

This dissertation has been  
microfilmed exactly as received 67-6932

HANDLER, Howard, 1936-  
HIGH-SPEED MONTE CARLO TECHNIQUE FOR HYBRID-  
COMPUTER SOLUTION OF PARTIAL DIFFERENTIAL  
EQUATIONS.

University of Arizona, Ph.D., 1967  
Engineering, electrical

University Microfilms, Inc., Ann Arbor, Michigan

HIGH-SPEED MONTE CARLO TECHNIQUE FOR HYBRID-COMPUTER  
SOLUTION OF PARTIAL DIFFERENTIAL EQUATIONS

by

Howard Handler

---

A Dissertation Submitted to the Faculty of the  
DEPARTMENT OF ELECTRICAL ENGINEERING  
In Partial Fulfillment of the Requirements  
For the Degree of  
DOCTOR OF PHILOSOPHY  
In the Graduate College  
THE UNIVERSITY OF ARIZONA

1 9 6 7

THE UNIVERSITY OF ARIZONA  
GRADUATE COLLEGE

I hereby recommend that this dissertation prepared under my  
direction by Howard Handler  
entitled High-Speed Monte Carlo Technique for Hybrid-Computer  
Solution of Partial Differential Equations  
be accepted as fulfilling the dissertation requirement of the  
degree of Doctor of Philosophy

CAKorn  
Dissertation Director

11/18/66  
Date

After inspection of the dissertation, the following members  
of the Final Examination Committee concur in its approval and  
recommend its acceptance:\*

Neil M Wigley

Nov 18, 66

P.K. Whittacharya

Nov 18, 1966

CAKorn

11/18/66

John V. Grant

11/18/66

Fredrick J. Hill

11/18/66

\*This approval and acceptance is contingent on the candidate's  
adequate performance and defense of this dissertation at the  
final oral examination. The inclusion of this sheet bound into  
the library copy of the dissertation is evidence of satisfactory  
performance at the final examination.

STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: \_\_\_\_\_

*Howard Handler*

## ACKNOWLEDGMENTS

The author would like to express his sincere appreciation to Dr. Granino A. Korn for his guidance and interest during this study and during the author's stay at the University of Arizona. A special note of thanks is due the author's wife, Dorothy, for her encouragement, assistance, and understanding during his career.

## TABLE OF CONTENTS

	Page
LIST OF ILLUSTRATIONS . . . . .	vii
LIST OF TABLES . . . . .	x
ABSTRACT . . . . .	xi
 CHAPTER	
1. INTRODUCTION . . . . .	1
2. MONTE CARLO METHODS FOR THE SOLUTION OF PARTIAL DIFFERENTIAL EQUATIONS . . . . .	5
2.1 Introduction . . . . .	5
2.2 Simplified Derivation of the Kolmogorov Partial Differential Equations for Linear Langevin Equations . . . . .	6
2.3 Markov Processes and the Kolmogorov Backward Partial Differential Equations . . . . .	15
2.4 Monte Carlo Solution of Parabolic Partial Differential Equations with Dirichlet- type Boundary Conditions . . . . .	19
2.5 Monte Carlo Solution of Elliptic Partial Differential Equations with Dirichlet- type Boundary Conditions . . . . .	22
2.6 Restrictions Limiting the Monte Carlo Method to Parabolic and Elliptic Second Order Partial Differential Equations . . . . .	25
2.7 Monte Carlo Solution of Nonhomogeneous Partial Differential Equations . . . . .	26
2.8 Extension of the Monte Carlo Technique to the Case where $\frac{\partial U}{\partial n} = 0$ Over a Portion of the Boundary . . . . .	32
2.9 Monte Carlo Procedure for the Case where $\frac{\partial U}{\partial n} = a$ Over a Straight Line Portion of the Boundary . . . . .	37
2.10 Monte Carlo Procedure for the Case where the Normal Derivative is Constant Along Orthogonal Segments on the Boundary . .	45

TABLE OF CONTENTS--Continued

	Page
3. ERROR CONSIDERATIONS IN MONTE CARLO COMPUTATIONS . . . . .	55
3.1 Introduction . . . . .	55
3.2 Amplifier Drift Errors . . . . .	55
3.3 Errors Due to Finite Noise-source and Integrator Bandwidths . . . . .	59
3.4 Integrator Gain Considerations for Binary Signals . . . . .	64
3.5 Effect of the Finite Computer-run Time Allotted to Each Random Walk . . . . .	67
3.6 Sweep Errors . . . . .	71
3.7 Averaging Time Requirements and Statistical Fluctuations . . . . .	75
4. ANALOG-HYBRID COMPUTER CONSIDERATIONS FOR MONTE CARLO SOLUTIONS OF PARTIAL DIFFERENTIAL EQUATIONS . . . . .	82
4.1 Introduction . . . . .	82
4.2 Description of ASTRAC II . . . . .	82
4.3 Noise Source Calibration . . . . .	85
4.4 Computer Mechanization of a Monte Carlo Problem . . . . .	90
4.5 Special Analog-hybrid Techniques in Monte Carlo Computations . . . . .	98
4.5A Boundary Detection . . . . .	98
4.5B Generation of Initial and Boundary Conditions . . . . .	102
4.5C Measurement of the Average Time of a Random Walk for Use in the Solution of Nonhomogeneous Partial Differential Equations . . . . .	106
5. MONTE CARLO SOLUTION OF ILLUSTRATIVE PROBLEMS . . . . .	108
5.1 Introduction . . . . .	108
5.2 Monte Carlo Solutions of One-dimensional Problems . . . . .	108
5.3 Monte Carlo Solutions of Two-dimensional Problems . . . . .	124
5.4 Monte Carlo Generation of Level-lines of Partial Differential Equations . . . . .	138
5.5 Discussion of Results . . . . .	147
6. CONCLUDING REMARKS . . . . .	150


TABLE OF CONTENTS--Continued

	Page
APPENDIX A. RELATIONSHIPS FOR MARKOV PROCESSES . . .	153
APPENDIX B. THE PARTIAL DIFFERENTIAL EQUATION SATISFIED BY THE AVERAGE TIME OF A RANDOM WALK . . . . .	162
APPENDIX C. PROOF OF THE MONTE CARLO TECHNIQUES FOR THE SOLUTION OF PARTIAL DIFFERENTIAL EQUATIONS . . . . .	165
APPENDIX D. BINARY NOISE SOURCE . . . . .	170
REFERENCES . . . . .	175

## LIST OF ILLUSTRATIONS

Figure		Page
2.1	A General Linear First-order System Excited by a Noise Signal . . . . .	14
2.2	Block Diagram of the Generalized Langevin Equations . . . . .	17
2.3	Portion of a Reflecting Boundary . . . . .	35
2.4	Region R and the Augmented Region R' . . . . .	35
2.5	Diagram Pertaining to the Problem of Section 2.9 . . . . .	38
2.6	Decomposition of the Problem Shown in Fig. 2.5 for Use in Procedure 2 . . . . .	42
2.7	Modification of the Problem of Fig. 2.6 . . . . .	43
2.8	Diagram Pertaining to the Problem of Section 2.10 . . . . .	47
2.9	Decomposition of the Problem Shown in Fig. 2.8 . . . . .	49
2.10	Modification of the Problem of Fig. 2.9 . . . . .	50
3.1	Integration of Noise Signal Showing Integrator Drift . . . . .	57
3.2	Autocorrelation Function of Binary-noise Waveform . . . . .	60
3.3	Block Diagram of a Langevin Equation Excited by Band-limited Noise . . . . .	62
3.4	Computer Diagram for Generating Level-lines . . . . .	76
3.5	An Analog Averaging Circuit . . . . .	78
3.6	Autocorrelation Function of the Random Binary Waveform . . . . .	78
4.1	ASTRAC II Timing Signals . . . . .	86

LIST OF ILLUSTRATIONS--Continued

	Page
4.2	Noise Generator Calibration Diagrams . . . . . 89
4.3	Hybrid Computer Symbols . . . . . 93
4.4	Hybrid Computer Implementation of the Monte Carlo Solution of the Diffusion Equation . . . . . 94
4.5	Typical Waveforms Appearing in a Monte Carlo Solution . . . . . 95
4.6	Photographs of Waveforms Appearing in a Monte Carlo Solution . . . . . 96
4.7	Circular Boundary Detector . . . . . 100
4.8	Two-Dimensional Region With Boundary C Described by Two Curves . . . . . 101
4.9	Hybrid Computer Diagram of a Boundary Detector for the Boundary Shown in Fig. 4.8 . . . . . 103
4.10	Hybrid Computer Arrangement for Introducing Initial Conditions of the Partial Differential Equation into a Monte Carlo Solution . . . . . 105
4.11	Average Random Walk Time Measurement . . . . . 107
5.1	Hybrid Computer Diagram for the Monte Carlo Solution of Examples 1 and 2 . . . . . 111
5.2	Monte Carlo Solution of Example 1 . . . . . 112
5.3	Digital Monte Carlo Error Solution of Example 1 . . . . . 113
5.4	Monte Carlo Solution of Example 2 . . . . . 115
5.5	Hybrid Computer Diagram for the Monte Carlo Solution of Example 3 . . . . . 117
5.6	Monte Carlo Solution of Example 3 . . . . . 118
5.7	Generation of the Variable Timing Signal  . . . . . 121

LIST OF ILLUSTRATIONS--Continued

	Page
5.8 Monte Carlo Solution of Example 4a . . . . .	122
5.9 Monte Carlo Solution of Example 4b . . . . .	123
5.10 Hybrid Computer Diagram for the Monte Carlo Solution of Example 5 . . . . .	125
5.11 Monte Carlo Solution of Example 5 . . . . .	126
5.12 Hybrid Computer Diagram for the Monte Carlo Solution of Example 6 . . . . .	128
5.13 Monte Carlo Solution of Example 6 . . . . .	129
5.14 Hybrid Computer Diagram for the Monte Carlo Solution of Example 7 . . . . .	131
5.15 Monte Carlo Solution of Example 7 . . . . .	132
5.16 Hybrid Computer Diagram for the Monte Carlo Solution of Example 8 . . . . .	135
5.17 Monte Carlo Solution of Example 8a . . . . .	136
5.18 Monte Carlo Solution of Example 8b . . . . .	137
5.19 Hybrid Computer Diagram for the Monte Carlo Solution of Example 9 . . . . .	139
5.20 Monte Carlo Solution of Example 9 . . . . .	140
5.21 Monte Carlo Solution of Example 10 . . . . .	142
5.22 Monte Carlo Solution of Example 11 . . . . .	144
5.23 Monte Carlo Solution of Example 12 . . . . .	146
5.24 Monte Carlo Solution of Example 13 . . . . .	148
D.1 Block Diagram of Noise Generator . . . . .	171
D.2 Measured Autocorrelation Function of the Noise Generator . . . . .	173
D.3 Noise Generator Schematic Diagram . . . . .	174

LIST OF TABLES

Table		Page
1.1	Definitions of Two Conditional Probability Functions . . . . .	7
4.1	ASTRAC II Performance Data . . . . .	84

## ABSTRACT

The analog-hybrid computer Monte Carlo technique for solving elliptic and parabolic partial differential equations has been implemented on a new hybrid computer capable of taking statistics over 1,000 two- or three-dimensional random walks each second. This exceptional computing speed and flexible digital control permit direct plotting of partial differential equation solutions; of perhaps even greater interest is the incorporation of such Monte Carlo routines in real-time analog computer setups in process control applications. In this connection, the Monte Carlo method has been extended to a wider class of boundary conditions especially applicable to heat conduction/diffusion problems.

## CHAPTER 1

### INTRODUCTION

A wide variety of computing methods has been applied in the solution of partial differential equations which occur in the modeling of continuous systems.<sup>1</sup> Both digital and analog computers have been used to solve finite difference equations resulting from the replacement of derivatives by finite differences.<sup>2</sup> This procedure requires the use of much computing equipment and/or computer time, depending upon which variables have been discretized. In addition, even if the solution is desired at one point only, it is still necessary in the finite-difference scheme to develop the solution over the entire region.

The observation that the conditional probability density function of a Markov process<sup>3</sup> satisfies a second order elliptic or parabolic partial differential equation lead early investigators to suggest a Monte Carlo method for the solution of such partial differential equations.<sup>4</sup>

A Markov process is generated on an analog computer by implementing the Langevin<sup>5</sup> stochastic differential equations of motion resulting from the excitation of  $n$  coupled first order non-linear systems with  $n$  zero-mean

white Gaussian noise signals. For partial differential equations with specified boundary values (Dirichlet's problem), random walks obtained as solutions to the Langevin equations are initiated from the point in space where the solution is desired. In the case of elliptical partial differential equations, the walks are continued until the boundary is crossed, whereupon the function value at the boundary is tallied. In the case of parabolic partial differential equations, the walks are allowed to continue for a fixed time, and one tallies the value of the function at the boundary crossing point, if indeed the boundary is crossed within the time interval allotted for the random walk; if the walk terminates before an absorption occurs one tallies the value of the initial condition function at the internal point where the walk terminated. In both cases, the sample average over a suitable number of random walks is an unbiased estimate for the desired solution.

This technique, while conceptually simple, was made feasible only by the development of high speed computing machines. Digital computers were utilized in the late 1940's and early 1950's<sup>7, 8, 9</sup> for Monte Carlo solutions and various procedures were developed to generate tallies of numbers which converged in a statistical sense to the solution of the parabolic and elliptic partial differential equations.

Chuang, Kazda, and Windeknecht<sup>10</sup> were the first to employ Monte Carlo procedures for the solution of partial differential equations on an analog computer. They solved the Langevin equations on a conventional "slow" analog computer with tape-recorded random noise inputs and applied  $x(t)$ ,  $y(t)$  to the horizontal and vertical plates of an oscilloscope. To demonstrate the feasibility of the Monte Carlo method, they restricted themselves to simple piecewise constant boundary functions equal to 100 on a portion  $C_1$  of the boundary  $C$  and zero on the remaining portion of the boundary. Boundary crossings of the oscilloscope beam marking the point  $(x,y)$  were detected by an arrangement of masks and photocells. The solution of the partial differential equation (the expected value of the tallies) was estimated by 100 times the fraction of the boundary crossings taking place across  $C_1$ , as determined by a decimal counter. Computer and noise source limitations limited computing speed to about one random walk per second; solution errors for samples of 300 to 2200 runs were of the order of a few per cent and were ascribed mainly to statistical fluctuations.

In modern analog-hybrid computers, accurate and convenient combinations of function generators, analog comparators, and digital logic replace the cumbersome photocell circuits used to detect boundary crossings. Little<sup>11</sup> utilized an analog-hybrid computer in which a

Pace 231R-V analog computer and an ALWAC III-E digital computer were linked together so that continuous random walks could be generated in the analog computer, while the digital computer generated timing signals, stored boundary values, and accumulated data; the solution speed was limited to about 1000 random walks per five minutes. In the present work, Monte Carlo solutions of partial differential equations are obtained on ASTRAC II,<sup>12</sup> a new iterative differential analyzer capable of generating 1000 complete random walks per second; it is this fact which makes the Monte Carlo method competitive with other methods for a restricted class of problems.

## CHAPTER 2

### MONTE CARLO METHODS FOR THE SOLUTION OF PARTIAL DIFFERENTIAL EQUATIONS

#### 2.1 Introduction

In this chapter the partial differential equations which can be treated by sampling or Monte Carlo methods will be presented. These equations must be of the second order, and may be either parabolic or elliptic; the limitations which prevent the consideration of hyperbolic equations will be examined.

The traditional approach leading to the Kolmogorov backward partial differential equation will not be employed in this chapter; instead, a rather simple parabolic partial differential equation in one dimension will be examined by determining the response of a linear system to a white Gaussian noise excitation. The partial differential equation satisfied by the conditional probability density function is obtained by standard random process theory without resorting to the theory of Markov processes. Our derivation may be more understandable for many engineers. The more general approach is, however, reviewed in Appendix A.

The treatment of nonhomogeneous partial differential equations and of partial differential equations with other

than Dirichlet boundary conditions will also be considered in this chapter.

## 2.2 Simplified Derivation of the Kolmogorov Partial Differential Equations for Linear Langevin Equations

To determine the class of problems which may be treated by Monte Carlo methods, it is necessary to investigate the behavior of the functions defined in Table 1.1.

The common procedure for finding the partial differential equations which can be treated by Monte Carlo methods has been to investigate the generalized Langevin equations representing the response of first order systems to white Gaussian noise forcing-functions.<sup>8, 11</sup> As in Appendix A, the conditional probability density function of the output variable (which completely describes the output process since it is a first order Markov process) satisfies a partial differential equation (Kolmogorov backward partial differential equation). The solution of second order boundary value problems of both the elliptic and parabolic type is based on these Kolmogorov backward partial differential equations, which follow directly from the Markovian property of the output random process.

An alternative way to derive the form of the partial differential equations which can be handled by the Monte Carlo method, is to consider the behavior of the conditional probability density function of the output of

Table 1.1. Definitions of Two Conditional Probability Functions

---



---

Definition 1

Let  $f(\underline{r}_k, t_k / \underline{r}_i, t_i)$  denote the conditional probability density function,<sup>13</sup> where  $\underline{r}_k$  and  $\underline{r}_i$  are generalized position vectors of a particle performing a random walk<sup>5</sup> in a finite-dimensional space. The quantity

$$\int_{R_a} f(\underline{r}_k, t_k / \underline{r}_i, t_i) d\underline{r}_k$$

represents the probability that a particle performing an n-dimensional random walk will be in the region  $R_a$  at the time  $t_k$ , given that it was at the point  $\underline{r}_i$  at the time  $t_i$ .

Definition 2

$g(\underline{r}_b, t_b / \underline{r}_i, t_i)$  denotes the conditional probability boundary function,<sup>11</sup> where

$$\int_{t_i}^{t_i+T} \int_C g(\underline{r}_b, t_b / \underline{r}_i, t_i) d\underline{r}_b dt_b$$

represents the probability that a particle performing a random walk will reach the boundary C at any time between  $t_i$  and  $t_i+T$ , given that the particle was at the point  $\underline{r}_i$  at time  $t_i$ .

---

a linear electrical network excited by white Gaussian noise. To this end, consider a linear electrical network characterized in terms of:

1. The impulse response function,  $h(t, t_i)$ , i.e., the response of the system at time  $t$  to a unit impulse applied at time  $t_i$ .
2. The mean value function,  $m(t, t_i)$ , i.e., the output voltage at time  $t$  given that the output voltage at time  $t_i$  was one volt.

Assuming that the linear system characterized by  $h(t, t_i)$  and  $m(t, t_i)$  is excited by a Gaussian white noise input voltage,  $N(t)$ , with autocorrelation function

$$E [N(t)N(t+\lambda)] = 2D\delta(\lambda) \quad (2.1)$$

it is of interest to determine the conditional probability density function of the output process.

Since a linear system is excited by a Gaussian signal, the output process is Gaussian<sup>13</sup> and is completely defined by its mean value and covariance function. If the input process starts at  $t=t_i$ , and it is known that the system output voltage at that time is  $y_i$  volts, the mean value of the output  $y(t)$

$$E[y(t, t_i)] = y_i m(t, t_i) \quad (2.2)$$

since the forcing function has a zero mean value.

The variance of  $y(t)$ ,

$$\text{Var}\{y(t, t_i)\} = 2D \int_{t_i}^t h^2(t-t_1) dt_1 = 2D \int_0^{t-t_i} h^2(x) dx \quad (2.3)$$

In terms of the mean value and variance just obtained, the conditional Gaussian probability density function of  $y(t)$  given  $y(t_i) = y_i$  is

$$f(y, t/y_i, t_i) = \frac{1}{\sqrt{2\pi \text{Var}\{y(t, t_i)\}}} e^{-\frac{(y-y_m(t, t_i))^2}{2\text{Var}\{y(t, t_i)\}}} \quad (2.4)$$

It is more convenient to consider the characteristic function

$$X(q, t/y_i, t_i) = \int_{-\infty}^{\infty} f(y, t/y_i, t_i) e^{-iqy} dy \quad (2.5)$$

Assume that the characteristic function satisfies the partial differential equation

$$\left[ L_{y_i, t_i} + \frac{\partial}{\partial t_i} \right] X(q, t/y_i, t_i) = 0 \quad (2.6)$$

where  $L_{y_i, t_i}$  denotes an operator which is a function of  $y_i$  and  $t_i$  and where differential operations are with respect to the coordinate of the initial position,  $y_i$ . It is of interest to determine the partial differential equation which the probability density function satisfies. Operating on the conditional probability density function given by

the inverse Fourier transformation of the characteristic function, with  $\left[ L_{y_i, t_i} + \frac{\partial}{\partial t_i} \right]$  yields,

$$\left[ L_{y_i, t_i} + \frac{\partial}{\partial t_i} \right] f(y, t/y_i, t_i) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left[ L_{y_i, t_i} + \frac{\partial}{\partial t_i} \right]$$

$$X(q, t/y_i, t_i) e^{iqy} dy \quad (2.7)$$

or

$$\left[ L_{y_i, t_i} + \frac{\partial}{\partial t_i} \right] f(y, t/y_i, t_i) = 0 \quad (2.8)$$

Thus the conditional probability density function satisfies the same partial differential equation as the characteristic function. More specifically, if the probability density function is given by (2.4), then

$$X(q, t/y_i, t_i) = e^{\left[ i q y_i m(t, t_i) - 1/2 q^2 \text{Var} \left[ y(t, t_i) \right] \right]} \quad (2.9)$$

Taking the partial derivative of the characteristic function with respect to both  $t_i$  and  $y_i$  and the second order partial derivative with respect to  $y_i$ , one obtains,

$$\frac{\partial X}{\partial t_i} = e^{\left[ i q y_i m(t, t_i) - 1/2 q^2 \text{Var} \left[ y(t, t_i) \right] \right]} \left( i q y_i m'(t, t_i) - 1/2 q^2 \text{Var}' y(t, t_i) \right) \quad (2.10)$$

where the prime denotes differentiation with respect to  $t_i$ .

$$\frac{\partial X}{\partial y_i} = e^{\left[ i q y_i m(t, t_i) - 1/2 q^2 \text{Var}\{y(t, t_i)\} \right]} \left( i q m(t, t_i) \right) \quad (2.11)$$

$$\frac{\partial^2 X}{\partial y_i^2} = e^{\left[ i q y_i m(t, t_i) - 1/2 q^2 \text{Var}\{y(t, t_i)\} \right]} \left( -q^2 m^2(t, t_i) \right) \quad (2.12)$$

By combining the above quantities, it follows that

$$\frac{\partial X}{\partial t_i} = y_i \frac{m'(t, t_i)}{m(t, t_i)} \frac{\partial X}{\partial y_i} + 1/2 \frac{\text{Var}'\{y(t, t_i)\}}{m^2(t, t_i)} \frac{\partial^2 X}{\partial y_i^2} \quad (2.13)$$

This same partial differential equation is also satisfied by the conditional probability density function.

It is possible to put (2.13) into a more convenient form, by expressing the derivative of the variance function in terms of the impulse response function. From (2.3)

$$\frac{\partial}{\partial t_i} \text{Var}\{y(t, t_i)\} = 2D \frac{\partial}{\partial t_i} \int_{t_i}^t h^2(t-t_1) dt_1 = -2Dh^2(t-t_i) \quad (2.14)$$

Rewriting (2.13),

$$\frac{\partial X}{\partial t_i} = y_i \frac{m'(t, t_i)}{m(t, t_i)} \frac{\partial X}{\partial y_i} - \frac{Dh^2(t, t_i)}{m^2(t, t_i)} \frac{\partial^2 X}{\partial y_i^2} \quad (2.15)$$

Since both the characteristic function and the conditional probability density function satisfy the same partial differential equation, the probability density function  $f(y, t/y_i, t_i)$  satisfies the equation

$$\left[ L_{y_i, t_i} + \frac{\partial}{\partial t_i} \right] f(y, t/y_i, t_i) = 0 \quad (2.16)$$

where the form of  $L_{y_i, t_i}$  is given in (2.15).

For subsequent discussions it will be necessary to have both the conditional probability density function and the conditional boundary function  $g(y_b, t/y_i, t_i)$  satisfy the same partial differential equation. Toward this end, consider (2.17) expressing the result that a particle performing a random walk within a closed region  $R$ , bound by the curve  $C$ , and starting at the time  $t_i$  from  $y_i$  will either reach the boundary  $C$  or still remain within  $R$  in a time interval  $t_2 - t_i$ .

$$\int_R f(y, t/y_i, t_i) dy + \int_{t_i}^{t_2} \int_C g(y_b, t_b/y_i, t_i) dy_b dt_b = 1 \quad (2.17)$$

Operating on (2.17) with  $L_{y_i, t_i} + \frac{\partial}{\partial t_i}$ , yields

$$\begin{aligned} & \int_R \left[ L_{y_i, t_i} + \frac{\partial}{\partial t_i} \right] f(y, t/y_i, t_i) dy \\ & + \int_{t_i}^{t_2} \int_C g(y_b, t_b/y_i, t_i) dy_b dt_b \\ & - \int_C g(y_b, t_i/y_i, t_i) dy_b = 0 \end{aligned} \quad (2.18)$$

or

$$\int_{t_1}^{t_2} \int_C \left[ L_{y_i, t_i} + \frac{\partial}{\partial t_i} \right] g(y_b, t_b / y_i, t_i) dy_b dt_b = 0 \quad (2.19)$$

The quantity  $g(y_b, t_b / y_i, t_i) dy_b dt_b$  is the probability that the particle executing a random walk from a point  $y_i$  at the time  $t_i$ , will reach the boundary point  $y_b$  between times  $t_b$  and  $t_b + dt_b$ . The conditional boundary function  $g(y_b, t_b / y_i, t_i)$  identically approaches zero as  $t_b \rightarrow t_i$ , since the particle cannot reach the boundary in an infinitesimal time if it starts at the interior point  $y_i$ . This initial condition forces the last integral of (2.18) to be zero.

For (2.19) to be valid for any reasonable region  $R$ , with boundary  $C$  and arbitrary times  $t_i$  and  $t_b$ , one must have

$$\left[ L_{y_i, t_i} + \frac{\partial}{\partial t_i} \right] g(y_b, t_b / y_i, t_i) = 0 \quad (2.20)$$

It follows that the conditional probability density function and the conditional boundary function satisfy the same partial differential equation.

It is instructive to consider a general first order linear system shown in the analog computer diagram of Fig. 2.1 excited by a white Gaussian noise input. To determine the partial differential equation satisfied by

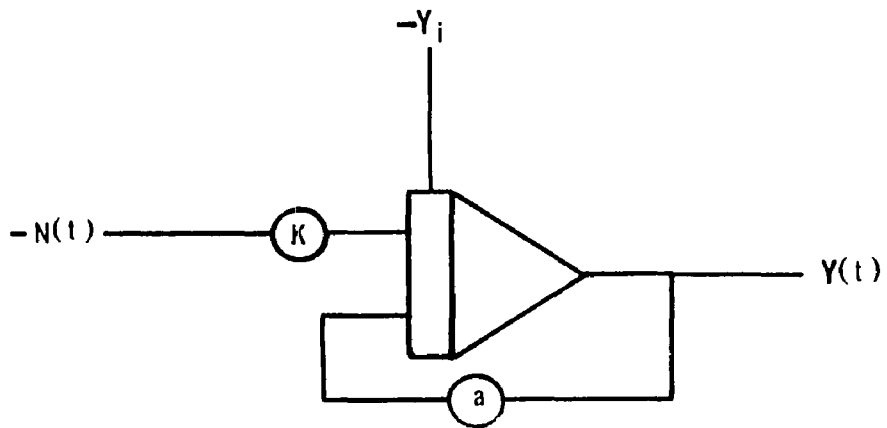


Figure 2.1 A General Linear First-order System Excited by a Noise Signal

the conditional density function, it is necessary to calculate both the mean value function  $m(t, t_i)$  and the impulse response function  $h(t, t_i)$  of the system. The impulse response function of the system is

$$h(t, t_i) = Ke^{-a(t-t_i)} u(t-t_i) \quad (2.21)$$

and the mean value function is

$$m(t, t_i) = e^{-a(t-t_i)} u(t-t_i) \quad (2.22)$$

where the  $u$  denotes the unit step function.<sup>13</sup>

Upon substitution of these terms into (2.15),

$$\frac{\partial X}{\partial t_i} = ay_i \frac{\partial X}{\partial y_i} - K^2 D \frac{\partial^2 X}{\partial y_i^2} \quad (2.23)$$

### 2.3 Markov Processes and the Kolmogorov Backward Partial Differential Equations

The partial differential equation which the conditional probability density function satisfies for a linear first order system excited by a white Gaussian noise signal was developed in Sec. 2.2 by simple random process theory. The restriction to linear systems can be removed by treating the output of a nonlinear first order system excited by a white Gaussian noise signal as a Markov process<sup>3</sup> for which the conditional probability density function satisfies the Kolmogorov backward partial

differential equation.<sup>14</sup> The development of this partial differential equation from the Markovian nature of the system output process is reserved for Appendix A, but the pertinent results will be presented in this section.

If  $n$  coupled first order systems are excited by  $n$  independent white Gaussian noise sources, the equations describing the behavior of the system are called the generalized Langevin<sup>5</sup> equations and are stated in matrix notation in (2.24).

$$\frac{d\mathbf{r}}{dt} + A(\mathbf{r},t) = B(\mathbf{r},t)N(t) \quad (2.24)$$

Each member of (2.24) is a column matrix, the component members of which are tabulated in (2.25).

$$\frac{d\mathbf{r}}{dt} = \frac{d}{dt} \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ \cdot \\ \cdot \\ \cdot \\ r_n \end{bmatrix} \quad A(\mathbf{r},t) = \begin{bmatrix} A_1(\mathbf{r},t) \\ A_2(\mathbf{r},t) \\ A_3(\mathbf{r},t) \\ \cdot \\ \cdot \\ \cdot \\ A_n(\mathbf{r},t) \end{bmatrix} \quad B(\mathbf{r},t)N(t) = \begin{bmatrix} B_1(\mathbf{r},t)N_1(t) \\ B_2(\mathbf{r},t)N_2(t) \\ B_3(\mathbf{r},t)N_3(t) \\ \cdot \\ \cdot \\ \cdot \\ B_n(\mathbf{r},t)N_n(t) \end{bmatrix} \quad (2.25)$$

The column matrix  $\begin{bmatrix} r \end{bmatrix}$  represents the coordinates of a point in  $n$  dimensional space. The block diagram corresponding to the Langevin equations is shown in Fig. 2.2.

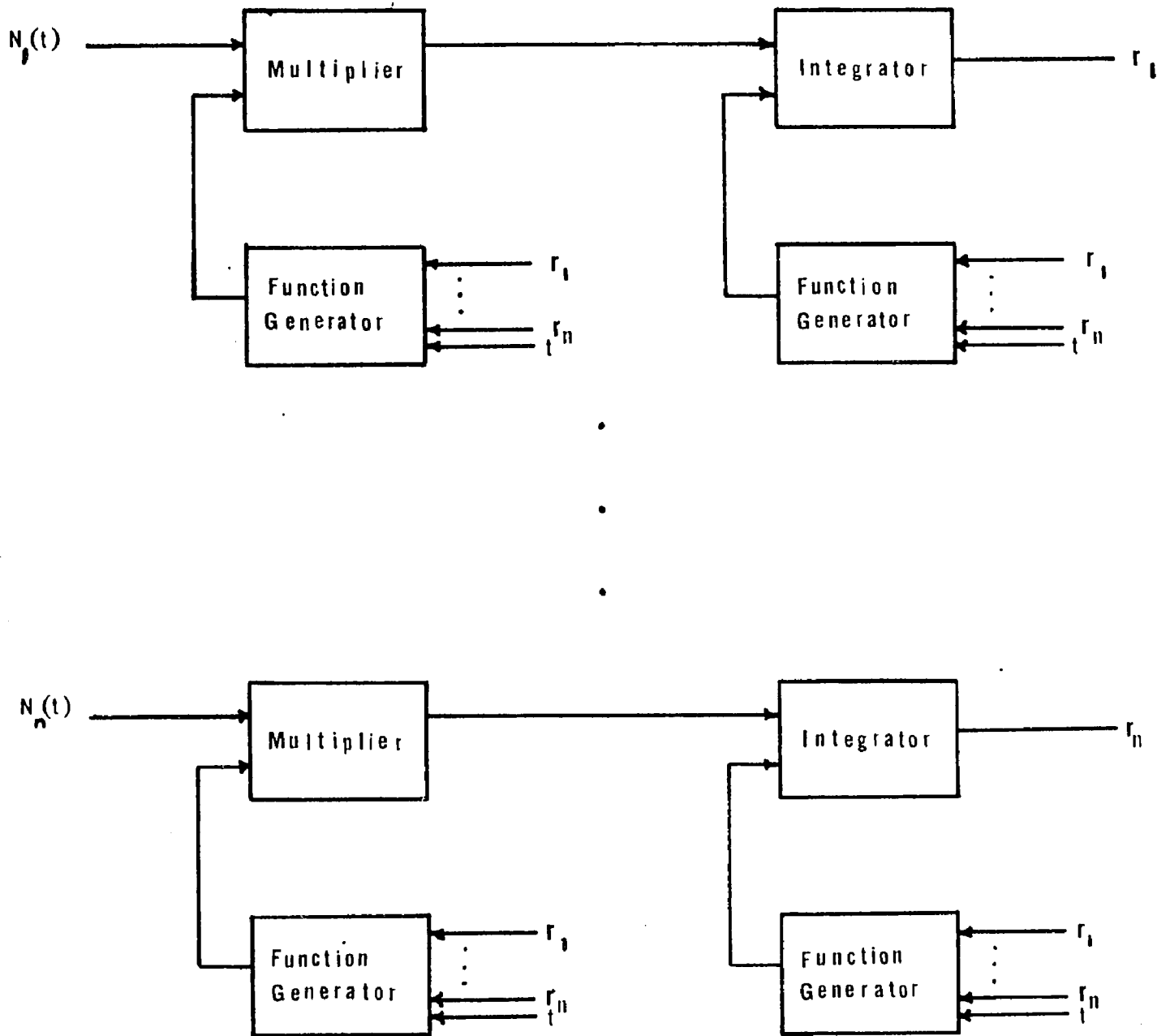


Figure 2.2 Block Diagram of the Generalized Langevin Equations

The conditional probability density function of a Markov process satisfies the Kolmogorov backward partial differential equation.

$$\begin{aligned} \frac{-\partial f}{\partial t_i} (r_k, t_k / r_i, t_i) &= \sum_{j=1}^n a_j(r_i, t_i) \frac{\partial f}{\partial (r_j)_i} \\ &+ b_j(r_i, t_i) \frac{\partial^2 f}{\partial (r_j)_i^2} \end{aligned} \quad (2.26)$$

Equation (2.26) rewritten for  $n=2$  and  $r_1=x$ ,  $r_2=y$  is presented in (2.27) to clarify the notation.

$$\frac{-\partial f}{\partial t_i} = a_1 \frac{\partial f}{\partial x_i} + a_2 \frac{\partial f}{\partial y_i} + b_1 \frac{\partial^2 f}{\partial x_i^2} + b_2 \frac{\partial^2 f}{\partial y_i^2} \quad (2.27)$$

The relationship between the terms in the Langevin equation (2.25) and the coefficients in (2.27) is as follows:

$$\begin{aligned} a_j(r_i, t_i) &= -A_j(r_i, t_i) \\ b_j(r_i, t_i) &= [B_j(r_i, t_i)]^2 D_j \end{aligned} \quad (2.28)$$

where the autocorrelation function of the noise sources is

$$E [N_j(t)N_j(t+\lambda)] = 2D_j \delta(\lambda). \quad (2.29)$$

$$E [N_j(t)N_k(t+\lambda)] = 0 \quad \text{for } j \neq k \quad (2.30)$$

## 2.4 Monte Carlo Solution of Parabolic Partial Differential Equations With Dirichlet-type Boundary Conditions

In this section, two Dirichlet problems will be presented along with the Monte Carlo techniques for approximating their solutions. The discussion will be limited to second-order parabolic partial differential equations; Sec. 2.5 will deal with elliptic partial differential equations. The mathematical justification for the Monte Carlo algorithms is found in Appendix C.

The parabolic partial differential equations to be treated will be divided into two classes.<sup>11</sup>

### Problem A

$U(\underline{r}_i, t_i)$  is to satisfy the partial differential equation

$$\frac{\partial U}{\partial t_i}(\underline{r}_i, t_i) = L_{\underline{r}_i, t_i} U(\underline{r}_i, t_i) \quad (2.31)$$

within a bounded region  $R$ , with boundary  $C$ , together with the boundary condition

$$U(\underline{r}_b, t_i) = U_b(\underline{r}_b) \quad (2.32)$$

and the initial condition

$$U(\underline{r}_i, 0) = U_i(\underline{r}_i) \quad (2.33)$$

where  $U_b(\underline{r}_b)$  is piecewise continuous and specified at all points of the boundary  $C$ . The operator  $L_{\underline{r}_i, t_i}$  is specified by

$$L_{\underline{r}_i, t_i} U(\underline{r}_i, t_i) = \sum_{k=1}^n a_k(\underline{r}_i, t_i) \frac{\partial U(\underline{r}_i, t_i)}{\partial (r_k)_i} + b_k(\underline{r}_i, t_i) \frac{\partial^2 U(\underline{r}_i, t_i)}{c(r_k)_i^2} \quad (2.34)$$

where  $(r_k)_i$  denotes the  $k^{\text{th}}$  coordinate of the initial position vector; the only difference between (2.34) and (2.26) is the replacement of the conditional density function  $f$  by  $U$ . A typical problem of this type is the heat conduction or the diffusion equation

$$\frac{\partial U}{\partial t}(\underline{r}, t) = \nabla^2 U(\underline{r}, t) \quad (2.35)$$

To obtain a solution to problem A at a point  $(\underline{r}_i, t_i)$ , a random walk is started at  $\underline{r}_i$  and is terminated either at a time  $t_b$  when the boundary  $C$  is reached, or at the time  $t=0$ ; the latter time being chosen because the random walk began at  $t_i$ , where  $t_i < t_b < 0$ . The random walk is generated by the analog computer solution of the generalized Langevin equations (2.24).

At the termination of each random walk, one tallies a number  $U_j$  equal to the boundary value  $U_b(\underline{r}_b)$ , or to the initial value  $U_i(\underline{r}_f)$ , depending upon whether the particle reached the absorbing boundary point  $\underline{r}_b$  or the interior point  $\underline{r}_f$ . The average over  $m$  random walks,

$$U_m(\underline{r}_i, t_i) = \frac{1}{m} \sum_{j=1}^m U_j \quad (2.36)$$

is a statistical estimate of  $U(\underline{r}_i, t_i)$ . The second type of parabolic partial differential equation which can be treated by Monte Carlo methods is given below.

Problem B

$U(\underline{r}_i, t_i)$  is to satisfy the partial differential equation

$$\frac{-\partial U(\underline{r}_i, t_i)}{\partial t_i} = L_{\underline{r}_i, t_i} U(\underline{r}_i, t_i) - d(\underline{r}_i, t_i) U(\underline{r}_i, t_i) \quad (2.37)$$

within a closed region R, with boundary C together with the boundary condition

$$U(\underline{r}_b, t_i) = U_b(\underline{r}_b) \quad (2.38)$$

and the initial condition

$$U(\underline{r}_i, 0) = U_i(\underline{r}_i) \quad (2.39)$$

where  $U_b(\underline{r}_b)$  is piecewise continuous and specified at all points of C. The first term of the right hand side of (2.39) is given in (2.34).

To obtain a solution to problem B at a point  $(\underline{r}_i, t_i)$ , a random walk similar to the one described for problem A is executed. At the end of the  $j^{\text{th}}$  walk, one tallies a number  $U_j$  equal to either the boundary value  $U_b(\underline{r}_b)$  or to the initial value  $U_i(\underline{r}_f)$ , depending upon

whether the particle reached the absorbing boundary point  $\underline{r}_b$  or the interior point  $\underline{r}_f$ . In addition, one calculates

$$v_j = \exp\left(\int_{t_i}^T -d[r(\tau), \tau] d\tau\right) \quad (2.40)$$

for each random walk, where  $T=0$  for walks terminating at  $t=0$  and  $T = t_b$  for walks terminating at a boundary point  $\underline{r}_b$  at time  $t_b$ . The Monte Carlo approximation to the solution of problem B is then the statistical average

$$U_m(\underline{r}_i, t_i) = \frac{1}{m} \sum_{j=1}^m v_j U_j \quad (2.41)$$

## 2.5 Monte Carlo Solution of Elliptic Partial Differential Equations with Dirichlet-type Boundary Conditions

Elliptic partial differential equations in a sense describe the steady state obtained by the solutions of the corresponding parabolic partial differential equations. In particular, Laplace's differential equation

$$\nabla^2 U = 0 \quad (2.42)$$

describes the steady state solution of the heat conduction or diffusion equation

$$\nabla^2 U = \frac{\partial U}{\partial t} \quad (2.43)$$

In this section, two additional problems are presented with Monte Carlo procedures for developing an approximate solution.

Problem C

$U(\underline{r}_i)$  is to satisfy the partial differential equation

$$L_{\underline{r}_i} U(\underline{r}_i) = 0 \quad (2.44)$$

within a bounded region  $R$  with boundary  $C$  together with the boundary condition

$$U(\underline{r}_b) = U_b(\underline{r}_b) \quad (2.45)$$

where  $U_b(\underline{r}_b)$  is piecewise continuous and specified at all points of  $C$ . The operator  $L_{\underline{r}_i}$  is defined by

$$L_{\underline{r}_i} U(\underline{r}_i) = \sum_{k=1}^n a_k(\underline{r}_i) \frac{\partial U(\underline{r}_i)}{\partial (r_k)_i} + b_k(\underline{r}_i) \frac{\partial^2 U(\underline{r}_i)}{\partial (r_k)_i^2} \quad (2.46)$$

To obtain a solution of problem C at a point  $\underline{r}_i$ , once again a random walk is executed; in the elliptic case, in contrast to the parabolic case, all walks are continued until the boundary is reached. At the end of the  $j^{\text{th}}$  walk, one tallies

$$U_j = U_b(\underline{r}_b) \quad (2.47)$$

The average value of  $U_j$

$$U_m(\underline{r}_i) = \frac{1}{m} \sum_{j=1}^m U_j \quad (2.48)$$

is an estimate of the solution of problem C at the point  $\underline{r}_i$ .

Problem D

$U(\underline{r}_i)$  is to satisfy the partial differential equation

$$L_{\underline{r}_i} U(\underline{r}_i) - d(\underline{r}_i)U(\underline{r}_i) = 0 \quad (2.49)$$

within a bounded region  $R$ , with boundary  $C$  together with the boundary condition

$$U(\underline{r}_b) = U_b(\underline{r}_b) \quad (2.50)$$

where  $U_b(\underline{r}_b)$  is piecewise continuous and specified on all points of  $C$ . The operator  $L_{\underline{r}_i}$  is defined by (2.46).

Monte Carlo solution of problem D is similar to that of problem C. A random walk is started at an interior point  $\underline{r}_i$  of  $R$  and continues until the absorbing boundary  $C$  is reached. At the end of the  $j^{\text{th}}$  walk from the point  $\underline{r}_i$ , one tallies

$$U_j = U_b(\underline{r}_b) \quad (2.51)$$

For each random walk, one calculates

$$v_j = \exp \left[ - \int_{t_i}^{t_b} d[\underline{r}(\tau)] d\tau \right] \quad (2.52)$$

where  $t_b$  denotes the time at which the boundary is reached. An estimate of the solution of problem D is the statistical average

$$U_m(\underline{r}_i) = \frac{1}{m} \sum_{j=1}^m v_j U_j \quad (2.53)$$

The mathematical justification of these results is presented in Appendix C.

## 2.6 Restrictions Limiting the Monte Carlo Method to Parabolic and Elliptic Second Order Partial Differential Equations

The results of Secs. 2.4 and 2.5 indicate how sampling procedures may be used to approximate the solution of parabolic and elliptical partial differential equations. To appreciate why the Monte Carlo method is not generally applicable to hyperbolic partial differential equations, consider the wave equation in one dimension:

$$\frac{\partial^2 U}{\partial t^2} = c^2 \frac{\partial^2 U}{\partial x^2} \quad (2.54)$$

The transformation

$$y = ict \quad (2.55)$$

reduces (2.54) to

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = 0 \quad (2.56)$$

which is Laplace's equation in two dimensions. To solve this equation by sampling procedures, it is necessary to know the behavior of the function on the boundary. If this were known, the Monte Carlo procedure could be used to estimate the behavior of the function within the bounded region. Assume that (2.54) describes the motion of a string fixed at both ends.

$$U(0,t) = 0 \quad (2.57)$$

$$U(d,t) = 0 \quad (2.58)$$

To estimate the solution of the elliptical partial differential equation (2.56), it is necessary to know the boundary conditions

$$U(x,t_1) \quad (2.59)$$

$$U(x,t_2) \quad (2.60)$$

i.e., the string displacement at two times,  $t_1$  and  $t_2$ . If these were known, it would be possible to calculate the displacement for any  $t_1 < t < t_2$ ,  $0 < x < d$ . In practice, however, one rarely knows the boundary conditions given by (2.59) and (2.60). In general, one is given the initial position  $U(x,0)$  and the initial velocity  $\frac{\partial U}{\partial t} \Big|_{t=0}$  of the string (Cauchy conditions), and is required to find the solution for a domain unlimited in time. The Monte Carlo method will then not apply.

## 2.7 Monte Carlo Solution of Nonhomogeneous Partial Differential Equations

A Monte Carlo procedure for estimating the solution of nonhomogeneous partial differential equations has been presented by Little.<sup>11</sup> Two problems are formed, one being a homogeneous time-independent boundary value problem, the other a time-dependent homogeneous boundary value problem. The solution of both problems are then combined to solve

the time-independent nonhomogeneous partial differential equation.

Problem E

$U(\underline{r}_i)$  is to satisfy the partial differential equation

$$L_{\underline{r}_i} U(\underline{r}_i) = -H(\underline{r}_i) \quad (2.61)$$

within a bounded region  $R$ , with boundary  $C$  together with the boundary condition

$$U(\underline{r}_b) = U_b(\underline{r}_b) \quad (2.62)$$

where  $U_b(\underline{r}_b)$  is piecewise continuous and defined on all points of  $C$ .

To generate a Monte Carlo solution to problem E, consider the two following component problems:

$$1. \quad L_{\underline{r}_i} U_1(\underline{r}_i) = 0 \quad (2.63)$$

within the region  $R$ , with the boundary condition on  $C$  given by

$$U_1(\underline{r}_b) = U_b(\underline{r}_b) \quad (2.64)$$

$$2. \quad \frac{-\partial U_2}{\partial t_i} = L_{\underline{r}_i} U_2(\underline{r}_i, t_i) \quad (2.65)$$

within the region  $R$ , with the boundary condition on  $C$  given by

$$U_2(\underline{r}_b, t_i) = 0 \quad (2.66)$$

The initial condition for this problem is

$$U_2(\underline{r}_i, 0) = H(\underline{r}_i) \quad (2.67)$$

The solution of problem E is

$$U(\underline{r}_i) = U_1(\underline{r}_i) + \int_{-\infty}^0 U_2(\underline{r}_i, t_i) dt_i \quad (2.68)$$

To verify that (2.68) is indeed a solution to the non-homogeneous partial differential equation (2.61), it is necessary to demonstrate that the expression satisfies both the differential equation and the boundary condition.

Operating on equation (2.68) with the operator  $L_{\underline{r}_i}$  yields

$$L_{\underline{r}_i} U(\underline{r}_i) = L_{\underline{r}_i} U_1(\underline{r}_i) + \int_{-\infty}^0 L_{\underline{r}_i} U_2(\underline{r}_i, t_i) dt_i \quad (2.69)$$

$$L_{\underline{r}_i} U(\underline{r}_i) = 0 + \int_{-\infty}^0 -\frac{\partial U_2}{\partial t_i} dt_i \quad (2.70)$$

$$L_{\underline{r}_i} U(\underline{r}_i) = -U_2(\underline{r}_i, 0) + U_2(\underline{r}_i, -\infty) \quad (2.71)$$

$U_2(\underline{r}_i, -\infty)$  equals zero, since the particle executing the random walk will reach the boundary after an infinite time with probability one.<sup>8</sup> Equation (2.71) reduces to

$$L_{\underline{r}_i} U(\underline{r}_i) = -U_2(\underline{r}_i, 0) = -H(\underline{r}_i) \quad (2.72)$$

which demonstrates that the composite function satisfies the partial differential equation (2.61). To show that  $U(\underline{x}_i)$  satisfies the boundary condition, form

$$U(\underline{x}_b) = U_1(\underline{x}_b) + \int_{-\infty}^0 U_2(\underline{x}_b, t_i) dt_i \quad (2.73)$$

From equations (2.64) and (2.76),

$$U(\underline{x}_b) = U_b(\underline{x}_b) \quad (2.74)$$

and hence the boundary condition is also satisfied.

The integration demanded by (2.68) is undesirable in any simple fast hybrid computation scheme. To circumvent this problem for a class of nonhomogeneous partial differential equations, the statistical behavior of the time interval of a random walk can be utilized in the case where  $H$  is a constant.

In Appendix B, it is shown that the expected time of a random walk  $T(\underline{x}_i, t_i)$  satisfies,

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{x}_i, t_i} \right] T(\underline{x}_i, t_i) = -1 \quad (2.75)$$

with the initial condition

$$T(\underline{x}_i, 0) = 0 \quad (2.76)$$

and the boundary condition

$$T(\underline{x}_b, t_i) = 0 \quad (2.77)$$

Consider the partial differential equation

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{x}_i} \right] U(\underline{x}_i, t_i) = -K \quad (2.78)$$

where  $U(\underline{x}_i, t_i)$  is defined in a region  $R$ , with boundary  $C$ .

The boundary condition on  $C$  is given by

$$U(\underline{x}_b, t_i) = U_b(\underline{x}_b) \quad (2.79)$$

and the initial condition is given by

$$U(\underline{x}_i, 0) = U_i(\underline{x}_i) \quad (2.80)$$

To develop a Monte Carlo algorithm, it is convenient to separate  $U(\underline{x}_i, t_i)$  into two functions,  $U_1(\underline{x}_i, t_i)$  and  $U_2(\underline{x}_i, t_i)$ .  $U_1(\underline{x}_i, t_i)$  satisfies the partial differential

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{x}_i, t_i} \right] U_1(\underline{x}_i, t_i) = 0 \quad (2.81)$$

with the piecewise continuous boundary condition on  $C$  given by

$$U_1(\underline{x}_b, t_i) = U_b(\underline{x}_b) \quad (2.82)$$

and the initial condition given by

$$U_1(\underline{x}_i, 0) = U_i(\underline{x}_i) \quad (2.83)$$

$U_2(\underline{x}_i, t_i)$  satisfies the partial differential equation

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{x}_i, t_i} \right] U_2(\underline{x}_i, t_i) = -K \quad (2.84)$$

with the boundary condition

$$U_2(\underline{r}_b, t_i) = 0 \quad (2.85)$$

and the initial condition

$$U_2(\underline{r}_i, 0) = 0 \quad (2.86)$$

It is easily verified that

$$U(\underline{r}_i, t_i) = U_1(\underline{r}_i, t_i) + U_2(\underline{r}_i, t_i) \quad (2.87)$$

The techniques for the Monte Carlo solution of  $U_1$  have been discussed in section 2.4, and no additional comments are necessary at this point.

To develop a Monte Carlo procedure for the estimation of the solution of (2.84), it is of interest to examine (2.75) which describes the behavior of the average time of a random walk. If the operator  $L$  is linear,

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right] KT(\underline{r}_i, t_i) = -K \quad (2.88)$$

Multiplying the time of the random walk by a constant  $K$  and denoting the product by  $V$ , equation (2.88) indicates that  $V$  satisfies the partial differential equation

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right] V(\underline{r}_i, t_i) = -K \quad (2.89)$$

with the boundary condition

$$V(\underline{r}_b, t_i) = 0 \quad (2.90)$$

and the initial condition

$$V(\underline{r}_i, 0) = 0 \quad (2.91)$$

Examination of equations (2.84) and (2.89) show that

$$V(\underline{r}_i, t_i) = U_2(\underline{r}_i, t_i) \quad (2.92)$$

To obtain a Monte Carlo estimate for the solution of (2.78), it is necessary to treat the partial differential equation as if it were a homogeneous partial differential equation, program the computer to solve the appropriate Langevin equations, and combine the homogeneous solution with a voltage proportional to the average time of a random walk.

## 2.8 Extension of the Monte Carlo Technique to the Case where $\frac{\partial U}{\partial n} = 0$ Over a Portion of the Boundary

Many physical problems involving partial differential equations have boundary conditions other than those of the Dirichlet type. In this and in the following section, some Monte Carlo computing procedures for such problems will be derived; although by no means general, these problems occur sufficiently often to be of considerable practical interest.

### Problem F

$U(\underline{r}_i, t_i)$  is to satisfy either the diffusion equation

$$\nabla^2 U(\underline{x}_i, t_i) = \frac{\partial U}{\partial t_i} \quad (2.93)$$

or Laplace's equation

$$\nabla^2 U(\underline{x}_i) = 0 \quad (2.94)$$

in a bounded region  $R$ , with boundary  $C$  composed of two parts,  $C_1$  and  $C_2$ . The value of  $U$  is specified on  $C_1$  and the normal derivative of  $U$  on  $C_2$  is zero:

$$U(\underline{x}_b) \Big|_{C_1} = U_{C_1}(\underline{x}_b) \quad (2.95)$$

$$\frac{\partial U}{\partial n} \Big|_{C_2} = 0 \quad (2.96)$$

Physically, the condition on the normal derivative represents a totally reflecting boundary in the heat-conduction equation.

To obtain an estimate of the solution for either (2.93) or (2.94), a random walk is started at a point  $\underline{x}_i$  and continues for a time  $t_i$  in the case of (2.93), or until the boundary is reached for (2.94). If boundary  $C_1$  is reached before  $C_2$ , the value of the boundary function  $U_{C_1}(\underline{x}_b)$  is tallied as in the Dirichlet problem. If the  $C_2$  portion of the boundary is encountered during the walk, then the particle executing the walk is reflected from the boundary, and the walk continues until the particle either arrives at the absorbing boundary  $C_1$  or until time  $t = 0$

(depending upon whether a solution is sought for the elliptic or parabolic partial differential equation).

To justify the above procedure, consider Fig. 2.3 which shows an enlarged view of a portion of  $C_2$  in addition to an arbitrary curve  $C_a$  and its reflection about  $C_2$ ,  $C_a'$ . If the value of the function  $U$  along  $C_a$  were known, then the particle executing the random walk would be absorbed upon reaching this curve, with the appropriate boundary value tallied. If the values of  $U$  on  $C_a$  are assigned to the reflected points on  $C_a'$ , and in the case of the diffusion equation, the initial conditions in the appended region are assigned the identical values of those of the mirror-image points, then from symmetry considerations, the actual boundary  $C_2$  would indeed satisfy the condition

$$\left. \frac{\partial U}{\partial n} \right|_{C_2} = 0 \quad (2.97)$$

Joining the collection of curves  $C_a'$  which cover the reflecting boundary  $C_2$  together, and attaching the endpoints of this collection of curves to  $C_1$ , a new enlarged region  $R'$  which includes  $R$  (see Fig. 2.4) is formed. It is assumed that the diffusion equation or Laplace's equation is valid in  $R'$ . The value of  $U$  is specified to be  $U_{C_1}(\vec{r}_b)$  on  $C_1$ , and the value of  $U$  along the collection of curves  $C_a'$  is assumed to be as described above. The solution to the diffusion or Laplace problem defined in  $R'$  will then be

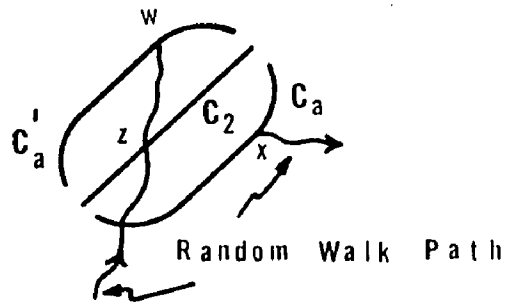


Figure 2.3 Portion of a Reflecting Boundary

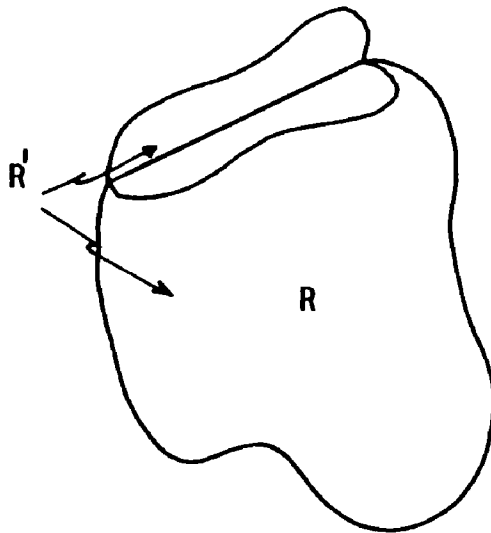


Figure 2.4 Region  $R$  and the Augmented Region  $R'$

identical to the solution of the original problem defined in the region common to both problems, namely,  $R'$ .

Consider a random walk in which  $C_2$  is crossed on the way towards  $C_a'$ , as shown in Fig. 2.3. When the point  $w$  is reached, the value of  $U(w)$  would normally be recorded; while this value is not known, by construction, the point  $x$  lying on curve  $C_a$  has an identical value

$$U(x) = U(w) \quad (2.98)$$

and the random walk resumes from point  $x$ . If, during the course of a random walk in the Monte Carlo solution of the diffusion or Laplace equation, a point is reached where the solution is known to be equal to that at another point in the region under consideration, the walk may be continued from the second point since the expected (absorbed) value is the same for both points which justifies resuming the random walk from point  $x$ .

Because curve  $C_a$  is arbitrary, the continuation of the walk from point  $x$  is equivalent to a reflection at point  $z$  lying on  $C_2$ . The boundary condition

$$\left. \frac{\partial U}{\partial n} \right|_{C_2} = 0 \quad (2.99)$$

dictates that a reflection about the vector, normal to the point of impact, must be made when the portion of the boundary  $C_2$  is reached during the course of a random walk.

## 2.9 Monte Carlo Procedure for the Case Where $\frac{\partial U}{\partial n} = a$ Over a Straight Line Portion of the Boundary

In this section we will consider the following problem:

### Problem G

$U(\underline{r}_i, t_i)$  is to satisfy either the diffusion equation

$$\nabla^2 U(\underline{r}_i, t_i) = \frac{\partial U}{\partial t_i} \quad (2.100)$$

or Laplace's equation

$$\nabla^2 U(\underline{r}_i) = 0 \quad (2.101)$$

in a bounded region  $R$ , with boundary  $C$  composed of two parts,  $C_1$  and  $C_2$  as shown in Fig. 2.5. The value of  $U$  is specified on  $C_1$  and the normal derivative of  $U$  is constant on  $C_2$ :

$$U(\underline{r}_b, t_i) \Big|_{C_1} = U_{C_1}(\underline{r}_b) \quad (2.102)$$

$$\frac{\partial U}{\partial n} \Big|_{C_2} = a \quad (2.103)$$

Two procedures will be presented to solve this problem by Monte Carlo techniques.

### Procedure 1

A new function  $V(\underline{r})$  is defined by

$$V(\underline{r}) = U(\underline{r}) - ay \quad (2.104)$$

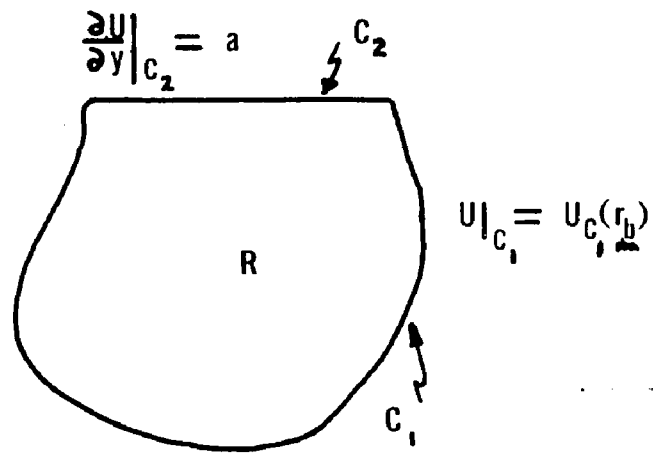


Figure 2.5 Diagram Pertaining to the Problem of Section 2.9

Operating on (2.104) with the Laplacian operator yields

$$\nabla^2 V(\underline{r}) = \nabla^2 U(\underline{r}) - \nabla^2 (ay) = 0 \quad (2.105)$$

Hence  $V(\underline{r})$  satisfies the Laplace equation in region  $R$ , with boundary  $C$ . If  $U(\underline{r})$  originally satisfied the diffusion equation, it is easy to show that  $V(\underline{r})$  also satisfies the diffusion equation. The boundary conditions on  $V(\underline{r})$  are

$$V(\underline{r}) \Big|_{C_1} = U(\underline{r}) \Big|_{C_1} - (ay) \Big|_{C_1} = U_{C_1}(\underline{r}_b) - ay_b \quad (2.106)$$

$$\frac{\partial V(\underline{r})}{\partial y} \Big|_{C_2} = \frac{\partial U(\underline{r})}{\partial y} \Big|_{C_2} - \frac{\partial (ay)}{\partial y} \Big|_{C_2} = 0 \quad (2.107)$$

The results of the previous section can be used to develop a Monte Carlo procedure which will generate an estimate for the solution of the problem involving  $V(\underline{r})$ : a particle starts a random walk at a point  $\underline{r}_i$ , is then reflected from  $C_2$  and finally absorbed on  $C_1$  where the boundary value given by (2.106) is tallied. To recover  $U(\underline{r}_i)$  from a knowledge of  $V(\underline{r}_i)$ , it is simply necessary to add  $ay_i$  to the estimate of  $V(\underline{r}_i)$ , because from (2.104)

$$U(\underline{r}_i) = V(\underline{r}_i) + ay_i \quad (2.108)$$

Rather than add  $ay_i$  on to the estimate of  $V(\underline{r}_i)$ , it is perhaps more suggestive to add  $ay_i$  to each tally of the random walk procedure in the calculation of  $V(\underline{r}_i)$ .

The preceding discussion dictates the Monte Carlo procedure to be followed when a boundary condition of the type described by (2.103) is encountered. A random walk is started at a point  $\underline{r}_i$  where the solution is sought. During the course of the walk, the particle is reflected at boundary  $C_2$  and is absorbed at boundary  $C_1$ . At the termination of each random walk, the quantity

$$U_{C_1}(\underline{r}_b) - a(y_b - y_i) \quad (2.109)$$

is tallied. After  $n$  walks have been tallied, the total score is calculated and the result divided by  $n$ .

### Procedure 2

A second Monte Carlo technique for the problem posed at the beginning of this section is to separate  $U(\underline{r})$  into two functions:

$$U(\underline{r}) = U_1(\underline{r}) + U_2(\underline{r}) \quad (2.110)$$

Both  $U_1(\underline{r})$  and  $U_2(\underline{r})$  are to satisfy either the diffusion equation or Laplace's equation in region  $R$  with boundary  $C$ . The boundary conditions on  $U(\underline{r})$  are repeated below for convenience:

$$U(\underline{r}_b, t_i) \Big|_{C_1} = U_{C_1}(\underline{r}_b) \quad (2.102)$$

$$\frac{\partial U}{\partial n} \Big|_{C_2} = a \quad (2.103)$$

The appropriate boundary conditions on  $U_1(\underline{r})$  and  $U_2(\underline{r})$  are given in equations (2.111) through (2.114) and are shown in Figs. 2.6a and 2.6b.

$$U_1(\underline{r}) \Big|_{C_1} = ay \quad (2.111)$$

$$\frac{\partial U_1}{\partial n} \Big|_{C_2} = a \quad (2.112)$$

$$U_2(\underline{r}) \Big|_{C_1} = U_{C_1}(\underline{r}_b) - ay \quad (2.113)$$

$$\frac{\partial U_2}{\partial n} \Big|_{C_2} = 0 \quad (2.114)$$

From (2.110), it is easily verified that  $U_1(\underline{r}) + U_2(\underline{r})$  satisfies the boundary conditions on  $U(\underline{r})$ .

The solution of the problem involving  $U_1(\underline{r})$  is

$$U_1(\underline{r}) = ay \quad (2.115)$$

since it satisfies both the diffusion equation, and Laplace's equation, and the appropriate boundary conditions. The  $U_1(\underline{r})$  problem shown in Fig. 2.6a can be replaced by the Dirichlet problem shown in Fig. 2.7a, since both problems have identical solutions. The problem involving  $U_2(\underline{r})$ , shown in Fig. 2.6b, is redrawn in Fig. 2.7b.

To obtain the Monte Carlo estimate of the solution  $U(\underline{r})$ , it is necessary to develop the procedure which will solve the component problems shown in Figs. 2.7a and 2.7b.

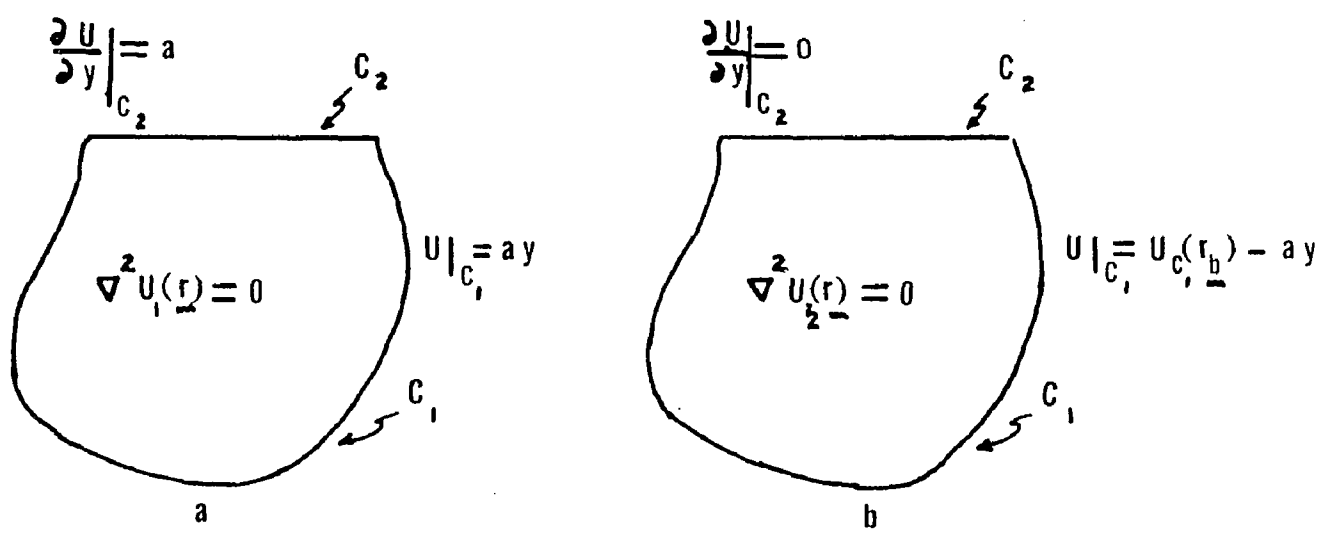


Figure 2.6 Decomposition of the Problem Shown in Fig. 2.5 for Use in Procedure 2

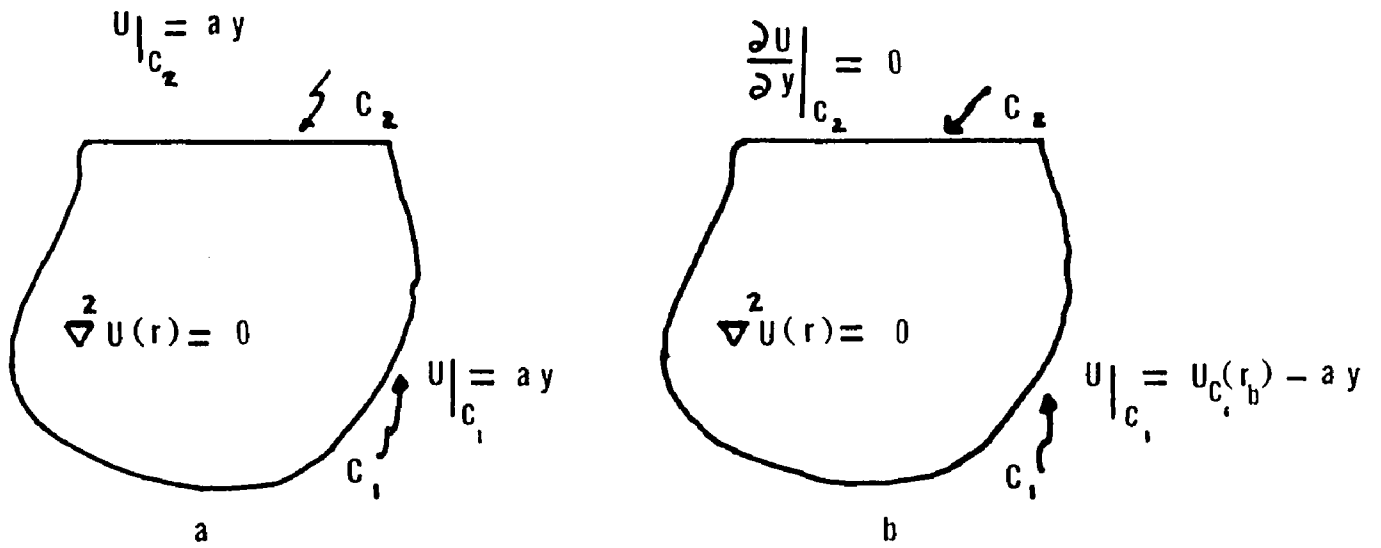


Figure 2.7 Modification of the Problem of Fig. 2.6

$U_1(\underline{r})$  and  $U_2(\underline{r})$  are obtained simultaneously by employing identical random walks for each problem.

Consider a random walk originating from a point  $\underline{r}_i$  within a closed region  $R$ . Two possibilities exist:

Case 1. The particle executing the random walk reaches  $C_1$  before  $C_2$ .

Case 2. The particle executing the random walk reaches  $C_2$  before  $C_1$ .

Because identical random walks are used for  $U_1(\underline{r})$  and  $U_2(\underline{r})$ , random walks belonging to case 1 will generate the following tallies:

Tallies for Case 1

$$U_1(\text{tally}) = ay_b \quad (2.116)$$

$$U_2(\text{tally}) = U_{C_1}(\underline{r}_b) - ay_b \quad (2.117)$$

Adding these together

$$U_1(\text{tally}) + U_2(\text{tally}) = U_{C_1}(\underline{r}_b) \quad (2.118)$$

For walks originating at  $\underline{r}_i$  and belonging to case 2, the following tallies are recorded:

Tallies for Case 2

$$U_1(\text{tally}) = ay_{\underline{r}_1} \quad (2.119)$$

$$U_2(\text{tally}) = U_{C_1}(\underline{r}_b) - ay_b \quad (2.120)$$

where  $y_{r_{m1}}$  denotes the point on  $C_2$  where the first reflection of the particle in the  $U_2(r)$  problem occurs, and where the particle is absorbed in the  $U_1(r)$  problem. Addition of the tallies for case 2 yields

$$U_1(\text{tally}) + U_2(\text{tally}) = U_{C_1}(r_b) + a(y_{r_{m1}} - y_b) \quad (2.121)$$

Equations (2.118) and (2.121) dictate the Monte Carlo procedure to be followed to obtain an estimate of the solution of the diffusion equation or of Laplace's equation with the boundary conditions as given by (2.102) and (2.103). To estimate  $U(r_i)$ , a random walk is started at  $r_i$  and, if the portion of the boundary where the normal derivative is specified is not encountered before the particle reaches  $C_1$ , then the boundary value  $U_{C_1}(r_b)$  is tallied. If  $C_2$  is reached before  $C_1$ , the final absorption value on  $C_1$  is tallied along with the product of the normal derivative  $a$  and the difference in displacement between the reflecting and absorbing points in the direction of the normal derivative, i.e.,  $a(y_{r_{m1}} - y_b)$ .

#### 2.10 Monte Carlo Procedure for the Case Where the Normal Derivative is Constant Along Orthogonal Segments on the Boundary

The results of the preceding section can be extended to cover the following problem.

Problem H

$U(\underline{r}_i)$  is to satisfy Laplace's equation

$$\nabla^2 U(\underline{r}_i) = 0 \quad (2.122)$$

in a bounded region  $R$ , with boundary  $C$  composed of three parts,  $C_1$ ,  $C_2$ , and  $C_3$  as shown in Fig. 2.8. The boundary conditions on  $U(\underline{r}_i)$  are

$$\left. \frac{\partial U}{\partial y} \right|_{C_1} = a \quad (2.123)$$

$$U(\underline{r}_i) \Big|_{C_2} = U_{C_2}(\underline{r}_b) \quad (2.124)$$

$$\left. \frac{\partial U}{\partial x} \right|_{C_3} = b \quad (2.125)$$

To develop a procedure for the Monte Carlo estimate of the solution of the partial differential equation with the above boundary conditions, it is convenient to define two functions,  $U_1(\underline{r}_i)$  and  $U_2(\underline{r}_i)$ , both satisfying Laplace's equation in the same region  $R$ . The boundary conditions for  $U_1(\underline{r}_i)$  are given in (2.126) through (2.128), and the boundary conditions for  $U_2(\underline{r}_i)$  are given in (2.129) through (2.131).

$$\left. \frac{\partial U_1}{\partial y} \right|_{C_1} = a \quad (2.126)$$

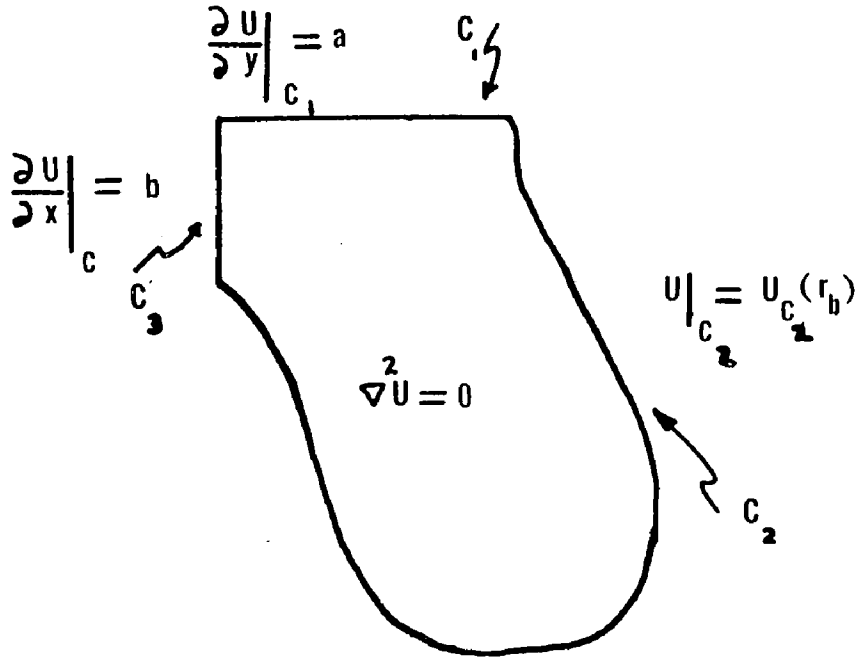


Figure 2.8 Diagram Pertaining to the Problem of Section 2.10

$$U_1(\underline{r}_i) \Big|_{C_2} = bx + ay \quad (2.127)$$

$$\frac{\partial U_1}{\partial x} \Big|_{C_3} = b \quad (2.128)$$

$$\frac{\partial U_2}{\partial y} \Big|_{C_1} = 0 \quad (2.129)$$

$$U_2(\underline{r}_i) \Big|_{C_2} = U_{C_2}(\underline{r}_b) - bx - ay \quad (2.130)$$

$$\frac{\partial U_2}{\partial x} \Big|_{C_3} = 0 \quad (2.131)$$

The solution of the  $U_1(\underline{r}_i)$  problem is simply

$$U_1(\underline{r}_i) = bx + ay \quad (2.132)$$

This solution is easily verified by substitution into Laplace's equation (2.122) and boundary conditions (2.126), (2.127), and (2.128).

The  $U_1(\underline{r}_i)$  problem shown in Fig. 2.9a may be replaced by a problem with Dirichlet conditions, that is, one for which all the boundary values of the function are specified. In Fig. 2.10a, this change is made with the appropriate boundary condition

$$U_1(\underline{r}_i) \Big|_C = bx + ay \quad (2.133)$$

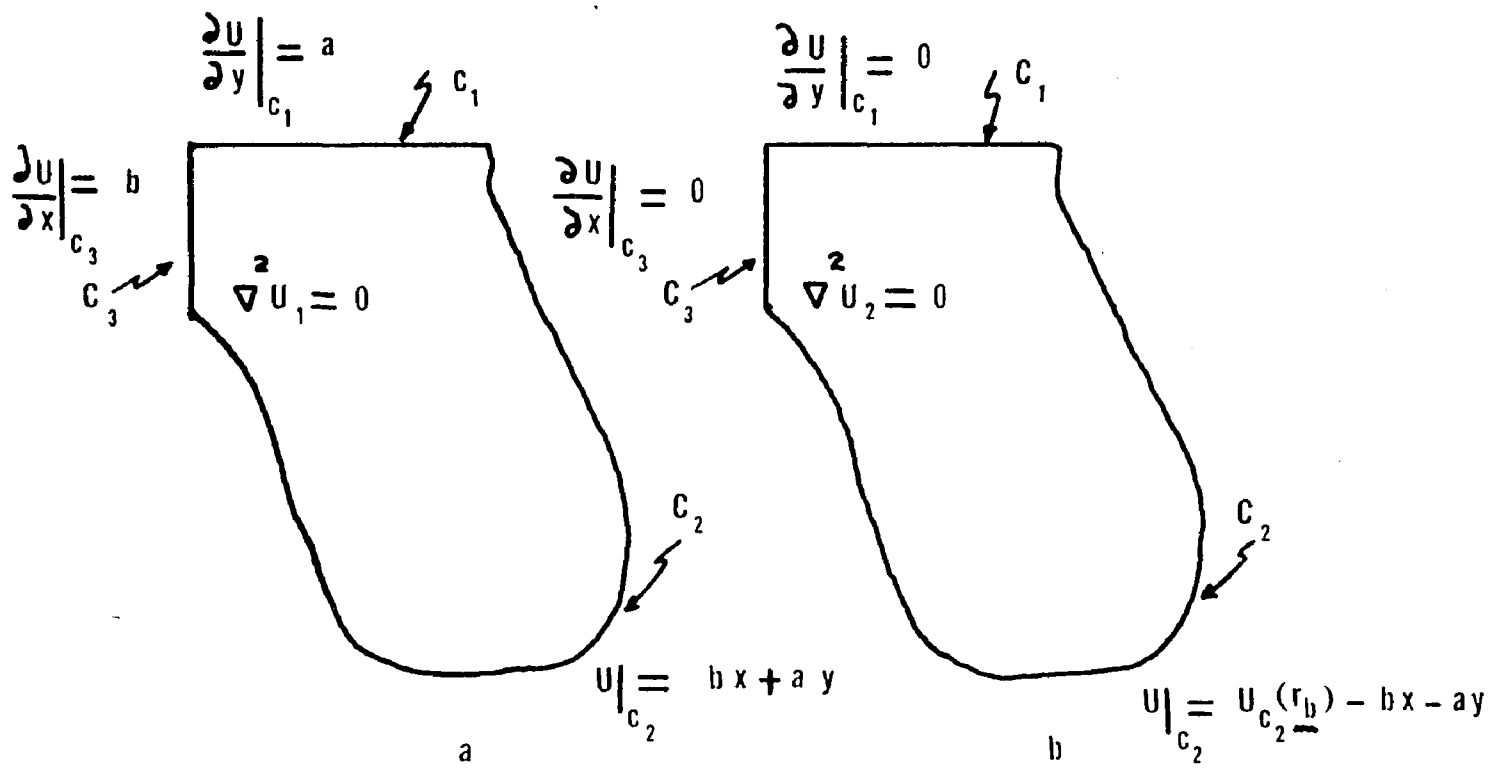


Figure 2.9 Decomposition of the Problem Shown in Fig. 2.8

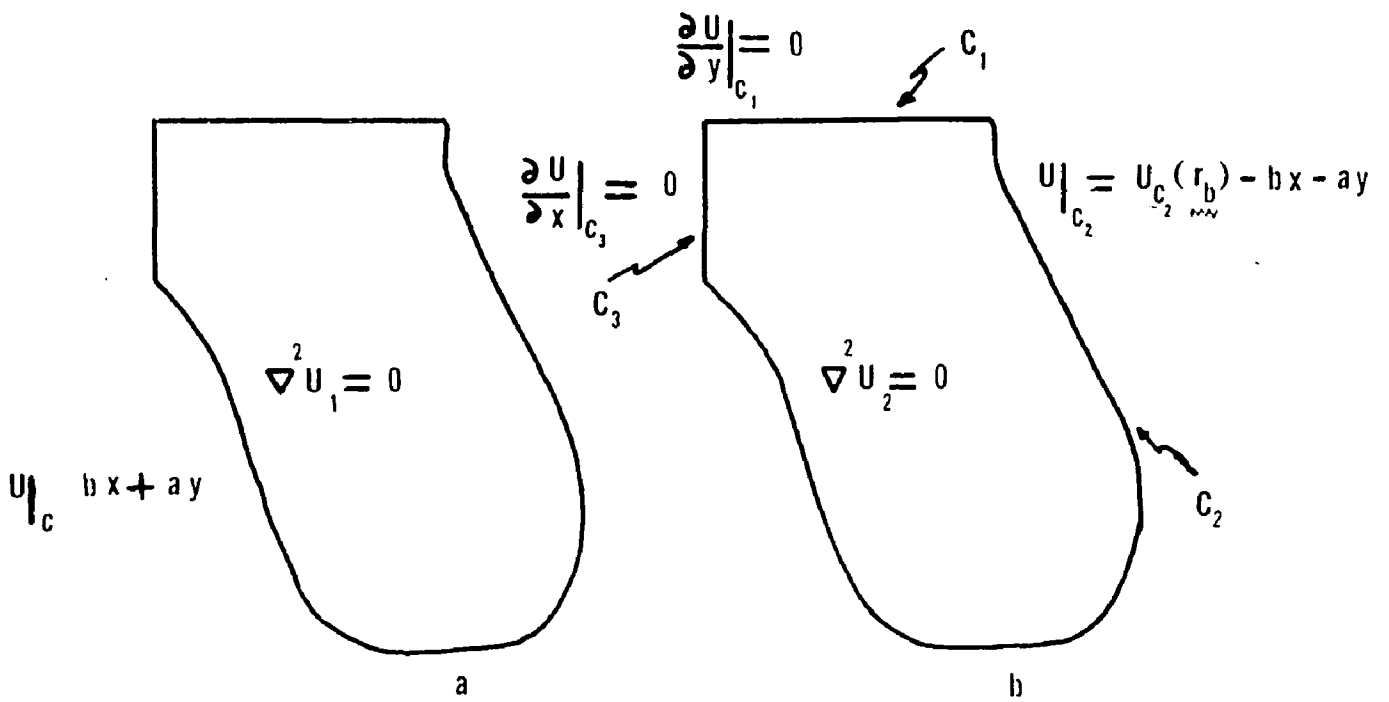


Figure 2.10 Modification of the Problem of Fig. 2.9

Fig. 2.10b is identical to Fig. 2.9b and is redrawn for convenience in the succeeding discussion.

To find  $U(\underline{r}_i)$ , it is sufficient to find the solutions for the component problems since

$$U(\underline{r}_i) = U_1(\underline{r}_i) + U_2(\underline{r}_i) \quad (2.134)$$

As in the previous section, the same random walk is used to obtain estimates of the solutions. For the problem involving  $U_1(\underline{r}_i)$ , the entire boundary  $C$  is absorbing, while in the  $U_2(\underline{r}_i)$  problem,  $C_1$  and  $C_3$  are reflecting boundaries and  $C_2$  is an absorbing boundary. By using the same random walk for both problems, a procedure can be formulated whereby the original problem can be solved without actually solving the component problems. To categorize the various situations which can occur during the course of a random walk, three cases will be considered which will exhaust the set of random paths.

Case 1. The particle executing the random walk starts at  $\underline{r}_i$  and reaches  $C_2$  before reaching any other part of the boundary first. The tallies for this case are the following:

$$U_1(\text{tally}) = bx_b + ay_b \quad (2.135)$$

$$U_2(\text{tally}) = U_{C_2}(\underline{r}_b) - bx_b - ay_b \quad (2.136)$$

$$U_1(\text{tally}) + U_2(\text{tally}) = U_{C_2}(\underline{r}_b) \quad (2.137)$$

where  $\underline{r}_b$  denotes the point of absorption on  $C_2$ .

Case 2. The particle reaches  $C_1$  initially, is absorbed in the  $U_1(\underline{r}_i)$  problem, is reflected in the  $U_2(\underline{r}_i)$  problem, and is eventually absorbed on  $C_2$  with the possibility of an arbitrary number of reflections allowed on  $C_1$  and  $C_3$ . The tallies for this case are given below:

$$U_1(\text{tally}) = bx_{\underline{r}_1} + ay_{C_1} \quad (2.138)$$

$$U_2(\text{tally}) = U_{C_2}(\underline{r}_b) - bx_b - ay_b \quad (2.139)$$

$$U_1(\text{tally}) + U_2(\text{tally}) = U_{C_2}(\underline{r}_b) + b(x_{\underline{r}_1} - x_b) + a(y_{C_1} - y_b) \quad (2.140)$$

where  $x_{\underline{r}_1}$  is the x coordinate of the point on  $C_1$  where the absorption occurs in the  $U_1(\underline{r}_i)$  problem,  $y_{C_1}$  is the y dimension of  $C_1$  (see Fig. 2.10), and  $\underline{r}_b$  denotes the final absorption point on  $C_2$ .

Case 3. The particle reaches  $C_3$  initially, is absorbed in the  $U_1(\underline{r}_i)$  problem, reflects in the  $U_2(\underline{r}_i)$  problem, and is eventually absorbed on  $C_2$  with the possibility of an arbitrary number of reflections allowed on  $C_1$  and  $C_3$ . The tallies for this case are given below:

$$U_1(\text{tally}) = bx_{C_3} + ay_{\underline{r}_1} \quad (2.141)$$

$$U_2(\text{tally}) = U_{C_2}(r_b) - bx_b - ay_b \quad (2.142)$$

$$U_1(\text{tally}) + U_2(\text{tally}) = U_{C_2}(r_b) + b(x_{C_2} - x_b) + a(y_{r_1} - y_b) \quad (2.143)$$

where  $y_{r_1}$  is the y coordinate of the point on  $C_3$  where the absorption occurs in the  $U_1(r_i)$  problem,  $x_{C_3}$  is the x dimension of  $C_3$  (see Fig. 2.10), and  $r_b$  denotes the final absorption point on  $C_2$ .

From the three cases considered above, a Monte Carlo procedure for obtaining an estimate of the solution of the problem can be formulated. A random walk is started at  $r_i$ ; if the particle executing the walk reaches  $C_2$  before any other part of the boundary, the value of the function at the point of absorption on  $C_2$  is tallied. If, however, the particle reaches  $C_1$  initially, then the particle is reflected, the random walk continues, and  $C_1$  and  $C_3$  are considered simple reflecting surfaces for the remainder of the random walk. Eventually  $C_2$  is reached and the random walk terminates. For this type of random walk, the tally is given by (2.140). Similarly, when the particle reaches  $C_3$  initially and thence goes to  $C_2$  with an arbitrary number of reflections occurring on  $C_1$  and  $C_3$ , the tally is given by (2.143).

From the preceding discussion, it is possible to extend this result to an n dimensional problem where the

boundary conditions are such that the normal derivative is constant on orthogonal line segments.

## CHAPTER 3

### ERROR CONSIDERATIONS IN MONTE CARLO COMPUTATIONS

#### 3.1 Introduction

Errors in any Monte Carlo procedure for the estimate of the solution of a partial differential equation fall into two categories: errors due to equipment limitations and those due to statistical fluctuations. Aside from calibration inaccuracies and errors in nonlinear computing elements, the most serious equipment limitation errors are due to amplifier drift, finite integrator bandwidth, compute-run time limits, and nonideal noise sources. The statistical errors are inherent in any estimation of expected values from finite samples.

#### 3.2 Amplifier Drift Errors

The effects of imperfections in hybrid computer equipment can be examined most realistically by a careful consideration of individual problems. A representative problem will be examined and solution errors resulting from equipment limitations will be considered for this problem.

Laplace's equation in one dimension is quite representative of the problems that one can solve by the Monte Carlo technique and, in addition, perhaps allows the greatest insight into the Monte Carlo sampling process.

The mathematics involved in this problem is quite tractable and no nonlinear equipment is used to implement the Langevin equations. The problem used to demonstrate the effects of drift is the following:

Example Problem

$U(x)$  satisfies Laplace's equation in one dimension.

$$\frac{d^2U(x)}{dx^2} = 0 \quad (3.1)$$

with the boundary conditions

$$U(0) = 0 \quad (3.2)$$

$$U(1) = 1 \quad (3.3)$$

The solution to this problem is readily seen to be

$$U(x) = x \quad (3.4)$$

Integrator current and voltage drift<sup>1</sup> can be taken into account by a fictitious voltage generator of magnitude  $K$  which is added along with the noise source into the integrator (Fig. 3.1). The stochastic differential equation actually being solved is then

$$\frac{dy}{dt} = n(t) + K \quad (3.5)$$

The elliptical partial differential equation corresponding to the above stochastic differential equation is,

$$D \frac{d^2U(x)}{dx^2} + K \frac{dU(x)}{dx} = 0 \quad (3.6)$$

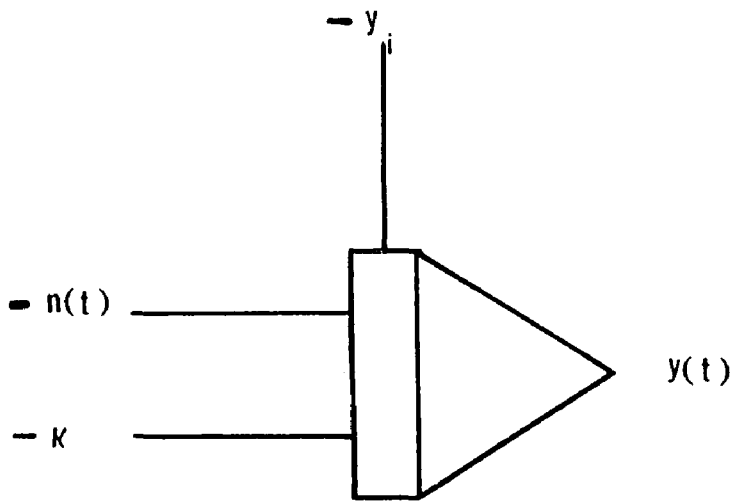


Figure 3.1 Integration of Noise Signal Showing Integrator Drift

where  $2D\delta(\tau)$  is the autocorrelation function of the ideal wideband noise source. The solution of (3.6) is

$$U_a(x) = \frac{10}{e^{-10K/D} - e^{10K/D}} 2e^{-K/Dx} - e^{10K/D} - e^{-10K/D} \quad (3.7)$$

Assuming that the offset term  $K$  is small compared with the noise power spectral density  $2D$ , i.e.,

$$K/D \ll 1 \quad (3.8)$$

one can approximate  $U_a(x)$  by expanding the exponential terms:

$$U_a(x) \approx x(1 - \frac{Kx}{2D}) \quad (3.9)$$

The error introduced into the solution by the presence of this offset term  $K$  is

$$e(x) = U(x) - U_a(x) = \frac{Kx^2}{2D} \quad (3.10)$$

To keep the error under 1% of 10 volts (100 millivolts) with unity gain integrators, it is necessary to enforce the condition

$$K \leq 10^{-3} 2D \quad (3.11)$$

If the integrator is operating with gain  $G$ , then the constraint on the allowable offset voltage is

$$K \leq 10^{-3} 2D G \quad (3.12)$$

Typical values for  $2D$  and  $G$  in the Monte Carlo computations performed with ASTRAC II are

$$2D = 30 \cdot 10^{-6} \text{ volts}^2/\text{cps}$$

$$G = 10^5 \tag{3.13}$$

hence

$$K \leq 30 \cdot 10^{-4} \tag{3.14}$$

The amplifiers used in ASTRAC II are Burr Brown type 1607a with

$$K \leq 20 \cdot 10^{-5}$$

which is well within the bounds of condition (3.14).

### 3.3 Errors Due to Finite Noise-source and Integrator Bandwidths

The application of white Gaussian noise signals into an electrical system is the basis for the Monte Carlo estimation of the solution of partial differential equations. In practice, any noise source will be bandlimited in addition to the bandwidth limitations peculiar to all computing elements. This finite noise-bandwidth can be accounted for by passing white noise through a simple bandlimiting filter and treating the filter output as the input noise to the system.

The random noise generator used in this study and described in Appendix D, produces a binary noise waveform whose autocorrelation function is shown in Fig. 3.2. The power spectral density corresponding to this autocorrelation function is<sup>15</sup>

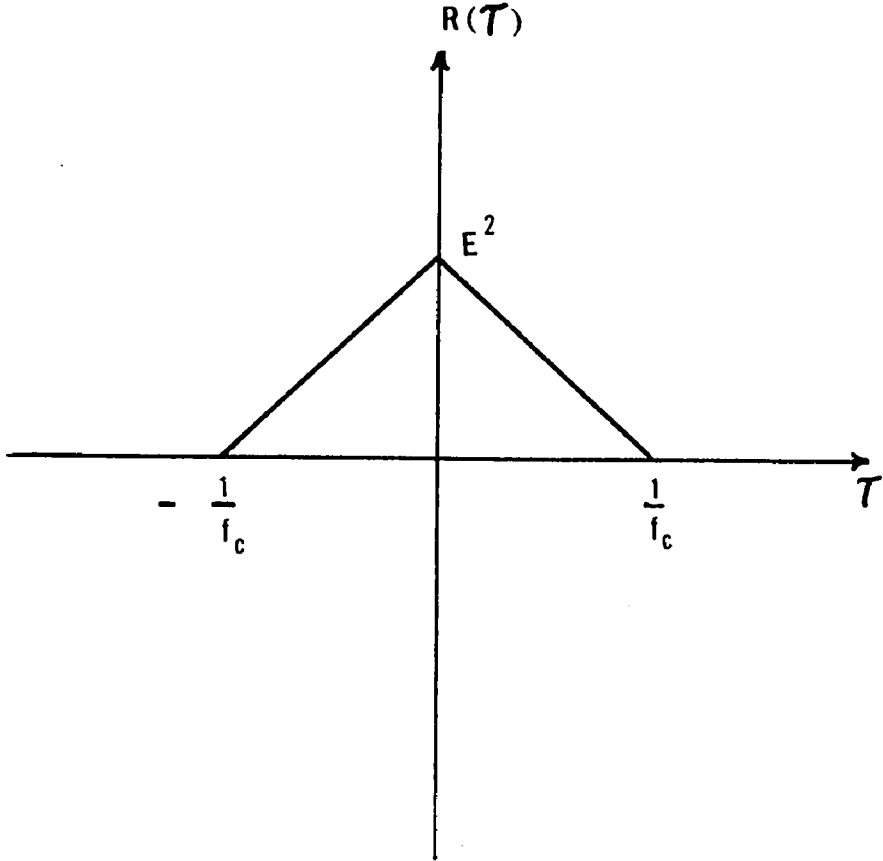


Figure 3.2 Autocorrelation Function of Binary-noise Waveform

$$S(\omega) = \frac{2f_c E^2 [1 - \cos(\omega/f_c)]}{\omega^2} \quad (3.16)$$

One can approximate the power spectral density of the binary signal by passing white noise through a low pass filter with a transfer function

$$H(s) = \frac{\omega_0}{s + \omega_0} \quad (3.17)$$

where  $\omega_0 = \pi f_c$ ,  $f_c$  is the pulse rate of the binary signal.

Integrator bandwidth limitations can be simulated by cascading a low pass filter with an ideal integrator<sup>1</sup> where the bandwidth of the low pass filter is proportional to the zero decibel frequency of the integrating operational amplifier. The amplifiers in ASTRAC II have a zero decibel frequency of 30 Mcps. (Burr Brown type 1607a); the noise source which has a 500 Kcps. bandwidth is therefore the significant bandlimiting device in the Monte Carlo procedure.

To study the effect of noise bandwidth limitations, it is convenient to consider the Monte Carlo solution of the diffusion equation in one dimension as a representative problem. The block diagram for the simulation of this problem taking the noise bandwidth into account is shown in Fig. 3.3. Referring to Sec. 2.2, the mean value function  $m(t, t_1)$  of the system shown in Fig. 3.3 is

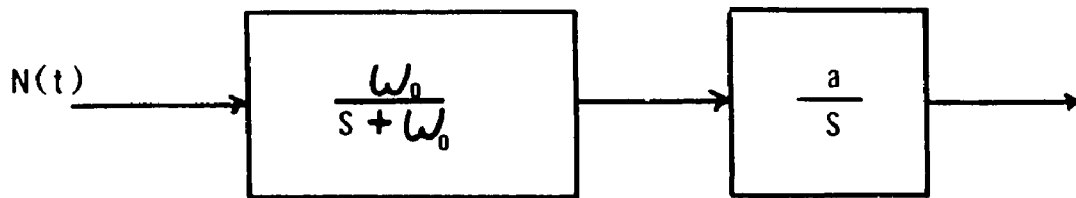


Figure 3.3 Block Diagram of a Langevin Equation Excited by Band-limited Noise

$$m(t, t_i) = u(t - t_i) \quad (3.18)$$

and the impulse response function  $h(t, t_i)$  is

$$h(t, t_i) = a \left[ 1 - e^{-\omega_o(t-t_i)} \right] u(t-t_i) \quad (3.19)$$

Using equation (2.15), the partial differential equation which the conditional probability density function of the output noise satisfies is:

$$\frac{\partial U}{\partial t_i} = -D a^2 \left[ 1 - e^{-\omega_o(t-t_i)} \right]^2 \frac{\partial^2 U}{\partial x^2} \quad (3.20)$$

If it is desired to solve the elliptical partial differential equation

$$\frac{\partial^2 U}{\partial x^2} = 0 \quad (3.21)$$

then only the "steady-state" solution of equation (3.20) is of interest. Note that the presence of the exponential term does not alter the steady-state behavior of the solution. If a solution of the diffusion equation is of interest, then the exponential term does disturb the solution and the random walks must be made sufficiently long to make the exponential term negligible compared with unity. It is convenient to choose

$$\omega_o(t-t_i) > 2\pi \quad (3.22)$$

or

$$t - t_i > \frac{2}{f_c} \quad (3.23)$$

where  $f_c$  is the pulse rate of the binary noise. In the parabolic diffusion equation, the particle walks for a total time  $t_i$ , and hence to insure proper solutions, the inequality

$$t_i > \frac{2}{f_c} = 4 \cdot 10^{-6} \quad (3.24)$$

should be enforced.

### 3.4 Integrator Gain Considerations for Binary Signals

To generate as many walks as possible, it is desirable to keep the integrator gain as large as possible so that the particle performing the random walk will reach the absorption boundaries quickly. When binary noise is used as the driving input, the integrator gain is set by allowable step size appearing at the integrator output, the statistical behavior of which can be treated as a discrete probability mass function which changes with time. Since the Monte Carlo method for obtaining the solutions of second order partial differential equations is predicated on the assumption that we are dealing with Gaussian probability distribution functions, it is of importance to consider the behavior of the random process appearing at the integrator output. Because the noise input to the

integrator is binary with independent segments, the integrator output is a binomial process.<sup>16</sup>

For a sufficiently large number of independent binary distributed events, the binomial distribution approaches a normal distribution. The rapidity of this convergence is highest if the probabilities of the mutually exclusive binary events are  $\frac{1}{2}$ . Since a normal distribution may be substituted for a binomially distributed random variable if the number of independent binary distributed events is large, it is important to choose the integrator gain so that a large number of binary noise pulses is required to complete each random walk.

The magnitude of the integrator response to one pulse of the binary noise signal is determined by the integrator gain:

$$V_s = \frac{E}{RC} \Delta_t \quad (3.25)$$

where  $E$  is the voltage level of the input noise,  $RC$  is the time constant and  $\Delta_t$  is the time interval during which the noise input remains constant. The range of voltages appearing at the output of the integrator is divided into segments of magnitude  $V_s$  and it is reasonable to expect that the quantizing error introduced by binary rather than Gaussian noise is of the order of  $(V_s/10)$  100%.<sup>17</sup>

To observe the relationship between the quantizing voltage level and the average time required for a random walk, once again it is convenient to consider Laplace's equation in one dimension with the boundary condition

$$U(10) = 10 \quad (3.26)$$

$$U(-10) = -10 \quad (3.27)$$

Now, the expected time for the completion of a random walk is

$$E [T(x)] = \frac{100 - x^2}{2D} \quad (3.28)$$

as shown in Appendix B. The maximum average time occurs when the starting point of the walk is at the origin, in this case,

$$E [T(x)]_{\max.} = \frac{100}{2D} \quad (3.29)$$

The power spectral density of the binary noise,  $2D$ , times the integrator gain  $1/RC$  is

$$\frac{2D}{(RC)^2} = \frac{E^2}{(RC)^2} \Delta_t \quad (3.30)$$

and therefore

$$E [T(x)]_{\max.} = \frac{100}{\left(\frac{E}{RC}\right) \Delta_t} \quad (3.31)$$

With the aid of (3.25), one obtains the following relationship between the quantizing voltage and the maximum average time for a random walk.

$$E \left[ T(x) \right]_{\max.} = \frac{100}{V_s} \frac{\Delta_t}{2} \quad (3.32)$$

If the maximum average time is chosen to be one millisecond so that 1000 random walks can be completed each second, then  $V_s$ , the quantizing interval, is 0.316 volts;  $\Delta_t$  has been set equal to one microsecond, which is the basic time interval of the ASTRAC II noise generator.

### 3.5 Effect of the Finite Computer-run Time Allotted to Each Random Walk

To obtain Monte Carlo estimates of the solutions of partial differential equations over any smooth curve in the region where the function is defined, analog rather than digital averaging is appropriate. To average the tallies of many random walks with a continuous analog averaging circuit, it is necessary to generate an analog voltage equal to the value of the function at the point of absorption on the boundary which is held constant for a fixed time. One can accomplish this most conveniently, by fixing the time allotted for a random walk and generating this voltage for the duration of the succeeding walk. Constraining the walk to a fixed computer-run time

introduces an error into the Monte Carlo routine which is examined below.

Consider a function  $U(\underline{r}_i)$  which satisfies Laplace's equation in  $n$  dimensions.  $U(\underline{r}_i)$  is defined within a region  $R$  with boundary  $C$ ; the value of  $U(\underline{r}_i)$  on  $C$  is

$$U(\underline{r}) \Big|_C = U_C(\underline{r}_b) \quad (3.33)$$

Assume that the solution to this problem is

$$U(\underline{r}_i) = F(\underline{r}_i) \quad (3.34)$$

where  $F(\underline{r}_i)$  remains to be found. To obtain an estimate of the solution of this by Monte Carlo methods, it is necessary to start a random walk at an interior point  $\underline{r}_i$  and continue the random walk until the boundary  $C$  is reached. After reaching the boundary, one records the value of the function at the point of absorption and a new random walk is begun. Since, as was mentioned above, a fixed time is allotted for each random walk, it is uncertain whether or not the particle will reach the boundary. To remedy this problem, rather than solve the elliptical Laplace equation, it is proposed to estimate the solution of the parabolic diffusion equation (3.35) which reduces to Laplace's equation in the steady state. We have

$$\nabla^2 v(\underline{r}_i, t_i) = \frac{\partial v}{\partial t_i} \quad (3.35)$$

with the boundary condition

$$V(\underline{r}_i, t_i) \Big|_C = U_C(\underline{r}_b) \quad (3.36)$$

and the initial condition

$$V(\underline{r}_i, 0) = a(\underline{r}_i) \quad (3.37)$$

where the function  $a(\underline{r}_i)$  is as yet undetermined. The problem posed above can be decomposed into two problems involving  $U_1(\underline{r}_i, t_i)$  and  $U_2(\underline{r}_i, t_i)$  where

$$V(\underline{r}_i, t_i) = U_1(\underline{r}_i, t_i) + U_2(\underline{r}_i, t_i) \quad (3.38)$$

$$\nabla^2 U_1(\underline{r}_i, t_i) = \frac{\partial U_1}{\partial t_i} \quad (3.39)$$

$$U_1(\underline{r}_i, t_i) \Big|_C = U_C(\underline{r}_b) \quad (3.40)$$

$$U_1(\underline{r}_i, 0) = F(\underline{r}_i) \quad (3.41)$$

$$\nabla^2 U_2(\underline{r}_i, t_i) = \frac{\partial U_2}{\partial t_i} \quad (3.42)$$

$$U_2(\underline{r}_i, t_i) \Big|_C = 0 \quad (3.43)$$

$$U_2(\underline{r}_i, 0) = a(\underline{r}_i) - F(\underline{r}_i) \quad (3.44)$$

Since the initial condition of the diffusion problem involving  $U_1(\underline{r}_i, t_i)$  was chosen to be the solution of the Laplace equation over the same region  $R$ , with the

same boundary conditions, the solution of this problem and the problem involving  $U(\underline{r}_i)$  is identical, i.e.,

$$U_1(\underline{r}_i, t_i) = F(\underline{r}_i) \quad (3.45)$$

In the decomposition (3.38) given above,  $F(\underline{r}_i)$  is the yet unknown solution of the original problem. The term  $a(\underline{r}_i)$ , the initial condition function of the problem involving  $V(\underline{r}_i, t_i)$  (which is the problem actually being solved because of the fixed time constraint), should be chosen to minimize the error made by solving this inexact problem. Since the solution of the original problem is the solution of the  $U_1(\underline{r}_i, t_i)$  problem, the solution of the  $U_2(\underline{r}_i, t_i)$  problem is the error resulting from the fixed time constraint. It remains to select the value of  $a(\underline{r}_i)$  to be used in the Monte Carlo solution of (3.35); ideally the optimum choice for  $a(\underline{r}_i)$  is

$$a(\underline{r}_i) = F(\underline{r}_i) \quad (3.46)$$

since this results in a null solution for the error function  $U_2(\underline{r}_i, t_i)$ . This, however, is in general not possible because it implies that the solution of the original problem is known in advance. A selection for  $a(\underline{r}_i)$  must be made which both maintains a reasonable error and lends itself to a convenient computing procedure. For computing simplicity, it is desirable to let  $a(\underline{r}_i)$  be a constant. Since the maximum and minimum values of a

harmonic function occur on the boundary,<sup>22</sup> a reasonable choice for  $a(\underline{r}_i)$  is the average of the maximum and minimum boundary values, or

$$a(\underline{r}_i) = \frac{U_C(\underline{r}_b)|_{\max.} + U_C(\underline{r}_b)|_{\min.}}{2} \quad (3.47)$$

Rather than choose  $a(\underline{r}_i)$  to be a constant given by (3.47), it would perhaps be more desirable to select an estimate of  $F(\underline{r}_i)$  to be equal to  $a(\underline{r}_i)$ . When the solution of Laplace's equation in the space  $R$  is sought and analog averaging is employed, the output of the averaging circuit is an estimate of the function in a neighborhood of the point of interest if the starting point of the random walks is slowly swept over a line segment containing the point in question. Because the solution of Laplace's equation is a well-behaved continuous function, a good choice for  $a(\underline{r}_i)$  would be the output of the averaging filter. The procedure to be followed in a time constrained random walk therefore is to average the absorbed boundary values whenever the particle reaches the absorbing boundary; for walks terminated before the boundary is reached, average the estimate of the value of the function.

### 3.6 Sweep Errors

If it is desired to sweep the starting point of the random walks over an arc in the region of definition of the partial differential equation and develop the Monte Carlo

estimate of the solution of the partial differential equation upon this arc, the effect of sweep speed is of interest. Two sources of error are associated with the sweeping arrangement:

1. Generation of errors due to an insufficient number of random walks originating from a specified interval.
2. Errors which result from the time delay of the averaging filter.

To obtain a Monte Carlo estimate of a solution of a partial differential equation at a point, it is necessary to originate a number of random walks from the point. The problem is complicated if the point in question is scanned, since, under this condition, really not more than one random walk originates from any point. The rate at which a region may be scanned will depend upon how rapidly the solution varies within this region. Once again Laplace's equation in one dimension

$$\frac{d^2U}{dx^2} = 0 \quad (3.48)$$

with the boundary conditions

$$U(10) = 10$$

$$U(-10) = -10 \quad (3.49)$$

is a particularly convenient choice for an examination of

the sweep problem. The solution of the above problem is of course

$$U(x) = x \quad (3.50)$$

For this problem, if the walk originates within an interval  $\pm \Delta x$  about some nominal point  $x_0$ , the expected error in the Monte Carlo estimate is

$$e = \pm \Delta x \quad (3.51)$$

To maintain this error, the sweep speed,  $n$  volts per second, must satisfy the following inequality:

$$n \leq \frac{xN}{N_0} \quad (3.52)$$

where  $N$  is the number of random walks per second and  $N_0$  the required number of random walks necessary for a suitable low variance estimate of the solution of the partial differential equation. If  $\Delta x$  is chosen to be 0.5 volts,  $N$  to be 1000 random walks and  $N_0$  1000 walks<sup>11</sup> then,

$$n \leq 0.5 \text{ volts/second.} \quad (3.53)$$

The error resulting from the time delay of the analog averaging filter is most easily considered by examining a basic averaging circuit as shown in Fig. 3.5. The transfer function of this filter is

$$H(s) = \frac{1}{sT_0 + 1} \quad (3.54)$$

If the starting point of the random walks in the Monte Carlo solution of (3.48) is scanned along the x axis, the expected value of the random waveform representing the absorbed boundary values is  $nt$  where  $n$  is the sweep speed and  $t$  is time. The steady state output of the averaging filter due to this average input is  $y(t)$  where

$$y(t) = n(t-T) \quad (3.55)$$

Since the desired output is  $nt$ , the error is

$$e(t) = nT \quad (3.56)$$

If the solution  $U(\underline{x}_i)$  changes abruptly over the arc segment which is being examined, then the time delay associated with the low pass filter could produce unacceptable errors. When a function with abrupt changes is the result of a Monte Carlo computation, it would be wise to repeat the solution with a slower scan rate and to compare the results obtained with the slow scan with the original results.

In addition to obtaining the estimate of the solution of the partial differential equation over a line segment, it is desirable to generate equipotential or level-lines of the solution. A procedure similar to the one described by Lym<sup>18</sup> in a different context was employed to find level-lines for the Laplace and diffusion equations. For a problem with two spatial dimensions, (3.57) and

(3.58) are the equations of a system generating equipotential lines.

$$\frac{dx}{dt} = a \quad (3.57)$$

$$\frac{dy}{dt} = b(A - U(r_i)) \quad (3.58)$$

The potential of the level-lines is denoted by A and the x sweep rate a, is suitably slow. The block diagram of this system is shown in Fig. 3.4 where the Monte Carlo computation routine is represented by one block.

The choice of

$$b = 1/3 \quad (3.59)$$

$$a = 0.03 \quad (3.60)$$

produces satisfactory overshoot behavior for an averaging time constant of one second in the low pass filter of the Monte Carlo routine. As Lym<sup>18</sup> indicates, the system yields unacceptable results if the solution of the partial differential equation is not a smooth function or if the equipotential lines are vertical ( $x = \text{constant}$ ).

### 3.7 Averaging Time Requirements and Statistical Fluctuations

An analog voltage equal to the value of the function at the absorption point is generated at the end of each random walk and held for a fixed time,  $T_w$ . The expected

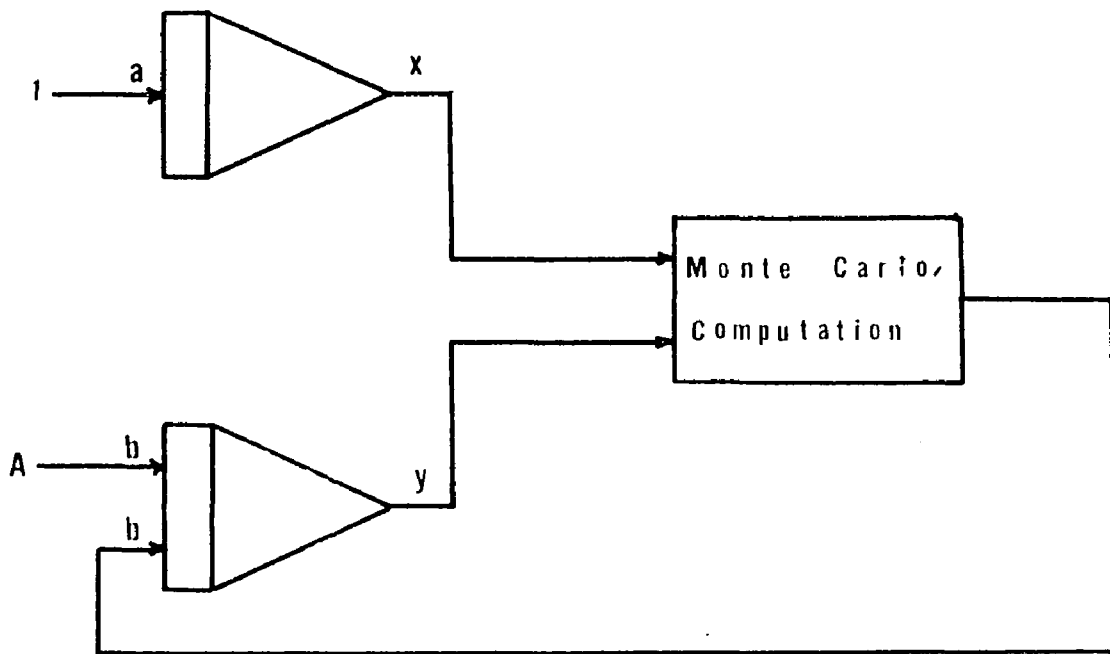


Figure 3.4 Computer Diagram for Generating Level-lines

value of this waveform is the solution of the partial differential equation at the starting point of the random walk.

Various analog circuits can be utilized for analog averaging; however, it was found that the simple low pass RC filter shown in Fig. 3.5 yielded acceptable results. The transfer function of the circuit is

$$H(s) = \frac{1}{sT_o + 1} \quad (3.61)$$

where  $T_o = RC$ . This averaging circuit has an impulse response

$$\begin{aligned} h(t) &= \frac{1}{T_o} e^{-\left(\frac{t}{T_o}\right)} \quad \text{for } t > 0 \\ &= 0 \quad \text{for } t < 0 \end{aligned} \quad (3.62)$$

If the average value of a random process is desired, a member function of the ensemble is put into the filter producing an output

$$y(t) = \frac{1}{T_o} \int_0^t f(t_1) e^{-\frac{(t-t_1)}{T_o}} dt_1 \quad (3.63)$$

When the integration time  $t$  is in the range  $4T_o < t < 10T_o$ ,  $y(t)$  is an unbiased estimate of  $E[f(t)]$ <sup>19</sup> with an estimate variance

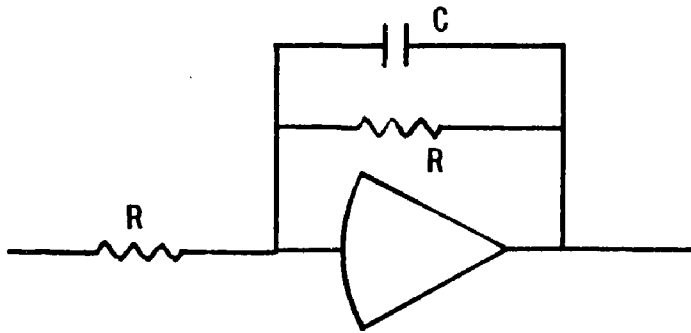


Figure 3.5 An Analog Averaging Circuit

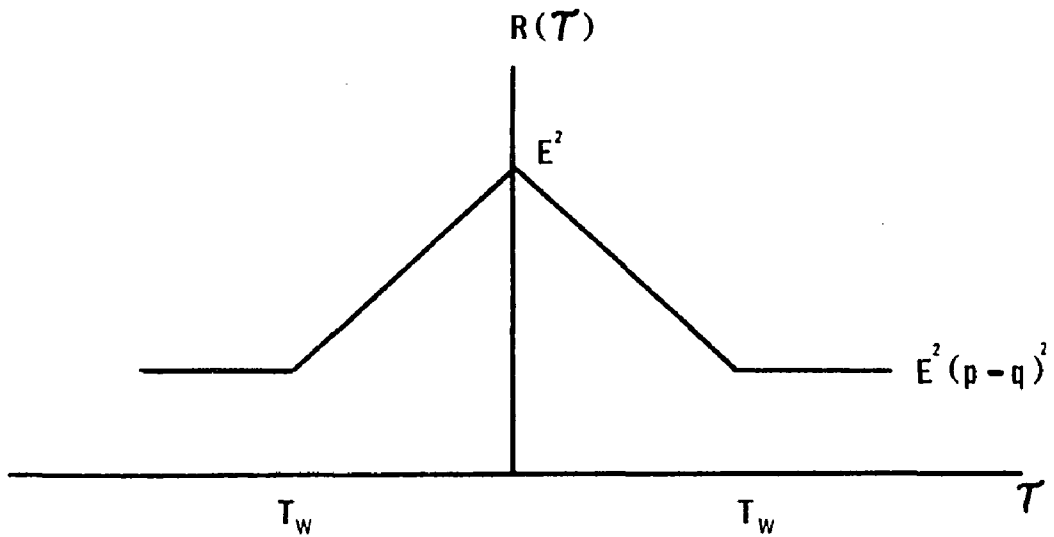


Figure 3.6 Autocorrelation Function of the Random Binary Waveform

$$\text{Var } f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{1}{(wT_0)^2 + 1} S(w) dw \quad (3.64)$$

where  $S(w)$  is the power spectral density of

$$\left( f(t) - E[f(t)] \right) \quad (3.65)$$

To examine the requirements of an averaging filter for use in Monte Carlo calculations, consider once again Laplace's equation in one dimension.

$$\nabla^2 U(x) = 0 \quad (3.66)$$

$$U(10) = 10 \quad (3.67)$$

$$U(-10) = -10 \quad (3.68)$$

The particle executing the random walk will be absorbed at either the +10 volt boundary or the -10 volt boundary. Since the boundary conditions take on only two values, the voltage waveform to be averaged is a random binary wave with a +10 volt amplitude which can only change at times  $nT_w$ , where  $T_w$  is the fixed time allotted for one random walk. If  $p$  is the probability that an absorption will occur on the +10 volt boundary and  $q$  is the probability that the absorption will occur on the -10 volt boundary, the autocorrelation function of the binary waveform is

$$R(\tau) = 100 \left[ 1 - \frac{\tau}{T_w} (1 - (p-q)^2) \right] \text{ for } \tau < T_w \quad (3.69)$$

$$R(\tau) = 100 (p-q)^2 \text{ for } \tau > T_w \quad (3.70)$$

This function is shown in Fig. 3.6. If the mean value of the binary wave  $E(p-q)$  is subtracted from the wave itself, the autocorrelation function of the resulting waveform is

$$R(\tau) = 100 \left[ 1 - (p-q)^2 \right] \left[ 1 - \frac{\tau}{T_w} \right] \text{ for } \tau < T_w \quad (3.71)$$

$$R(\tau) = 0 \text{ for } \tau > T_w \quad (3.72)$$

This is the autocorrelation function of a random square wave with zero mean value and peak amplitude  $A$ .

$$A = 10 \left[ 1 - (p-q)^2 \right]^{1/2} \quad (3.73)$$

It is possible to treat this sampled data process as a random telegraph<sup>15</sup> wave with an average zero crossing rate of

$$K = \frac{1}{T_w} \quad (3.74)$$

crossings per second. For a specified variance  $e_{rms}^2$ , the filter time constant  $T_o$  must satisfy the inequality

$$T_o \geq \frac{T_w}{2} \frac{100 \left[ 1 - (p-q)^2 \right]^2}{e_{rms}^2} \quad (3.75)$$

The largest value for  $T_o$  occurs when  $p = q = 1/2$ . For this condition, the inequality assumes the form

$$T_o \geq \frac{T_w}{2} \frac{1}{E^2} \quad (3.76)$$

where  $E = \left( \frac{e_{rms}}{10} \right) 100\%$  is the percentage error.

For  $E$  to be 5% and the time allotted for a random walk  $T_w$  to be one millisecond,

$$T_o \geq 0.2 \text{ seconds} \quad (3.77)$$

## CHAPTER 4

### ANALOG-HYBRID COMPUTER CONSIDERATIONS FOR MONTE CARLO SOLUTIONS OF PARTIAL DIFFERENTIAL EQUATIONS

#### 4.1 Introduction

The attractiveness of a Monte Carlo solution of a partial differential equation depends upon the accuracy and solution speeds attainable. Aside from the inevitable statistical fluctuations, the nature and flexibility of the computer determine the feasibility of a Monte Carlo solution.

This study was conducted with the digitally controlled iterative differential analyzer ASTRAC II;<sup>12</sup> the results are general enough to be used as guideposts for programming any analog-hybrid computer. In this chapter, the ASTRAC II computer is described and the implementation of a typical Monte Carlo computation is presented. In addition, various computing schemes useful in Monte Carlo studies are examined and computer diagrams for their generation are given.

#### 4.2 Description of ASTRAC II

Most analog computers capable of repetitive operation can be adapted to Monte Carlo computations. High computing speed is desirable for the solution of partial

differential equations, because approximately 1000 random walks are necessary to obtain an estimate of the solution of a partial differential equation at each point with reasonable accuracy.

ASTRAC II, developed at the University of Arizona Analog-Hybrid Computer Laboratory, employs  $\pm 10$  volt transistor amplifiers with unity-gain bandwidth beyond 20 Mcps. and extremely low computing impedances. Summing-resistor values are 5K and 1K, as compared to the 1M and 100K resistors employed in "slow" analog computers. In addition, all amplifiers and multipliers are plugged directly into the rear of a shielded patchbay which eliminates signal-wiring capacitances and crosstalk. Table 4.1 summarizes ASTRAC II performance.<sup>20</sup>

ASTRAC II employs digital circuits for iteration control and statistics accumulation. Integrator and track-hold mode control, comparator outputs, analog switches and free logic elements are terminated in a control patchbay. A digital control panel allows control of repetition rate, computer-run length, readout timing and sample sizes for statistical computations.

The digital control unit of ASTRAC II, in addition to providing free digital logic elements such as flip-flops, gates, etc., provides a group of timing pulses derived from a crystal oscillator to be used for integrator

Table 4.1. ASTRAC II Performance Data

---



---

<u>Amplifiers:</u>	$\pm 10\text{v}$ ; 50ma at dc; $\pm 10\text{v}$ , 30ma at 1 Mc.
	Dc gain, $10^6$ (120 db.) with chopper stabilization.
	Drift, $10\mu\text{v}/^\circ\text{C}$ , $1\text{na}/^\circ\text{C}$ with chopper stabilization.
	0-db. bandwidth, 20Mc; 6 db./octave rolloff.
	Unity-inverter phase error, 0.16% ( $0.09^\circ$ ) at 16 Kc.
	Integrator phase error, 0.08% ( $0.05^\circ$ ) at 16 Kc.
	Resistance ratios specified within 0.2%.
<u>Integrator/track-hold circuits</u>	( $C=1\mu\text{f}$ , $0.1\mu\text{f}$ or $0.01\mu\text{f}$ ;
data for $C=0.01\mu\text{f}$ ):	
	Phase error in TRACK $< 0.5\%$ ( $0.3^\circ$ ) at 10 Kc.
	Holding error $\leq 0.1\%$ of half-scale in 1 msec.
	Switching time into HOLD or COMPUTE $< 80$ nsec;
	two circuits switch within 20 nsec of one another.
<u>Diode quarter-square multipliers:</u>	
	Static error $< 0.2\%$ of half-scale.
	Drift $< 0.7$ mv/ $^\circ\text{C}$ .
	Dynamic error, 0.5% of half-scale at 10 Kc. ( $0.3^\circ$ phase error).

Table 4.1--ContinuedAnalog comparators:

Chopper stabilized; response time  $< 120$  nsec.

Static hysteresis,  $\pm 15$  mv.

Drift  $< 25\mu\text{v}/^\circ\text{C}$ .

mode control and for timing signals for analog switches.

The ASTRAC II timing signals are shown in Fig. 4.1.

All timing signals in ASTRAC II are derived from the basic clock signal  $C_1$ , which in turn originates from a crystal oscillator. Every 1000 pulses, a flip-flop is set; it is reset by a thumbwheel switch  $n_T$  pulses later to generate the reset waveform R, shown in Fig. 4.1. Both the  $S_1$  and  $S_2$  sampling-pulse trains are of fixed width (100  $C_1$  pulses) and are positioned relative to R by two independent digit switches. The  $S_2$  signal can also be swept both forward and backward in time relative to R.  $S_{1D}$  and  $S_{2D}$  are derived from  $S_1$  and  $S_2$  respectively as shown in Fig. 4.1. The R' signal is used to fix the number of computer runs desired for a particular problem.

#### 4.3 Noise Source Calibration

In the Monte Carlo method for the estimation of the solution of partial differential equations, the quantity

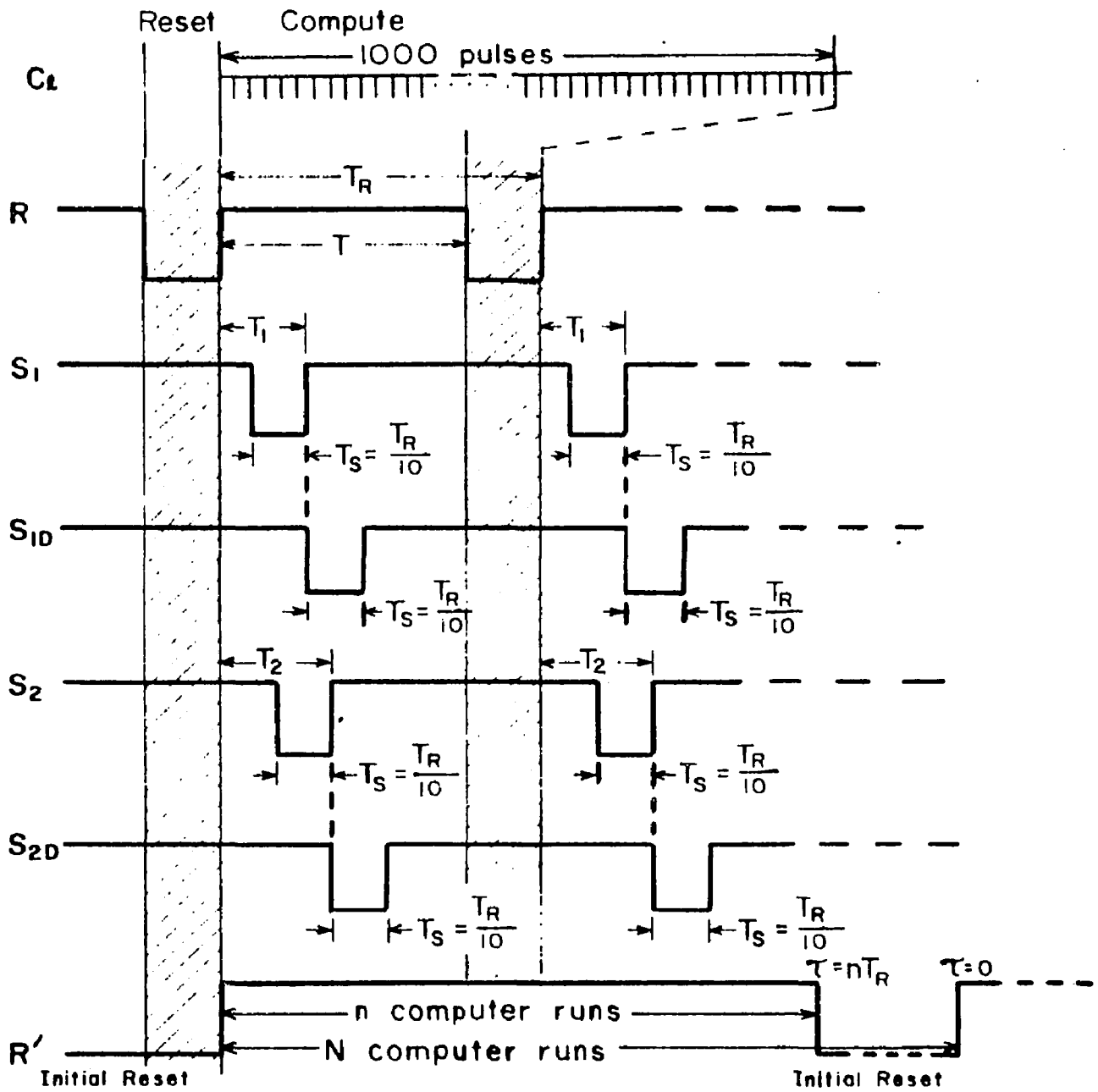


Figure 4.1 ASTRAC II Timing Signals

2D, the power spectral density of the noise source, appears quite frequently in the equations associated with the process. In all equations other than Laplace's equation, it is important that this quantity be known accurately. Several techniques which permit its measurement have been described in the literature;<sup>10, 11</sup> the procedure to be given here is basically the one described by Chuang et al.<sup>10</sup>

To measure the power spectral density of the noise source, it is convenient to solve the following problem:

$$D \frac{d^2U}{dx^2} + K \frac{dU}{dx} = 0 \quad (4.1)$$

with the boundary conditions,

$$U(10) = 10 \quad (4.2)$$

$$U(-10) = -10 \quad (4.3)$$

The solution of this problem is:

$$U(x) = \frac{10}{e^{-a} - e^a} \left[ 2e^{-\frac{ax}{10}} - e^a - e^{-a} \right] \quad (4.4)$$

$$\text{where } a = 10 K/D \quad (4.5)$$

If  $U(x)$  is evaluated at the origin, one obtains,

$$U(0) = \frac{10}{e^{-a} - e^a} \left[ 2 - e^a - e^{-a} \right] \quad (4.6)$$

For small  $a$ ,  $U(0)$  can be approximated by expansion of the exponential function in a Taylor series. Preserving the

first significant terms, for  $a \ll 1$

$$U(0) \approx 5a \quad (4.7)$$

The stochastic differential equation which must be implemented on the analog computer to produce a Monte Carlo solution of (4.1) is,

$$\frac{dy}{dt} = n(t) - V_{in} \quad (4.8)$$

where  $n(t)$  is white Gaussian noise with zero mean value and  $V_{in}$  is a constant. The analog computer diagram showing the implementation of (4.8) is given in Fig. 4.2a; Fig. 4.2b is the corresponding block diagram. From Fig. 4.2b,

$$K = G_1 V_{in} \quad (4.9)$$

$$n(t) = G_2 n'(t) \quad (4.10)$$

If the power spectral density of the input noise  $n'(t)$  is  $2D_N$ , then the power spectral density of  $n(t)$  is

$$\text{Var} [n(t)] = 2D_N G_2^2 \quad (4.11)$$

The partial differential equation which the conditional probability density function of  $y$  satisfies is,

$$D_N G_2^2 \frac{d^2 U}{dx^2} + V_{in} G_1 \frac{dU}{dx} = 0 \quad (4.12)$$

The solution at the origin is,

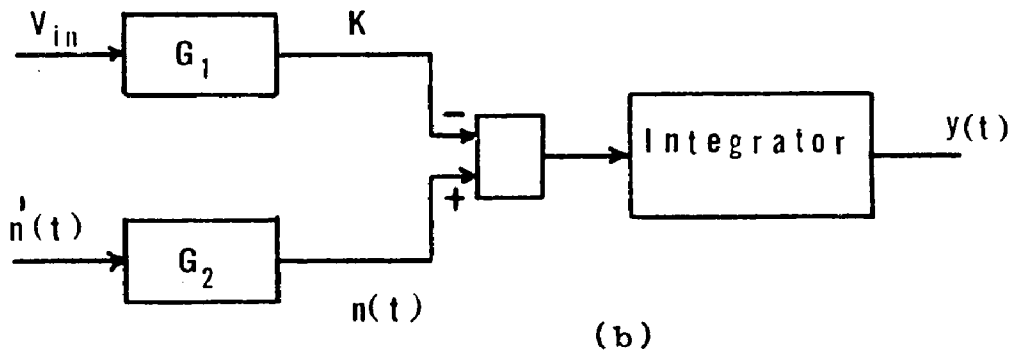
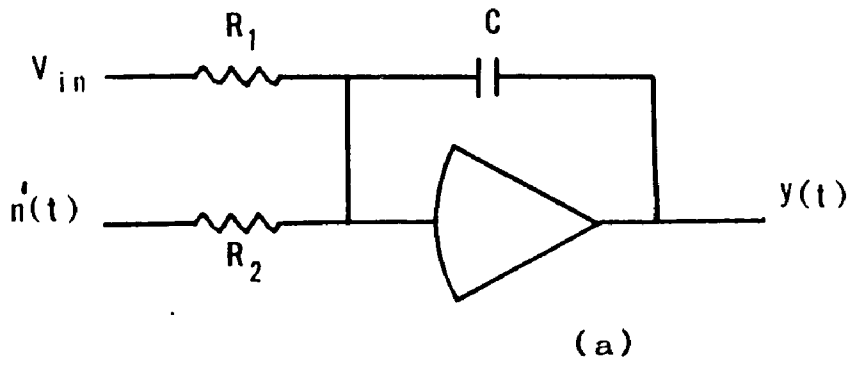


Figure 4.2 Noise Generator Calibration Diagrams

a. Analog Computer Diagram

b. Block Diagram

$$U(0) = \frac{50 V_{in} G_1}{D_N G_2^2} \quad (4.13)$$

If the above problem is programmed on the hybrid computer, and  $U(0)$  is the output voltage  $V_{out}$ , measurement of the slope  $m$  of the input-output characteristic at the origin enables one to calculate  $D_N$ .

$$D_N = \frac{50 G_1}{m G_2^2} \quad (4.14)$$

Using this technique, the power spectral density of the ASTRAC II noise generators was measured to be,

$$2D_N = 30 \cdot 10^{-6} \frac{\text{volts}^2}{\text{cps.}} \quad (4.15)$$

#### 4.4 Computer Mechanization of a Monte Carlo Problem

To obtain Monte Carlo estimates of the solution of partial differential equations on a hybrid computer, it is necessary to implement the Langevin equations, detect when absorbing or reflecting boundaries have been reached, and generate the proper absorbing-boundary voltages to be averaged. Since these steps are common to all partial differential equations which can be treated by the Monte Carlo method, it is convenient to consider the implementation of one problem, the diffusion equation, in great detail and then note the relatively minor modifications

necessary for different individual problems. The partial differential equation to be examined is,

$$\frac{\partial^2 U}{\partial x^2} = \frac{\partial U}{\partial t} \quad (4.16)$$

with the boundary conditions

$$U(10) = 10 \quad (4.17)$$

$$U(-10) = -10 \quad (4.18)$$

and the initial condition

$$U(x,0) = 0 \quad (4.19)$$

When white Gaussian noise with zero mean value and power spectral density  $2D_N$  is applied to an integrator with gain  $G$ , the stochastic differential equation implemented by this configuration is,

$$\frac{dy}{dt} = G n(t) \quad (4.20)$$

The corresponding partial differential equation satisfied by the conditional probability density function of  $y$ , and therefore the equation which is to be solved by Monte Carlo methods is,

$$G^2 D_N \frac{\partial^2 U}{\partial x^2} = - \frac{\partial U}{\partial t} \quad (4.21)$$

If one chooses

$$T = G^2 D_N \quad (4.22)$$

where  $T$  is a normalized time variable, equation (4.21) may be rewritten,

$$\frac{\partial^2 U}{\partial x^2} = - \frac{\partial U}{\partial t} \quad (4.23)$$

The hybrid computer diagram implementing the Monte Carlo solution of this equation is shown in Fig. 4.4. The analog-hybrid computer diagram symbols appearing in Fig. 4.3 and in the Figs. to follow, utilize standard analog and digital computer symbols, in addition to the symbols shown in Fig. 4.3. Typical waveforms appearing in the Monte Carlo simulation are given in Fig. 4.5. Actual photographs depicting the random walks appear in Fig. 4.6.

Fig. 4.5a shows the basic timing pulse  $R$ , which sets the repetition rate and the length of the random walks, thereby determining the value of  $T$  for which (4.23) is being solved. Typical random walks (integrator output) are shown in Fig. 4.5b; comparator outputs corresponding to a +10 volt and -10 volt absorption are given in Figs. 4.5c and d respectively. In the first compute-run interval, the first frame of the diagram, an absorption at the +10 volt level has occurred, in the second frame, no absorption has occurred, and in the third frame, an absorption at the -10

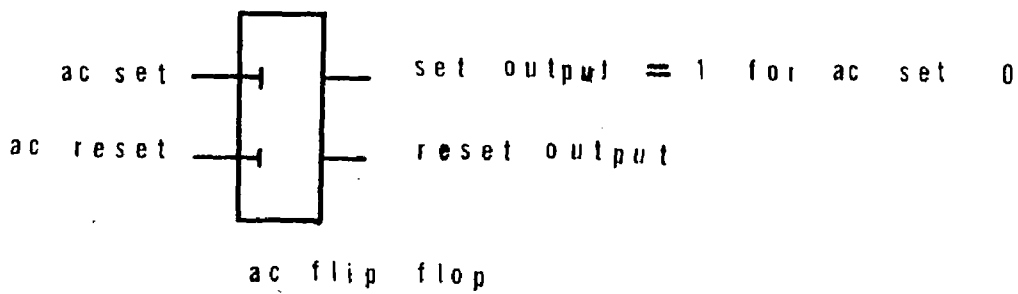
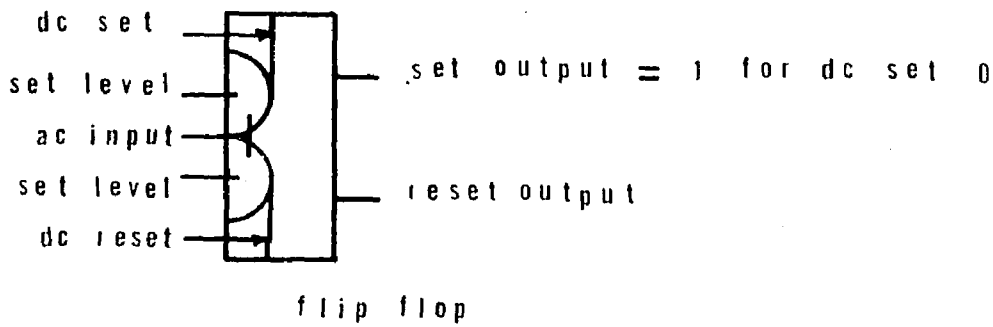
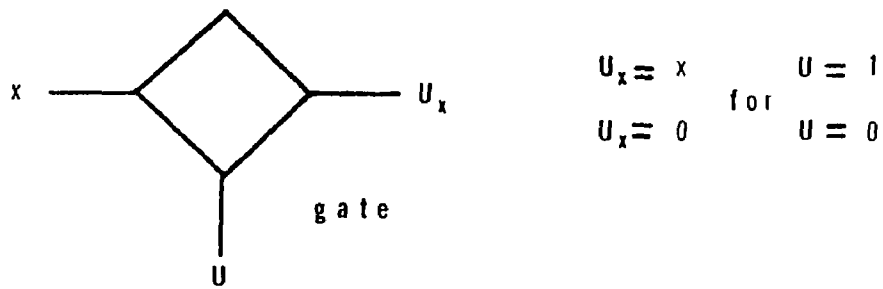
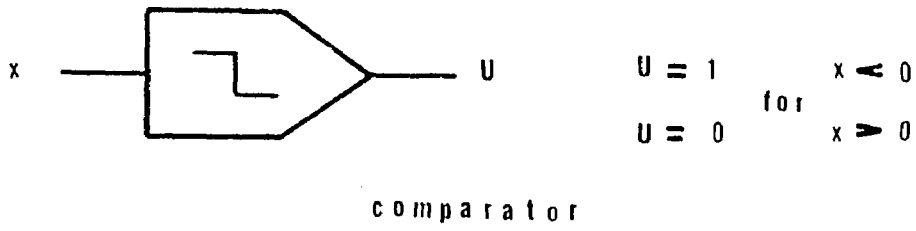


Figure 4.3 Hybrid Computer Symbols

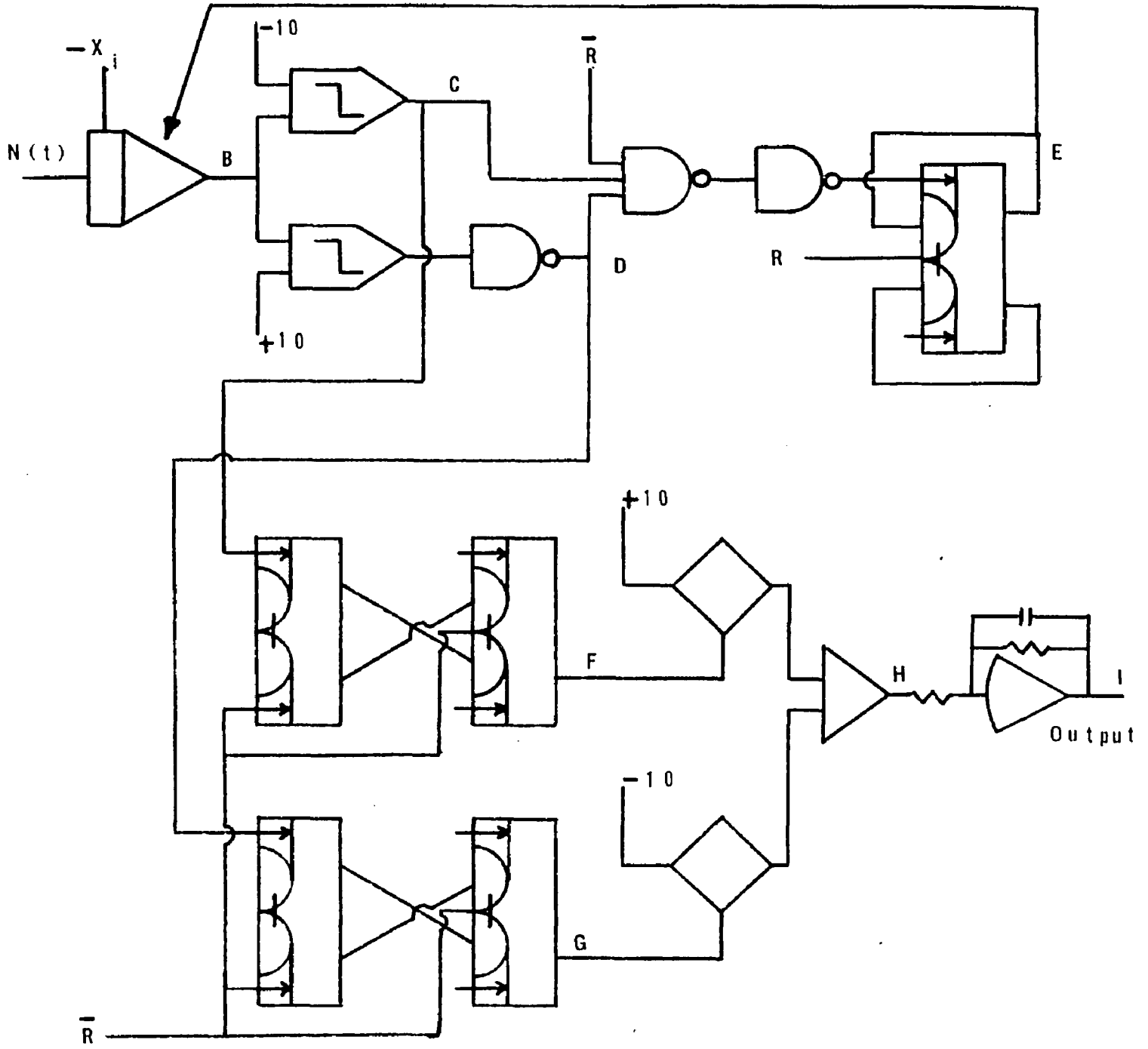


Figure 4.4 Hybrid Computer Implementation of the Monte Carlo Solution of the Diffusion Equation

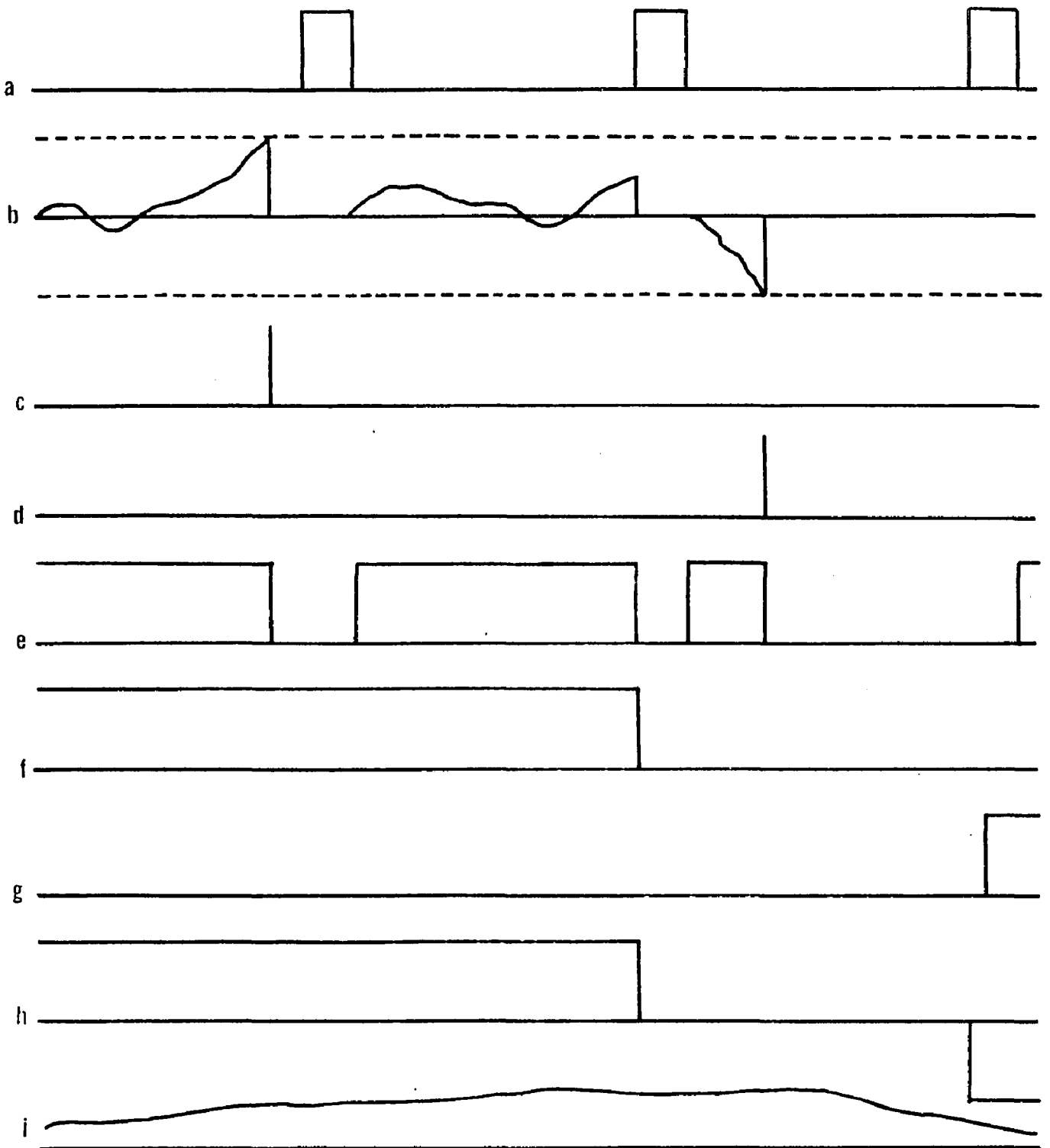
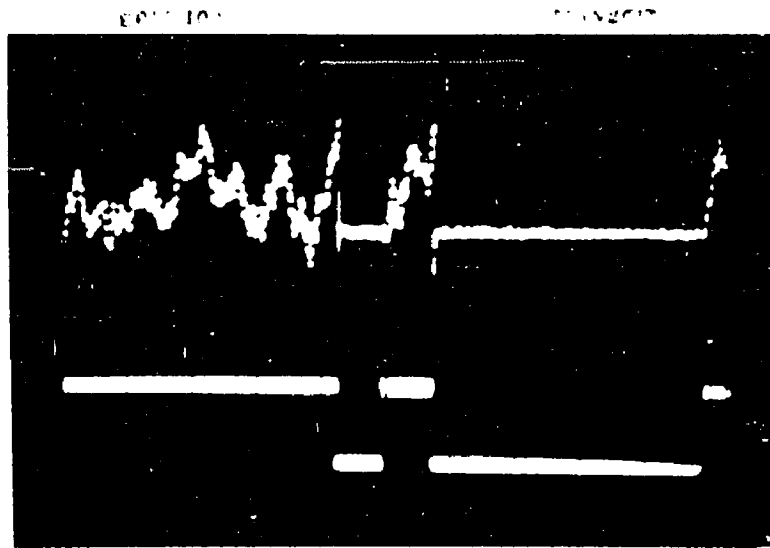
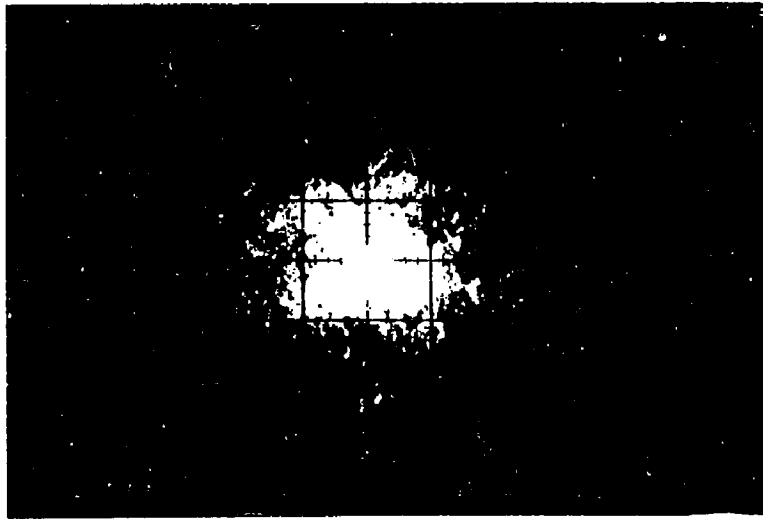


Figure 4.5. Typical Waveforms Appearing in a Monte Carlo Solution



(a)



(b)

Figure 4.6 Photographs of Waveforms Appearing in a Monte Carlo Solution

- a. Vertical scale; 5 volts per division
- b. Horizontal scale; 200 microseconds per division

volt level is shown. Figure 4.5e shows the integrator mode timing signal corresponding to the signals shown in Figs. 4.5b, c, and d. Figure 4.5f shows the waveform which appears at point F of Fig. 4.4. This signal responds to absorptions at the +10 volt boundary; if the +10 volt boundary is reached during the course of a random walk, as shown in frame 1, the signal at F will remain in the 0 (high) state during the time interval set by the two succeeding R pulses. The waveform shown in Fig. 4.5g is similar to the one shown in Fig. 4.5f; however, this signal responds to absorptions at the -10 volt boundary. The signal shown in Fig. 4.5h is set at +10 volts if the preceding random walk was terminated due to a +10 volt absorption, -10 volts if an absorption occurred at the -10 volt boundary, and 0 volts (the initial condition) if no absorption occurred during the time interval allotted for the random walk. The waveform shown in Fig. 4.5i, the final output of the computation, is the average value of the signal shown in Fig. 4.5h. The solution of the diffusion equation may be obtained for all  $-10 < x < +10$  by sweeping the initial condition input into the integrator; the value of T may be altered by adjusting the duty cycle of the R signal which changes the time allotted for a random walk.

The photographs in Figs. 4.6a and b are actual random walks implemented on ASTRAC II. Fig. 4.6a shows the

random walk which was obtained during a Monte Carlo solution of the diffusion equation in one dimension; the top waveform corresponds to Fig. 4.5b while the bottom trace is the integrator reset signal similar to Fig. 4.5e. A two-dimensional random walk which was implemented to obtain a Monte Carlo solution of Laplace's equation is given in Fig. 4.6b; the two-dimensional boundary for this problem is a circle.

#### 4.5 Special Analog-hybrid Techniques in Monte Carlo Computations

For rapid Monte Carlo estimation of the solution of partial differential equations on a hybrid computer, it is necessary to employ somewhat specialized computing techniques. The implementation of the random walk itself is straightforward, but, the detection of boundaries, the generation of initial and boundary conditions, average time measurements used in the solution of nonhomogeneous partial differential equations and the procedure necessary for generating equipotential lines require special attention.

##### 4.5A Boundary Detection

In the Monte Carlo method for estimating the solution of partial differential equations, it is necessary to sense when a particle executing a random walk has reached a boundary point. In the one-dimensional problem, boundary detection requires simple comparison of the particle

coordinate with the two boundary points; the walk is terminated whenever either comparator indicates a boundary transition. This arrangement is shown in the hybrid computer configuration for the Monte Carlo solution of the diffusion equation, Fig. 4.4.

In two-dimensional problems, one must compare the particle coordinates with the curve defining the boundary. For example, if the boundary is a circle of radius 10, centered about the origin, one calculates  $(x^2 + y^2)$ , where  $(x,y)$  are the particle coordinates and compares the result with 100. The computer circuit for this operation is shown in Fig. 4.7.

If the boundary cannot be described by a simple mathematical expression, as many curves as are necessary to define the boundary must be generated and the particle coordinates compared with these curves; the comparator outputs are then combined with a logical OR operation. An example is presented below to clarify this procedure.

Consider the two-dimensional region R with boundary C shown in Fig. 4.8. The boundary C is composed of two curves:  $C_1(x)$  describes C for  $y > 0$  and  $C_2(x)$  is used for  $y < 0$ . The following decisions must be made to detect a boundary crossing:

1. Is  $y$  greater or less than zero?
2. If  $y$  is greater than zero, detect when  $y = C_1(x)$ .
3. If  $y$  is less than zero, detect when  $y = C_2(x)$ .

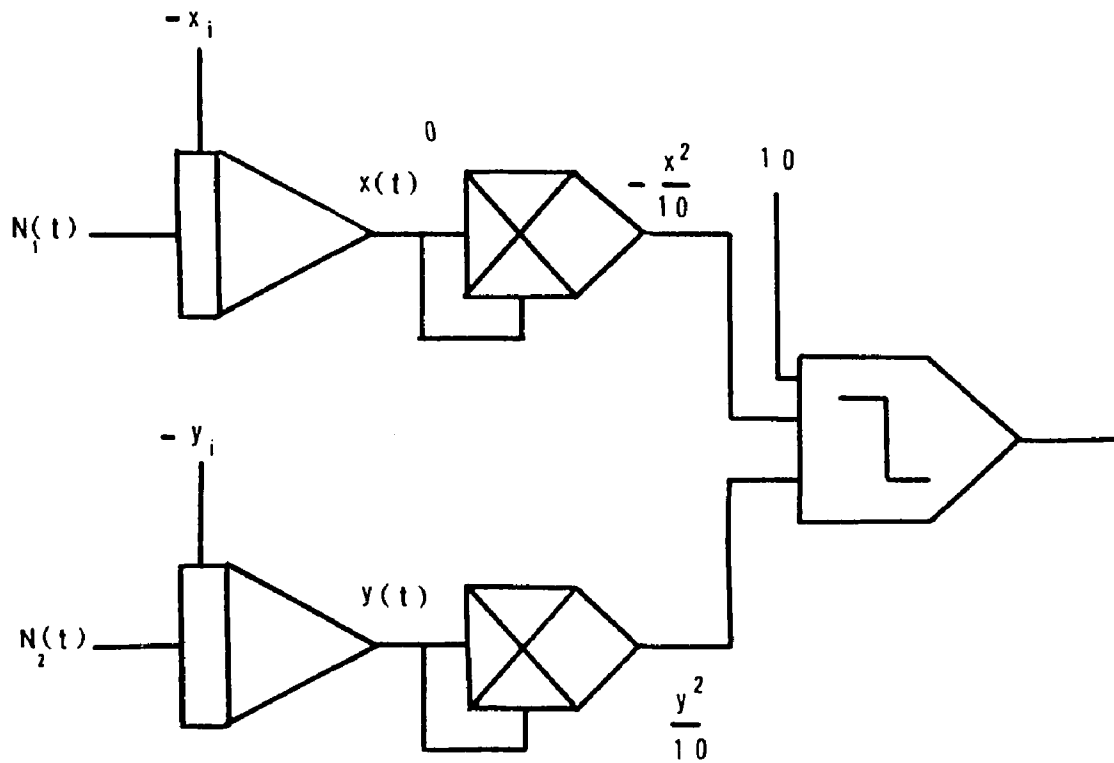


Figure 4.7 Circular Boundary Detector

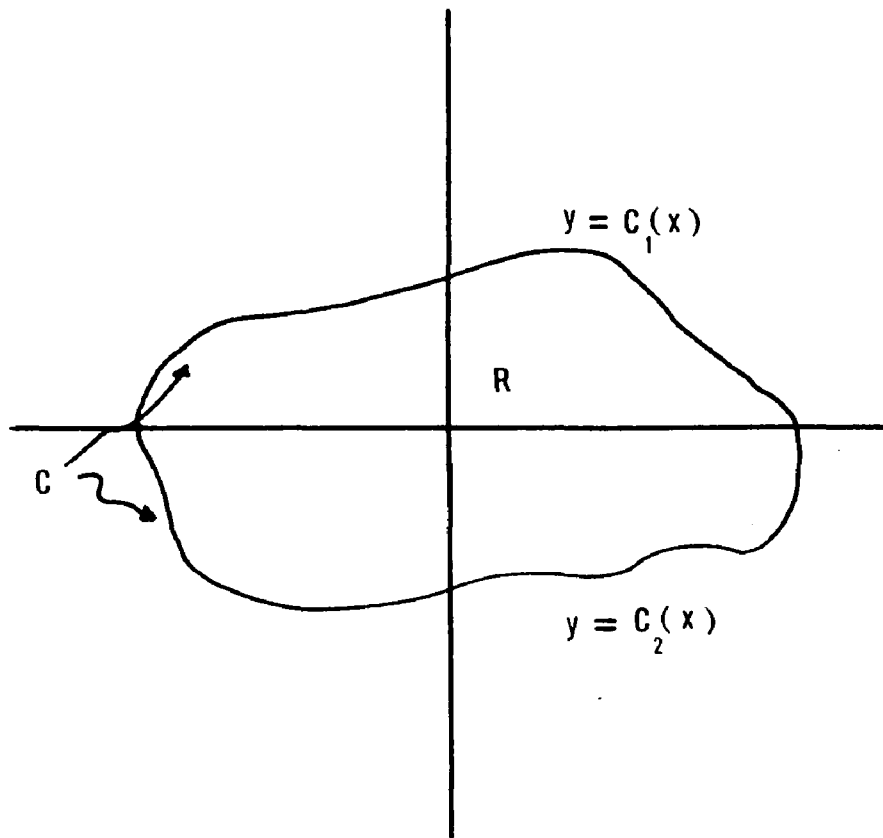


Figure 4.8 Two-Dimensional Region With Boundary  $C$  Described by Two Curves

The analog-hybrid computer implementation of this decision process is given in Fig. 4.9.

This procedure, with minor modification, can be used for any two-dimensional problem where a curve can be found which divides the boundary  $C$  into two single valued curves; the technique is readily extended to three or more dimensions. E.g., for a sphere,

$$x^2 + y^2 + z^2 \geq r^2 \quad (4.24)$$

In many cases, it may be desirable or necessary to employ digital function generation.

#### 4.5B Generation of Initial and Boundary Conditions

In the Monte Carlo estimation of the solution of a parabolic partial differential equation, unless the boundary is reached prior to time zero in the course of a random walk, it is necessary to generate a number equal to the initial condition evaluated at the internal point at which the random walk terminated. Similarly, the value of the boundary function at the point where an absorption has occurred must also be generated and averaged in the Monte Carlo process. If the functions to be generated depend upon a single parameter, they can be obtained with analog function generators. The equipment involved is inexpensive in comparison with the equipment necessary for use with digital techniques. When other than simple piecewise

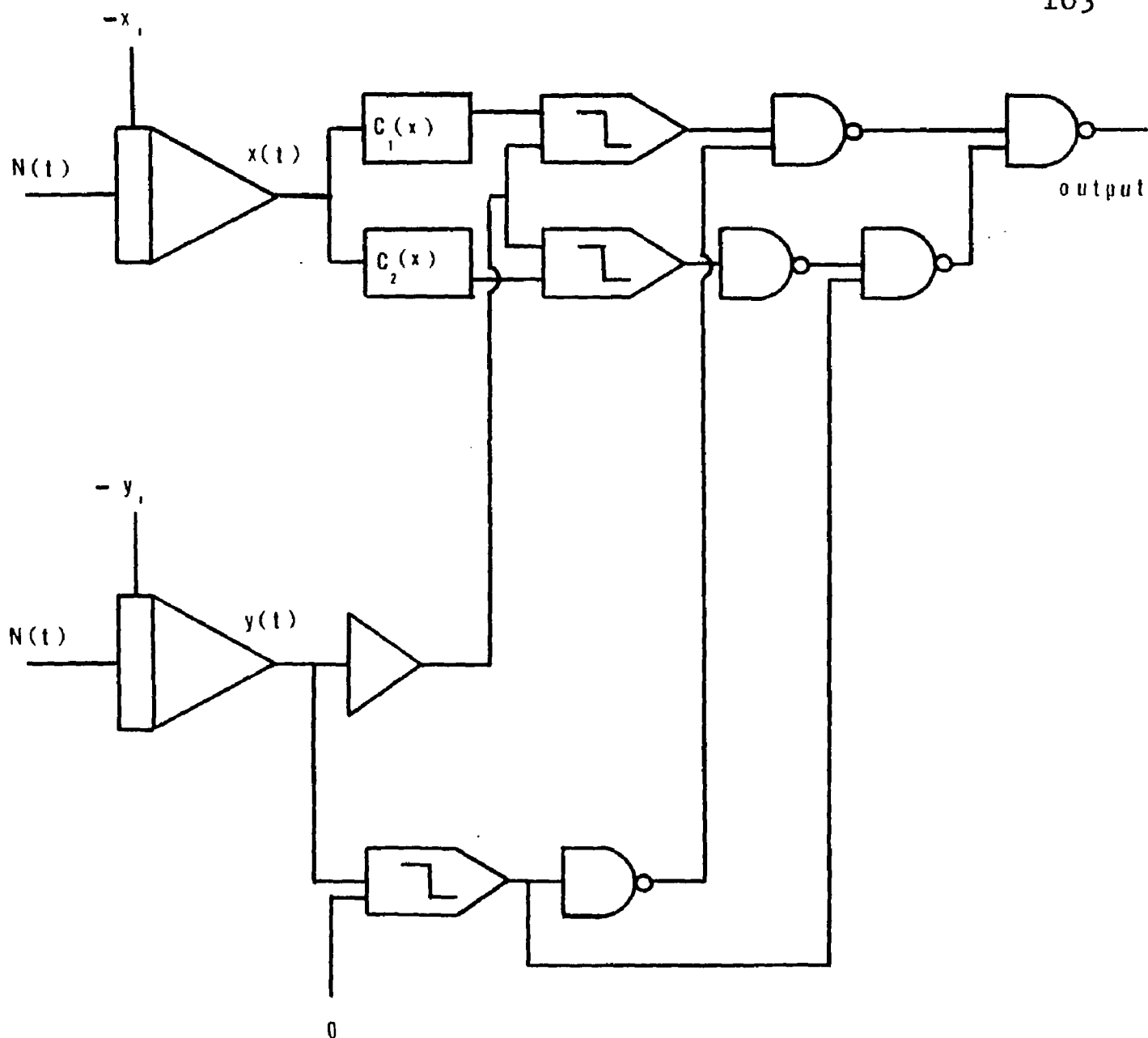
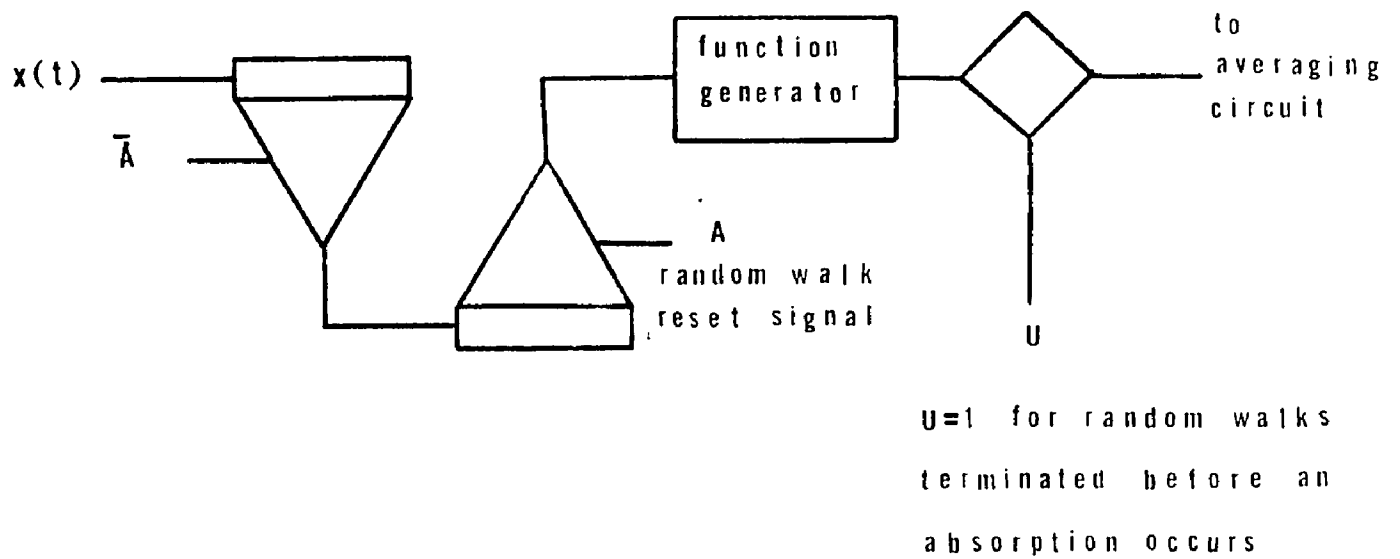


Figure 4.9 Hybrid Computer Diagram of a Boundary Detector for the Boundary Shown in Fig. 4.8

constant functions of several parameters are required for a Monte Carlo solution, digital function may be mandatory.

To illustrate the technique, consider the Monte Carlo solution of a parabolic partial differential equation in one dimension. The arrangement of the computing elements necessary for introducing the initial conditions appear in Fig. 4.10.

The position of the particle is tracked in the track-hold amplifier; at the time that the walk ends,  $t_b$  when an absorption occurs or at  $t = 0$  when no absorption occurs, its value is held. The output of the initial condition function generator after this time is available for averaging and is inserted into the averaging filter if indeed the walk did not terminate due to a boundary absorption. A scheme similar to the one just described is used for generating boundary conditions along curves bounding two-dimensional regions. For problems formulated for regions described by more than two coordinates, it is perhaps most convenient to sample the coordinates of the path of the random walk, store their value at either time  $t_b$  or 0, and insert these coordinates into a digital computer where suitable functions may be calculated for averaging.



**Figure 4.10** Hybrid Computer Arrangement for Introducing Initial Conditions of the Partial Differential Equation into a Monte Carlo Solution

4.5C Measurement of the Average Time of a Random Walk for Use in the Solution of Nonhomogeneous Partial Differential Equations

The average time  $T(\underline{r}_i, t_i)$  of a random walk, satisfies the partial differential equation (see Appendix B),

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right] T(\underline{r}_i, t_i) = -1 \quad (4.25)$$

with the boundary condition,

$$T(\underline{r}_b, t_i) = 0 \quad (4.26)$$

and the initial condition

$$T(\underline{r}_i, 0) = 0 \quad (4.27)$$

Nonhomogeneous partial differential equations can be decomposed into two partial differential equations (see Section 2.7), one of which satisfies an equation similar to (4.25). Therefore, a Monte Carlo solution may be obtained for (4.25) by measuring the average time of a random walk. This measurement is made by converting the time interval of a random walk into a voltage, and inserting this voltage into an averaging filter. Figure 4.11 is the analog-hybrid computer diagram showing this measurement for Poisson's equation in one dimension.

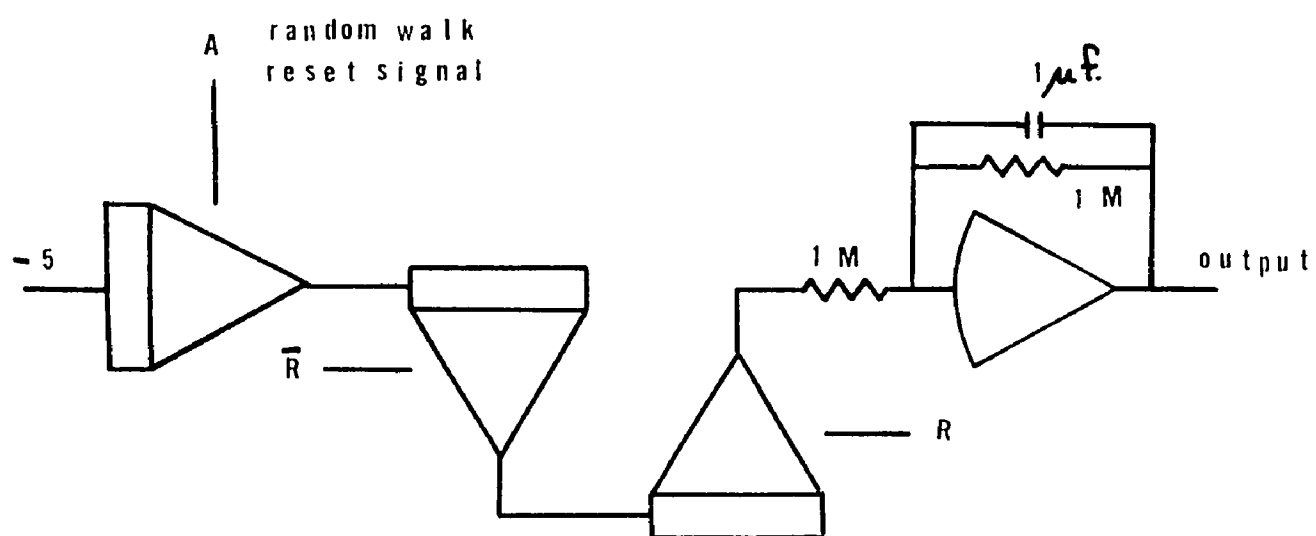


Figure 4.11 Average Random Walk Time Measurement

## CHAPTER 5

### MONTE CARLO SOLUTION OF ILLUSTRATIVE PROBLEMS

#### 5.1 Introduction

In this chapter, the Monte Carlo solutions of both parabolic and elliptical partial differential equations are presented. Problems were chosen for which solutions are known either in an analytic or graphical form, so that comparisons with Monte Carlo results could be made. Although Monte Carlo estimates of solutions were obtained for one- and two-dimensional problems, no theoretical difficulty obstructs the extension of the method to problems involving more than two dimensions. The limitation to two dimensions in the present study was necessary, since there was no access to a digital computer, necessary for function generation.

#### 5.2 Monte Carlo Solutions of One-dimensional Problems

Five examples are presented in this section in tabular form along with the hybrid computing setup used to obtain the Monte Carlo estimates of the solution of the partial differential equations. Each example lists the statement and region of definition of the problem and the desired solution; if a nongraphical solution of the problem is available, it will also be given. The stochastic

differential equations (Langevin equations) are presented along with both the integrator gain and noise source characteristics which determine the time scale factor.

The sweep speed specifies the rate at which the horizontal axis voltage is scanned and the averaging time constant is the time constant of the low pass filter used for analog averaging. The maximum errors specify both the systematic error which is the difference between the exact solution and the smoothed curve and the statistical fluctuations which denote the variation of the Monte Carlo solution about the smoothed curve.

Example 1

Statement of problem:

$$\nabla^2 U(x) = 0$$

$$U(-10) = -10$$

$$U(+10) = +10$$

Desired solution:

U(x) for all  $-10 < x < +10$

Langevin equation:

$$\frac{dx}{dt} = G N(t)$$

Sweep speed:

0.1 volts/second

Region of definition:

$$-10 < x < +10$$

Exact solution:

$$U(x) = x$$

where

$$G = 10^5$$

$$E[N(t)N(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

Averaging time constant:

1 second

Maximum errors:

- (a) Systematic errors 4.3%
- (b) Statistical fluctuations 3.5%

The analog-hybrid computer diagrams for both Examples 1 and 2 are identical and are shown in Fig. 5.1. The solution to the Laplace equation given in Example 1 appears in Fig. 5.2. A Monte Carlo digital solution of this problem utilizing the uncommitted counters and free digital logic of ASTRAC II was obtained, the results of which are given in the error curve of Fig. 5.3; a maximum total error of 2.6% was obtained.

Example 2Statement of problem:

$$\nabla^2 U(x,t) = \frac{\partial U}{\partial t}$$

$$U(-10,t) = -10$$

$$U(+10,t) = +10$$

$$U(x,0) = 0$$

Region of Definition:

$$-10 < x < +10$$

$$0 < t$$

Desired solution:

$U(x,t)$  for all  $-10 < x < +10$  and for

$t = 12, t = 6, t = 1.5$ , seconds

Exact solution:

$$U(x,t) = x + \frac{20}{\pi} \sum_{n=1}^{\infty} \frac{(-1)^n}{n} e^{-\left(\frac{n\pi}{10}\right)^2 t} \sin \frac{n\pi x}{20}$$

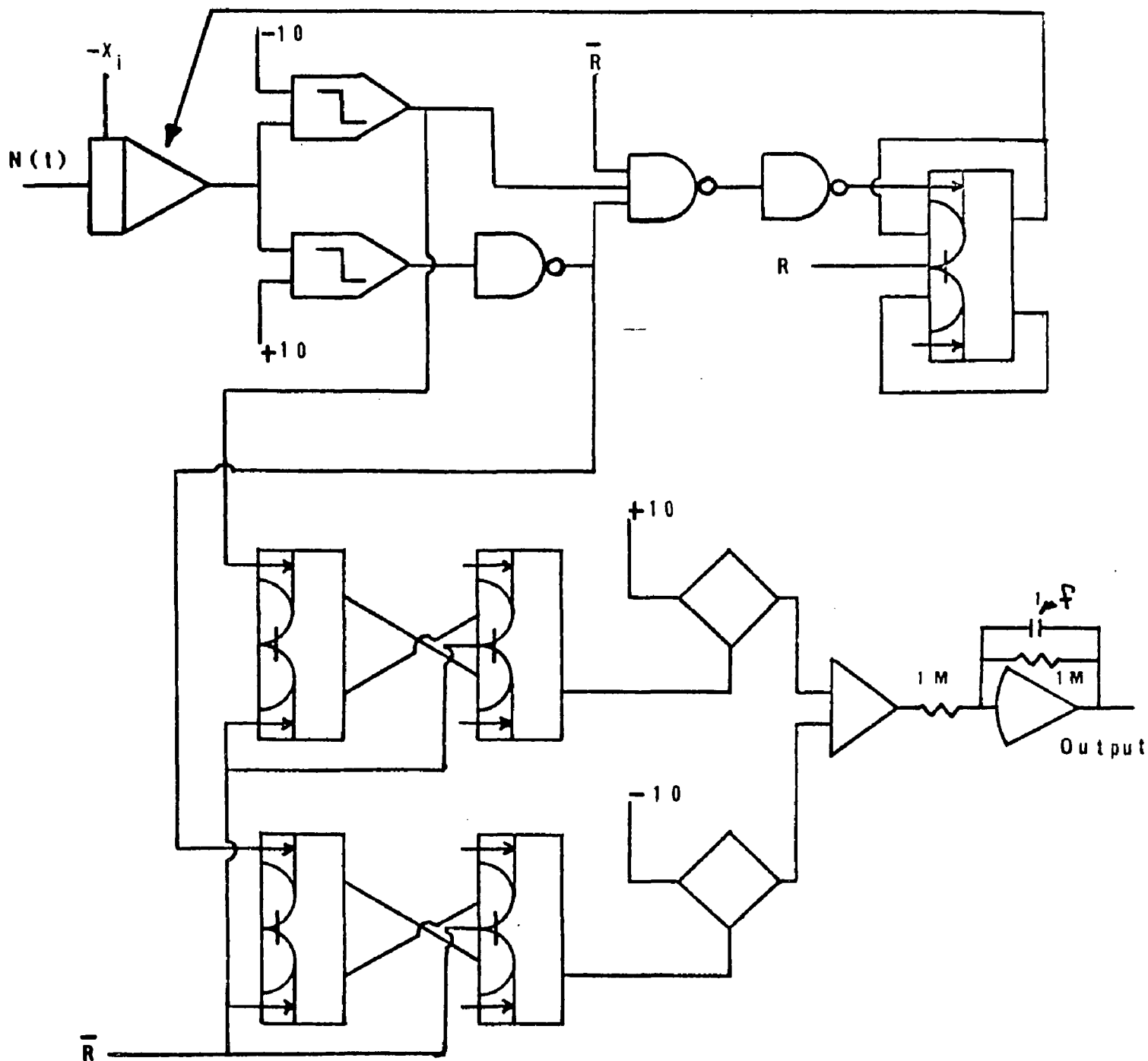


Figure 5.1 Hybrid Computer Diagram for the Monte Carlo Solution of Examples 1 and 2

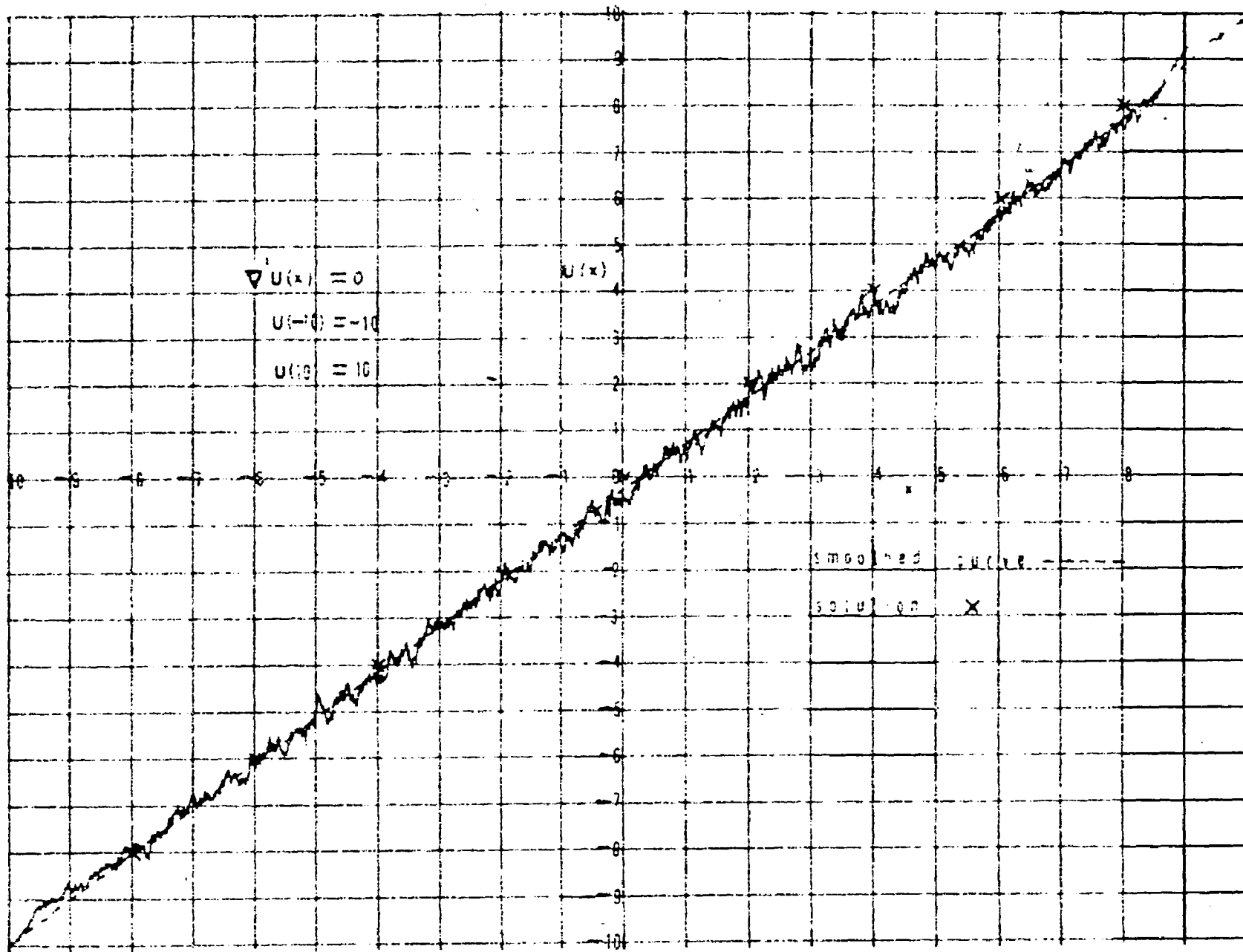


Figure 5.2 Monte Carlo Solution of Example 1

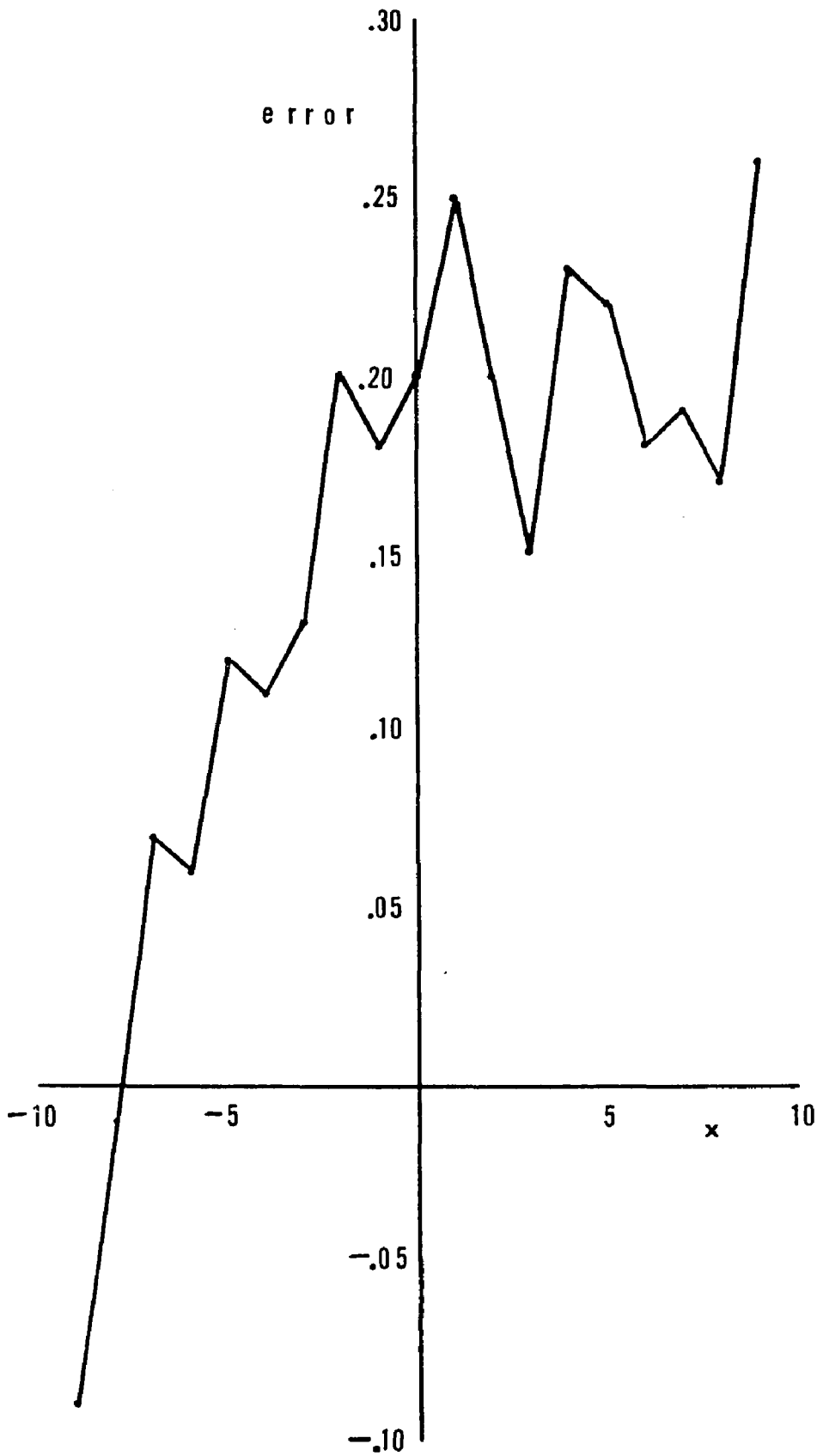


Figure 5.3 Digital Monte Carlo Error Solution of Example 1

Langevin equation:

$$\frac{dx}{dt} = G N(t)$$

where

$$G = 10^5$$

$$E [N(t)N(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

Time scale factor,  $T_s$ :

$$T_s = t/\tau = 15 \cdot 10^4$$

where  $t$  is real time and  $\tau$  is computer time

Sweep speed:

0.1 volts/second

Averaging time constant:

1 second

Maximum errors:

(a) Systematic errors 6.4%

(b) Statistical fluctuations 2.2%

The analog-hybrid computer diagram for this problem appears in Fig. 5.1 and the Monte Carlo solutions are given in Fig. 5.4.

Example 3Statement of problem:

$$\nabla^2 U(x,t) = \frac{\partial U}{\partial t}$$

$$\left. \frac{\partial U}{\partial x} \right|_{x=0} = 0$$

$$U(10,t) = 10$$

$$U(x,0) = 0$$

Region of definition:

$$0 < x < +10$$

$$0 < t$$

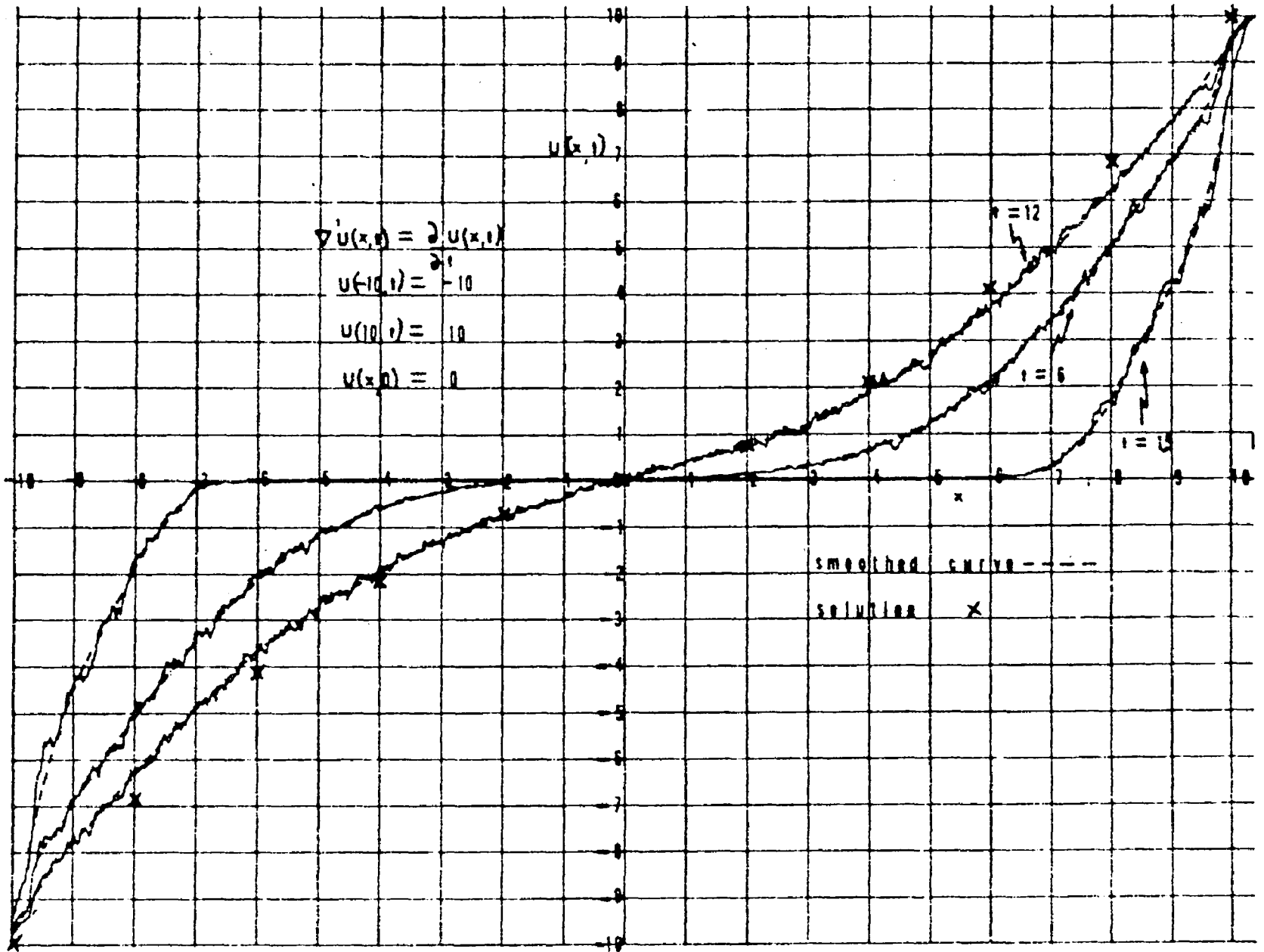


Figure 5.4 Monte Carlo Solution of Example 2

Desired solution:

$U(x,t)$  for all  $0 < x < +10$  and for

$t = 7.5, t = 22.5, t = 60, t = 144$

Exact solution:

$$U(x,t) = 10 - \frac{40}{\pi} \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{2n-1} e^{-\left[\frac{(2n-1)\pi}{20}\right]^2 t} \cos \frac{(2n-1)\pi x}{20}$$

Langevin equation:

$$\frac{dx}{dt} = G N(t)$$

where

$$G = 10^5$$

$$E[N(t)N(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

Time scale factor,  $T_s$ :

$$T_s = t/\tau = 15 \cdot 10^4$$

where  $t$  is real time and  $\tau$  is computer time

Sweep speed:

0.1 volts/second

Averaging time constant:

1 second

Maximum errors:

(a) Systematic errors 11.6% for  $t = 7.5$  seconds

(b) Statistical fluctuations 4.2% for  $t = 7.5$  seconds

The analog-hybrid computer diagram for this problem appears in Fig. 5.5 and the Monte Carlo solutions are given in Fig. 5.6.

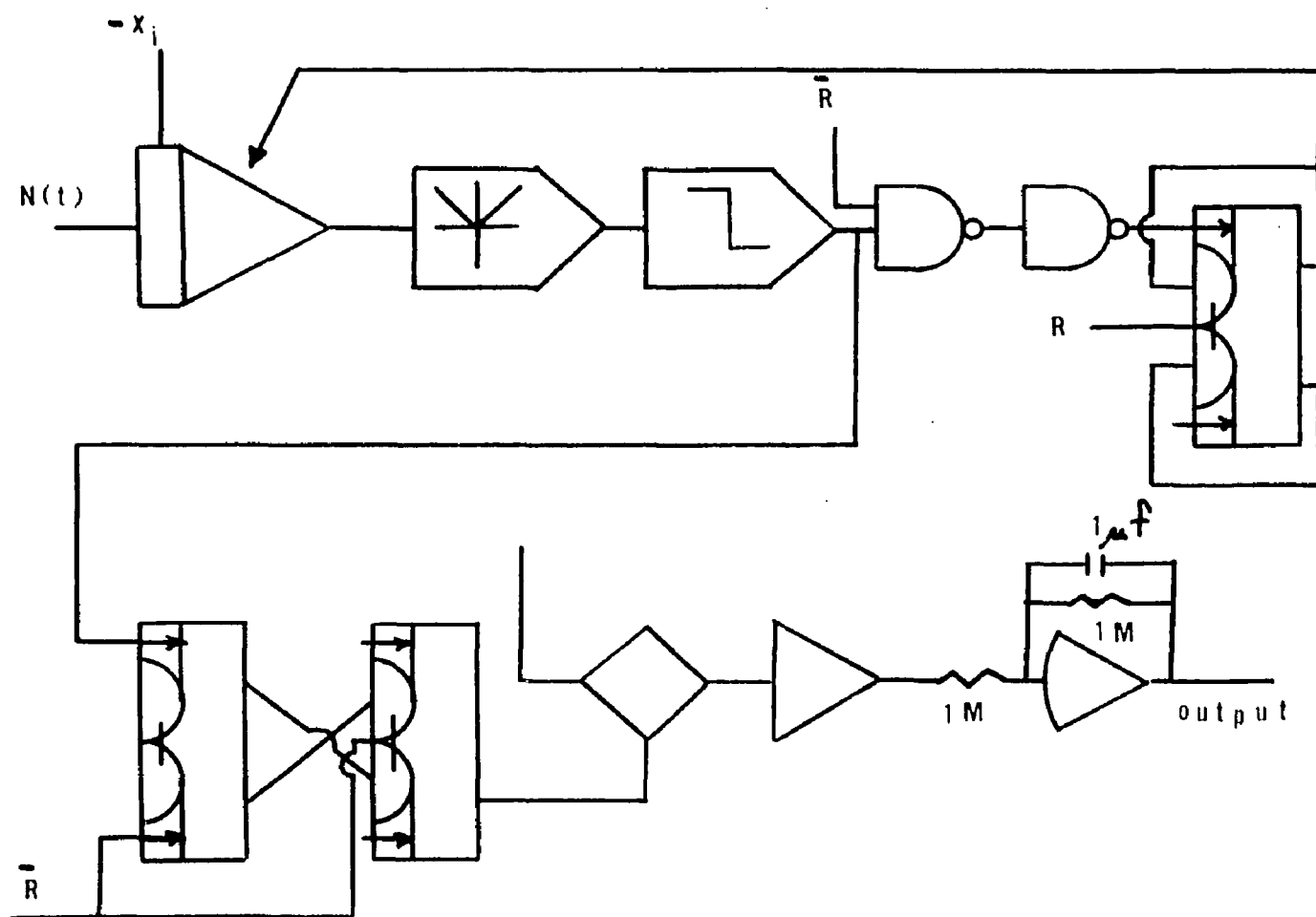


Figure 5.5 Hybrid Computer Diagram for the Monte Carlo Solution of Example 3

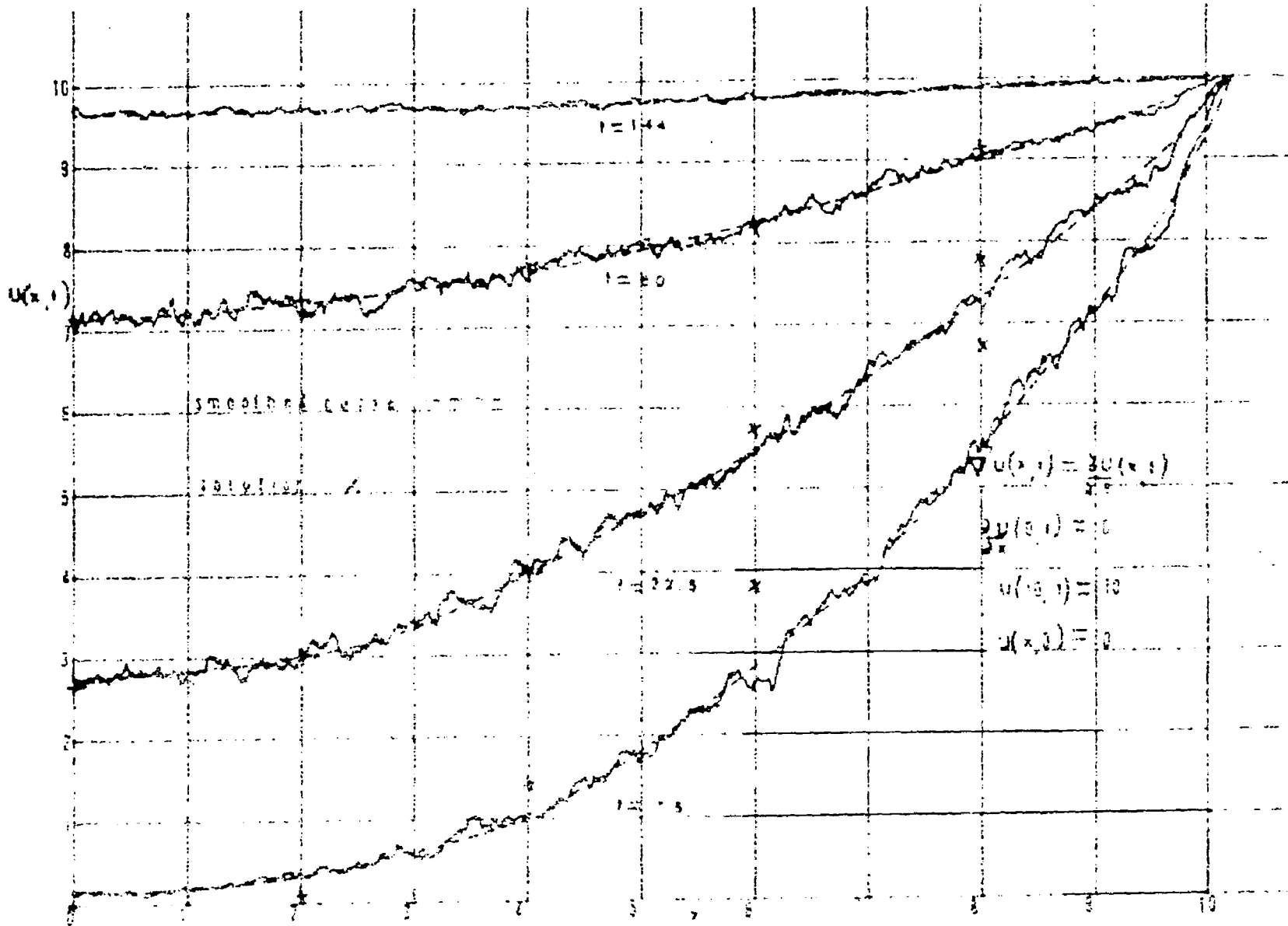


Figure 5.6 Monte Carlo Solution of Example 3

Example 4Statement of problem:

$$\nabla^2 U(x,t) = \frac{\partial U}{\partial t}$$

$$U(-10,t) = -10$$

$$U(+10,t) = +10$$

$$U(x,0) = 0$$

Region of definition:

$$-10 < x < +10$$

$$0 < t$$

Desired solutions:

$$(a) U(5,t)$$

$$(b) U(7.5,t)$$

Exact solutions:

$$(a) U(5,t) = 5 + \frac{20}{\pi} \sum_{n=1}^{\infty} \frac{(-1)^n}{n} e^{-\left(\frac{n\pi}{10}\right)^2 t} \sin \frac{n\pi}{4}$$

$$(b) U(7.5,t) = 7.5 + \frac{20}{\pi} \sum_{n=1}^{\infty} \frac{(-1)^n}{n} e^{-\left(\frac{n\pi}{10}\right)^2 t} \sin \frac{3n\pi}{8}$$

Langevin equation:

$$\frac{dx}{dt} = G N(t)$$

where

$$G = 3/5 \cdot 10^5$$

$$E[N(t)N(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

Time scale factor,  $T_s$ :

$$T_s = t/\tau = 5.4 \cdot 10^4$$

where  $t$  is real time and  $\tau$  is computer time

Sweep speed:

(a) 200 seconds/millisecond

(b) 300 seconds/millisecond

Averaging time constant:

1 second

Maximum errors:

(a) Systematic errors 5% for problem a, 6.3% for problem b.

(b) Statistical fluctuations 4% for problem a, 2.3% for problem b.

The analog-hybrid computer diagram for the Monte Carlo solution of this problem is similar to the one shown in Fig. 5.1, the only difference is in the generation of the integrator reset signal,  $a$ . Since the solution is desired at fixed points in space as a function of time, it is necessary to vary the duty cycle of the reset signal,  $R$ . The analog computer diagram showing the generation of  $a$ , the variable timing signal, is given in Fig. 5.7. The Monte Carlo estimates of the solution of the problems posed in Example 4, are presented in Figs. 5.8 and 5.9.

Example 5Statement of problem:

$$\nabla^2 U(x) = 1/15$$

$$U(-10) = +10$$

$$U(+10) = -10$$

Desired solution:

$$U(x) \text{ for all } -10 < x < +10$$
Region of definition:

$$-10 < x < +10$$

Exact solution:

$$U(x) = \frac{x^2 - 100}{30} - x$$

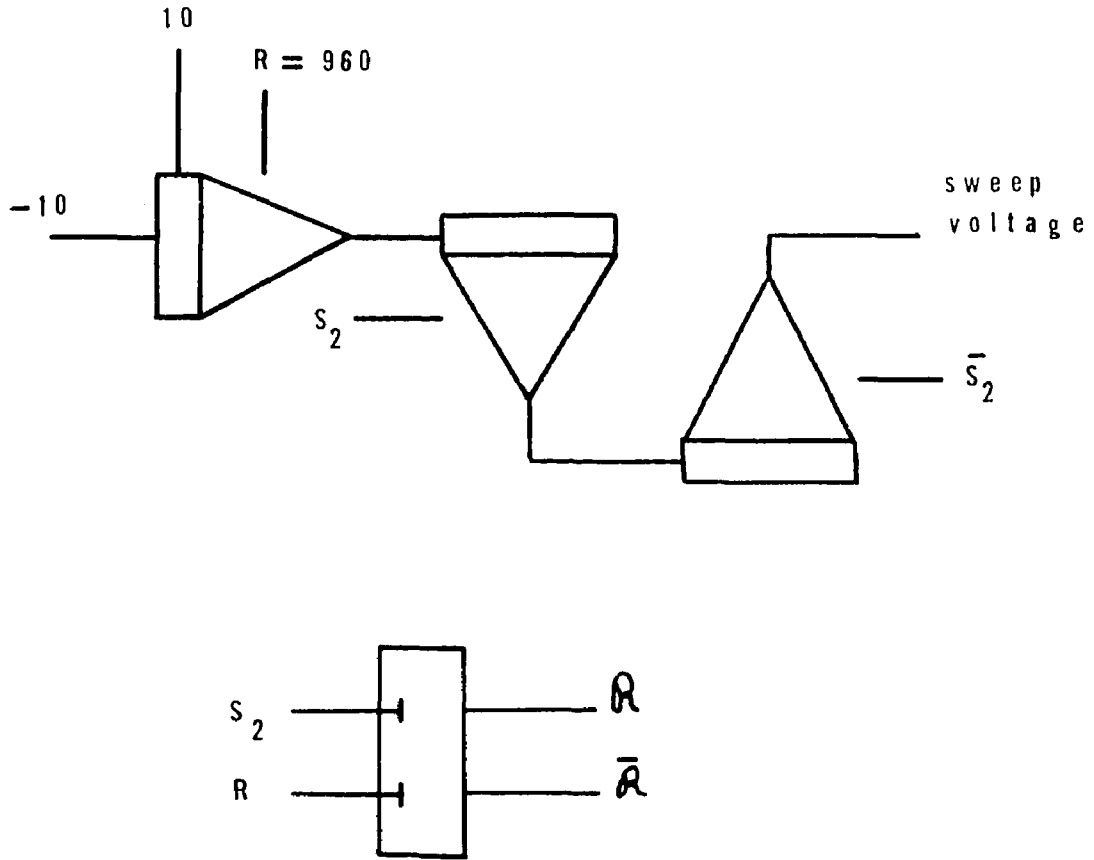


Figure 5.7 Generation of the Variable Timing Signal

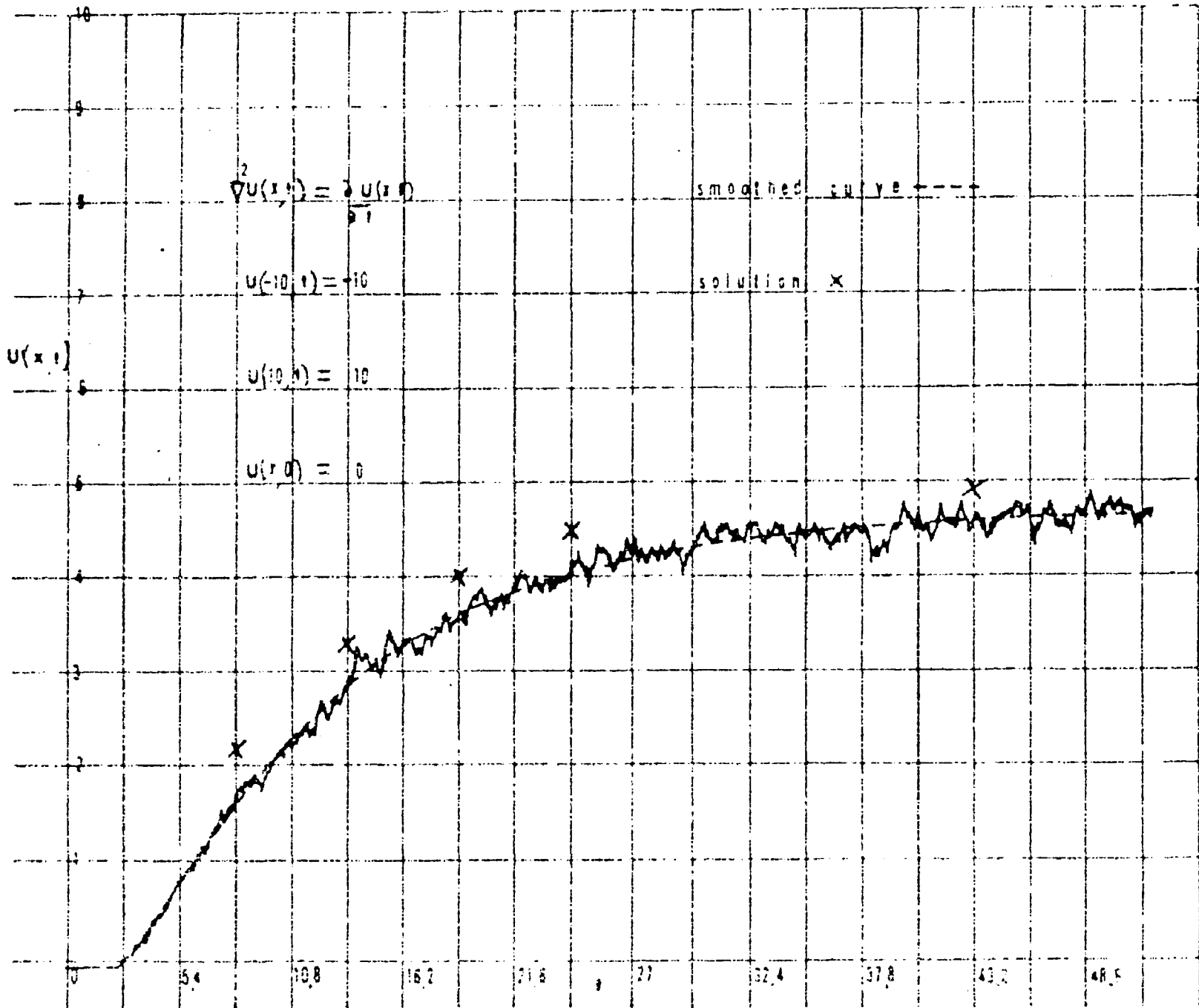


Figure 5.8 Monte Carlo Solution of Example 4a

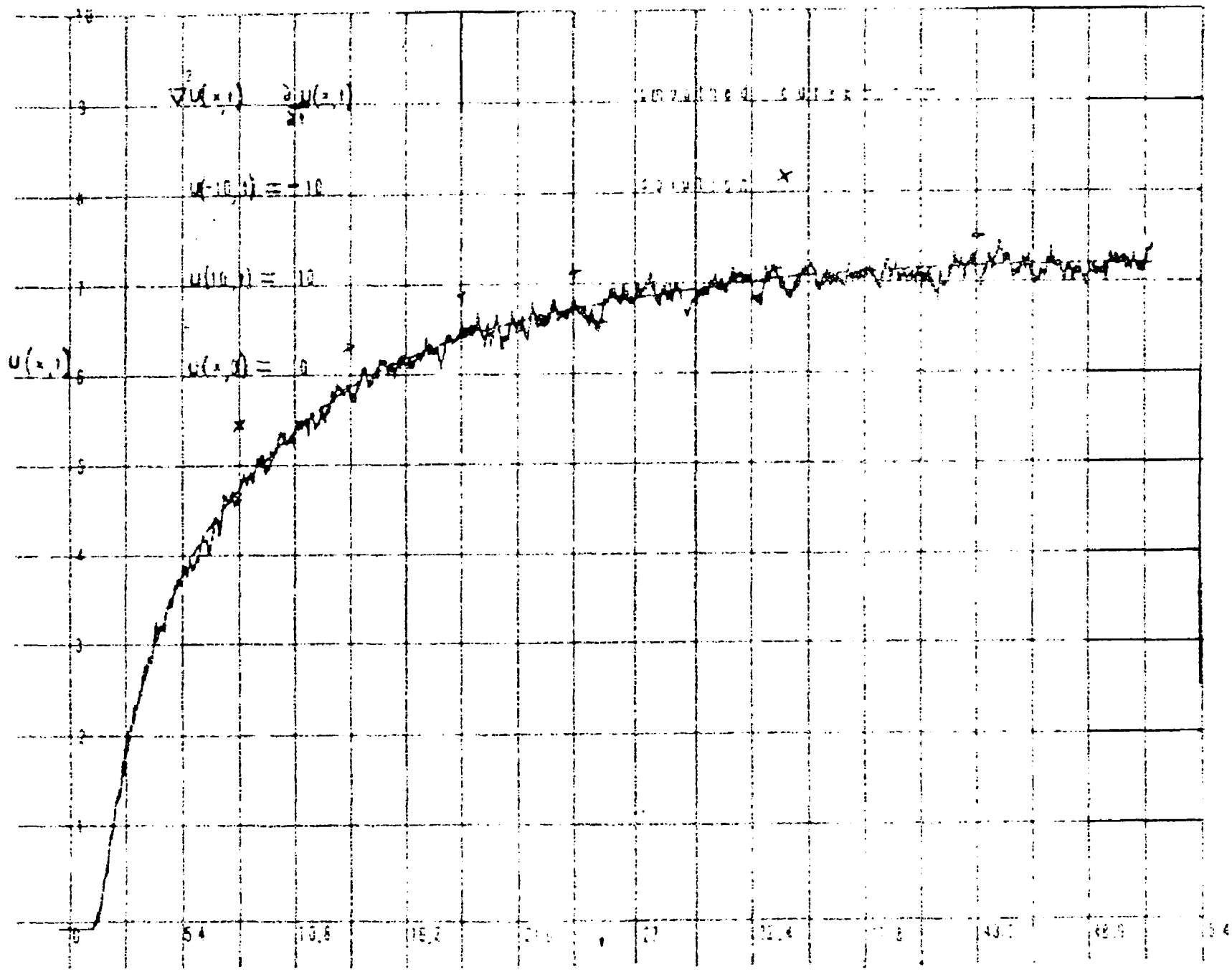


Figure 5.9 Monte Carlo Solution of Example 4b

Langevin equation:

$$\frac{dx}{dt} = G N(t)$$

where

$$G = 10^5$$

$$E[N(t)N(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

Sweep speed:

0.1 volts/second

Averaging time constant:

1 second

Maximum errors:

(a) Systematic errors 7%

(b) Statistical fluctuations

3.8%

The Monte Carlo solution of Example 5 was solved by the average time measurements described in Section 2.9. The analog-hybrid computer diagram for the Monte Carlo solution of this problem appears in Fig. 5.10 and the Monte Carlo solution is given in Fig. 5.11.

5.3 Monte Carlo Solutions of Two-dimensional ProblemsExample 6Statement of problem:

$$\nabla^2 U(x,y) = 0$$

$$U(x,y) \Big|_{x^2 + y^2 = 100} = 10$$

$$U(x,y) \Big|_{x^2 + y^2 = 16} = -10$$

Region of definition:

$$16 < x^2 + y^2 < 100$$

Exact solution:

$$u(x,y) = 10 \frac{\ln(y^2/40)}{\ln(10/4)}$$

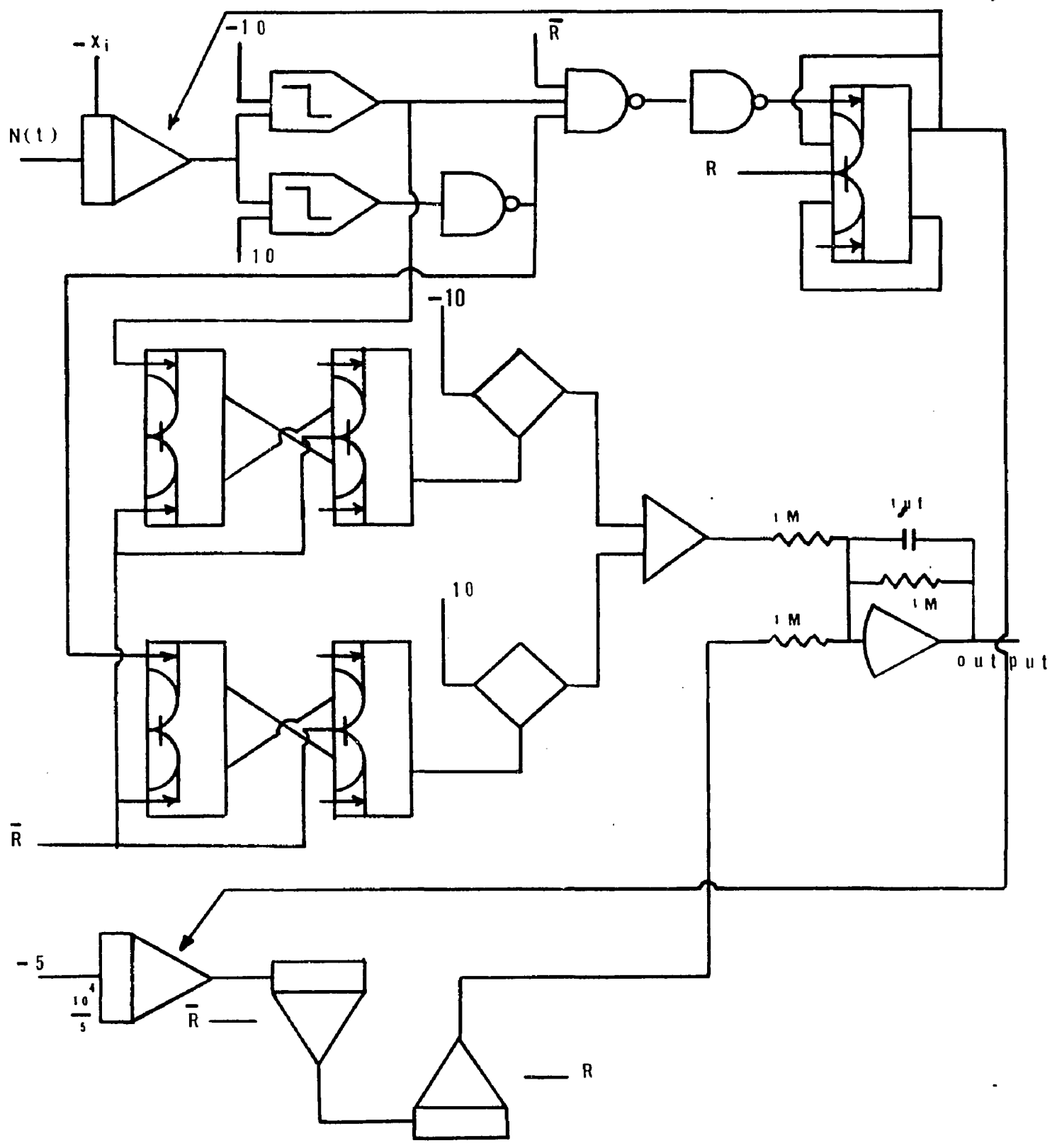


Figure 5.10 Hybrid Computer Diagram for the Monte Carlo Solution of Example 5

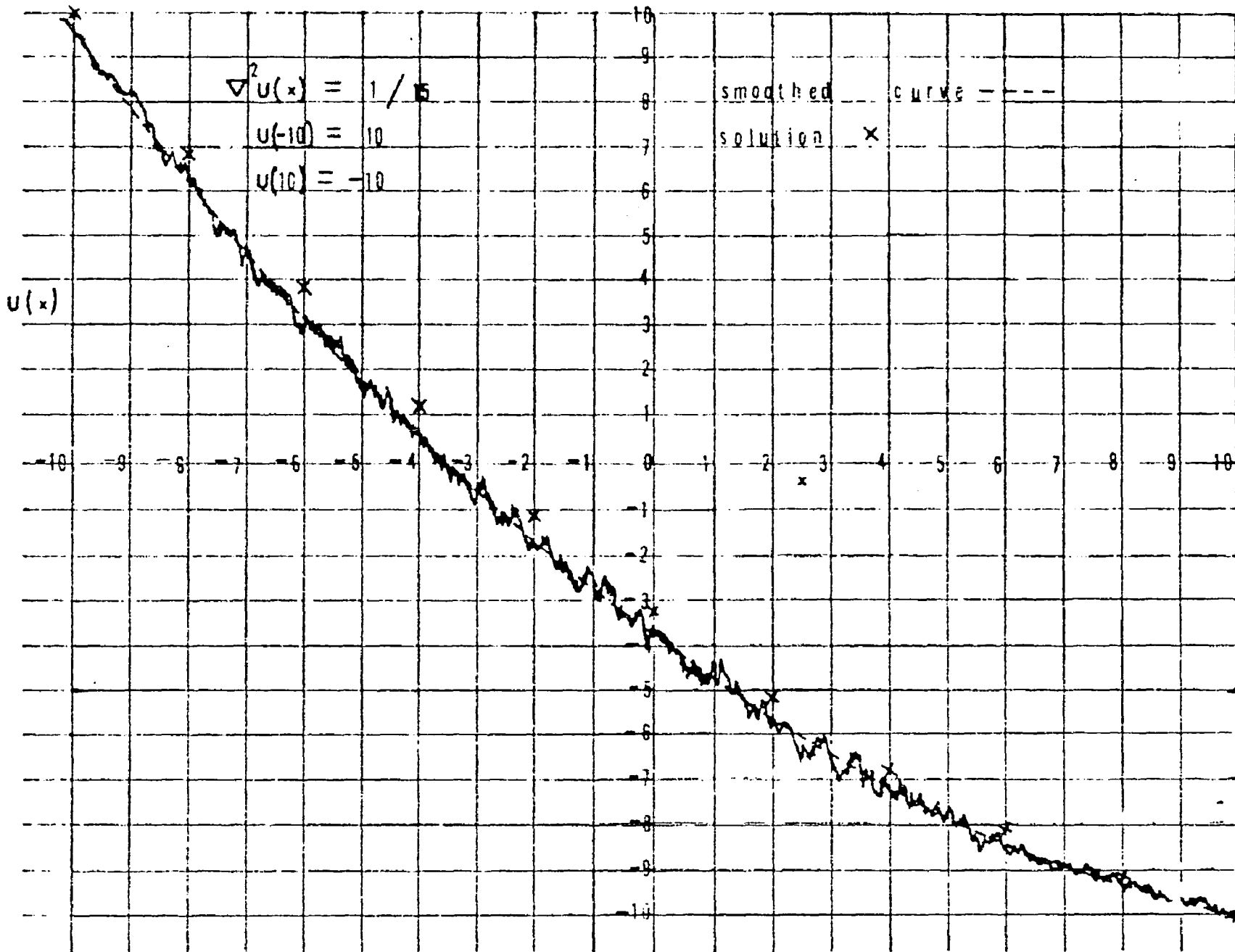


Figure 5.11 Monte Carlo Solution of Example 5

Desired solution:

$U(x,y)$  along the line  $x = 0$

Langevin equations:

$$\frac{dx}{dt} = G N_1(t)$$

$$\frac{dy}{dt} = G N_2(t)$$

where

$$G = 10^5$$

$$E[N_i(t)N_i(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

$$E[N_1(t)N_2(t+\lambda)] = 0$$

Sweep speed:

0.1 volts/second

Averaging time constant:

1 second

Maximum errors:

(a) Systematic errors 9.6%

(b) Statistical fluctuations 4.7%

The analog-hybrid computer diagram for the Monte Carlo solution of this problem appears in Fig. 5.12 and the solution which was obtained is shown in Fig. 5.13.

Example 7Statement of problem:

$$\nabla^2 U(x,y) = 0$$

$$U(x,y) \Big|_{x^2 + y^2 = 100} = \begin{cases} +10 & \text{for } y > 0 \\ -10 & \text{for } y < 0 \end{cases}$$

Region of definition:

$$0 < x^2 + y^2 < 100$$

Desired solution:

$U(x,y)$  along the line

(a)  $y = 0$

(b)  $x = 0$

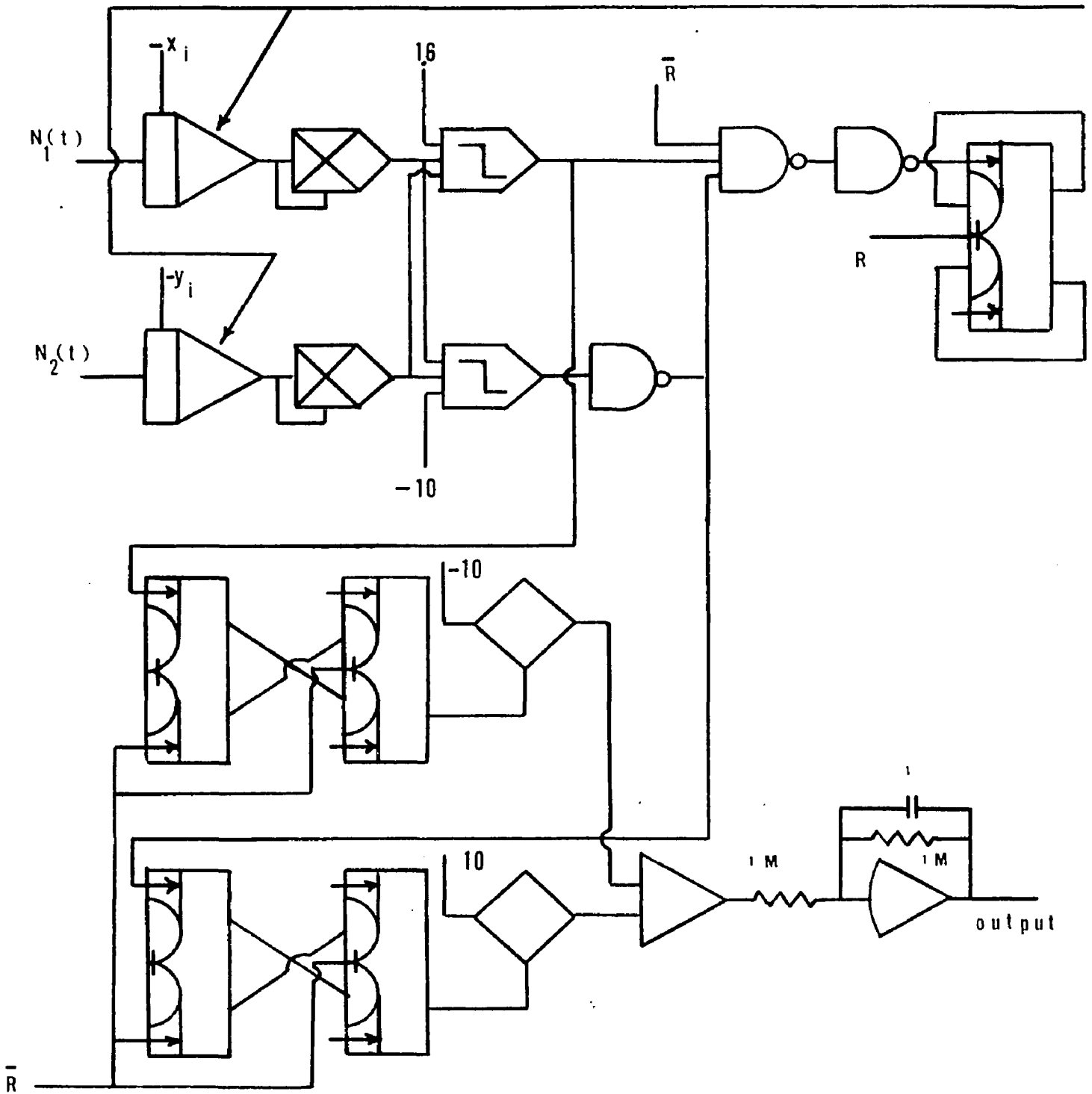


Figure 5.12 Hybrid Computer Diagram for the Monte Carlo Solution of Example 6

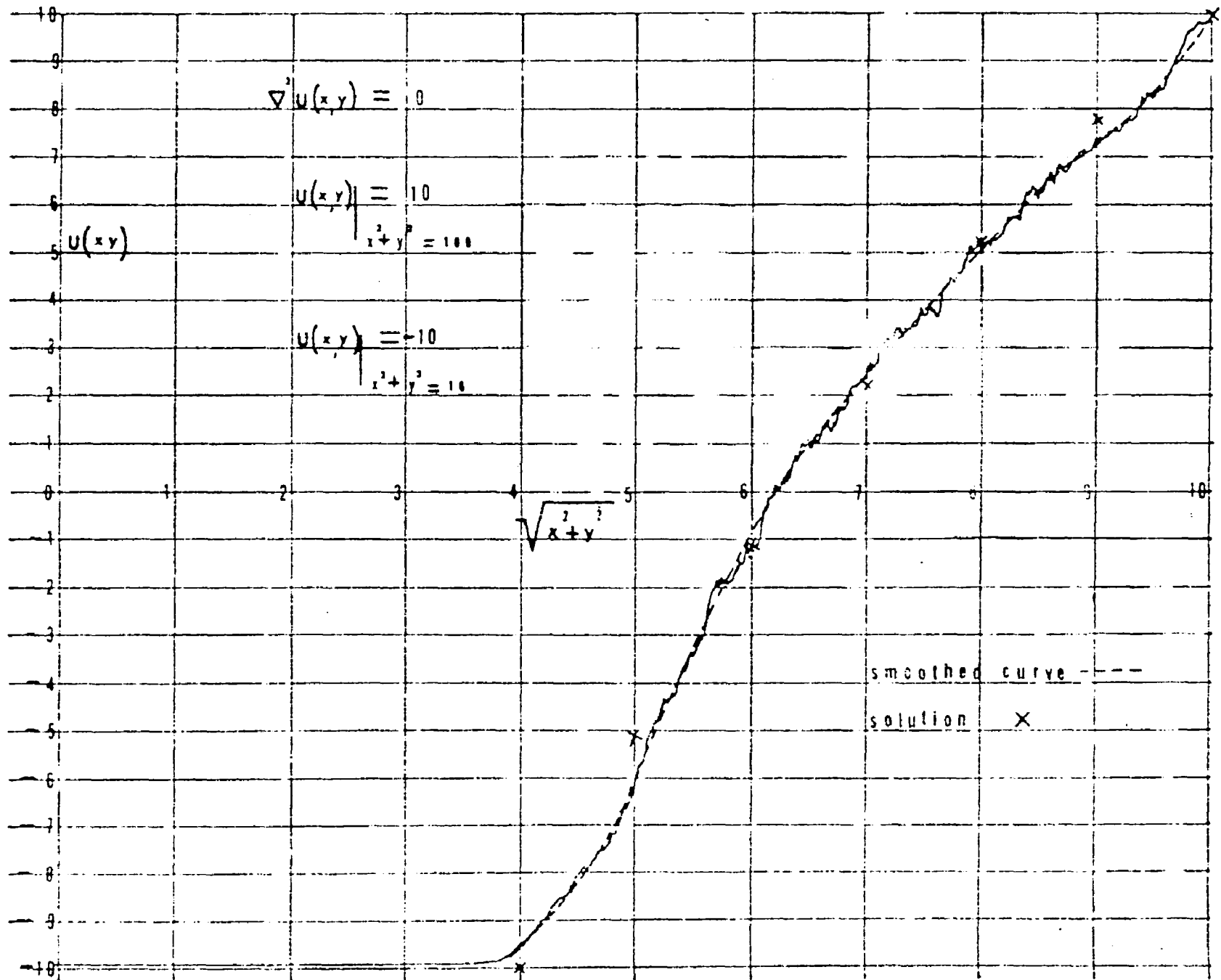


Figure 5.13 Monte Carlo Solution of Example 6

Exact solution:

$$U(x,y) = 10 - \frac{20}{\pi} \tan^{-1} \frac{100 - (x^2 + y^2)}{20y} \text{ for } y > 0$$

$$U(x,y) = -U(x,-y)$$

Langevin equations:

$$\frac{dx}{dt} = G N_1(t)$$

where

$$G = 10^5$$

$$\frac{dy}{dt} = G N_2(t)$$

$$E[N_1(t)N_1(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

$$E[N_1(t)N_2(t+\lambda)] = 0$$

Sweep speed:

0.1 volts/second

Averaging time constant:

1 second

Maximum errors:

(a) Systematic errors 4.6%

(b) Statistical fluctuations 5.4%

The analog-hybrid computer diagram for the Monte Carlo solution of this problem appears in Fig. 5.14 and the solutions which were obtained appear in Fig. 5.15.

Example 8Statement of problem:

$$\begin{aligned} \nabla^2 U(x,y,t) &= \frac{\partial U}{\partial t} \\ U(x,y,t) \Big|_{t=0} &= 10 \\ U(x,y,t) \Big|_{x^2 + y^2 = 100} &= 0 \end{aligned}$$

$$U(x,y,0) = 0$$

Region of definition:

$$0 < x^2 + y^2 < 100$$

$$0 < t$$

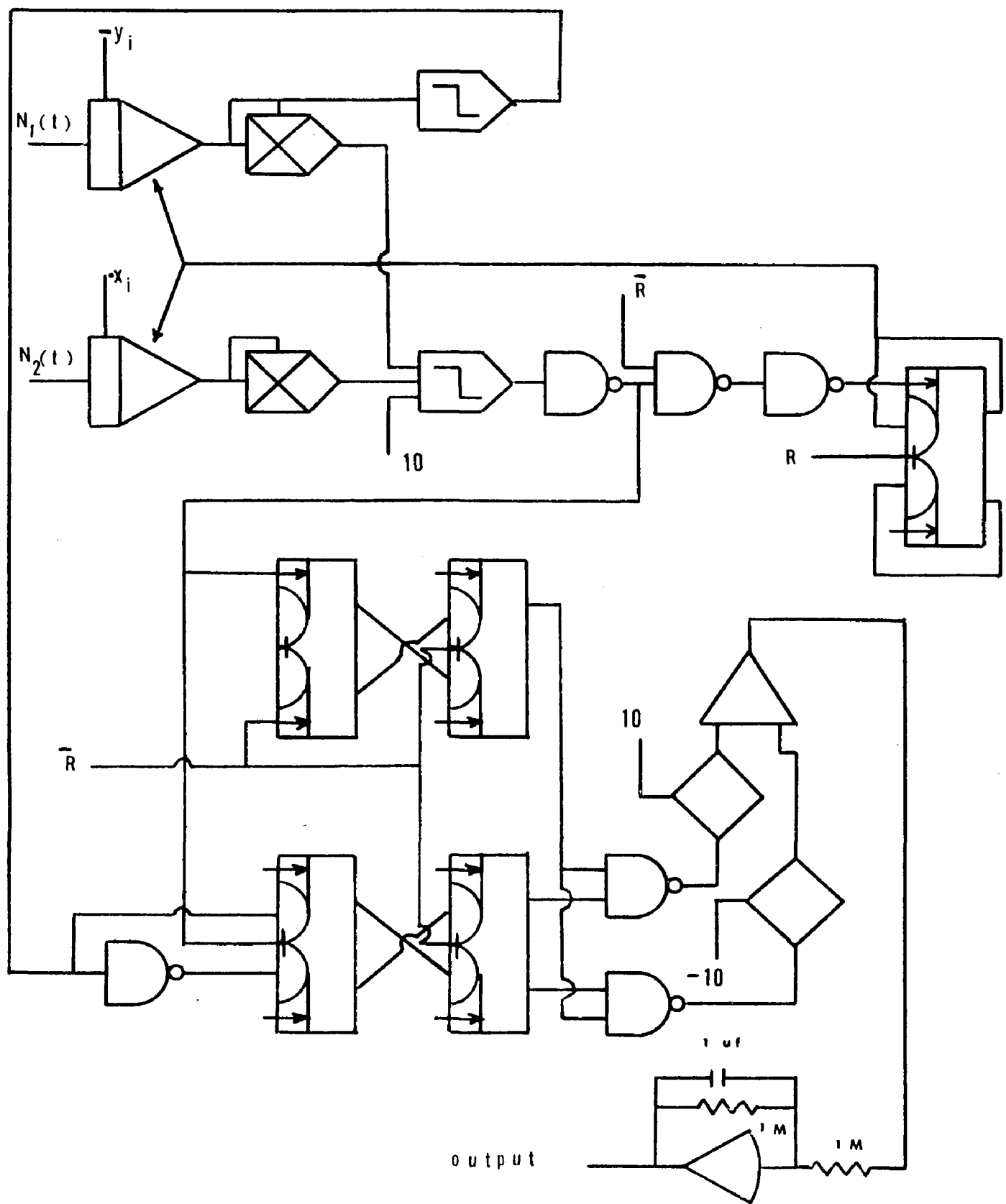


Figure 5.14 Hybrid Computer Diagram for the Monte Carlo Solution of Example 7

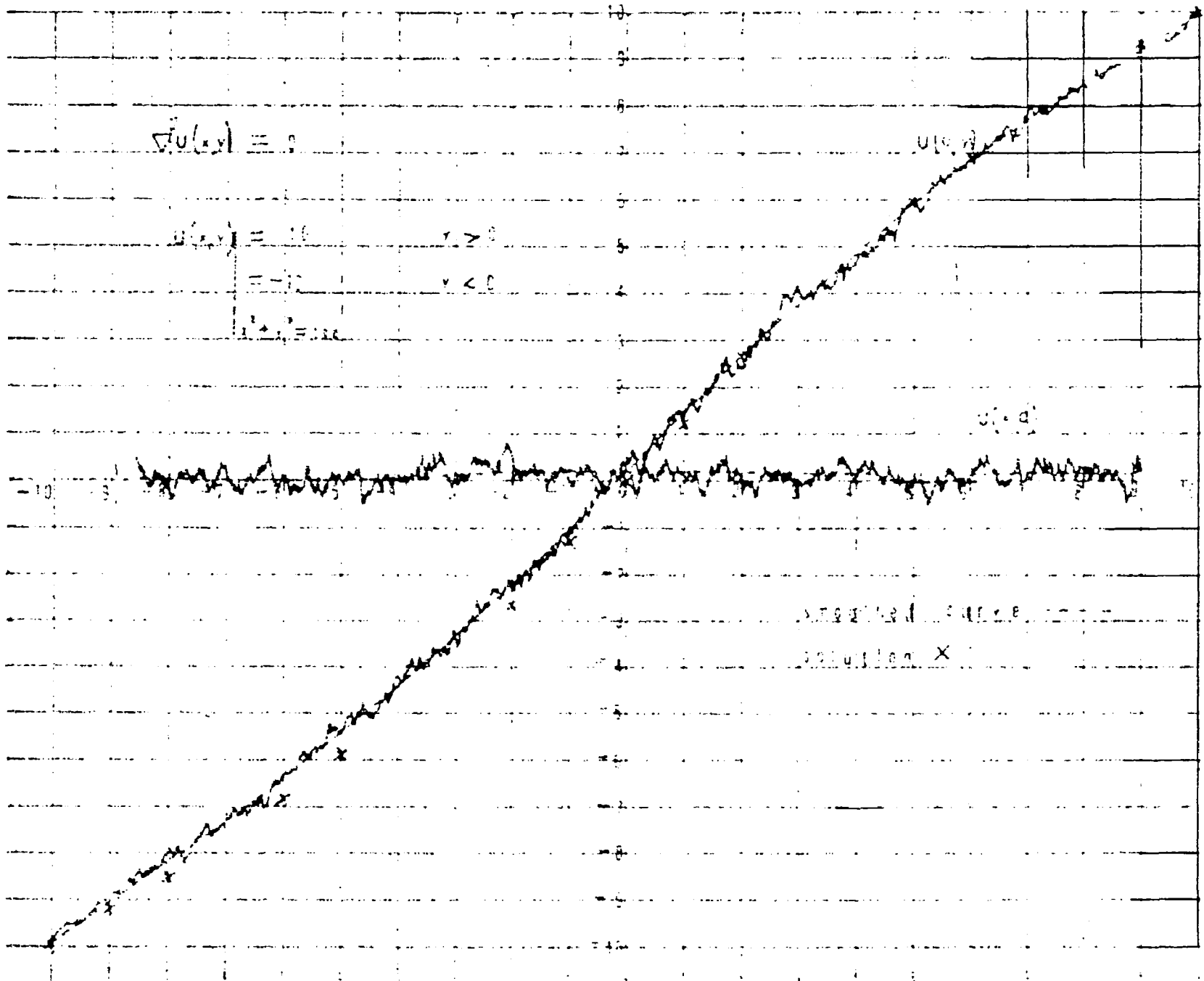


Figure 5.15 Monte Carlo Solution of Example 7

Desired solutions:

(a)  $U(0,y,t)$  for  $t = 3.75, t = 7.5, t = 15, t = 22.5,$   
 $t = 37.5, t = 75$  seconds

(b)  $U(0,0,t)$

Exact solution:<sup>23</sup>

$$U(x,y,t) = 10 - \frac{1}{5} \sum_{j=1}^{\infty} \frac{J_0(\lambda_j r)}{[J_1(10\lambda_j)]^2} e^{-(\lambda_j t)} \int_0^{10} p J_0(\lambda_j p) dp$$

where  $J$  denotes the Bessel function and  $r^2 = x^2 + y^2$

Langevin equations:

$$\frac{dx}{dt} = G N_1(t)$$

where

$$G = 10^5$$

$$\frac{dy}{dt} = G N_2(t)$$

$$E[N_i(t)N_i(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

$$E[N_1(t)N_2(t+\lambda)] = 0$$

Time scale factor,  $T_s$ :

$$T_s = t/\tau = 15 \cdot 10^4$$

where  $t$  is real time and  $\tau$  is  
 computer time

Averaging time constant:

1 second

Sweep speed:

(a) 300 seconds/millisecond

(b) 0.1 volts/second

Maximum errors:

(a) Systematic errors problem (a) 5%, problem (b) 2.9%

(b) Statistical fluctuations problem (a) 2.8%, problem (b)  
 1.9%

The analog-hybrid computer diagram for the Monte Carlo solution of the problem given in Example 8 appears in Fig. 5.16 and the solutions which were obtained are shown in Figs. 5.17 and 5.18.

Example 9

Statement of problem:

$$\nabla^2 U(x, y) = 0$$

$$U(x, 0) = +10$$

$$U(0, y) = U(6, y) = U(x, 10) = -10$$

Desired solution:

$$U(3, y)$$

Exact solution:<sup>24</sup>

$$U(x, y) = -10 + 40 \sum_{n=1}^{\infty} \frac{\sinh(\frac{n\pi}{6})(10-y)}{(n\pi) \sin(\frac{n\pi 5}{3})} \sin(\frac{n\pi x}{6}) [1 + (-1)^{n-1}]$$

Langevin equations:

$$\frac{dx}{dt} = G N_1(t)$$

where

$$G = 10^5$$

$$\frac{dy}{dt} = G N_2(t)$$

$$E[N_i(t)N_i(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

$$E[N_i(t)N_i(t+\lambda)] = 0$$

Sweep speed:

0.1 volts/second

Averaging time constant:

1 second

Maximum errors:

(a) Systematic errors 5%

(b) Statistical fluctuations 2.6%

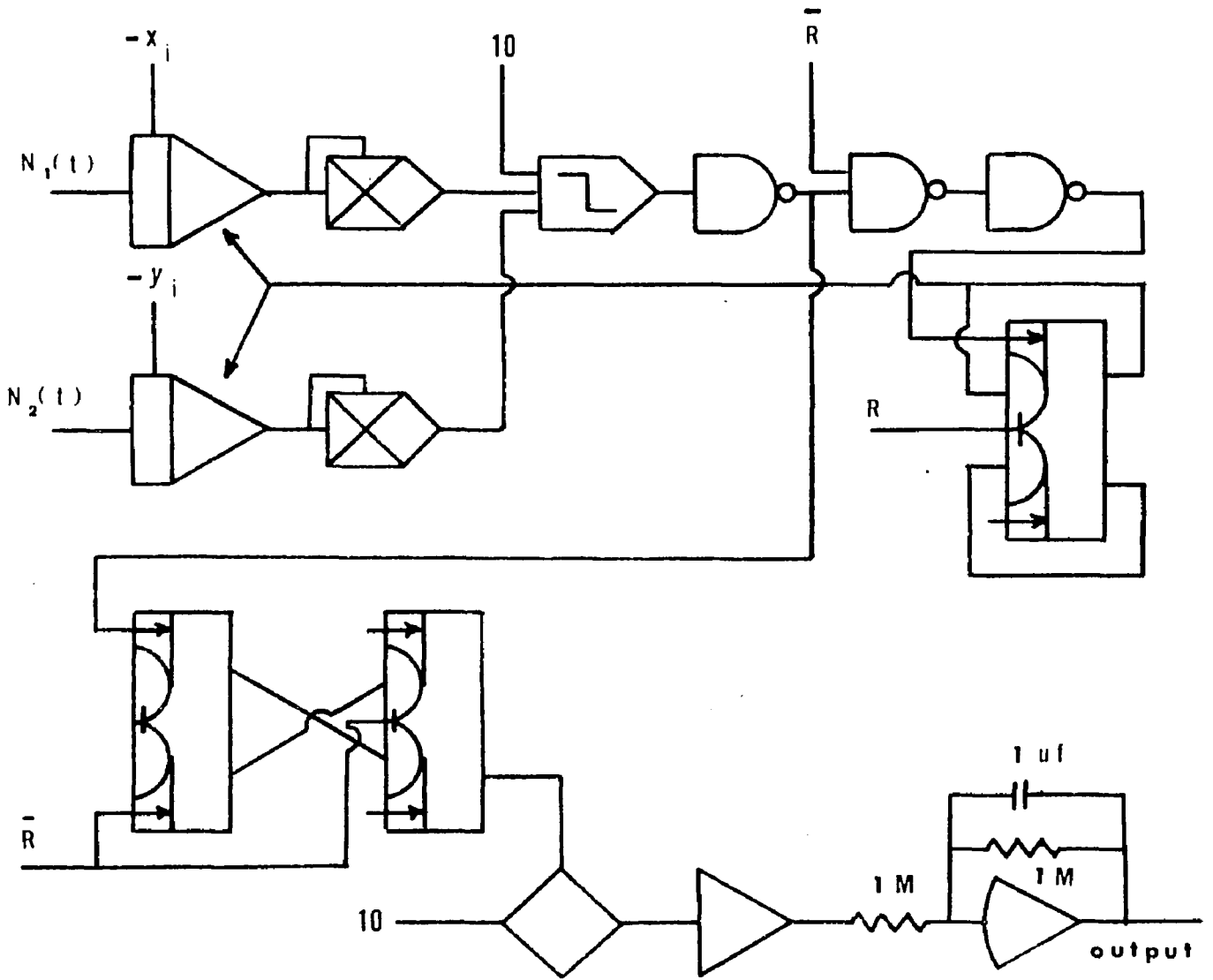


Figure 5.16 Hybrid Computer Diagram for the Monte Carlo Solution of Example 8

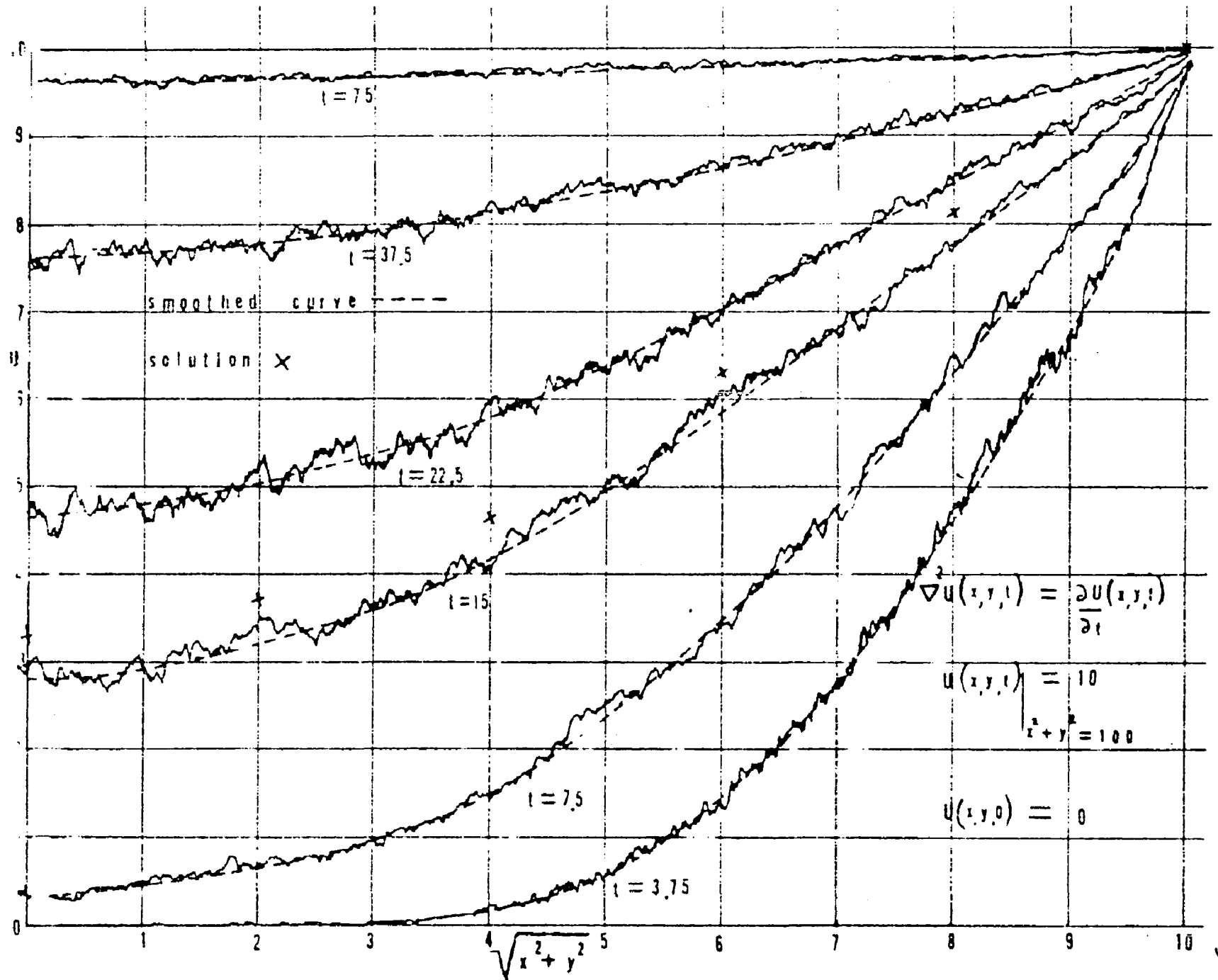


Figure 5.17 Monte Carlo Solution of Example 8a

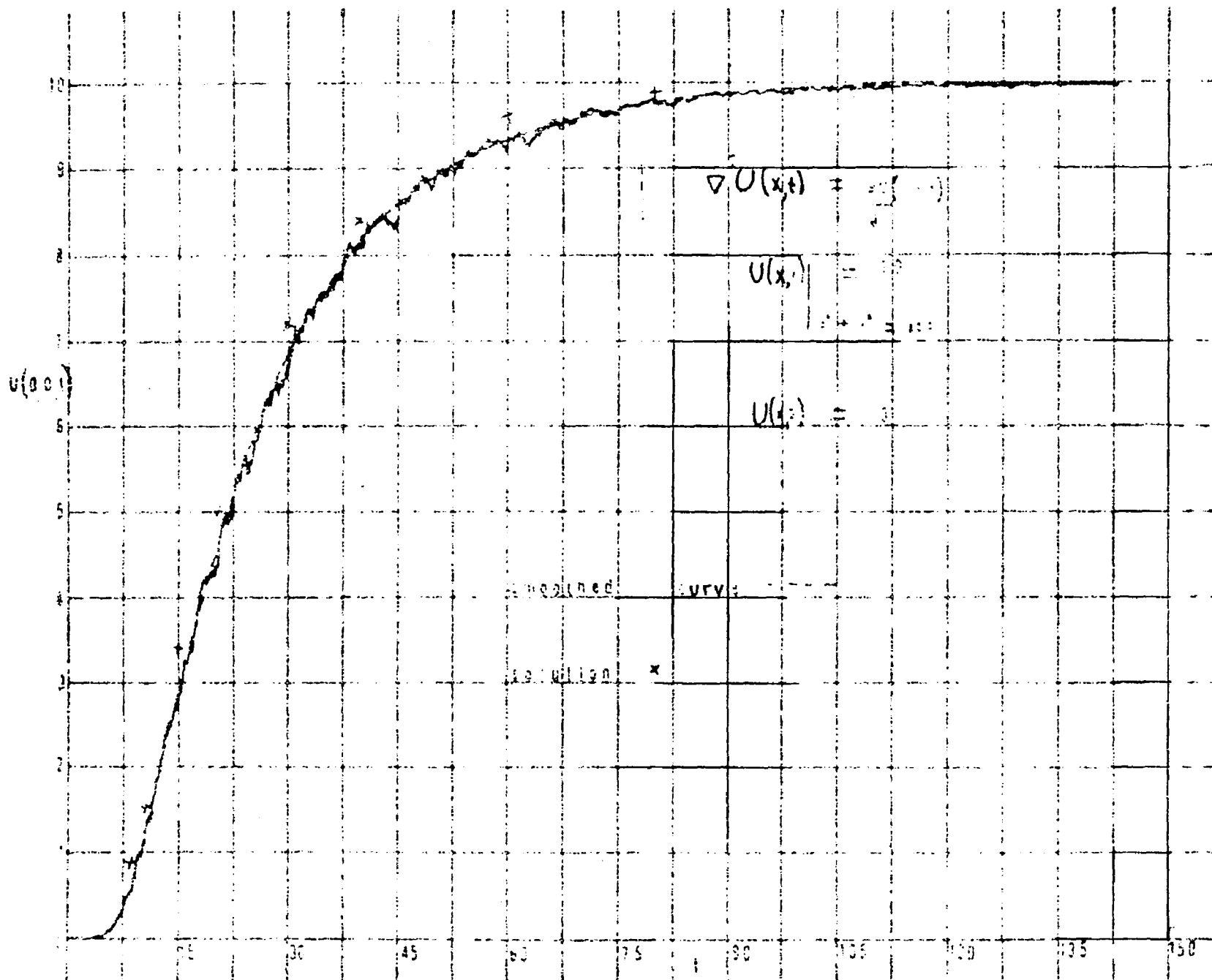


Figure 5.18 Monte Carlo Solution of Example 8b

The analog-hybrid computer diagram for the Monte Carlo solution of the problem given in Example 9 is found in Fig. 5.19 and the solution which was obtained appears in Fig. 5.20.

#### 5.4 Monte Carlo Generation of Level-lines of Partial Differential Equations

##### Example 10

Statement of problem:

$$\nabla^2 U(x,y) = 0$$

$$U(x,y) \Big|_{x^2 + y^2 = 100} = 10$$

$$U(x,y) \Big|_{x^2 + y^2 = 16} = -10$$

Region of definition:

$$16 < x^2 + y^2 < 100$$

Desired solutions:

Level curves for  $U(x,y) = 5$ ,  $U(x,y) = 0$ ,  $U(x,y) = -5$

Exact solution:

$$x^2 + y^2 = 40(10/4)^{(U_0/10)}$$

where  $U_0$  is the desired level

Langevin equations:

$$\frac{dx}{dt} = G N_1(t)$$

where

$$G = 10^5$$

$$\frac{dy}{dt} = G N_2(t)$$

$$E[N_1(t)N_1(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

$$E[N_1(t)N_2(t+\lambda)] = 0$$

Sweep speed:

0.02 volts/second

Averaging time constant:

3 seconds

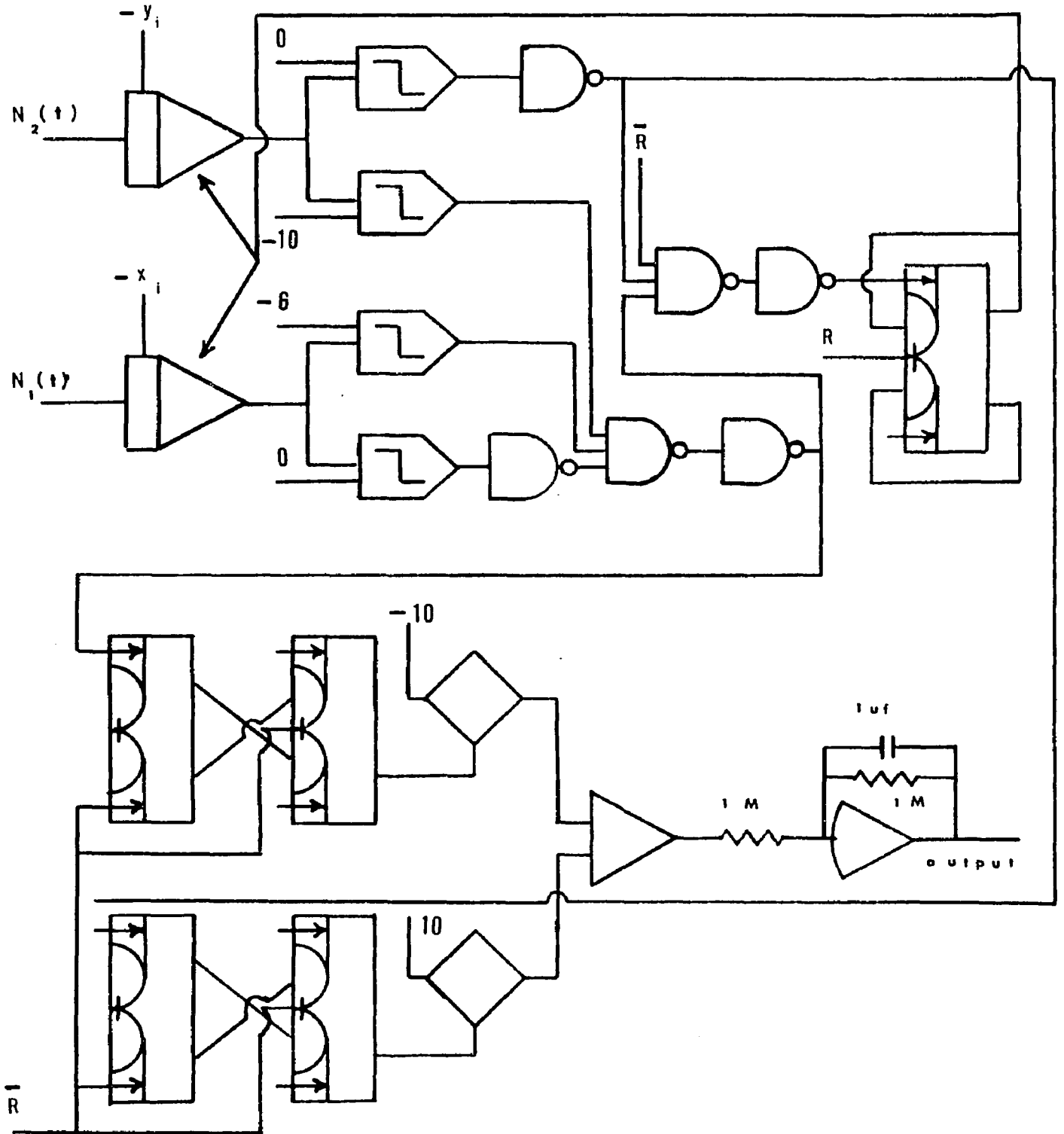


Figure 5.19 Hybrid Computer Diagram for the Monte Carlo Solution of Example 9

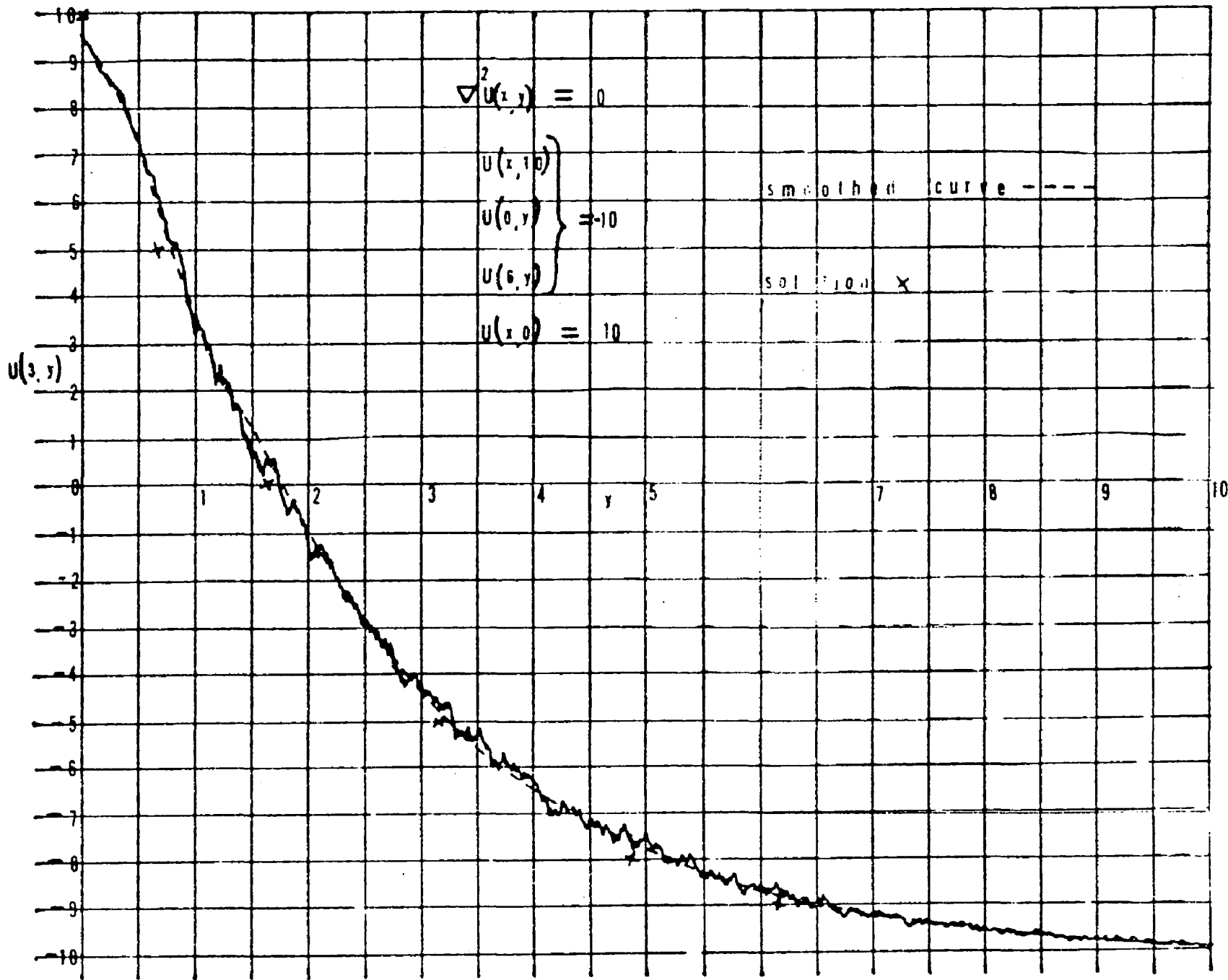


Figure 5.20 Monte Carlo Solution of Example 9

Maximum errors:

- (a) Systematic errors 5.5%
- (b) Statistical fluctuations 1.9%

The analog-hybrid computer diagram for the Monte Carlo solution of this problem is similar to the one given in Fig. 5.12. In the present case, however, the time constant of the averaging filter is 3 seconds in contrast with the 1 second time constant used in Example 6. The computer setup required to generate the level-lines (see Section 3.6) is given in Fig. 3.4. The Monte Carlo solutions are shown in Fig. 5.21.

Example 11Statement of problem:

$$\nabla^2 U(x,y) = 0$$

$$U(x,y) \Big|_{x^2 + y^2 = 100} = \begin{cases} +10 & \text{for } y > 0 \\ -10 & \text{for } y < 0 \end{cases}$$

Region of definition:

$$0 < x^2 + y^2 < 100$$

Desired solution:

Level curves for  $U(x,y) = \underline{+9}$ ,  $U(x,y) = \underline{+5}$ ,  $U(x,y) = 0$

Exact solution:

$$x^2 + \left[ y + \tan \frac{(10 - U_0)\pi}{20} \right]^2 = \left[ \frac{10}{\cos \frac{(10 - U_0)\pi}{20}} \right]^2$$

where  $U_0$  is the desired level.

Langevin equations:

$$\frac{dx}{dt} = G N_1(t)$$

where

$$G = 10^5$$

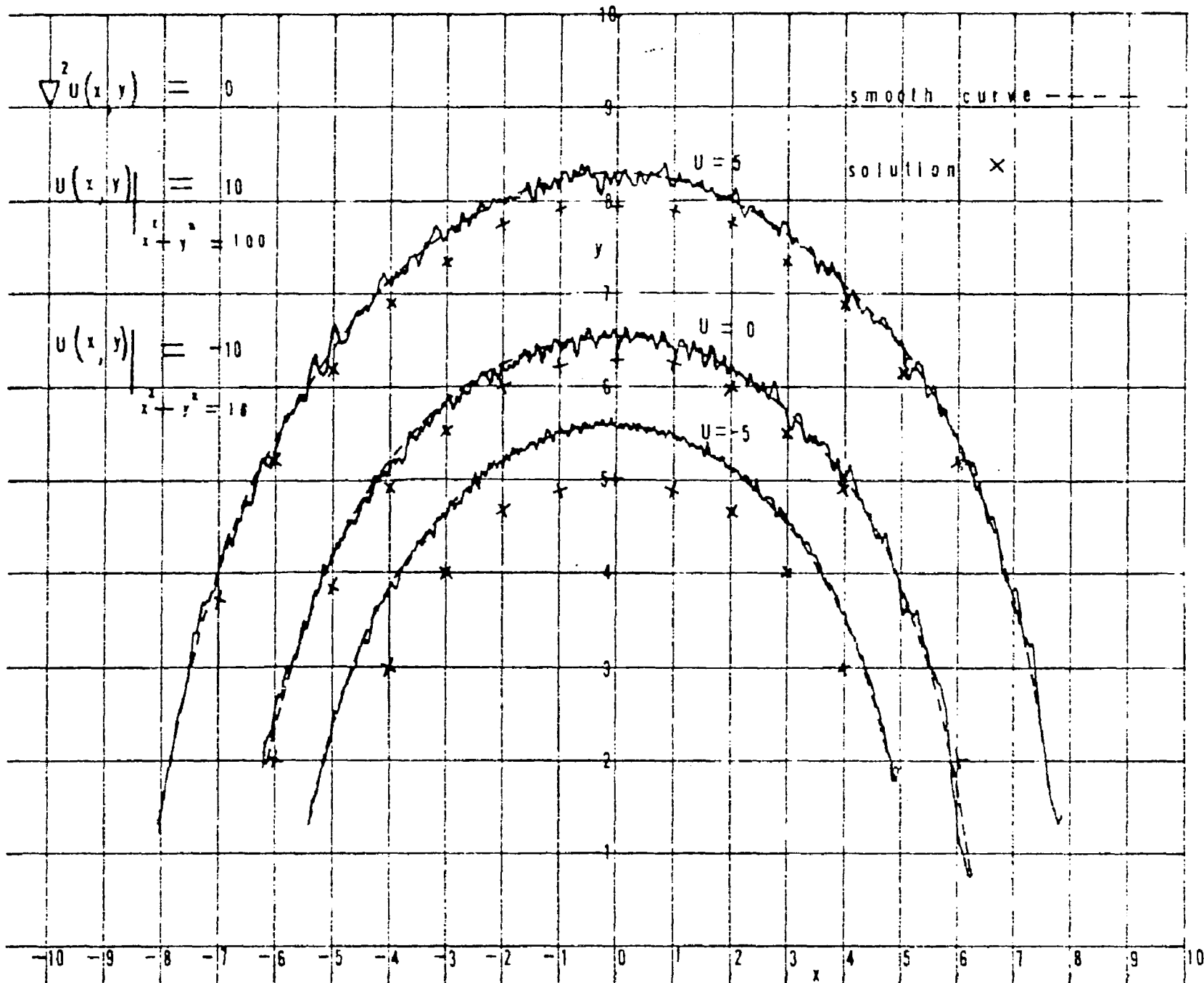


Figure 5.21 Monte Carlo Solution of Example 10

$$\frac{dy}{dt} = G N_2(t)$$

$$E[N_i(t)N_i(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

$$E[N_1(t)N_2(t+\lambda)] = 0$$

Sweep speed:

0.02 volts/second

Averaging time constant:

3 seconds

Maximum errors:

(a) Systematic errors 4.6%

(b) Statistical fluctuations 2.7%

The analog-hybrid computer diagram for the Monte Carlo solution of the problem given in Example 11 is similar to the one given in Fig. 5.14; however, as in Example 10, the time constant of the averaging filter is 3 seconds. The Monte Carlo solution of Example 11 appears in Fig. 5.22.

### Example 12

Statement of problem:

$$\nabla^2 U(x,y,t) = \frac{\partial U}{\partial t}$$

$$U(x,y,t) \Big|_{x^2 + y^2 = 100} = \begin{cases} +10 & \text{for } y > 0 \\ -10 & \text{for } y < 0 \end{cases}$$

$$U(x,y,0) = 0$$

Region of definition:

$$0 < x^2 + y^2 < 100, \quad 0 < t$$

Desired solution:

Level curves for  $U(x,y,31) = \underline{+2}$ ,  $U(x,y,77.5) = \underline{+2}$ ,

$U(x,y,147) = \underline{+2}$

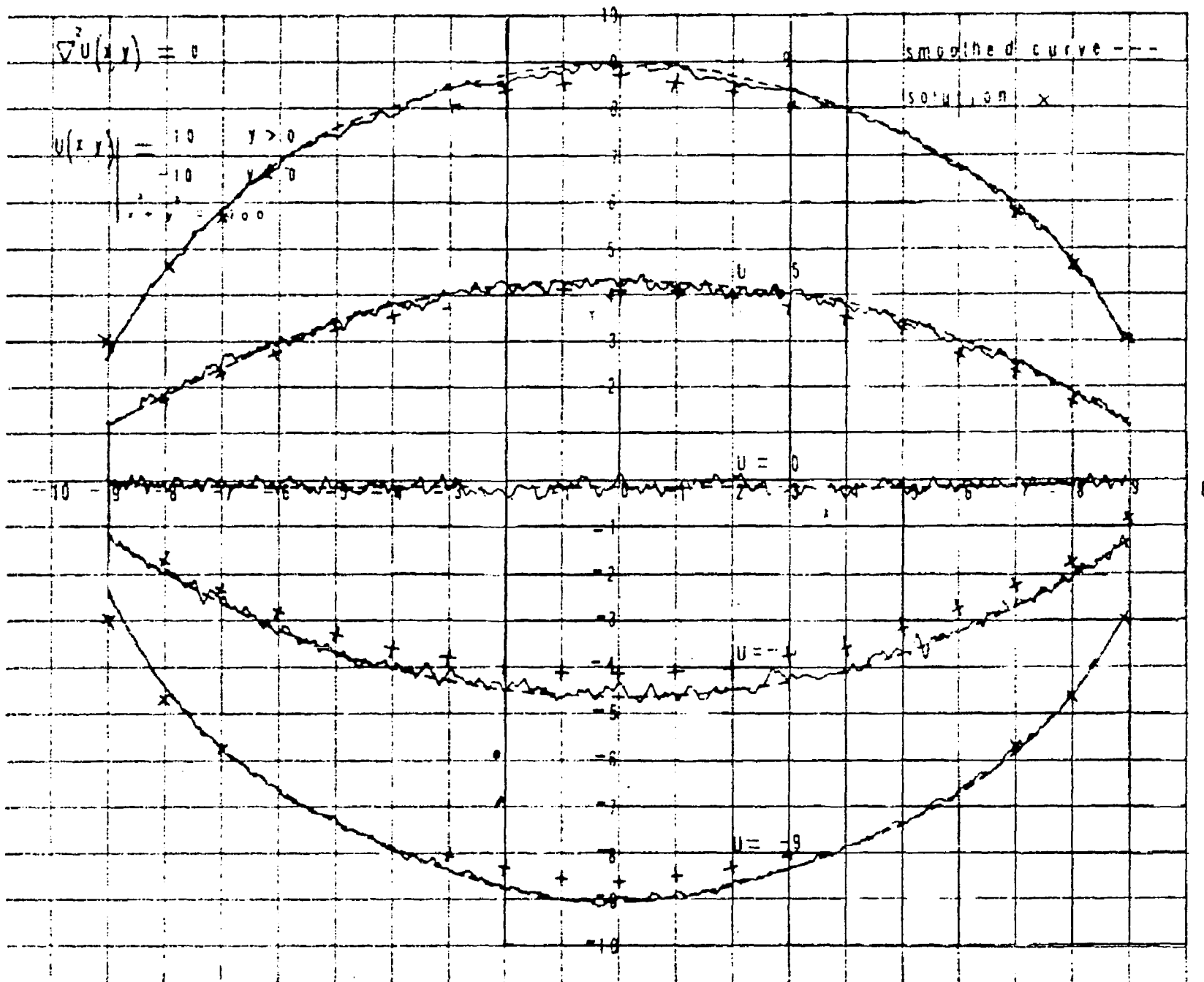


Figure 5.22 Monte Carlo Solution of Example 11

Langevin equations:

$$\frac{dx}{dt} = G N_1(t)$$

$$\frac{dy}{dt} = G N_2(t)$$

where

$$G = 10^5$$

$$E[N_i(t)N_i(t+\lambda)] = 30 \cdot 10^{-6} \delta(\lambda)$$

$$E[N_1(t)N_2(t+\lambda)] = 0$$

Time scale factor,  $T_s$ :

$$T_s = t/\tau = 15 \cdot 10^4$$

where  $t$  is real time and  $\tau$  is computer time.

Sweep speed:

0.02 volts/second

Averaging time constant:

3 seconds

Maximum errors:

- (a) Systematic errors 13%
- (b) Statistical fluctuations 2.4%

Both analog-hybrid computer diagrams for the Monte Carlo solution and the level-line generator of the problem given in Example 12 are identical to the ones given for the problem in Example 11. The Monte Carlo solutions which were obtained for Example 12 are given in Fig. 5.23.

Example 13Statement of problem:

$$\nabla^2 U(x,y) = 0$$

$$U(x,0) = +10, U(0,y) = U(6,y) = U(x,10) = -10$$

Region of definition:

$$0 < x < 6, 0 < y < 10$$

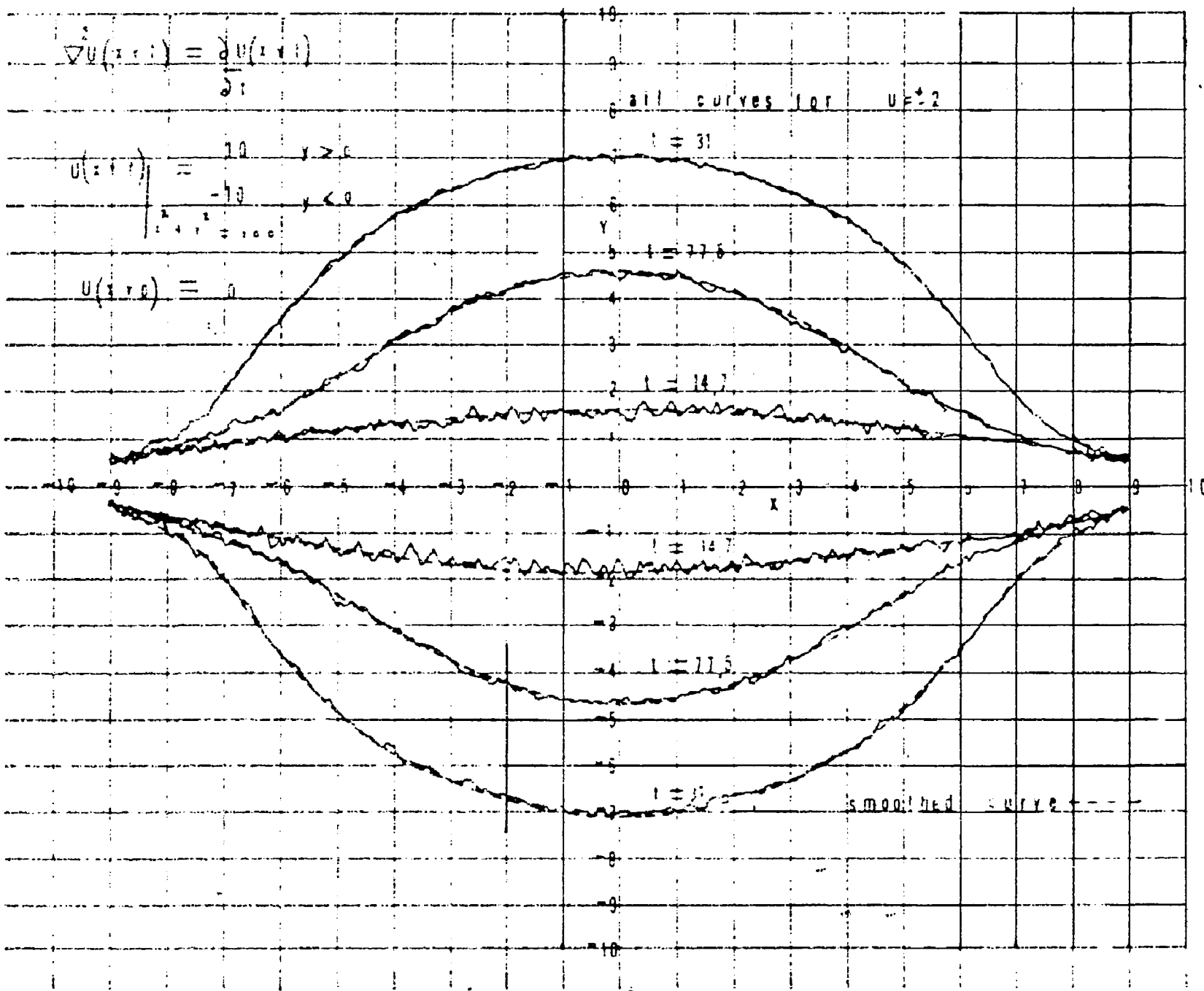


Figure 5.23 Monte Carlo Solution of Example 12

Desired solution:

Level curves for  $U(x,y) = -9$ ,  $U(x,y) = -8$ ,  $U(x,y) = +5$ ,  
 $U(x,y) = 0$

Langevin equations:

$$\frac{dx}{dt} = G N_1(t)$$

where

$$G = 10^5$$

$$\frac{dy}{dt} = G N_2(t)$$

$$E [ N_1(t)N_1(t+\lambda) ] = 30 \cdot 10^{-6} \mathcal{E}(\lambda)$$

$$E [ N_1(t)N_2(t+\lambda) ] = 0$$

Sweep speed:

0.02 volts/second

Averaging time:

3 seconds

The analog-hybrid computer diagram for the Monte Carlo solution of this problem is similar to the one given in Fig. 5.19; however, as in the previous examples, the time constant of the averaging filter is 3 seconds. The Monte Carlo solution which was obtained for the problem given in Example 13 appears in Fig. 5.24.

### 5.5 Discussion of Results

The Examples presented in the previous sections demonstrate that the Monte Carlo estimates of the solutions of the partial differential equations are in agreement with the theoretical solutions. The solutions of the elliptic partial differential equations exhibit typical systematic errors of approximately 3.7% of the boundary values and statistical errors of approximately 2.3% of the boundary

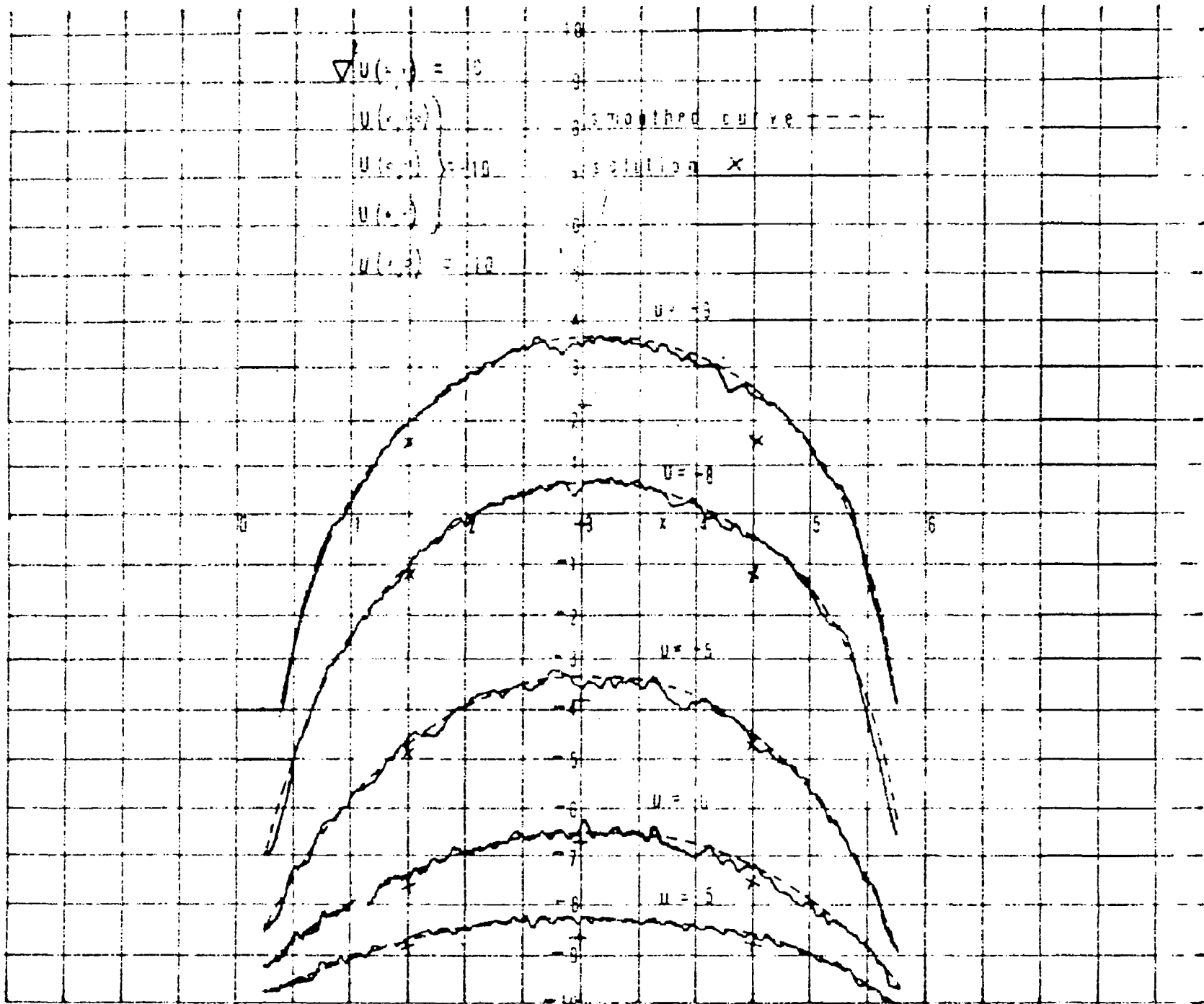


Figure 5.24 Monte Carlo Solution of Example 13

values. Parabolic partial differential equations exhibit typical systematic errors of approximately 3.8% of the boundary values and typical statistical errors of approximately 1.8% of the boundary values, the larger errors occurring for solutions where the time  $t$ , is small (see Sec. 3.3).

This study was performed upon ASTRAC II just after the machine was built and before some technical difficulties could be eliminated. In addition to the errors discussed in Chapter 3, large integrator overshoots, while not particularly critical for most computations, indicate false boundary crossings in Monte Carlo routines. Excessive computer noise in addition alters comparison levels of analog comparators producing solution errors. It is felt that by careful control of spurious pulses and by decreasing noise levels present, the systematic errors could be significantly reduced.

## CHAPTER 6

### CONCLUDING REMARKS

In this study, Monte Carlo estimates of solutions of partial differential equations were obtained on ASTRAC II, a high speed iterative differential analyzer. Random walks were generated at the rate of 1000 per second in contrast with digital computer Monte Carlo solutions where 10 random walks per second are executed<sup>26</sup> and with previous analog-hybrid Monte Carlo procedures where 4 random walks per second were achieved.<sup>11</sup>

In previous Monte Carlo computations, solutions of partial differential equations at specific points were obtained by digitally averaging the values of the function at the boundary absorption points. In the present study, voltages equal to the function at the boundary absorption points are developed and averaged in a low pass filter. The rapid random walk rate and the use of analog, rather than digital averaging allows the solution to be rapidly scanned over a line in the region of interest.

Where digital averaging is used, boundary values of the function need only be quantized to within, say 8 bits, for accurate estimates of the solution of the partial differential equation, since quantizing theory indicates

that the biasing effects of quantization average out approximately, or can be approximately predicted and corrected even with surprising coarse quantization.<sup>27, 28</sup> The coarse quantization of the boundary values yields best results when the probability density function of the absorbed boundary values is unimodal. For unfavorable probability density functions, dither can be added to yield satisfactory results.

Coarse quantization immediately realizes savings in equipment, since it permits primitive table-lookup rather than function generation; it is not applicable to finite-difference schemes for the solution of partial differential equations since it leads to unacceptable errors.

In addition to the class of problems discussed by Little,<sup>11</sup> partial differential equations with specified conditions on the normal derivative of the boundary function were examined, computing algorithms given, and some examples worked. It is hoped that future work in this area will enlarge the class of boundary conditions on partial differential equations which can be treated by Monte Carlo techniques.

The level-lines of the solutions of partial differential equations were generated by utilizing high-speed Monte Carlo estimates of the solutions of the partial differential equations. The construction of such

level-lines is perhaps the most illuminating form into which a solution of a partial differential equation may be put. The rapidity of the random walks, and the characteristic behavior of Monte Carlo methods which enable one to obtain a solution at a point independently from all other points, permitted the generation of these level-lines.

Although Monte Carlo estimates of solutions of partial differential equations are applicable to a restricted class of problems, and convergence of the solutions are probabilistic, the ease of programming, and modification of equation parameters, boundaries, initial conditions, and boundary conditions make the Monte Carlo method appear quite attractive for a selected class of problems.

## APPENDIX A

### RELATIONSHIPS FOR MARKOV PROCESSES

The conditional probability density function of a continuous Markov process satisfies the Chapman-Kolmogorov integral equation,<sup>3</sup>

$$f(\underline{r}_k, t_k / \underline{r}_i, t_i) = \int_R f(\underline{r}_k, t_k / \underline{r}_j, t_j) f(\underline{r}_j, t_j / \underline{r}_i, t_i) d\underline{r}_j \quad (\text{A.1})$$

If  $t_j = t_i + \Delta t$  then,

$$f(\underline{r}_k, t_k / \underline{r}_i, t_i) = \int_R f(\underline{r}_k, t_k / \underline{r}_j, t_i + \Delta t) f(\underline{r}_j, t_i + \Delta t / \underline{r}_i, t_i) d\underline{r}_j \quad (\text{A.2})$$

To develop the Kolmogorov backward partial differential equation, the first term in the integrand of (A.2) is developed into a Taylor series about the point  $\underline{r}_i$ , i.e.,

$$\begin{aligned}
f(\underline{r}_k, t_k / \underline{r}_j, t_i + \Delta t) &= f(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t) \\
&+ (x_j - x_i) \frac{\partial f}{\partial x_i}(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t) \\
&+ (y_j - y_i) \frac{\partial f}{\partial y_i}(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t) \\
&+ \frac{1}{2!} (x_j - x_i)^2 \frac{\partial^2 f}{\partial x_i^2}(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t) \\
&+ \frac{1}{2!} (y_j - y_i)^2 \frac{\partial^2 f}{\partial y_i^2}(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t) \\
&+ (x_j - x_i)(y_j - y_i) \frac{\partial^2 f}{\partial x_i \partial y_i}(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t) \\
&+ \text{higher order terms.} \tag{A.3}
\end{aligned}$$

Although the Taylor series has been developed for a two-dimensional space, the extension to  $n$  dimensions is straightforward.

Inserting (A.3) into (A.2) and noting that

$$f(\underline{r}_j, t_i + \Delta t / \underline{r}_i, t_i) \rightarrow \delta(\underline{r}_j - \underline{r}_i) \text{ as } \Delta t \rightarrow 0 \tag{A.4}$$

one finds,

$$\begin{aligned}
& \frac{f(\underline{r}_k, t_k / \underline{r}_i, t_i) - f(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t)}{\Delta t} \\
&= \frac{\partial f}{\partial \underline{x}_i}(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t) \frac{1}{\Delta t} \int_R (x_j - x_i) f(\underline{r}_j, t_i + \Delta t / \underline{r}_i, t_i) d\underline{r}_j \\
&+ \frac{\partial f}{\partial \underline{y}_i}(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t) \frac{1}{\Delta t} \int_R (y_j - y_i) f(\underline{r}_j, t_i + \Delta t / \underline{r}_i, t_i) d\underline{r}_j \\
&+ \frac{\partial^2 f}{\partial \underline{x}_i^2}(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t) \frac{1}{2! \Delta t} \int (x_j - x_i)^2 f(\underline{r}_j, t_i + \Delta t / \underline{r}_i, t_i) d\underline{r}_j \\
&+ \frac{\partial^2 f}{\partial \underline{y}_i^2}(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t) \frac{1}{2! \Delta t} \int (y_j - y_i)^2 f(\underline{r}_j, t_i + \Delta t / \underline{r}_i, t_i) d\underline{r}_j \\
&+ \frac{\partial^2 f}{\partial \underline{x}_i \partial \underline{y}_i}(\underline{r}_k, t_k / \underline{r}_i, t_i + \Delta t) \frac{1}{\Delta t} \int (x_j - x_i)(y_j - y_i) \\
&f(\underline{r}_j, t_i + \Delta t / \underline{r}_i, t_i) d\underline{r}_j + \text{higher order terms.} \quad (\text{A.5})
\end{aligned}$$

Since the integrals represent conditional expectation, it is perhaps more suggestive to rewrite (A.5) in the form

$$\begin{aligned}
\frac{\partial f}{\partial t_i}(\underline{r}_k, t_k / \underline{r}_i, t_i) &= \lim_{\Delta t \rightarrow 0} \left[ \frac{\partial f}{\partial \underline{x}_i} \frac{1}{\Delta t} E \left\{ (x_j - x_i) / \underline{r}_i, t_i \right\} \right. \\
&+ \frac{\partial f}{\partial \underline{y}_i} \frac{1}{\Delta t} E \left\{ (y_j - y_i) / \underline{r}_i, t_i \right\} + \frac{1}{2!} \frac{\partial^2 f}{\partial \underline{x}_i^2} \frac{1}{\Delta t} \\
&E \left\{ (x_j - x_i)^2 / \underline{r}_i, t_i \right\} + \frac{1}{2!} \frac{\partial^2 f}{\partial \underline{y}_i^2} \frac{1}{\Delta t} E \left\{ (y_j - y_i)^2 / \underline{r}_i, t_i \right\} \\
&+ \frac{\partial^2 f}{\partial \underline{x}_i \partial \underline{y}_i} \frac{1}{\Delta t} E \left\{ (x_j - x_i)(y_j - y_i) / \underline{r}_i, t_i \right\} \\
&\left. + \text{higher order terms.} \right] \quad (\text{A.6})
\end{aligned}$$

The following limits are assumed to exist:

$$\lim_{\Delta t \rightarrow 0} E \left\{ \frac{(x_j - x_i)/r_i, t_i}{\Delta t} \right\} = a_1(r_i, t_i) \quad (\text{A.7})$$

$$\lim_{\Delta t \rightarrow 0} E \left\{ \frac{(y_j - y_i)/r_i, t_i}{\Delta t} \right\} = a_2(r_i, t_i) \quad (\text{A.8})$$

$$\lim_{\Delta t \rightarrow 0} E \left\{ \frac{(x_j - x_i)^2 / r_i, t_i}{2! \Delta t} \right\} = b_1(r_i, t_i) \quad (\text{A.9})$$

$$\lim_{\Delta t \rightarrow 0} E \left\{ \frac{(y_j - y_i)^2 / r_i, t_i}{2! \Delta t} \right\} = b_2(r_i, t_i) \quad (\text{A.10})$$

$$\lim_{\Delta t \rightarrow 0} E \left\{ \frac{(x_j - x_i)(y_j - y_i) / r_i, t_i}{\Delta t} \right\} = 0 \quad (\text{A.11})$$

$$\lim_{\Delta t \rightarrow 0} \frac{\text{all higher order moments}}{\Delta t} = 0 \quad (\text{A.12})$$

Rewriting (A.6) using the above limits, one finds

$$\begin{aligned} -\frac{\partial f}{\partial t_i} &= a_1(r_i, t_i) \frac{\partial f}{\partial x_i} + a_2(r_i, t_i) \frac{\partial f}{\partial y_i} + b_1(r_i, t_i) \frac{\partial^2 f}{\partial x_i^2} \\ &+ b_2(r_i, t_i) \frac{\partial^2 f}{\partial y_i^2} \end{aligned} \quad (\text{A.13})$$

Equation (A.13) is the partial differential equation satisfied by the Markov process. To determine the

relationship between the terms of (A.13) and the coefficients of the generalized Langevin equations, consider the matrix stochastic differential equation,

$$\frac{dr}{dt} + A(\underline{r}, t) = B(\underline{r}, t)N(t) \quad (\text{A.14})$$

In the sequel, it is assumed that the signals  $N_i(t)$  are mutually uncorrelated white Gaussian noise signals with zero mean values. The set of equations (A.14) define a Markov process. Integration of (A.14) yields

$$\underline{r}(t_j) = \underline{r}(t_i) + \int_{t_i}^{t_j} [B(\underline{r}, t)N(t) - A(\underline{r}, t)] dt \quad (\text{A.15})$$

The value of  $\underline{r}$  at the time  $t_j$  depends on the value of  $\underline{r}$  at time  $t_i$  and upon the behavior of  $A(\underline{r}, t)$ ,  $B(\underline{r}, t)$  and  $N(t)$  during the interval  $t_i < t < t_j$ . Knowledge of  $\underline{r}$  for times prior to time  $t_i$  does not yield any additional information about the behavior of  $\underline{r}$  after time  $t_i$ ; this is, indeed, the characteristic property of a Markov process.

To find the parameters of the Markov process which appear in (A.13), consider the first row of the matrix equation (A.14)

$$\frac{dx}{dt} + A_1(\underline{r}, t) = B_1(\underline{r}, t)N_1(t) \quad (\text{A.16})$$

Integrating once again over the time interval  $t_i$  to  $t_j$ , and allowing

$$t_j = t_i + \Delta t \quad (\text{A.17})$$

one finds,

$$x_j - x_i = - \int_{t_i}^{t_i + \Delta t} A_1(\underline{r}, t) dt + \int_{t_i}^{t_i + \Delta t} B_1(\underline{r}, t) N(t) dt \quad (\text{A.18})$$

From (A.7),

$$a_1(\underline{r}_i, t_i) = \lim_{\Delta t \rightarrow 0} E \left\{ \frac{(x_j - x_i) / \underline{r}_i, t_i}{\Delta t} \right\} \quad (\text{A.7})$$

Calculation of  $a_1(\underline{r}_i, t_i)$  from (A.18) yields

$$\begin{aligned} a_1(\underline{r}_i, t_i) &= \lim_{\Delta t \rightarrow 0} \Delta t^{-1} \int_{t_i}^{t_i + \Delta t} A_1(\underline{r}, t) dt \\ &= - A_1(\underline{r}_i, t_i) \end{aligned} \quad (\text{A.19})$$

since

$$E \{ N_1(t) \} = 0 \quad (\text{A.20})$$

By a similar calculation,

$$\begin{aligned} a_2(\underline{r}_i, t_i) &= - A_2(\underline{r}_i, t_i) \\ a_3(\underline{r}_i, t_i) &= - A_3(\underline{r}_i, t_i) \\ &\cdot \\ &\cdot \\ &\cdot \end{aligned} \quad (\text{A.21})$$

Using (A.9), one may calculate  $b_1(\underline{r}_i, t_i)$ ,

$$b_1(\underline{r}_i, t_i) = \lim_{\Delta t \rightarrow 0} E \left\{ \frac{(x_j - x_i)^2 / \underline{r}_i, t_i}{2! \Delta t} \right\} \quad (\text{A.9})$$

$$\begin{aligned} &= \lim_{\Delta t \rightarrow 0} \frac{1}{2! \Delta t} \left\{ A_1^2(\underline{r}_i, t_i) (\Delta t)^2 + 2D_1 B_1^2(\underline{r}_i, t_i) \Delta t \right\} \\ &= D_1 B_1^2(\underline{r}_i, t_i) \end{aligned} \quad (\text{A.22})$$

The last result follows since,

$$E \left\{ N_1(t) N_1(t+t_1) \right\} = 2D_1 \delta(t_1) \quad (\text{A.23})$$

Once again, by a similar calculation,

$$\begin{aligned} b_2(\underline{r}_i, t_i) &= D_2 B_2^2(\underline{r}_i, t_i) \\ b_3(\underline{r}_i, t_i) &= D_3 B_3^2(\underline{r}_i, t_i) \\ &\vdots \\ &\vdots \\ &\vdots \end{aligned} \quad (\text{A.24})$$

The calculation of a third order moment is given in (A.25)

$$\begin{aligned} \frac{1}{3! \Delta t} E \left( x_j - x_i \right)^3 / \underline{r}_i, t_i &= \lim_{\Delta t \rightarrow 0} \left[ \frac{A_1^3(\underline{r}_i, t_i) (\Delta t)^3}{3! \Delta t} \right. \\ &\quad - \frac{3A_1(\underline{r}_i, t_i) 2D_1 B_1^2(\underline{r}_i, t_i) (\Delta t)}{3! \Delta t} + \frac{1}{3! \Delta t} \int_{t_i}^{t_i + \Delta t} \int_{t_i}^{t_i + \Delta t} \int_{t_i}^{t_i + \Delta t} \\ &\quad \left. B_1(\underline{r}_i, \lambda) B_1(\underline{r}_i, \eta) B_1(\underline{r}_i, \xi) E \left[ N_1(\lambda) N_1(\eta) N_1(\xi) \right] d\lambda d\eta d\xi \right] \end{aligned} \quad (\text{A.25})$$

By assumption,  $N_1(t)$  is white Gaussian noise with zero mean value and exhibits the following properties:<sup>25</sup>

$$E[N_i(t)] = 0 \quad (\text{A.26})$$

$$E[N_i(t_1)N_i(t_2)] = 2D_i \delta(t_2 - t_1) \quad (\text{A.27})$$

$$E[N_i(t_1)N_i(t_2)\dots N_i(t_{2m+1})] = 0 \quad (\text{A.28})$$

$$E[N_i(t_1)N_i(t_2)\dots N_i(t_{2m})] \\ = \sum_{\text{all pairs}} E\{N_i(t_k)N_i(t_j)\} E\{N_i(t_p)N_i(t_n)\} \quad (\text{A.29})$$

The last summation contains  $\frac{(2m)!}{2^m m!}$  terms,<sup>25</sup> which is the number of ways the  $2m$  points in time can be divided into  $m$  pairs.

From (A.28), the left hand member of (A.20) is zero. To calculate the general  $n^{\text{th}}$  order parameter of the process, consider

$$\lim_{\Delta t \rightarrow 0} \frac{1}{n! \Delta t^n} E \left\{ (x_j - x_i)^n / r_i, t_i \right\} = \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t^n} \\ \sum_{k=0}^n \frac{1}{k!(n-k)!} \left[ -A_i(r_i, t_i) \Delta t \right]^{n-k} \int_{t_i}^{t_i + \Delta t} \dots \int_{t_i}^{t_i + \Delta t} \\ B_1(r_i, t_1) \dots B_1(r_i, t_k) E \left\{ N_1(t_1) \dots N_2(t_k) \right\} dt_1 \dots dt_k \quad (\text{A.30})$$

Now

$$\begin{aligned}
 E\left[N_1(t_1)N_1(t_2)\dots N_1(t_k)\right] &= 0 \text{ for } k \text{ odd} \\
 &= \frac{k!(2D_1)^{k/2}(\Delta t)^{k/2}}{2^{k/2}(k/2)!} \text{ for } k \text{ even}
 \end{aligned}
 \tag{A.31}$$

Therefore, for  $n \geq 3$ , each term in the summation (A.31) is either identically zero or goes to zero as  $\Delta t \rightarrow 0$ .

Summarizing the result:

$$\begin{aligned}
 \frac{1}{n!\Delta t} E \left\{ (x_j - x_i)^n / r_i, t_i \right\} &= -A_1(r_i, t_i) \text{ for } n=1 \\
 &= D_1 B_1^2(r_i, t_i) \text{ for } n=2 \\
 &= 0 \text{ for } n \geq 3
 \end{aligned}
 \tag{A.32}$$

## APPENDIX B

### THE PARTIAL DIFFERENTIAL EQUATION SATISFIED BY THE AVERAGE TIME OF A RANDOM WALK

It is of interest for both the solution of non-homogeneous partial differential equations (Sec. 2.7) and in error analysis (Sec. 3.4) to find the partial differential equation satisfied by the average time taken for a random walk. From definition 2 of Sec. 2.2, the quantity

$$\int_C g(\underline{r}_b, t_b / \underline{r}_i, t_i) d\underline{r}_b \quad (\text{B.1})$$

is the probability density for a particle executing a random walk to reach any portion of the boundary C at the time  $t_b$ , given that the walk originated at  $\underline{r}_i$  at time  $t_i$ . From the Markov property of the random walk,

$$\int_C g(\underline{r}_b, t_b / \underline{r}_i, t_i) d\underline{r}_b = \int_C \int_R g(\underline{r}_b, t_b / \underline{r}_j, t_j) f(\underline{r}_j, t_j / \underline{r}_i, t_i) d\underline{r}_j d\underline{r}_b \quad (\text{B.2})$$

The particle is within R at time  $t_j$  since  $t_i < t_j < t_b$ . The average time for a particle to reach the boundary C assuming that the random walk originates from  $\underline{r}_i$  at time  $t_i$  is

$$T(\underline{r}_i, t_i) = E(t_b - t_i) = \int_{\text{all } t_b} \int_C \int_R (t_b - t_i) g(\underline{r}_b, t_b / \underline{r}_j, t_j) f(\underline{r}_j, t_j / \underline{r}_i, t_i) d\underline{r}_j d\underline{r}_b dt_b \quad (\text{B.3})$$

Since

$$t_b - t_i = t_b - t_j + t_j - t_i \quad (\text{B.4})$$

$$T(\underline{r}_i, t_i) = \int_{\text{all } t_b} \int_C \int_R (t_b - t_j) g(\underline{r}_b, t_b / \underline{r}_j, t_j) f(\underline{r}_j, t_j / \underline{r}_i, t_i) d\underline{r}_j d\underline{r}_b dt_b + \int_{\text{all } t_b} \int_C \int_R (t_j - t_i) g(\underline{r}_b, t_b / \underline{r}_j, t_j) f(\underline{r}_j, t_j / \underline{r}_i, t_i) d\underline{r}_j d\underline{r}_b dt_b \quad (\text{B.5})$$

Hence,

$$T(\underline{r}_i, t_i) = \int_R T(\underline{r}_j, t_j) f(\underline{r}_j, t_j / \underline{r}_i, t_i) d\underline{r}_j + (t_j - t_i) \int_R f(\underline{r}_j, t_j / \underline{r}_i, t_i) d\underline{r}_j \quad (\text{B.6})$$

This last result follows from (B.7) which expresses the result that all walks eventually terminate at a boundary point.

$$\int_{\text{all } t_b} \int_C g(\underline{r}_b, t_b / \underline{r}_j, t_j) d\underline{r}_b dt_b = 1. \quad (\text{B.7})$$

Equation (B.6) may be further simplified by noting that

$$\int_R f(\underline{r}_j, t_j / \underline{r}_i, t_i) d\underline{r}_j = 1 \quad (\text{B.8})$$

because  $t_j$  is earlier in time than  $t_i$  and therefore the particle is within region  $R$  with probability one.

Rewriting equation (B.6),

$$T(\underline{r}_i, t_i) = \int_R T(\underline{r}_j, t_j) f(\underline{r}_j, t_j / \underline{r}_i, t_i) d\underline{r}_j + (t_j - t_i) \quad (\text{B.9})$$

If the conditional probability density function satisfies the partial differential equation,

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right] f(\underline{r}_j, t_j / \underline{r}_i, t_i) = 0 \quad (\text{B.10})$$

$T(\underline{r}_i, t_i)$  satisfies (B.11) which can be verified by operating upon (B.10) with  $\left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right]$ .

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right] T(\underline{r}_i, t_i) = -1 \quad (\text{B.11})$$

## APPENDIX C

### PROOF OF THE MONTE CARLO TECHNIQUES FOR THE SOLUTION OF PARTIAL DIFFERENTIAL EQUATIONS

In this Appendix, the procedures which were discussed in Chapter 2 for the Monte Carlo solution of partial differential equations will be proved. Consider problem A defined by equation (2.31) which is repeated below.

#### Problem A

$U(\underline{r}_i, t_i)$  is to satisfy the partial differential equation

$$\frac{-\partial U}{\partial t_i} = L_{\underline{r}_i, t_i} U(\underline{r}_i, t_i) \quad (C.1)$$

within a bounded region R, with boundary C together with the initial condition

$$U(\underline{r}_i, 0) = U_i(\underline{r}_i) \quad (C.2)$$

and the boundary condition,

$$U(\underline{r}_b, t) = U_b(\underline{r}_b) \quad (C.3)$$

where  $U_b(\underline{r}_b)$  is piecewise continuous and specified at all points of the boundary C. The operator  $L_{\underline{r}_i, t_i}$  is defined by

$$\begin{aligned}
L_{\vec{r}_i, t_i} U(\vec{r}_i, t_i) &= \sum_{k=1}^n a_k(\vec{r}_i, t_i) \frac{\partial U}{\partial (\vec{r}_k)_i} \\
&+ b_k(\vec{r}_i, t_i) \frac{\partial^2 U}{\partial (\vec{r}_k)_i^2}
\end{aligned} \tag{C.4}$$

where  $(\vec{r}_k)_i$  denotes the coordinates of the initial position vector.

In accordance with the procedure given in section 2.4 for the Monte Carlo solution, the expected value of either the boundary value function or the initial condition function, depending upon whether the walk terminated due to a boundary absorption, or at time zero, is the solution to the problem. The expectation in question is:

$$\begin{aligned}
U(\vec{r}_i, t_i) &= \int_R U_i(\vec{r}_j) f(\vec{r}_j, 0 / \vec{r}_i, t_i) d\vec{r}_j \\
&+ \int_{t_i}^0 \int_C U_b(\vec{r}_b) g(\vec{r}_b, t_b / \vec{r}_i, t_i) d\vec{r}_b dt_b
\end{aligned} \tag{C.5}$$

If the partial differential equation is elliptic rather than parabolic, the second integral is the solution, since all walks are continued until an absorption occurs.

To demonstrate that (C.5) is indeed a solution to the problem given in (C.1), it is necessary to show that the partial differential equation, the boundary conditions,

and the initial conditions are all satisfied by (C.5).

Operating upon (C.5) with  $\left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right]$  yields

$$\begin{aligned} \left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right] U(\underline{r}_i, t_i) &= \int_R U_i(\underline{r}_j) \left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right] \\ & f(\underline{r}_j, 0 / \underline{r}_i, t_i) d\underline{r}_j + \int_{t_i}^0 \int_C U_b(\underline{r}_b) \left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right] \\ & g(\underline{r}_b, t_b / \underline{r}_i, t_i) d\underline{r}_b dt_b - \int_C^{t_i} U_b(\underline{r}_b) g(\underline{r}_b, t_i / \underline{r}_i, t_i) d\underline{r}_b \end{aligned} \quad (C.6)$$

Both  $f(\underline{r}_j, 0 / \underline{r}_i, t_i)$  and  $g(\underline{r}_b, t_b / \underline{r}_i, t_i)$  satisfy the same partial differential equation, namely,

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right] f(\underline{r}_j, 0 / \underline{r}_i, t_i) = 0 \quad (C.7)$$

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right] g(\underline{r}_b, t_b / \underline{r}_i, t_i) = 0 \quad (C.8)$$

In addition,

$$\lim_{t_b \rightarrow t_i} g(\underline{r}_b, t_b / \underline{r}_i, t_i) = 0 \quad (C.9)$$

Hence,

$$\left[ \frac{\partial}{\partial t_i} + L_{\underline{r}_i, t_i} \right] U(\underline{r}_i, t_i) = 0 \quad (C.10)$$

To show that  $U(\underline{r}_i, t_i)$  as given by (C.5) also satisfies the initial conditions as well as the boundary conditions, consider

$$\lim_{t_i \rightarrow 0} U(\underline{r}_i, t_i) = \lim_{t_i \rightarrow 0} 0$$

$$\left[ \int_R U_i(\underline{r}_j) f(\underline{r}_j, 0 / \underline{r}_i, t_i) d\underline{r}_j + \int_{t_i}^0 \int_C U_b(\underline{r}_b) g(\underline{r}_b, t_b / \underline{r}_i, t_i) d\underline{r}_b dt_b \right] \quad (C.11)$$

$$= \int_R U_i(\underline{r}_j) \delta(\underline{r}_j - \underline{r}_i) d\underline{r}_j = U_i(\underline{r}_i) \quad (C.12)$$

and

$$\lim_{\underline{r}_i \rightarrow \underline{r}_b} U(\underline{r}_i, t_i) = \lim_{\underline{r}_i \rightarrow \underline{r}_b} U_b(\underline{r}_i)$$

$$\left[ \int_R U_i(\underline{r}_j) f(\underline{r}_j, 0 / \underline{r}_i, t_i) d\underline{r}_j + \int_{t_i}^0 \int_C U_b(\underline{r}_b) g(\underline{r}_b, t_b / \underline{r}_i, t_i) d\underline{r}_b dt_b \right] \quad (C.13)$$

$$= \int_C U_b(\underline{r}_b) \delta(\underline{r}_i - \underline{r}_b) d\underline{r}_b = U_b(\underline{r}_i) \quad (C.14)$$

Hence (C.5) is the solution of Problem A.

In the Monte Carlo procedure, random walks are generated in order to approximate (C.5) by a finite-sample average. The proof of the other Monte Carlo procedures given in Chapter 2 is similar to the one just presented.<sup>11</sup>

## APPENDIX D

### BINARY NOISE SOURCE

The requirements upon noise sources for use in Monte Carlo computations are quite stringent;<sup>10, 11</sup> a rather novel technique described by Kohne et al.<sup>15</sup> was utilized in this study to obtain a very stable noise source. The form of this noise source is shown in Fig. D.1.

The low quality exciting noise source has rather mild restrictions placed upon it; it is necessary that the output of this source be equally likely to be above and below the comparison level of the comparator and that the number of transitions about this level occur at a high rate with respect to the clock frequency. Under these conditions, the comparator output signal is equally likely to pass or inhibit a clock pulse through the gate. The flip-flop output is a random square wave with transitions occurring only at points in time determined by the clock.

For random walks limited to less than one millisecond, the clock frequency should be of the order of one Mcps. (see Sec. 3.3), the frequency chosen for this study. As a result, it is necessary to use a low quality exciting noise source exhibiting energy density to beyond 10 Mcps.

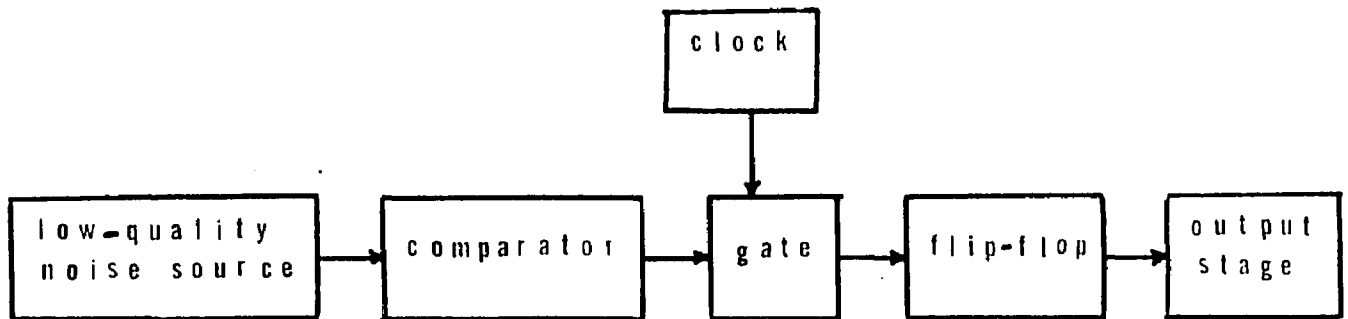


Figure D.1 Block Diagram of Noise Generator

Certain zener noise diodes exhibit this high frequency noise characteristic; however, these diodes are usually expensive and require high voltage power supplies. It was experimentally found that certain families of transistors, usually the epoxy case variety, generate significant amounts of high frequency noise when the base-emitter junction is operated in the reverse bias region. Since the breakdown voltage of the base-emitter junction is usually between three and ten volts, the low voltage power supplies of ASTRAC II, + 15 volts are appropriate for the operation of the transistor in this mode.

The amplitude of the output pulse signal can be held constant; therefore, the quality of the noise generator is only dependent upon the correlation existing between adjacent segments of the flip-flop waveform. This is specified by the autocorrelation function of the output signal. Ideally, the normalized autocorrelation function should appear as shown in Fig. 3.2; the measured autocorrelation function of the noise generator is given in Fig. D.2.

The high speed and packaging requirements make fast integrated circuits attractive for use in this application. The overall noise generator schematic utilizing Motorola MECL integrated circuits appears in Fig. D.3.

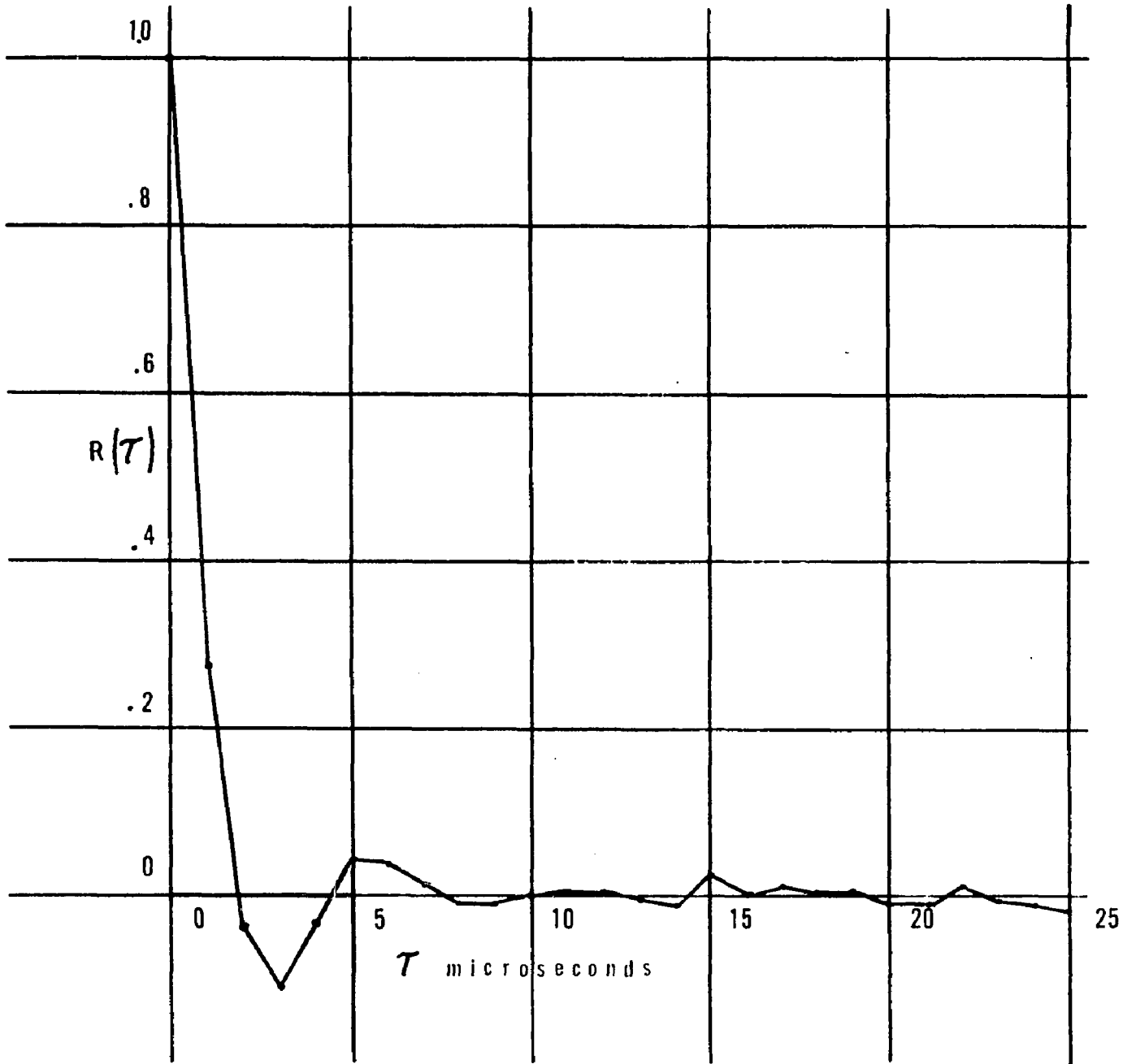


Figure D.2 Measured Autocorrelation Function of the Noise Generator



## REFERENCES

1. Tomovic, R., and Karplus, W. J. High-Speed Analog Computers. New York: John Wiley and Sons, Inc., 1962.
2. Forsythe, G. E., and Wasow, W. R. Finite-Difference Methods for Partial Differential Equations. New York: John Wiley and Sons, Inc., 1960.
3. Bharucha-Reid, A. T. Elements of the Theory of Markov Processes and Their Applications. New York: McGraw-Hill Book Company, Inc., 1960.
4. Korn, G. A. "Hybrid Computer Monte Carlo Techniques," Simulation, Vol. 5, No. 4 (October, 1965), pp. 234-245.
5. Wang, Ming Chen, and Ulenbeck, G. E. "On the Theory of Brownian Motion II," Reviews of Modern Physics, Vol. 17, Nos. 2 and 3 (April-July, 1945), pp. 323-342.
6. Churchill, R. V. Complex Variables and Applications. New York: McGraw-Hill Book Company, Inc., 1960.
7. Wasow, W. "Random Walks and the Eigenvalues of Elliptic Difference Equations," Journal of Research of the National Bureau of Standards, Vol. 46, No. 1 (January, 1951), pp. 65-73.
8. Curtiss, T. J. H. "Sampling Methods Applied to Differential and Difference Equations," Proc. Seminar on Scientific Computation, International Business Machines Corp. New York, N. Y., November, 1959.
9. Metroplis, N., and Ulam, S. "The Monte Carlo Method," Journal American Statistical Association, Vol. 44 (1949), pp. 335-341.
10. Chuang, K., Kazda, L. F., and Windeknecht, T. "A Stochastic Method of Solving Partial Differential Equations Using an Electronic Analog Computer," Project Michigan Report 2900-91-T, Willow Run Laboratories, University of Michigan, June, 1960.

11. Little, W. D. Hybrid Computer Solutions of Partial Differential Equations by the Monte Carlo Method, University of British Columbia, Ph.D. Thesis, British Columbia, Canada, October, 1965.
12. Korn, G. A. "New High-Speed Analog and Analog-Digital Computing Techniques: The ASTRAC System," Proceedings 3rd AsICA Conference, Opatija, Yugoslavia, 1961, Presses Académiques Européennes, Brussels, 1962.
13. Davenport, W. B., and Root, W. L. An Introduction to the Theory of Random Signals and Noise. New York: McGraw-Hill Book Company, Inc., 1958.
14. Parzen, E. Stochastic Processes. San Francisco: Holden-Day, Inc., 1962.
15. Kohne, H., Little, W. D., Soudack, A. C. "An Economical Multichannel Noise Source," Simulation, Vol. 5, No. 8 (November, 1965).
16. Parzen, E. Modern Probability Theory and Its Application. New York: John Wiley and Sons, Inc., 1960.
17. Shreider, Yu A. Method of Statistical Testing. Amsterdam: Elsevier Publishing Company, 1964.
18. Lym, W. R. "Plotting the Equipotential Lines of a Potential Field," Simulation, Vol. 4, No. 4 (February, 1965).
19. Korn, G. A. Random-Process Simulation and Measurements. New York: McGraw-Hill Book Company, 1966.
20. Eckes, H. R., and Korn, G. A. "Digital Program Control for Iterative Differential Analyzers," Simulation, Vol. 2, No. 2 (February, 1964).
21. Korn, G. A., and Korn, T. M. Mathematical Handbook for Scientists and Engineers. New York: McGraw-Hill Book Company, Inc., 1961.
22. Churchill, R. V. Fourier Series and Boundary Value Problems. New York: McGraw-Hill Book Company, Inc., 1941.
23. Grober, H., and Erk, S. Fundamentals of Heat Transfer. New York: McGraw-Hill Book Company, Inc., 1961.

24. Chapman, A. J. Heat Transfer. New York: The Macmillan Company, 1960.
25. Middleton, D. Statistical Communication Theory. New York: McGraw-Hill Book Company, Inc., 1960.
26. Jaede, E. K. An Application of the Monte Carlo Method to the Solution of Elliptic Partial Differential Equations, Grumman Research Department Memorandum R-238, July, 1964.
27. Korn, G. A. "Hybrid-Computer Techniques for Measuring Statistics from Quantized Data," Simulation, Vol. 4, No. 4 (April, 1965).
28. Korn, G. A. Private Communication, August, 1966.