1.612	20.74	21.13
1.5	1.0	23.43	23.58 ± 1.2129	22.99	23.43
1.5	0.5	10.52	10.13 ± 0.8789	10.21	10.52
1.5	1.5	59.36	58.89 ± 2.1382	59.06	59.34

8.6 Regional Monte Carlo Methods

A major limitation inherent with the standard Monte Carlo methods discussed above is that they only permit single point calculations. In view of this limitation, several techniques have been proposed for using Monte Carlo for whole field computation. The popular ones are the *shrinking boundary method* [37] and *inscribed figure method* [38].

The shrinking boundary method is similar to the regular fixed random walk except that once the potential at an interior point is calculated, that point is treated as a boundary or absorbing point. This way, the random walking particles will have more points to terminate their walks and the walking time is reduced.

The inscribed figure method is based on the concept of subregion calculation. It involves dividing the solution region into standard shapes or inscribing standard shapes into the region. (By standard shapes is meant circles, squares, triangles, rectangles, etc. for which Green's function can be obtained analytically or numerically.) Then, a Monte Carlo method is used in computing potential along the dividing lines between the shapes and the regions that have nonstandard shapes. Standard analytical methods are used to compute the potential in the subregions.

Both the shrinking boundary method and the inscribed figure method do not make Monte Carlo methods efficient for whole field calculation. They still require point-by-point calculations and a number large of tests as standard Monte Carlo techniques. Therefore, they offer no significant advantage over the standard Monte Carlo methods. Using Markov chains for whole field calculations has been found to be more efficient than the shrinking boundary method and the inscribed figure method. The technique basically calculates the transition probabilities using absorbing Markov chains [55, 56].

A Markov chain is a sequence of random variables $X^{(0)}, X^{(1)}, \ldots,$ where the probability distribution for $X^{(n)}$ is determined entirely by the probability distribution of $X^{(n-1)}$. A Markov process is a type of random process that is characterized by the memoryless property [57]–[60]. It is a process evolving in time that remembers only the most recent past and whose conditional distributions are time invariant. Markov chains are mathematical models of this kind of process. The Markov chain of interest to us are *discrete-state, discrete-time* Markov chains. In our case, the Markov chain is the random walk and the states are the grid nodes. The transition probability P_{ij} is the probability that a random-walking particle at node i moves to node j. It is expressed by the Markov property

$$\begin{aligned} P_{ij} &= P\left(x_{n+1} = j | x_o, x_1, \ldots, x_n\right) \\ &= P\left(x_{n+1} = j | x_n\right) \qquad j \in \mathbf{X},\ n = 0, 1, 2, \ldots \end{aligned} \tag{8.94}$$

The Markov chain is characterized by its transition probability matrix **P**, defined by

$$\mathbf{P} = \begin{bmatrix} P_{00} & P_{01} & P_{02} & \cdots \\ P_{10} & P_{11} & P_{12} & \cdots \\ P_{20} & P_{21} & P_{22} & \cdots \\ \vdots & \vdots & \vdots & \ldots \end{bmatrix} \tag{8.95}$$

P is a stochastic matrix, meaning that the sum of the elements in each row is unity, i.e.,

$$\sum_{j \in \mathbf{X}} P_{ij} = 1 \qquad i \in \mathbf{X} \tag{8.96}$$

We may also use the state transition diagram as a way of representing the evolution of a Markov chain. An example is shown in Fig. 8.17 for a three-state Markov chain.

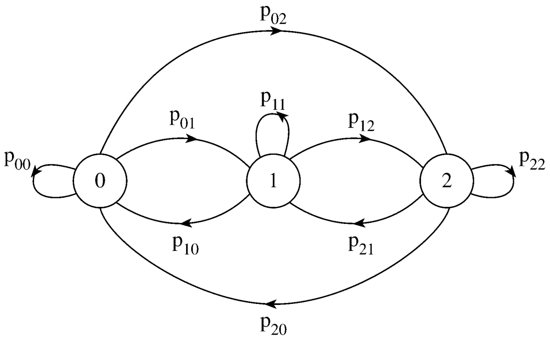


Figure 8.17
State Transition diagram for a three-state Markov Chain.

If we assume that there are n_f free (or nonabsorbing) nodes and n_p fixed (prescribed or absorbing) nodes, the size of the transition matrix **P** is n, where

$$n = n_f + n_p \tag{8.97}$$

If the absorbing nodes are numbered first and the nonabsorbing states are numbered last, the $n \times n$ transition matrix becomes

$$\mathbf{P} = \begin{bmatrix} \mathbf{I} & 0 \\ \mathbf{R} & \mathbf{Q} \end{bmatrix} \tag{8.98}$$

where the $n_f \times n_p$ matrix **R** represents the probabilities of moving from nonabsorbing nodes to absorbing ones; the $n_f \times n_f$ matrix **Q** represents the probabilities of moving from one nonabsorbing node to another; **I** is the identity matrix representing transitions between the absorbing nodes ($P_{ii} = 1$ and $P_{ij} = 0$); and 0 is the null matrix showing that there are no transitions from absorbing to nonabsorbing nodes.

For the solution of Laplace's equation, we obtain the elements of **Q** from Eq. (8.55b) as

$$Q_{ij} = \begin{cases} \dfrac{1}{4}, & \text{if } i \text{ is directly connected to } j, \\ 0, & \text{if } i = j \text{ or } i \text{ is not directly connected to } j \end{cases} \tag{8.99}$$

The same applies to R_{ij} except that j is an absorbing node.

For any absorbing Markov chain, **I** − **Q** has an inverse. This is usually referred as the fundamental matrix

$$\mathbf{N} = (\mathbf{I} - \mathbf{Q})^{-1} \tag{8.100}$$

where N_{ij} is the average number of times the random-walking particle starting from node i passes through node j before being absorbed. The absorption probability matrix **B** is

$$\mathbf{B} = \mathbf{NR} \tag{8.101}$$

where B_{ij} is the probability that a random-walking particle originating from a non-absorbing node i will end up at the absorbing node j. **B** is an $n_f \times n_p$ matrix and is stochastic like the transition probability matrix, i.e.,

$$\sum_{j=1}^{n_p} B_{ij} = 1, \qquad i = 1, 2, \ldots, n_f \tag{8.102}$$

If $\mathbf{V}_f$ and $\mathbf{V}_p$ contain potentials at the free and fixed nodes, respectively, then

$$\boxed{\mathbf{V}_f = \mathbf{B}\mathbf{V}_p} \tag{8.103}$$

In terms of the prescribed potentials $V_1, V_2, \ldots, V_{n_p}$, Eq. (8.103) becomes

$$V_i = \sum_{j=1}^{n_p} B_{ij} V_j, \qquad i = 1, 2, \ldots, n_f \tag{8.104}$$

where V_i the potential at any free node i. Unlike Eq. (8.57), Eq. (8.103) or (8.104) provides the solution at all the free nodes at once.

An alternative way to obtain the solution in Eq. (8.103) is to exploit a property of the transition probability matrix **P**. When **P** is multiplied by itself repeatedly for a large number of times, we obtain

$$\lim_{n \to \infty} \mathbf{P}^n = \begin{bmatrix} \mathbf{I} & 0 \\ \mathbf{B} & 0 \end{bmatrix} \tag{8.105}$$

Thus

$$\begin{bmatrix} \mathbf{V}_p \\ \mathbf{V}_f \end{bmatrix} = \mathbf{P}^n \begin{bmatrix} \mathbf{V}_p \\ \mathbf{V}_f \end{bmatrix} = \begin{bmatrix} \mathbf{I} & 0 \\ \mathbf{B} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V}_p \\ \mathbf{V}_f \end{bmatrix} \tag{8.106}$$

Either Eq. (8.103) or (8.106) can be used to find $\mathbf{V}_f$ but it is evident that Eq. (8.103) will be more efficient and accurate. From Eq. (8.103) or (8.104), it should be noticed that if $\mathbf{N}$ is calculated accurately, the solution is "exact."

There are several other procedures for whole field computation [37, 38], [61]–[64]. One technique involves using Green's function in the floating random walk [42].

The random walk MCMs and the Markov chain MCM applied to elliptic PDEs in this chapter can be applied to parabolic PDEs as well [65, 66].

The following two examples will corroborate Markov chain Monte Carlo method. The first example requires no computer programming and can be done by hand, while the second one needs computer programming.

Example 8.8

Rework Example 8.5 using Markov chain. The problem is shown in Fig. 8.11. We wish to determine the potential at points $(a/3, a/3)$, $(a/3, 2a/3)$, $(2a/3, a/3)$, and $(2a/3, 2a/3)$. Although we may assume that $a = 1$, that is not necessary. □

Solution

In this case, there are four free nodes ($n_f = 4$) and eight fixed nodes ($n_p = 8$) as shown in Fig. 8.18. The transition probability matrix is obtained by inspection as

$$\mathbf{P} = \begin{array}{c} \\ 1\\2\\3\\4\\5\\6\\7\\8\\9\\10\\11\\12 \end{array}
\begin{array}{c} \begin{array}{cccccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \end{array} \\
\left[\begin{array}{cccccccccccc}
1 & & & & & & & & & & & \\
 & 1 & & & & & & & & & & \\
 & & 1 & & & & & & & & & \\
 & & & 1 & & & & & & & & \\
 & & & & 1 & & & & & & & \\
 & & & & & 1 & & & & & & \\
 & & & & & & 1 & & & & & \\
 & & & & & & & 1 & & & & \\
\frac{1}{4} & & & & & & & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 \\
0 & \frac{1}{4} & \frac{1}{4} & & & & & & \frac{1}{4} & 0 & 0 & \frac{1}{4} \\
0 & 0 & 0 & & & \frac{1}{4} & \frac{1}{4} & & \frac{1}{4} & & & \frac{1}{4} \\
0 & & & \frac{1}{4} & \frac{1}{4} & & 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & 0
\end{array}\right] \end{array}$$

Other entries in $\mathbf{P}$ shown vacant are zeros.

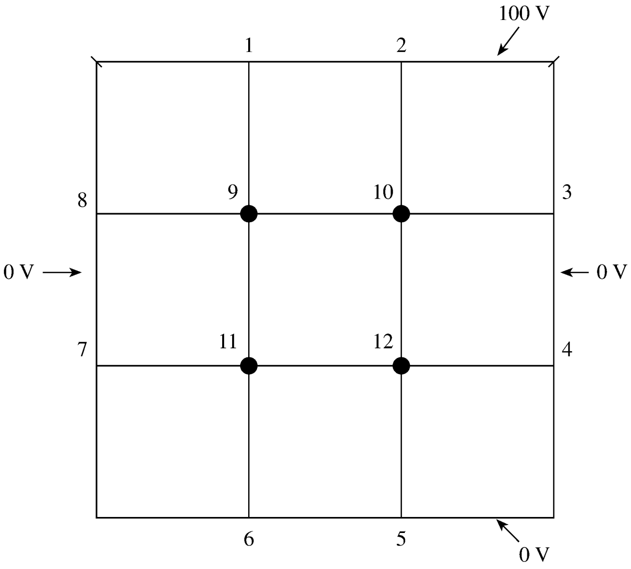


Figure 8.18
For Example 8.8.

From **P**, we obtain

$$\mathbf{R} = \begin{array}{c} \\ 9 \\ 10 \\ 11 \\ 12 \end{array} \begin{array}{c} \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \begin{bmatrix} \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{4} \\ 0 & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$$\mathbf{Q} = \begin{array}{c} \\ 9 \\ 10 \\ 11 \\ 12 \end{array} \begin{array}{c} \begin{array}{cccc} 9 & 10 & 11 & 12 \end{array} \\ \begin{bmatrix} 0 & \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & 0 & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & 0 & \frac{1}{4} \\ 0 & \frac{1}{4} & \frac{1}{4} & 0 \end{bmatrix} \end{array}$$

The fundamental matrix **N** is obtained as

$$\mathbf{N} = (\mathbf{I} - \mathbf{Q})^{-1} = \begin{bmatrix} 1 & -\frac{1}{4} & -\frac{1}{4} & 0 \\ -\frac{1}{4} & 1 & 0 & -\frac{1}{4} \\ -\frac{1}{4} & 0 & 1 & -\frac{1}{4} \\ 0 & -\frac{1}{4} & -\frac{1}{4} & 1 \end{bmatrix}^{-1}$$

or

$$\mathbf{N} = \frac{1}{6}\begin{bmatrix} 7 & 2 & 2 & 1 \\ 2 & 7 & 1 & 2 \\ 2 & 1 & 7 & 2 \\ 1 & 2 & 2 & 7 \end{bmatrix}$$

The absorption probability matrix **B** is obtained as

$$\mathbf{B} = \mathbf{NR} = \begin{array}{c} \\ 9 \\ 10 \\ 11 \\ 12 \end{array} \begin{array}{c} \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \begin{bmatrix} \frac{7}{24} & \frac{1}{12} & \frac{1}{12} & \frac{1}{24} & \frac{1}{24} & \frac{1}{12} & \frac{1}{12} & \frac{7}{24} \\ \frac{1}{12} & \frac{7}{24} & \frac{7}{24} & \frac{1}{12} & \frac{1}{12} & \frac{1}{24} & \frac{1}{24} & \frac{1}{12} \\ \frac{1}{12} & \frac{1}{24} & \frac{1}{24} & \frac{1}{12} & \frac{1}{12} & \frac{7}{24} & \frac{7}{24} & \frac{1}{12} \\ \frac{1}{24} & \frac{1}{12} & \frac{1}{12} & \frac{7}{24} & \frac{7}{24} & \frac{1}{12} & \frac{1}{12} & \frac{1}{24} \end{bmatrix} \end{array}$$

Notice that Eq. (8.102) is satisfied. We now use Eq. (8.104) to obtain the potentials at he free nodes. For example,

$$V_9 = \frac{7}{24}V_1 + \frac{1}{12}V_2 + \frac{1}{12}V_3 + \frac{1}{24}V_4 + \frac{1}{24}V_5 + \frac{1}{12}V_6 + \frac{1}{12}V_7 + \frac{7}{24}V_8$$

Since $V_1 = V_2 = 100$ while $V_3 = V_4 = \cdots = V_8 = 0$,

$$V_9 = \left(\frac{7}{24} + \frac{1}{12}\right)100 = 37.5$$

By symmetry, $V_{10} = V_9 = 37.5$. Similarly,

$$V_{11} = V_{12} = \left(\frac{1}{24} + \frac{1}{12}\right)100 = 12.5$$

Table 8.5 compares these results with the finite difference solution (with 10 iterations) and the exact solution using Eq. (2.31b) or (8.91). It is evident that the Markov chain solution compares well. ∎

Table 8.5 Results of Example 8.8

Node	Finite Difference Solution	Markov Chain Solution	Exact Solution
9	37.499	37.5	38.074
10	37.499	37.5	38.074
11	12.499	12.5	11.926
12	12.499	12.5	11.926

Example 8.9

Consider the potential problem shown in Fig. 8.19. Let

$$V_o = 100, \quad \epsilon_1 = \epsilon_o, \quad \epsilon_2 = 3\epsilon_o$$
$$a = b = 0.5, \quad h = w = 1.0 \qquad \square$$

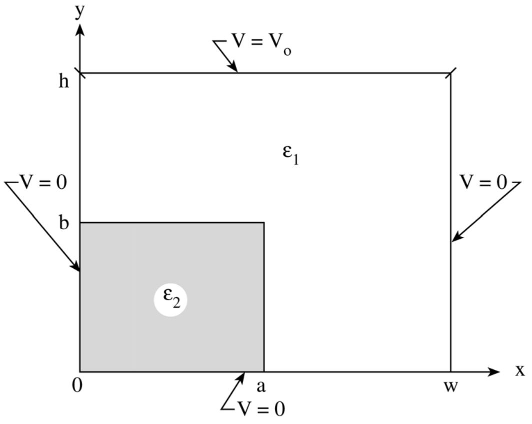


Figure 8.19
Potential system for Example 8.9.

Solution

The Markov chain solution was implemented using MATLAB. The approach involved writing code that generated the transition probability matrices using the random walk probabilities, computing the appropriate inverse, and manipulating the solution matrix. The use of MATLAB significantly reduced the programming complexity by the way the software internally handles matrices. The Q–matrix was selected as a timing index since the absorbing Markov chain algorithm involves inverting it. In this example, the Q–matrix is 361×361 and the running time was 90 and 34 seconds on 486DX2 and Pentium, respectively. $\Delta = 0.05$ was assumed. At the corner point $(x, y) = (a, b)$, the random walk probabilities are

$$p_{x+} = p_{y+} = \frac{\epsilon_1}{3\epsilon_1 + \epsilon_2}, \quad p_{x-} = p_{y-} = \frac{\epsilon_1 + \epsilon_2}{2\,(3\epsilon_1 + \epsilon_2)}$$

The plot of the potential distribution is portrayed in Fig. 8.20. Since the problem has no exact solution, the results at five typical points are compared with those from the Exodus method and finite difference in Table 8.6. It should be observed that the Markov chain approach provides a solution that is close to that by the Exodus method. ■

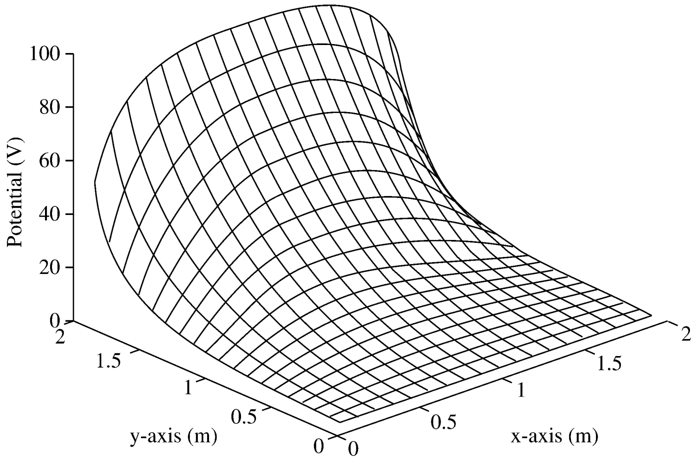


Figure 8.20
Potential distribution obtained by Markov chains; for Example 8.9.

Table 8.6 Results of Example 8.9

Node		Markov Chain	Exodus Method	Finite Difference
x	y			
0.25	0.5	10.2688	10.269	10.166
0.5	0.5	16.6667	16.667	16.576
0.75	0.5	15.9311	15.931	15.887
0.5	0.75	51.0987	51.931	50.928
0.5	0.25	6.2163	6.2163	6.1772

8.7 Concluding Remarks

The Monte Carlo technique is essentially a means of estimating expected values and hence is a form of numerical quadrature. Although the technique can be applied to simple processes and estimating multidimensional integrals, the power of the technique rests in the fact that [66]:

- it is often more efficient than other quadrature formulas for estimating multi-dimensional integrals,
- it is adaptable in the sense that variance reduction techniques can be tailored to the specific problem, and
- it can be applied to highly complex problems for which the definite integral formulation is not obvious and standard analytic techniques are ineffective.

For rigorous mathematical justification for the methods employed in Monte Carlo simulations, one is urged to read [32, 67]. As is typical with current MCMs, other numerical methods of solutions appear to be preferable when they may be used. Monte Carlo techniques often yield numerical answers of limited accuracy and are therefore employed as a last resort. However, there are problems for which the solution is not feasible using other methods. Problems that are probabilistic and continuous in nature (e.g., neutron absorption, charge transport in semiconductors, and scattering of waves by random media) are ideally suited to these methods and represent the most logical and efficient use of the stochastic methods. Since the recent appearance of vector machines, the importance of the Monte Carlo methods is growing.

It should be emphasized that in any Monte Carlo simulation, it is important to indicate the degree of confidence of the estimates or insert error bars in graphs illustrating Monte Carlo estimates. Without such information, Monte Carlo results are of questionable significance.

Applications of MCMs to other branches of science and engineering are summarized in [14, 15, 25, 68]. EM-related problems, besides those covered in this chapter, to which Monte Carlo procedures have been applied include:

- diffusion problems [62, 64, 70]
- strip transmission lines [40]
- random periodic arrays [71]
- waveguide structures [72]–[77]
- scattering of waves by random media [78]–[84]
- noise in magnetic recording [85, 86]
- induced currents in biological bodies [87].

We conclude this chapter by referring to two new Monte Carlo methods. One new MCM, known as the equilateral triangular mesh fixed random walk, has been proposed to handle Neumann problems [88]. Another new MCM, known as Neuro-Monte Carlo solution, is an attempt at whole field computation [89]. It combines an artificial neural network and a Monte Carlo method as a training data source. For further exposition on Monte Carlo techniques, one should consult [25, 26, 61, 90, 91].

References

[1] R. Hersch and R.J. Griego, "Brownian motion and potential theory," *Sci. Amer.,* Mar. 1969, pp. 67–74.

[2] T.F. Irvine and J.P. Hartnett (eds.), *Advances in Heat Transfer.* New York: Academic Press, 1968.

[3] H.A. Meyer (ed.), *Symposium on Monte Carlo Methods.* New York: John Wiley, 1956.

[4] D.D. McCracken, "The Monte Carlo method," *Sci. Amer.,* vol. 192, May 1955, pp. 90–96.

[5] T.E. Hull and A.R. Dobell, "Random number generators," *SIAM Review,* vol. 4, no. 3, July 1982, pp. 230–254.

[6] D.E. Knuth, *The Art of Computer Programming,* vol. 2. Reading, MA: Addison-Wesley, 1969, pp. 9, 10, 78, 155.

[7] J. Banks and J. Carson, *Discrete Event System Simulation.* Englewood Cliffs, NJ: Prentice-Hall, 1984, pp. 257–288.

[8] P.A.W. Lewis, et al., " A pseudo-random number generator for the system/360," *IBM System Jour.,* vol. 8, no. 2, 1969, pp. 136–146.

[9] S.S. Kuo, *Computer Applications of Numerical Methods.* Reading, MA: Addison-Wesley, 1972, pp. 327–345.

[10] A.M. Law and W.D. Kelton, *Simulation Modeling and Analysis.* New York: McGraw-Hill, 1982, pp. 219–228.

[11] D.E. Raeside, "An introduction to Monte Carlo methods," *Amer. Jour. Phys.,* vol. 42, Jan. 1974, pp. 20–26.

[12] C. Jacoboni and L. Reggiani, "The Monte Carlo method for the solution of charge transport in semiconductors with applications to covalent materials," *Rev. Mod. Phys.,* vol. 55, no. 3, July 1983, pp. 645–705.

[13] H. Kobayashi, *Modeling and Analysis: An Introduction to System Performance Evaluation Methodology.* Reading, MA: Addison-Wesley, 1978, pp. 221–247.

[14] I.M. Sobol, *The Monte Carlo Method.* Chicago: University of Chicago Press, 1974, pp. 24–30.

[15] Y.A. Shreider, *Method of Statistical Testing (Monte Carlo Method),* Amsterdam: Elsevier, 1964, pp. 39–83. Another translation of the same Russian text: Y.A. Shreider, *The Monte Carlo Method (The Method of Statistical Trials).* Oxford: Pergamon, 1966.

[16] E.E. Lewis and W.F. Miller, *Computational Methods of Neutron Transport.* New York: John Wiley, 1984, pp. 296–360.

[17] I.S. Sokolinkoff and R.M. Redheffer, *Mathematics of Physics and Modern Engineering.* New York: McGraw-Hill, 1958, pp. 644–649.

[18] M.H. Merel and F.J. Mullin, "Analytic Monte Carlo error analysis," *J. Spacecraft,* vol. 5, no. 11, Nov. 1968, pp. 1304–1308.

[19] A.J. Chorin, "Hermite expansions in Monte-Carlo computation," *J. Comp. Phys.,* vol. 8, 1971, pp. 472–482.

[20] R.Y. Rubinstein, *Simulation and the Monte Carlo Method.* New York: John Wiley, 1981, pp. 20–90.

[21] W.J. Graybeal and U.W. Pooch, *Simulation: Principles and Methods.* Cambridge, MA: Winthrop Pub., 1980, pp. 77–97.

[22] B.J.T. Morgan, *Elements of Simulation.* London: Chapman & Hall, 1984, pp. 77–81.

[23] C.W. Alexion, et al., "Evaluation of radiation fields using statistical methods of integration," *IEEE Trans. Ant. Prog.,* vol. AP-26, no. 2, Mar. 1979, pp. 288–293.

[24] R.C. Millikan, "The magic of the Monte Carlo method," *BYTE,* vol. 8, Feb. 1988, pp. 371–373.

[25] M.H. Kalos and P.A. Whitlook, *Monte Carlo Methods.* New York: John Wiley, 1986, vol. 1, pp. 89–116.

[26] J.M. Hammersley and D.C. Handscomb, *Monte Carlo Methods.* London: Methuen, 1964.

[27] S. Haber, "A modified Monte-Carlo quadrature II," *Math. Comp.,* vol. 21, July 1967, pp. 388–397.

[28] S. Haber, "Numerical evaluation of multiple integrals," *SIAM Rev.,* vol. 12, no. 4, Oct. 1970, pp. 481–527.

[29] J.M. Hammersley and K.W. Morton, "A new Monte Carlo technique: antithetic variates," *Proc. Camb. Phil. Soc.,* vol. 52, 1955, pp. 449–475.

[30] J.H. Halton and D.C. Handscom, "A method for increasing the efficiency of Monte Carlo integration," *J. ACM,* vol. 4, 1957, pp. 329–340.

[31] S.J. Yakowitz, *Computational Probability and Simulation.* Reading, MA: Addison-Wesley, 1977, pp. 192, 193.

[32] S. Kakutani, "Two-dimensional Brownian motion harmonic functions," *Proc. Imp. Acad.,* (Tokyo), vol. 20, 1944, pp. 706–714.

[33] A. Haji-Sheikh and E.M. Sparrow, "The floating random walk and its application to Monte Carlo solutions of heat equations," *J. SIAM Appl. Math.,* vol. 14, no 2, Mar. 1966, pp. 370–389.

[34] A. Haji-Sheikh and E.M. Sparrow, "The solution of heat conduction problems by probability methods," *J. Heat Transfer,* Trans. ASME, Series C, vol. 89, no. 2, May 1967, pp. 121–131.

[35] G.E. Zinsmeiter, "Monte Carlo methods as an aid in teaching heat conduction," *Bull. Mech. Engr. Educ.,* vol. 7, 1968, pp. 77–86.

[36] R. Chandler, et al., "The solution of steady state convection problems by the fixed random walk method," *J. Heat Transfer,* Trans. ASME, Series C, vol. 90, Aug. 1968, pp. 361–363.

[37] G.E. Zinsmeiter and S.S. Pan, "A method for improving the efficiency of Monte Carlo calculation of heat conduction problems," *J. Heat Transfer,* Trans. ASME, Series C, vol. 96, 1974, pp. 246–248.

[38] G.E. Zinsmeiter and S.S. Pan, "A modification of the Monte Carlo method," *Inter. J. Num. Meth. Engr.,* vol. 10, 1976, pp. 1057–1064.

[39] G.M. Royer, "A Monte Carlo procedure for potential theory of problems," *IEEE Trans. Micro. Theo. Tech.,* vol. MTT-19, no. 10, Oct. 1971, pp. 813–818.

[40] R.M. Bevensee, "Probabilistic potential theory applied to electrical engineering problems," *Proc. IEEE,* vol. 61, no. 4, April 1973, pp. 423–437.

[41] R.L. Gibbs and J.D. Beason, "Solutions to boundary value problems of the potential type by random walk method," *Am. J. Phys.,* vol. 43, no. 9, Sept. 1975, pp. 782–785.

[42] J.H. Pickles, "Monte Carlo field calculations," *Proc. IEEE,* vol. 124, no. 12, Dec. 1977, pp. 1271–1276.

[43] F. Sanchez-Quesada, et al., "Monte-Carlo method for discrete inhomogeneous problems," *Proc. IEEE,* no. 125, no. 12, Dec. 1978, pp. 1400–1402.

[44] R. Schlott, "A Monte Carlo method for the Dirichlet problem of dielectric wedges," *IEEE Trans. Micro. Theo. Tech.,* vol. 36, no. 4, April 1988, pp. 724–730.

[45] M.N.O. Sadiku, "Monte Carlo methods in an introductory electromagnetic class," *IEEE Trans. Educ.,* vol. 33, no. 1, Feb. 1990, pp. 73–80.

[46] M.D.R. Beasley, et al., "Comparative study of three methods for computing electric fields," *Proc. IEEE,* vol. 126, no. 1, Jan. 1979, pp. 126–134.

[47] J.R. Currie, et al., "Monte Carlo determination of the frequency of lightning strokes and shielding failures on transmission lines," *IEEE Trans. Power Appl. Syst.,* vol. PAS-90, 1971, pp. 2305–2310.

[48] R.S. Velazquez, et al., "Probabilistic calculations of lightning protection for tall buildings," *IEEE Trans. Indust. Appl.,* vol. IA-18, no. 3, May/June 1982, pp. 252–259.

[49] M.N.O. Sadiku, "Monte Carlo Solution of Axisymmetric Potential Problems," *IEEE Trans. on Industry Applications,* vol. 29, no. 6, 1993, Nov./Dec. pp. 1042–1046.

[50] J.N. Jere and Y.L.L. Coz, "An Improved Floating-random-walk Algorithm for Solving Multi-dielectric Dirichlet Problem," *IEEE Trans. Micro. Theo. & Tech.,* vol. 41, no. 2, Feb. 1993, pp. 252–329.

[51] A.F. Emery and W.W. Carson, “A modification to the Monte Carlo method—the Exodus method,” *J. Heat Transfer,* Trans. ASME, Series C, vol. 90, 1968, pp. 328–332.

[52] M.N.O. Sadiku and D. Hunt, “Solution of Dirichlet Problems by the Exodus Method,” *IEEE Trans. Microwave Theory and Techniques,* vol. 40, no. 1, Jan. 1992, pp. 89–95.

[53] M.N.O. Sadiku, S.O. Ajose, and Zhibao Fu, “Appling the Exodus Method to Solve Poisson’s Equation,” *IEEE Trans. Microwave Theory and Techniques,* vol. 42, no. 4, April 1994, pp. 661–666.

[54] W.H. McCrea and F.J.W. Whipple, “Random paths in two and three dimensions,” *Proc. Roy. Soc. Edinb.,* vol. 60, 1940, pp. 281–298.

[55] Fusco, V.F. and Linden, P.A., “A Markov Chain Approach for Static Field Analysis,” *Microwave and Optical Technology Letters,* vol. 1, no. 6, Aug. 1988, pp. 216–220.

[56] M.N.O. Sadiku and R. Garcia, “Whole field computation using Monte Carlo Method,” *Inter. Jour. Num. Model.,* vol. 10, 1997, pp. 303–312.

[57] M.E. Woodward, *Communication and Computer Networks.* Los Alamitos, CA: IEEE Computer Society Press, 1994, pp. 53–57.

[58] J.G. Kemeny and J.L. Snell, *Finite Markov Chains.* New York: Spring-Verlag, 1976, pp. 43–68.

[59] M. Iosifescu, *Finite Markov Processes and Their Applications.* New York: John Wiley & Sons, 1980, pp. 45, 99–106.

[60] G.J. Anders, *Probability Concepts in Electric Power Systems.* New York: John Wiley & Sons, 1990, pp. 160–170.

[61] T.J. Hoffman and N.E. Banks, “Monte Carlo surface density solution to the Dirichlet heat transfer problem,” *Nucl. Sci. Engr.,* vol. 59, 1976, pp. 205–214.

[62] T.J. Hoffman and N.E. Banks, “Monte Carlo solution to the Dirichlet problem with the double-layer potential density,” *Trans. Amer. Nucl. Sci.,* vol. 18, 1974, pp. 136, 137. See also vol. 19, 1974, p. 164; vol. 24, 1976, p. 181.

[63] T.E. Booth, “Exact Monte Carlo solution of elliptic partial differential equations,” *J. Comp. Phys.,* vol. 39, 1981, pp. 396–404.

[64] T.E. Booth, “Regional Monte Carlo solution of elliptic partial differential equations,” *J. Comp. Phys.,* vol. 47, 1982, pp. 281–290.

[65] A.F. Ghoniem, “Grid-free simulation of diffusion using random walk methods,” *J. Comp. Phys.,* vol. 61, 1985, pp. 1–37.

[66] E.S. Troubetzkoy and N.E. Banks, “Solution of the heat diffusion equation by Monte Carlo,” *Trans. Amer. Nucl. Soc.,* vol. 19, 1974, pp. 163, 164.

[67] A.W. Knapp, "Connection between Brownian motion and potential theory," *J. Math. Analy. Appl.,* vol. 12, 1965, pp. 328–349.

[68] J.H. Halton, "A retrospective and prospective survey of the Monte Carlo method," *SIAM Rev.,* vol. 12, no. 1, Jan. 1970, pp. 1–61.

[69] A.T. Bharucha-Reid (ed.), *Probabilistic Methods in Applied Mathematics.* New York: Academic Press, vol. 1, 1968; vol. 2, 1970; vol. 3, 1973.

[70] G.W. King, "Monte Carlo method for solving diffusion problems," *Ind. Engr. Chem.,* vol. 43, no. 11, Nov. 1951, pp. 2475–2478.

[71] Y.T. Lo, "Random periodic arrays," *Rad. Sci.,* vol. 3, no. 5, May 1968, pp. 425–436.

[72] T. Troudet and R.J. Hawkins, "Monte Carlo simulation of the propagation of single-mode dielectric waveguide structures," *Appl. Opt.,* vol. 27, no. 24, Feb. 1988, pp. 765–773.

[73] T.R. Rowbotham and P.B. Johns, "Waveguide analysis by random walks," *Elect. Lett.,* vol. 8, no. 10, May 1972, pp. 251–253.

[74] P.B. Johns and T.R. Rowbothan, "Solution of resistive meshes by deterministic and Monte Carlo transmission-line modelling," *IEEE Proc.,* vol. 128, Part A, no. 6, Sept. 1981, pp. 453–462.

[75] R.G. Olsen, "The application of Monte Carlo techniques to the study of impairments in the waveguide transmission system," *B. S. T. J.,* vol. 50, no. 4, April 1971, pp. 1293–1310.

[76] H.E. Rowe and D.T. Young, "Transmission distortion in multimode random waveguides," *IEEE Trans. Micro. Theo. Tech.,* vol. MMT-20, no. 6, June 1972, pp. 349–365.

[77] C. Huang et al., "Stationary phase Monte Carlo path integral analysis of electromagnetic wave propagation in graded-index waveguides," *IEEE Trans. Micro. Theo. Tech.,* vol. 42, no. 9, Sept. 1994, pp. 1709–1714.

[78] H.T. Chou and J.T. Johnson, "A novel acceleration algorithm for the computation of scattering from a rough surfaces with the forward-backward method," *Radio Science,* vol. 33, no. 5, Sept./Oct. 1998, pp. 1277–1287.

[79] M. Nieto-Vesperinas and J.M. Soto-Crespo, "Monte Carlo simulations for scattering of electromagnetic waves from perfectly conductive random rough surfaces," *Opt. Lett.,* vol. 12, no. 12, Dec. 1987, pp. 979–981.

[80] G.P. Bein, "Monte Carlo computer technique for one-dimensional random media," *IEEE Trans. Ant. Prop.,* vol. AP-21, no. 1, Jan. 1973, pp. 83–88.

[81] N. Garcia and E. Stoll, "Monte Carlo calculation for electromagnetic-wave scattering from random rough surfaces," *Phy. Rev. Lett.,* vol. 52, no. 20, May 1984, pp. 1798–1801.

[82] J. Nakayama, "Anomalous scattering from a slightly random surface," *Rad. Sci.,* vol. 17, no. 3, May-June 1982, pp. 558–564.

[83] A. Ishimaru, *Wave Propagation and Scattering in Random Media.* New York: Academic Press, vol. 2, 1978.

[84] A.K. Fung And M.F. Chen, "Numerical simulation of scattering from simple and composite random surfaces," *J. Opt. Soc. Am.,* vol. 2, no. 12, Dec. 1985, pp. 2274–2284.

[85] R.A. Arratia and H.N. Bertram, "Monte Carlo simulation of particulate noise in magnetic recording," *IEEE Trans. Mag.,* vol. MAG-20, no. 2, Mar. 1984, pp. 412–420.

[86] P.K. Davis, "Monte Carlo analysis of recording codes," *IEEE Trans. Mag.,* vol. MAG-20, no. 5, Sept. 1984, p. 887.

[87] J.H. Pickles, "Monte-Carlo calculation of electrically induced human-body currents," *IEEE Proc.,* vol. 134, Pt. A, no. 9, Nov. 1987, pp. 705–711.

[88] K. Gu and M.N.O. Sadiku, "A triangular mesh random walk for Dirichlet problems," *Jour. Franklin Inst.,* vol. 332B, no. 5, 1995, pp. 569–578.

[89] R.C. Garcia and M.N.O. Sadiku, "Neuro-Monte Carlo solution of electrostatic problems," *Jour. Franklin Inst.,* vol. 335B, no. 1, 1998, pp. 53–69.

[90] M.N. Barber and B.W. Ninham, *Random and Restricted Walks.* New York: Gordon and Breach, 1970.

[91] K.K. Sabelfeld, *Monte Carlo Methods in Boundary Value Problems.* New York: Springer-Verlag, 1991.

Problems

8.1 Write a program to generate 1000 pseudorandom numbers U uniformly distributed between 0 and 1. Calculate their mean and compare the calculated mean with the expected mean (0.5) as a test of randomness.

8.2 Generate 10,000 random numbers uniformly distributed between 0 and 1. Find the percentage of numbers between 0 and 0.1, between 0.1 and 0.2, etc., and compare your results with the expected distribution of 10% in each interval.

8.3 (a) Using the linear congruential scheme, generate 10 pseudorandom numbers with $a = 1573$, $c = 19$, $m = 10^3$, and seed value $X_0 = 89$.

(b) Repeat the generation with $c = 0$.

8.4 For $a = 13, m = 2^6 = 64$, and $X_0 = 1, 2, 3$, and 4, find the period of the random number generator using the multiplicative congruential method.

8.5 Develop a subroutine that uses the inverse transformation method to generate a random number from a distribution with the probability density function

$$f(x) = \begin{cases} 0.25, & 0 \leq x \leq 1 \\ 0.75, & 1 \leq x \leq 1 \end{cases}$$

8.6 It is not easy to apply the inverse transform method to generate normal distribution. However, by making use of the approximation

$$e^{-x^2/2} \simeq \frac{2e^{-kx}}{\left(1 + e^{-kx}\right)^2}, \; x > 0$$

where $k = \sqrt{\frac{8}{\pi}}$, the inverse transform method can be applied. Develop a subroutine to generate normal deviates using inverse transform method.

8.7 Using the rejection method, generate a random variable from $f(x) = 5x^2, \; 0 \leq x \leq 1$.

8.8 Use the rejection method to generate Gaussian (or normal) deviates in the truncated region $-a \leq X \leq a$.

8.9 Use sample mean Monte Carlo integration to evaluate:

(a) $\int_0^1 4\sqrt{1 - x^2}\, dx$,

(b) $\int_0^1 \sin x \, dx$,

(c) $\int_0^1 e^x \, dx$,

(d) $\int_0^1 \frac{1}{\sqrt{x}} \, dx$

8.10 Evaluate the following four-dimensional integrals:

(a) $\int_0^1 \int_0^1 \int_0^1 \int_0^1 \exp(x^1 x^2 x^3 x^4 - 1)\, dx^1\, dx^2\, dx^3\, dx^4$,

(b) $\int_0^1 \int_0^1 \int_0^1 \int_0^1 \sin(x^1 + x^2 + x^3 + x^4)\, dx^1\, dx^2\, dx^3\, dx^4$

8.11 The radiation from a rectangular aperture with constant amplitude and phase distribution may be represented by the integral

$$I(\alpha, \beta) = \int_{-1/2}^{1/2} \int_{-1/2}^{1/2} e^{j(\alpha x + \beta y)}\, dx\, dy$$

Evaluate this integral using a Monte Carlo procedure and compare your result for $\alpha = \beta = \pi$ with the exact solution

$$I(\alpha, \beta) = \frac{\sin(\alpha/2)\ \sin(\beta/2)}{\alpha\beta/4}$$

8.12 Consider the differential equation

$$\frac{\partial^2 W}{\partial x^2} + \frac{\partial^2 W}{\partial y^2} + \frac{k}{y}\frac{\partial W}{\partial y} = 0$$

where $k =$ constant. By finding its finite difference form, give a probabilistic interpretation to the equation.

8.13 Given the one-dimensional differential equation

$$y'' = 0, \qquad 0 \leq x \leq 1$$

subject to $y(0) = 0$, $y(1) = 10$, use an MCM to find $y(0.25)$ assuming $\Delta x = 0.25$ and the following 20 random numbers:

0.1306, 0.0422, 0.6597, 0.7905, 0.7695, 0.5106, 0.2961, 0.1428, 0.3666, 0.6543, 0.9975, 0.4866, 0.8239, 0.8722, 0.1330, 0.2296, 0.3582, 0.5872, 0.1134, 0.1403.

8.14 Consider N equal resistors connected in series as in Fig. 8.21. By making $V(0) = 0$ and $V(N) = 10$ volts, find $V(k)$ using the fixed random walk for the following cases: (a) $N = 5,\ k = 2$, (b) $N = 10,\ k = 7$, (c) $N = 20,\ k = 11$.

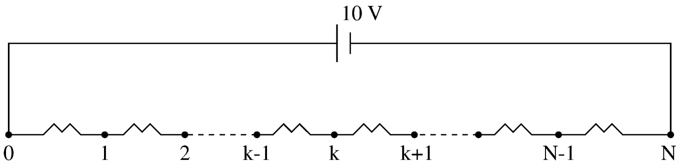


Figure 8.21
For Problem 8.14.

8.15 Use a Monte Carlo method to solve Laplace's equation in the triangular region $x \geq 0,\ y \geq 0,\ x + y \leq 1$ with the boundary condition $V(x, y) = x + y + 0.5$. Determine V at (0.4, 0.2), (0.35, 0.2), (0.4, 0.15), (0.45, 0.2), and (0.4, 0.25).

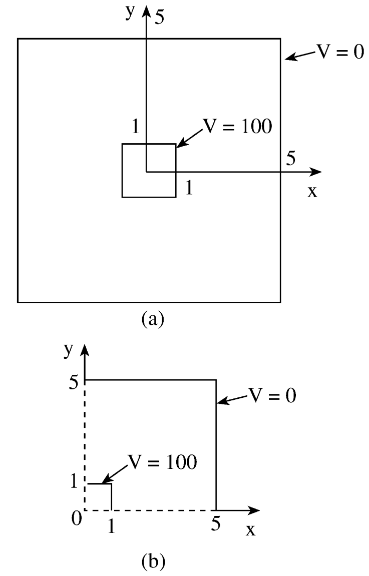


Figure 8.22
For Problem 8.16.

8.16 Use a Monte Carlo procedure to determine the potential at points (2,2), (3,3), and (4,4) in the problem shown in Fig. 8.22(a). By virtue of double symmetry, it is sufficient to consider a quarter of the solution region as shown in Fig. 8.22(b).

8.17 In the solution region of Fig. 8.23, $\rho_s = x(y-1)\,n\mathrm{C}/m^2$. Find the potential at the center of the region using a Monte Carlo method.

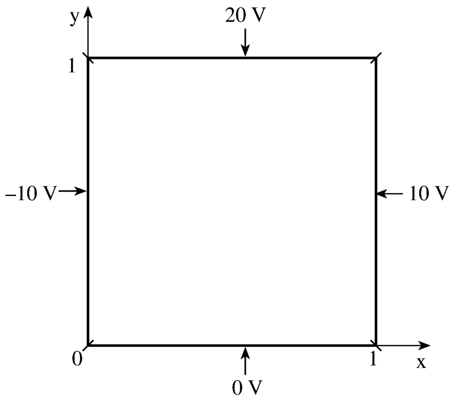
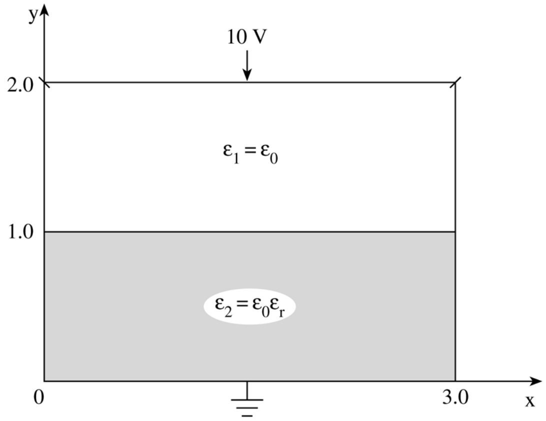


Figure 8.23
For Problem 8.17.

8.18 Consider the potential system shown in Fig. 8.24. Determine the potential at the center of the solution region. Take $\epsilon_r = 2.25$.



y
10 V
2.0
$\varepsilon_1 = \varepsilon_0$
1.0
$\varepsilon_2 = \varepsilon_0 \varepsilon_r$
0
3.0
x

Figure 8.24
For Problem 8.18.

8.19 Apply an MCM to solve Laplace's equation in the three-dimensional region

$$|x| \le 1, \quad |y| \le 0.5, \quad |z| \le 0.5$$

subject to the boundary condition

$$V(x, y, z) = x + y + z + 0.5$$

Find the solution at (0.5, 0.1, 0.1).

8.20 Consider the interface separating two homogeneous media in Fig. 8.25. By applying Gauss's law

$$\oint_S \epsilon \frac{\partial V}{\partial n} \, dS = 0$$

show that

$$\begin{aligned} V(\rho, z) = {} & p_{\rho+} V(\rho + \Delta, z) + p_{\rho-} V(\rho - \Delta, z) \\ & + p_{z+} V(\rho, z + \Delta) + p_{z-} V(\rho, z - \Delta) \end{aligned}$$

where

$$p_{z+} = \frac{\epsilon_1}{2(\epsilon_1 + \epsilon_2)}, \qquad p_{z-} = \frac{\epsilon_2}{2(\epsilon_1 + \epsilon_2)}$$
$$p_{\rho+} = p_{\rho-} = \frac{1}{4}$$

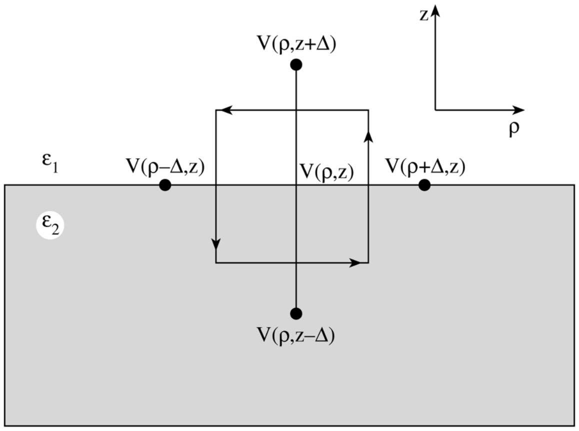


Figure 8.25
For Problem 8.20.

8.21 Consider the finite cylindrical conductor held at $V = 100$ enclosed in a larger grounded cylinder. The axial symmetric problem is portrayed in Fig. 8.26 for your convenience. Using a Monte Carlo technique, write a program to determine the potential at points $(\rho, z) = (2,10), (5,10), (8,10), (5,2)$, and $(5,18)$.

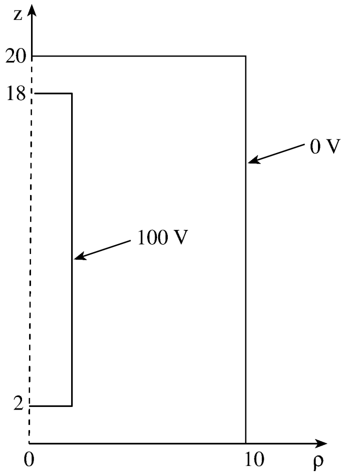


Figure 8.26
For Problem 8.21.

8.22 Figure 8.27 shows a prototype of an electrostatic particle focusing system employed in a recoil-mass time-of-flight spectometer. It is essentially a finite cylindrical conductor that abruptly expands radius by a factor of 2. Write a program based on an MCM to calculate the potential at points $(\rho, z) = (5,18)$, (5,10), (5,2), (10,2), and (15,2).

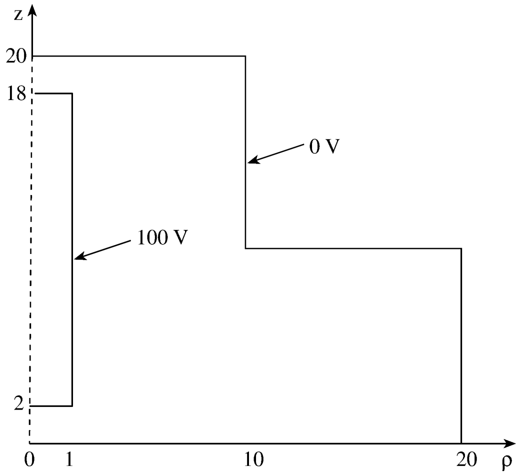


Figure 8.27
For Problem 8.22.

8.23 Consider the square region shown in Fig. 8.28. The transition probability $p(Q, S_i)$ is defined as the probability that a randomly walking particle leaving point Q will arrive at side S_i of the square boundary. Using the Exodus method, write a program to determine:

(a) $p(Q_1, S_i)$, $\quad i = 1, 2, 3, 4,$

(b) $p(Q_2, S_i)$, $\quad i = 1, 2, 3, 4.$

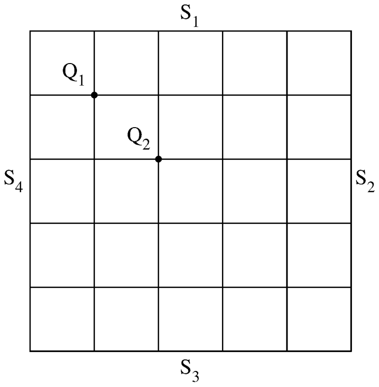


Figure 8.28
For Problem 8.23.

8.24 Given the one-dimensional differential equation

$$\frac{d^2\Phi}{dx^2} = 0, \quad 0 \leq x \leq 1$$

subject to $\Phi(0) = 0, \Phi(1) = 10$, use the Exodus method to find $\Phi(0.25)$ by injecting 256 particles at $x = 0.25$. You can solve this problem by hand calculation.

8.25 Use the Exodus method to find the potential at node 4 in Fig. 8.29. Inject 256 particles at node 4 and scan nodes in the order 1, 2, 3, 4. You can solve this problem by hand calculation.

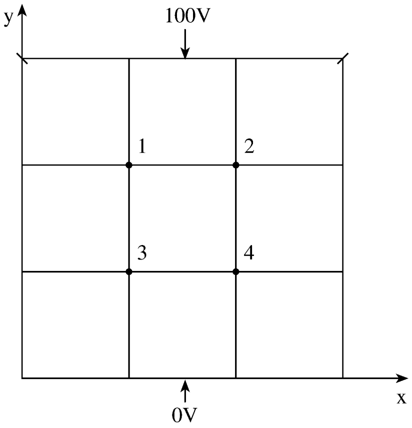


Figure 8.29
For Problem 8.25

8.26 Using the Exodus method, write a program to calculate $V(0.25, 0.75)$ in Example 8.5.

8.27 Write a program to calculate $V(1.0, 1.5)$ in Example 8.6 using the Exodus method.

8.28 Write a program that will apply the Exodus method to determine the potential at point (0.2, 0.4) in the system shown in Fig. 8.30.

8.29 Use Markov chain MCM to determine the potential at node 5 in Fig. 8.31.

8.30 Rework Problem 8.18 using Markov chain.

8.31 Rework Problem 8.22 using Markov chain.

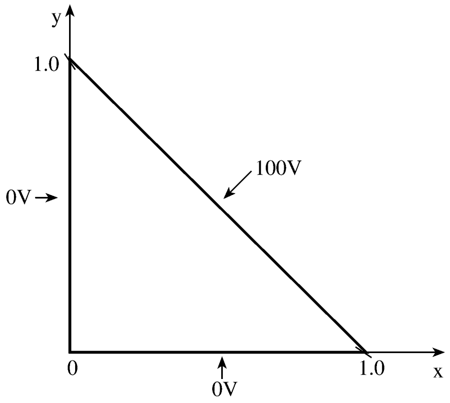


Figure 8.30
For Problem 8.28.

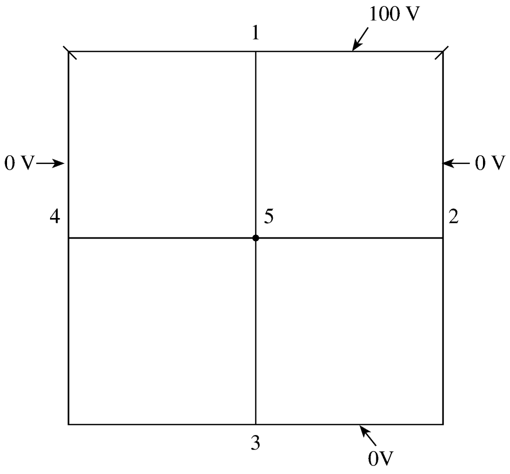


Figure 8.31
For Problem 8.29.