SENSITIVITY ANALYSIS APPLIED TO COMPUTER AIDED CIRCUIT DESIGN

by

Ka-Ho Leung

A thesis submitted for the degree of
Doctor of Philosophy in the Faculty
of Engineering, University of London.

Department of Electrical Engineering,
Imperial College of Science and Technology,
University of London.

April 1976

ABSTRACT

This thesis is concerned with the development of efficient techniques for various computational requirements relating to linear circuit sensitivities. Attention is paid to the situations where the changes in component values cannot be assumed to be infinitesimal.

Two efficient methods for predicting the effect on circuit response of simultaneous changes in a number (m) of components are examined. It is shown that the computational cost of predicting the effect of each set of component changes is approximately G_m, where G_m is the cost of a Gaussian elimination of an m x m matrix. However, if the sets represent combinations of relatively few individual component changes, then systematic exploration is shown to reduce the computational cost *per set* to the order of one. This remarkable saving in computational cost makes systematic exploration a powerful tool in computer aided circuit design.

The use of systematic exploration leads to the discovery of an efficient method for the generation of multi-dimensional performance contours. The efficiency of the method makes it particularly suitable for interactive graphic display implementation. This is illustrated with a 3-dimensional example.

For the situation where a number of components in a circuit exhibit the same form of functional dependance on a global variable such as temperature, the concept of tracking sensitivity is introduced. An algorithm for the computation of tracking sensitivity is devised. Rapid exploration of the effect is permitted by the small number of operations associated with each value of the global variable following an initial calculation.

For the prediction of manufacturing yield, a modelling technique based on the regionalization of a variable component space is combined with the computational technique of systematic exploration to provide a very efficient prediction of the spread in circuit performance due to spread in a number of components. Illustrative examples enable the new approach to be compared with the Monte Carlo technique.

Finally, a method is proposed for design centering and tolerance assignment. This method, termed orthogonal silhouette method, is based on the fact that in a regionalized variable component space, all edges leading to a common vertex are orthogonal. Being compatible with systematic exploration, it enables design centering, in a space where the number of dimensions is large, to be performed with great accuracy and economy.

ACKNOWLEDGEMENTS

The author wishes to express his deep gratitude to his Supervisor, Dr. R. Spence, for his guidance and encouragement during the course of the work reported in this thesis.

He also wishes to acknowledge the stimulating discussions he has had with Mr. M. Chen, Dr. J.C. Allwright, Dr. R.K. Brayton, Prof. P.R. Bryant and various other people too numerous to mention.

The programming assistance and advice of Jadwiga Dunn is also gratefully acknowledged.

To my uncle,

Mr. T. M. Liang.

CONTENTS

LIST OF FIGURES

PAGE

PAGE

LIST OF TABLES

STATEMENT OF ORIGINALITY

The contributions listed below are original as far as the author is aware. Any results which are not original to the author are acknowledged by references in the text.

1. The analysis of multi-parameter large change sensitivity. The exploitation of the computational and conceptual aspect of two available techniques - the current substitutional method and matrix modification method - for the computation of large change sensitivity. The implementation of the multi-parameter large change sensitivity algorithm within an interactive graphic circuit design facility [32] .

2. The concept and computational method of "Systematic Exploration" which is shown to reduce the computational cost per set of variable components to the order of one, if the effect on linear circuit response of simultaneous changes in a number of components is computed and the sets represent combinations of relatively few individual component changes [32] .

3. The efficient method for generating the multi-dimensional performance contour. The implementation of this method within an interactive graphic circuit design facility [32,8] .

4. The concept and computational method of "Tracking Sensitivity" which permits the effect on linear circuit response of the global variable (e.g. temperature of other parameters) to be efficiently computed. The implementation of this method within an interactive graphic circuit design facility [36] .

5. The concept of the regionalized variable component space and its application to efficient statistical circuit analysis [4] . The method for the worst case simulation of a regionalized variable component space.

6. The concept of "Idealized Statistical Model" and its application to the efficient prediction of manufacturing yield. The implementation of this technique within an interactive graphic circuit design facility [5,46] .

7. The proposal of "Orthogonal Silhouette Method" and its application to design centering and tolerance assignment.

CHAPTER 1

COMPUTER AIDED CIRCUIT DESIGN

1.1 Introduction

The traditional design process of an electronic circuit may be characterized by a simple block diagram as shown in Fig. 1.1. From available technology, design technique, and experience, the design engineer formulates a possible design to meet the specifications. The design must then be analysed to find out if the specifications are met and if the proposed design is reasonable. Then, the next step is to construct a breadboard model and test the design experimentally. If all proves satisfactory, the designer can ask for the design to be laid out ready for a prototype construction.

As electronic circuits have become increasingly complicated, designers have had to rely on more sophisticated design techniques. In recent years, engineers have turned with ever increasing attention to the use of computers for electronic circuit design and it has become common practice to simulate circuit designs, both discrete and integrated, on digital computers. An obvious example of the application of computer aided circuit design is for integrated circuit design, where simulation using discrete components may prove to be unrealistic at best, if not impossible in practice.

As many copies of a circuit are to be manufactured in a modern mass production factory, the need to reduce the manufacturing cost and predict the manufacturing yield are of great commercial significance. To arrive at a minimum cost design (i.e., one using maximum component tolerances) commensurate with minimum specification failure of the manufactured circuit, the process known as toleranced circuit design [1,2] must be invoked. The manufacturing yield may be determined from a statistical analysis of the circuit [3,4,5] . Although the above processes are greatly beneficial to a better circuit design, they require a huge number of computational operations which can only be practically performed with the aid of a digital computer.

The use of interactive graphic facilities [6,7,8] in computer aided circuit design enables the designer to exploit the computer's potential to the full. For this reason, throughout this thesis, the discussion of computer aided circuit design is within the context of an interactive graphic facility.

1.2 A Typical CACD Process

The principal tools at the designer's disposal for a typical computer aided circuit design package are programs which perform ac,dc, transient, sensitivity and statistical analysis and other functions as shown in Fig. 1.2. Over many years, much research work has been done in improving and extending the functions within this package [8] .

At the initial step of a computer aided circuit design, the devices such as transistors and operational amplifiers in the circuit are suitably simulated by models. These models must be able to describe the behaviour of the device embedded in its circuit environment. Once suitable device models are available, a circuit can be simulated on a digital computer. The circuit description for a computer input can be highly user oriented and allow considerable freedom in the description of numerical values of the components, and the topology of the circuit [9] . A set of circuit analysis programs, which are usually based on nodal analysis can now produce the desired response measure which is then presented to the designer graphically or numerically.

Having obtained the initial analysis result, the designer often wants to speculate and explore, and to ask 'what if?' questions: "What happens if I change these components by a large amount?". The change can be made either manually by the designer as in the process of exploring the effect of the change of one or a group of components dynamically as in the use of an interactive graphic facility, or automatically as in an optimisation or Monte Carlo process. Knowing the analysis result enables tolerances to be assigned to the variable components, or the circuit to be modified by the designer.

For the above mentioned processes, sensitivity analysis is used for the exploration of a design with respect to component perturbation. For truly interactive computer aided design to be feasible, efficient methods for the computation of circuit sensitivity are required. Since a minimization of computational effort allows results to be returned promptly, the remaining chapters of this thesis will be devoted to the sensitivity analysis of a linear circuit.

1.3 Device Modelling

To facilitate the description of an electronic circuit for computer aided circuit design, complex devices such as transistors and operational amplifiers must be represented with an acceptable degree of approximation. There are a number of ways to describe the electrical behaviour of a device. In terms of network analysis by computer, however, these different models can be grouped into five classifications:

1. A numerical constant.
2. An analytic expression.
3. A table.
4. A four terminal black box.
5. An equivalent circuit.

In performing an analysis, not all five models will be equivalent in terms of accuracy and computation. Usually, the equivalent circuit type model is chosen for computer aided circuit design. The reasons are: Firstly, the equivalent circuits are easily interpretable with the known circuit theory, which allows the designer to have better insight into the model. Secondly, many techniques which have proved themselves very useful in the field of circuit analysis and design can also be applied to the analysis and synthesis of models.

Several points must be considered when representing an active device by an equivalent circuit:

1. Operating region. If the device performs only over a small range of voltage and current, the parameters in the models may be regarded as approximately constant.

2. Frequency range. When the analysis is over a limited frequency range, additional simplification may be made in representing the device.

3. Accuracy. Generally, in any analysis, the degree of accuracy required will greatly affect the complexity of the equivalent circuit used in the modelling.

4. Measurability of parameters. Not only must numerical values for the circuit elements in the model be determined, but their dependence on electrical and physical conditions must be determined as well.

One example of device modelling is illustrated in Fig. 1.3. The active device - a transistor - used in the different examples which follow is modelled using the hybrid-π circuit. The component values in this model are quoted in the manufacturers technical data of π-parameters for the device in question.

Models which allow accurate analysis are normally very complex, and call for a large number of device measurements and calculations. Thus, to reduce the computing time involved in network analysis, and to gain insight into the behaviour of the device, it is necessary to employ models of reduced complexity. Therefore, one important aspect of device modelling is the problem of realizing maximum accuracy with minimum model complexity. The model simplification technique due to Villalaz et al [10] which allows the complexity of linear equivalent circuit models to be reduced to a minimum level is shown in Fig. 1.4. In this example, the equivalent circuit model of an operational amplifier is simplified according to the particular application and to the tolerance on the circuit response.

1.4 Basic Network Analysis Procedure

The development of a general network analysis computer program must be based on the framework of a mathematical description, of the general node (or loop) system of equations [11,12] . Otherwise the program will be reduced to solving special cases and would consequently be of limited usefulness.

Physically, a general branch of an electrical network may consist of four distinct electrical devices:

1. A passive element.
2. An ideal voltage source in series with the passive element.
3. An ideal current source in parallel with the above combination.
4. A voltage controlled current source in parallel with the above combination.

The relative position and orientations of these electrical elements for a general branch are shown in Fig. 1.5. If we use the notation of that figure as definitions for voltages and currents, then the branch is described by

$$i_b = y_b(v_b - e_s) - i_s + i_{cs} \qquad (1.1)$$

where i_b and v_b are the branch current and branch voltage respectively, i_s and e_s are independent current and voltage sources and i_{cs} is a voltage controlled current source.

Equation 1.2 is obtained by forming all the branch equations into the matrix form

$$I_b = Y_b \, (V_b - E_s) - I_s + I_{cs} \qquad (1.2)$$

where Y_b is a diagonal matrix and the remaining quantities are column vectors. On pre-multiplying equation 1.2 by the reduced incidence matrix A, [13], the left hand side of the equation goes to zero, we get

$$0 = A\, Y_b (V_b - E_s) - A\, I_s + A\, I_{cs} \qquad (1.3)$$

This leads to an expanded version of the nodal analysis equation.

$$A\ Y_b\ A^t\ V_N\ +\ A\ G_b\ A^t\ V_N\ =\ A\ I_s\ +\ A\ Y_b\ E_s \qquad (1.4)$$

In this equation, V_N is the vector of nodal voltages, G_b is the non-diagonal matrix of mutual conductances, and the term for the controlled current sources has been expressed explicitly as $G_b\ A^t\ V_N$. If a comparison with the usual compact form of the expression for nodal analysis

$$Y_N\ V_N\ =\ I_N \qquad (1.5)$$

is made, it is seen that

$$Y_N\ V_N\ =\ A\ Y_b\ A^t\ V_N\ +\ A\ G_b\ A^t\ V_N \qquad (1.6)$$

and that

$$I_N\ =\ A\ I_s\ +A\ Y_b\ E_s \qquad (1.7)$$

Matrix equation 1.5 may be solved by one of the techniques which yield all or some of the nodal voltages depending on the method chosen. In the following work of the thesis, when computational cost is to be assessed, it is assumed that the Gaussian method [14] is used for the solution of linear equations and the Gaussian-Jordan complete elimination method [14] is used for the inversion of a matrix.

1.5 Man-Computer Dialogue

In order to make interactive computer aided circuit design feasible, an effective communcation dialogue must be established between man and machine [15] . In circuit design the natural language of the engineer is the circuit diagram. Therefore, the communication language between man and machine should be graphical. Such communication is possible with an input-output device known as a graphics console. A man sitting at the CRT display could enter data (such as a circuit diagram) directly on the CRT face by using a light pen. These data are then converted into an internal form which is acceptable by the circuit analysis program residing in the computer. The circuit analysis program calculates the requested response, and outputs data which are in turn translated into pictorial form (Fig. 1.6). The designer views the graphic output and makes changes in his original circuit diagram. Then the process is repeated until a satisfactory design is achieved. The process can be analogous to an engineer making actual changes in his breadboard circuit and viewing the result on an oscilloscope.

How does a designer "draw" his circuit on the face of the display scope using a light pen and programmed control buttons? In drawing a circuit on the scope face, the user should not have to label a resistor such as R_1, R_2 .....R_n and specify that R_1 is connected between nodes number 10 and 13. The user needs only pick a resistor light button, fetch it to the display surface, connect it to the appropriate nodes, and give the resistor a value. The computer is delegated the task of

1. assigning node numbers to keep track internally of circuit data,
2. keeping a list of which components are connected to which nodes, and the values and types of the components,
3. assisting the composing of the circuit in step-by-step fashion.

Fig. 1.7 shows how a drawn circuit diagram appears on the face of the display scope and Fig. 1.8 is the response of the circuit found from the analysis. Fig. 1.9 is the computer print out description of the circuit.

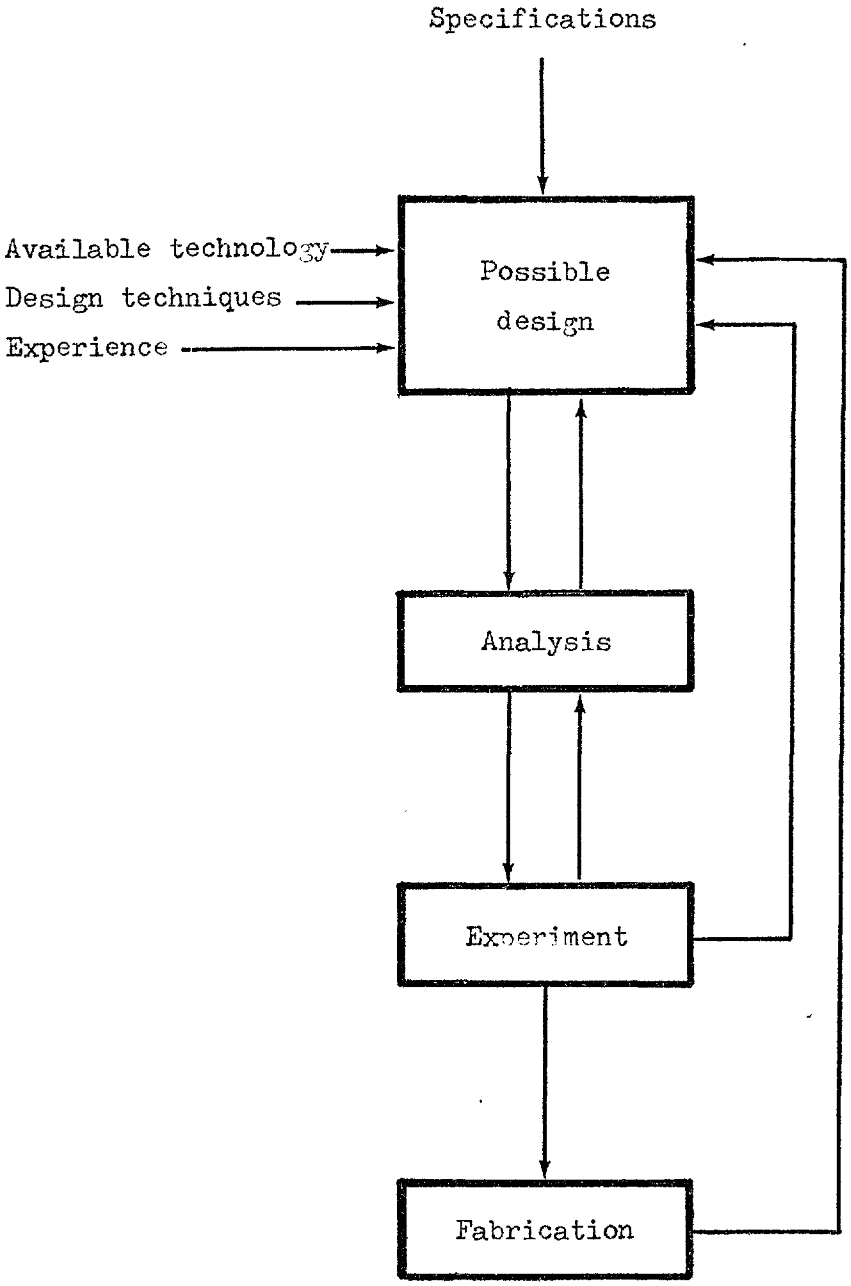


Fig. 1.1. The design process.

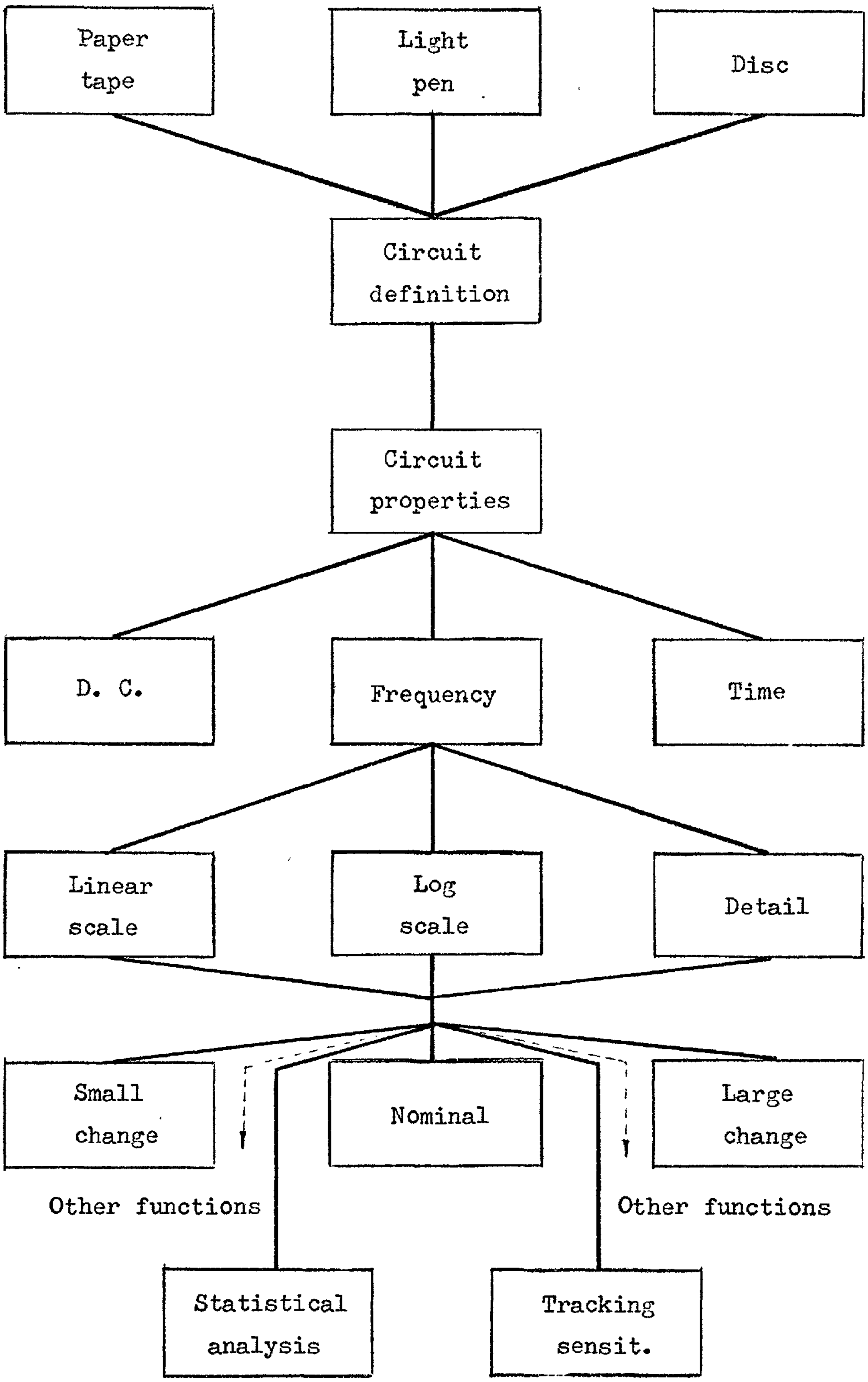


Fig. 1.2. The principal functional blocks of a typical computer aided circuit design facility

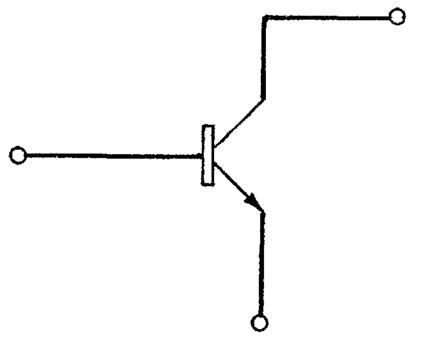


Fig. 1.3(a). A Transistor

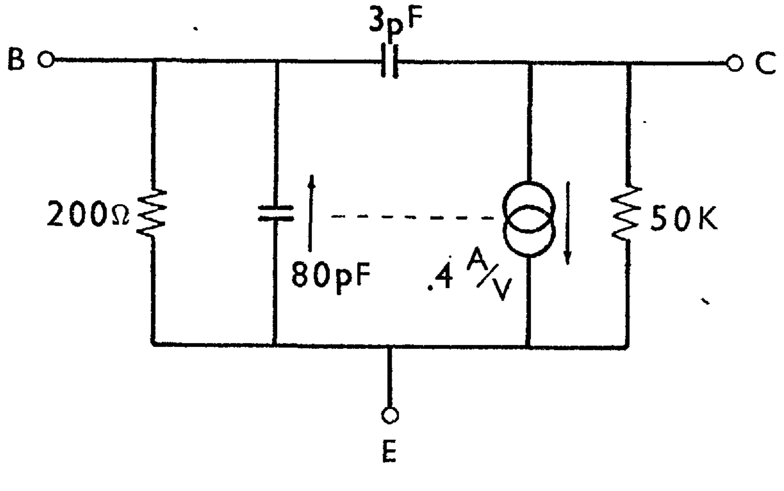


Fig. 1.3(b). Hybrid-π model of the transistor in common emitter configuration.

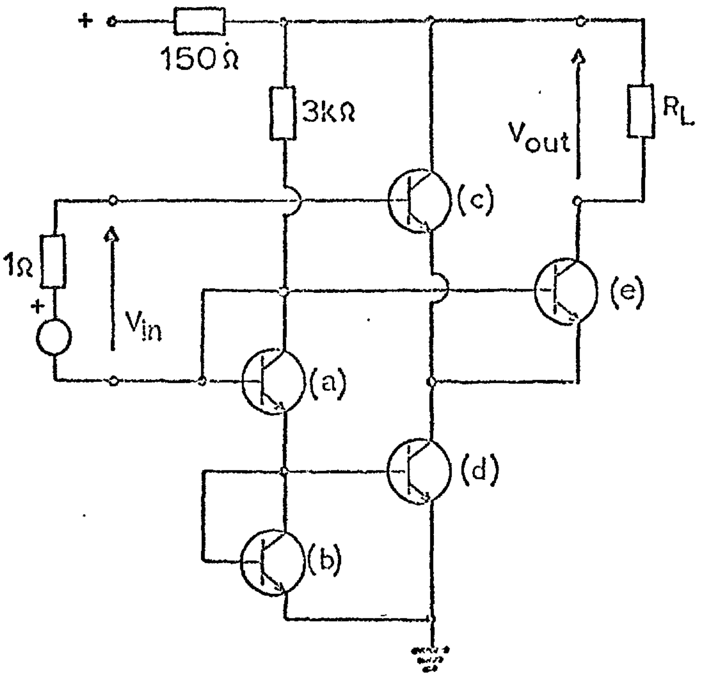


Fig. 1.4(a). Original operational amplifier circuit.

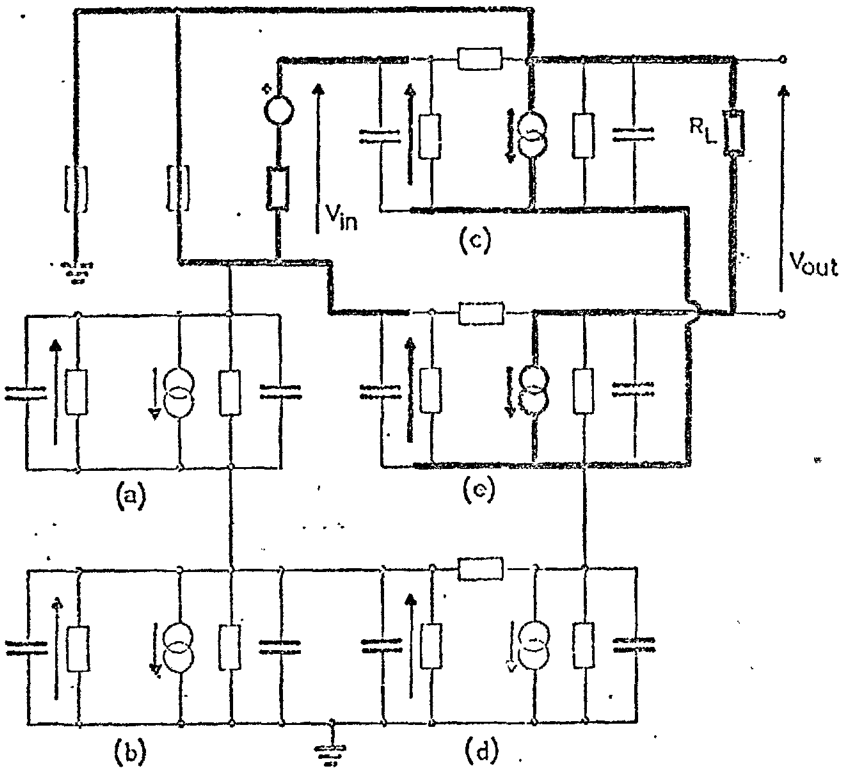


Fig. 1.4(b). Small signal equivalent circuit model of the operational amplifier, the heavy lines identifying the simplified model

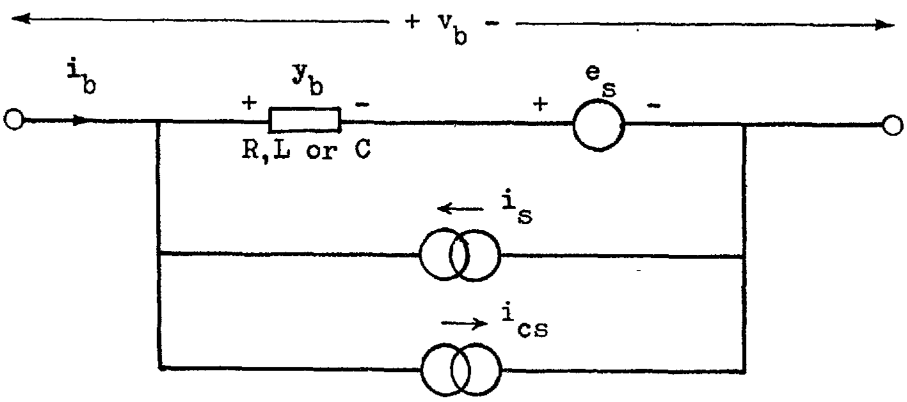


Fig. 1.5. A general network branch.

y_b - passive element.

e_s - ideal current source.

i_s - ideal voltage source.

i_{cs} - voltage controlled current source.

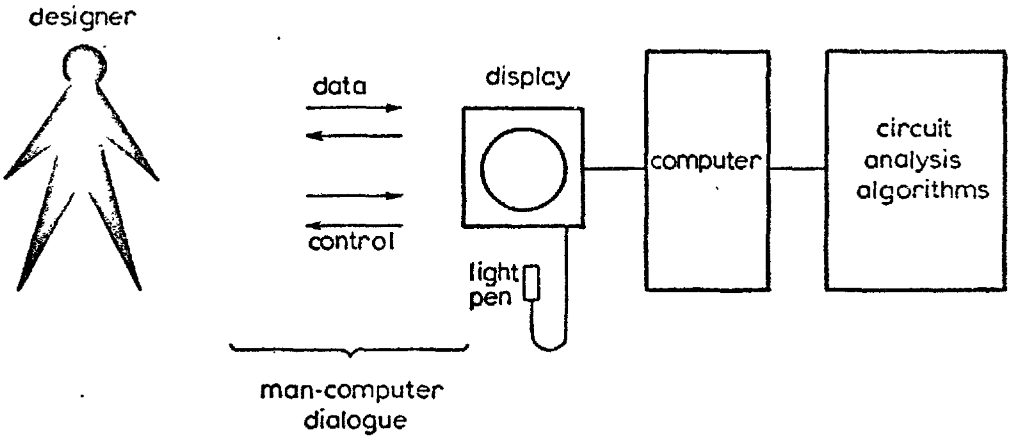


Fig. 1.6. Man-computer interaction in the process of computer aided circuit design.

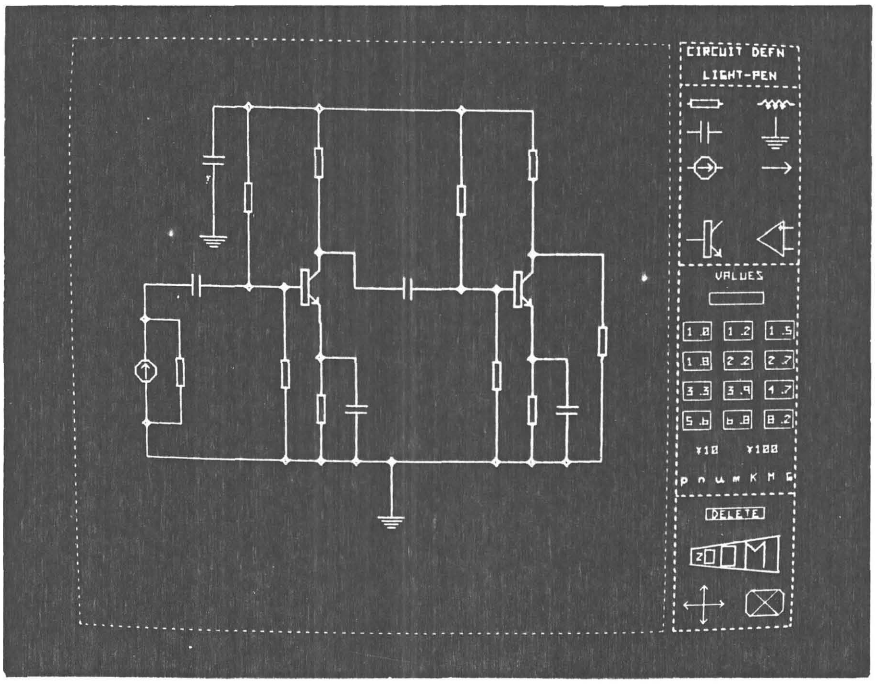


Fig. 1.7. A circuit diagram drawn on the face of the display scope.

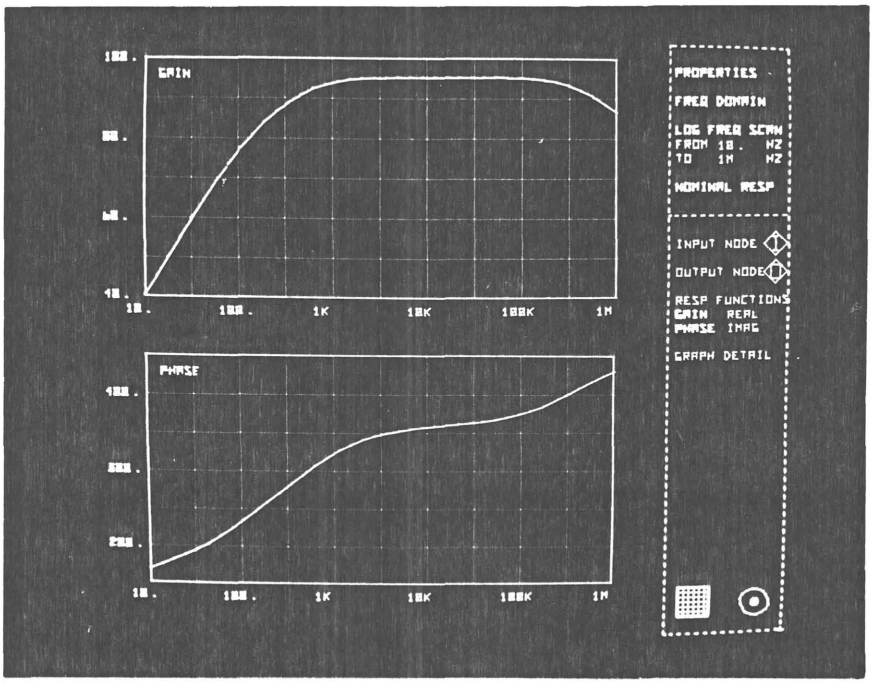


Fig. 1.8. The response curves of the drawn circuit.

```
**** MINNIE CIRCUIT DATA ****    DATE  1/ 4/1976

11 NODES,  18 BRANCHES. NODE  4 IS INPUT, NODE 11 IS OUTPUT.
```

```
                     **3****3****3****3****3****3****3****3***
                     3         3         3         3         3
                     *         *         *         *         *
                     *         *         *         *         *
                     3        R2*       C18        3        R11
                     *        R2*       C18        *        R11
                     *         *         *         *         *
                    R3*        2         E        R12        1
                    R3*        *         E        R12        1
                     *         **2***    *         *         **11***11**
                     1        C*+   2   EEE        8        C*+        1
                     *        C*+   *    E         *        C*+        1
 **4****C4***1****1****1****T1**    **2****C9***8****8****T10*         *
 4      C4*     1           T1E*-          C9*          8 T1E*-        1
 *              *             E*-                       *   E*-        1
 **4***         *              *                        *    *         *
 4    4         1              5                        8    1        R17
 *    *         *              *                        *    0        R17
 *    *         *              **5***                   *    **10**    *
15*  R8*       R6*             5    5                  R14   1    1    E
I5*  R8*       R6*             *    *                  R14   0    0    E
 *    *         *              *    *                   *    *    *    *
 E    E         E             R7*  C16                  E   R13  C15  EEE
 E    E         E             R7*  C16                  E   R13  C15   E
 **EE**         *              *    *                   *    *    *    *
 E              E              E    E                   E    E    E
 E              E              E    E                   E    E    E
 **EE***EE***EE***EE***EE***EE***E E***EE***EE***EE***EE***EE***EE**
                                         E
                                         E
                                         *
                                        EEE
                                         E
                                         *
```

```
BRANCH      BETWEEN          VALUE              CONTROLLED
             NODES                               BY NODES

 T 1    1(B)  2(C)  5(E)     @@@@@@
 R 2        2     3       2. 200E+03 OHMS
 R 3        1     3       1. 000E+05 OHMS
 C 4        4     1       1. 000E-05 FARAD
 I 5        0     4       1. 000E+00 AMPS
 R 6        0     1       2. 200E+04 OHMS
 R 7        0     5       4. 700E+02 OHMS
 R 8        0     4       5. 600E+02 OHMS
 C 9        2     8       1. 000E-05 FARAD
 T10    8(B) 11(C) 10(E)     @@@@@@
 R11       11     3       2. 200E+03 OHMS
 R12        8     3       1. 000E+05 OHMS
 R13        0    10       4. 700E+02 OHMS
 R14        0     8       2. 200E+04 OHMS
 C15        0    10       1. 000E-04 FARAD
 C16        0     5       1. 000E-04 FARAD
 R17        0    11       1. 000E+06 OHMS
 C18        0     3       1. 000E-04 FARAD
```

Fig. 1.9. Computer print out description of the drawn circuit.

CHAPTER 2

SMALL CHANGE SENSITIVITY

2.1 Introduction

Once an initial design has been completed, this design should be simulated to find the various factors affecting the performance of the circuit. One factor which determines the quality of the circuit is the effect on its output of a small change in any component value. The small change sensitivity of a component is defined as the normalized change in output for a normalized minute change in component value.

Consider a network (Fig. 2.1(a)) with branches described by (1.1) which can be rewritten with the term for the controlled current source expressed as mutual conductance g_m, and controlling voltage v_b'

$$i_b = y_b(v_b - e_s) - i_s + g_m v_b' \qquad (2.1)$$

Differentiation of this branch equation with respect to a dummy variable, χ , provides the starting point of all the well known methods of small change sensitivity analysis.

The differentiation of (2.1) has three possible results depending on what component χ represents.

1. If χ is not a component of the branch under consideration ($\chi \neq y_b$ and $\chi \neq g_m$) then the differential is

$$\frac{\partial i_b}{\partial \chi} = y_b \frac{\partial v_b}{\partial \chi} + g_m \frac{\partial v_b'}{\partial \chi} \qquad (2.2)$$

2. If χ is the branch admittance ($\chi = y_b$) then the differential is

$$\frac{\partial i_b}{\partial \chi} = y_b \frac{\partial v_b}{\partial \chi} + g_m \frac{\partial v_b'}{\partial \chi} + (v_b - e_s) \qquad (2.3)$$

3. The last alternative is that in which χ is the branch mutual conductance ($\chi = g_m$), then the differential is

$$\frac{\partial i_b}{\partial \chi} = y_b \frac{\partial v_b}{\partial \chi} + g_m \frac{\partial v_b'}{\partial \chi} + v_b' \qquad (2.4)$$

Use of the Kronicker - Delta function,

$$\delta_{ij} = 1 \qquad \text{if} \qquad i = j$$

$$\delta_{ij} = 0 \qquad \text{if} \qquad i \neq j$$

allows (2.2) to (2.4) to be expressed far more compactly as

$$\frac{\partial i_b}{\partial \chi} = y_b \frac{\partial v_b}{\partial \chi} + g_m \frac{\partial v_b'}{\partial \chi} + \delta_{\chi y_b}(v_b - e_s) + \delta_{\chi g_m} v_b' \qquad (2.5)$$

Pre-multiplication of the generalized matrix form of the equation by the reduced incidence matrix, A, gives

$$A\ Y_b \left[\frac{\partial v_b}{\partial \chi}\right] + A\ G_b \left[\frac{\partial v_b}{\partial \chi}\right] = -A\left[\delta_{\chi y_b}(v_b - e_s) + \delta_{\chi g_m} v_b'\right] \qquad (2.6)$$

which may also be rewritten as

$$A\ Y_b\ A^t \left[\frac{\partial v_N}{\partial \chi}\right] + A\ G_b\ A^t \left[\frac{\partial v_N}{\partial \chi}\right] = -A\left[\delta_{\chi y_b}(v_b - e_s) + \delta_{\chi g_m} v_b'\right] \qquad (2.7)$$

The left hand sides of this equation are identical in form with (1.5). This fact means that (2.7) may be interpreted as a new network, topologically identical to the original and with the same component values, from which the voltage differentials of the original network may be found. These differentials with respect to a component χ , are equal to the nodal voltages of the new network when the network is driven by a suitably valued current source across the same component χ . As shown by the right side of equation 2.7, this one current source, of value $v_b - e_s$, or v_b', across χ is the only independent source in the new network. The new network, shown in Fig. 2.1(b), is the auxiliary network proposed by J.V. Leeds [16] . Hence, by analysing one extra circuit, the sensitivities of all possible transfer impedances to one chosen component can be found.

In a sensitivity analysis procedure, we normally require the sensitivity of one particular output to all the components. If the network under consideration is limited to the reciprocal set, then, using the principle of reciprocity [17] , the auxiliary network may be used to find all the component sensitivities with respect to one nodal voltage. Consider forcing the auxiliary network across the output port by a unit current source (Fig. 2.1(c)). The voltage, Ψ , appearing across any component, χ , is identically equal to the voltage that would appear across the output if the current source had been placed across the component in question. Therefore, if the auxiliary network is analysed when the output is being forced by a unit current source, then the sensitivity of the output voltage to any component is the product of the original component voltage and the auxiliary component voltage.

2.2 Tellegen's Theorem

The restriction of the Leeds' technique to reciprocal networks was lifted by the introduction of the 'adjoint network' [18,19]. In order to examine the adjoint circuit theory in detail, the concept of Tellegen's theorem [20] must first be reviewed.

Tellegen's theorem states that if $i_1', i_2', \ldots\ldots, i_b'$ are the branch currents of a b-branch network N', and $v_1'', v_2'', \ldots\ldots, v_b''$ are the branch voltages of another b-branch network N'', where N' and N'' have a common linear topology but may otherwise be different, then

$$\sum_{\alpha} i_\alpha' v_\alpha'' = 0 \tag{2.8}$$

where the summation is over all branches of the network. The sign convention adopted for branch voltage and current is such that if N' and N'' were identical, the product $i_\alpha' v_\alpha''$ would be the instantaneous power supplied to the branch.

If some branches are, in fact, ports of the network, the products associated with the ports can conveniently be placed on the opposite side of the equality sign to yield

$$\sum_{\alpha} i_\alpha' v_\alpha'' = \sum_{p} i_p' v_p'' \tag{2.9}$$

where α and p now denote internal branches and ports respectively. Equation 2.9 is the theorem originally presented by Tellegen and has since been known as Tellegen's theorem. However, it is a special case of a more general form of the theorem. The general form of Tellegen's theorem was not found until 1970 when it was first introduced [21]. This form of Tellegen's theorem has been shown to be of particular value in small change sensitivity computation.

The generalized form of Tellegen's theorem is expressed in terms of "Kirchhoff operators". The purpose of these operators is to derive from one set of currents (or voltages) that obeys Kirchhoff's current (or voltage) law, another set of quantities that obeys the law. For example if the currents $\left[i_a(t)\right]$ obey Kirchhoff's current law, then so do their derivatives $\left[di_a(t)/dt\right]$.

Let Λ' be a Kirchhoff current operator whose effect upon the set of branch currents i_α of a b-branch network is the generation of a new set of a b-branch "currents" $\Lambda' i_\alpha$ that obeys Kirchhoff's current law. Similarly let Λ'', a Kirchhoff voltage operator, operate upon the set of branch voltages v_α to generate a new set of branch "voltages" $\Lambda'' v_\alpha$ that obeys Kirchhoff's voltage law. For a network with ports, it then follows immediately from this, that

$$\sum_\alpha \Lambda' i_\alpha \Lambda'' v_\alpha = \sum_p \Lambda' i_p \Lambda'' v_p \tag{2.10}$$

where i_p and v_p are the port currents and voltages respectively and the indices α and p are over all the branches and ports of the network. Equation 2.10 is the generalized form of Tellegen's theorem.

In many applications of the generalized form of Tellegen's theorem, it is simpler to apply what is called the difference form of the theorem. The difference form of Tellegen's theorem is obtained by interchanging the roles of Λ' and Λ'' in (2.10) and by subtracting the result from (2.10):

$$\sum_\alpha (\Lambda' i_\alpha \Lambda'' v_\alpha - \Lambda'' i_\alpha \Lambda' v_\alpha) = \sum_p (\Lambda' i_p \Lambda'' v_p - \Lambda'' i_p \Lambda' v_p) \tag{2.11}$$

This form of Tellegen's theorem has shown particular usefulness as a basis for the efficient computation of small change sensitivities.

2.3 Adjoint Network

Consider a network N containing linear reciprocal or nonreciprocal two terminal elements described by their branch admittance y_α. Let the branch voltages and currents of N be denoted by v_α and i_α respectively. Beside the original network, consider another adjoint network denoted as $\tilde{N}$. We impose the requirement that both N and $\tilde{N}$ have the same topology, but not necessarily the same component type in corresponding branches. This requirement is made so that Tellegen's theorem can be applied. If $\tilde{v}_\alpha$ and $\tilde{i}_\alpha$ denote branch voltages and currents in the adjoint network, we may rewrite the left side of equation 2.11 in which Λ' selects the small variations in the actual network and Λ'' selects the adjoint network, to get

$$\sum_\alpha (\delta i_\alpha \tilde{v}_\alpha - \tilde{i}_\alpha \delta v_\alpha) \tag{2.12}$$

If the circuit contains two terminal reciprocal components which are described by an admittance y_α, then, substituting i_α and $\tilde{i}_\alpha$ with $v_\alpha y_\alpha$ and $\tilde{v}_\alpha \tilde{y}_\alpha$ respectively yields

$$\sum_\alpha \left[\delta\ v_\alpha\ \tilde{v}_\alpha\ (y_\alpha - \tilde{y}_\alpha) + v_\alpha\ \tilde{v}_\alpha\ \delta\ y_\alpha\right] \tag{2.13}$$

Since the choice of the component types in the adjoint network is arbitrary, we may choose them in such a way so as to render (2.13) independent of all the $\delta\ i_\alpha$ or $\delta\ v_\alpha$ terms that are associated with nonsource branches. The reason behind this step is that for sensitivity calculations, we are interested in the variation of output response with respect to component variations and not with respect to branch voltage or current variations. By setting $\tilde{y}_\alpha = y_\alpha$ we get

$$\sum_{\alpha} v_\alpha \tilde{v}_\alpha \delta y_\alpha \qquad (2.14)$$

If the circuit contains multi-terminal non-reciprocal components, these components can be described by an admittance matrix $Y_{\alpha\beta}$ such that

$$i_\alpha = \sum_{\alpha\beta} Y_{\alpha\beta} v_\beta \qquad (2.15)$$

Substituting (2.15) into (2.12) yields

$$\sum_{\alpha\beta} \left[\delta(Y_{\alpha\beta} v_\beta) \tilde{v}_\alpha - \tilde{Y}_{\alpha\beta} \tilde{v}_\beta \delta v_\alpha \right] \qquad (2.16)$$

or

$$\sum_{\alpha\beta} \left[\delta v_\alpha \tilde{v}_\beta (Y_{\beta\alpha} - \tilde{Y}_{\alpha\beta}) + v_\beta \tilde{v}_\alpha \delta Y_{\alpha\beta} \right] \qquad (2.17)$$

By setting $\tilde{Y}_{\alpha\beta} = Y_{\beta\alpha}$ [17] , we get

$$\sum_{\alpha\beta} v_\beta \tilde{v}_\alpha \delta Y_{\alpha\beta} \qquad (2.18)$$

If a circuit contains two terminal reciprocal and multi-terminal non-reciprocal components, we have the following general expression for (2.12)

$$\sum_{\alpha} v_\alpha \tilde{v}_\alpha \delta y_\alpha + \sum_{\alpha\beta} \tilde{v}_\alpha v_\beta \delta Y_{\alpha\beta} \qquad (2.19)$$

Consider a two port network whose input port is forced by an independent current source and whose output voltage is of interest to the designer. The full expression of equation 2.11 has now become

$$\sum_{\alpha} v_{\alpha} \tilde{v}_{\alpha} \delta y_{\alpha} + \sum_{\alpha\beta} v_{\beta} \tilde{v}_{\alpha} \delta Y_{\alpha\beta} = - \tilde{i}_p \delta v_p \tag{2.20}$$

If the adjoint network is analysed with a unit current source forcing the output port, $(\tilde{i}_p = 1)$, the differential of the output voltage of the original circuit with respect to any component is the negative product of the original network component voltage and the adjoint network component voltage. Therefore we have

$$\frac{\delta v_p}{\delta y_{\alpha}} = - v_{\alpha} \tilde{v}_{\alpha} \tag{2.21}$$

for two terminal components, or

$$\frac{\delta v_p}{\delta Y_{\alpha\beta}} = - v_{\beta} \tilde{v}_{\alpha} \tag{2.22}$$

for multi-terminal non-reciprocal components.

It can now be noted that, for non-reciprocal circuits, if the auxiliary network proposed by Leeds is substituted with an adjoint network, the small change sensitivity of one network response to all elements can also be found by the analysis of two circuits - the original and the adjoint - as illustrated in Fig. 2.2(a) - (c).

The relation $\tilde{y}_{\alpha} = y_{\alpha}$ expresses the relationship for the branch impedance of the actual network and the adjoint network, and the relation $\tilde{Y}_{\alpha\beta} = Y_{\beta\alpha}$ expresses the relationship for the admittance matrix describing the multi-terminal elements of the actual network and the adjoint network.

2.4 Engineering Applications

The small change sensitivity algorithm described in the previous section has been implemented within an interactive graphic circuit design facility [8]. The small number of operations associated with the calculation enables the designer to explore dynamically the small change sensitivities due to frequency variations.

In the process of the interactive graphic circuit design, once the circuit has been input satisfactorily and displayed on the screen, the designer may activate the appropriate light button for the small change sensitivity display and indicate the components whose small change sensitivity is to be displayed. After a time lapse of roughly half a second*, the computer is ready to display the normalized sensitivity of the voltage response to the component.

Circles having radius proportional to sensitivity are now superimposed on component symbols indicating the magnitude of the sensitivity of the corresponding components. A light potentiometer appears on the right hand side of the screen (Fig. 2.3). The arrowed position indicates the frequency under which the sensitivities are evaluated. The display changes at roughly half second intervals as the pointer on the potentiometer automatically cycles through the frequency range of interest.

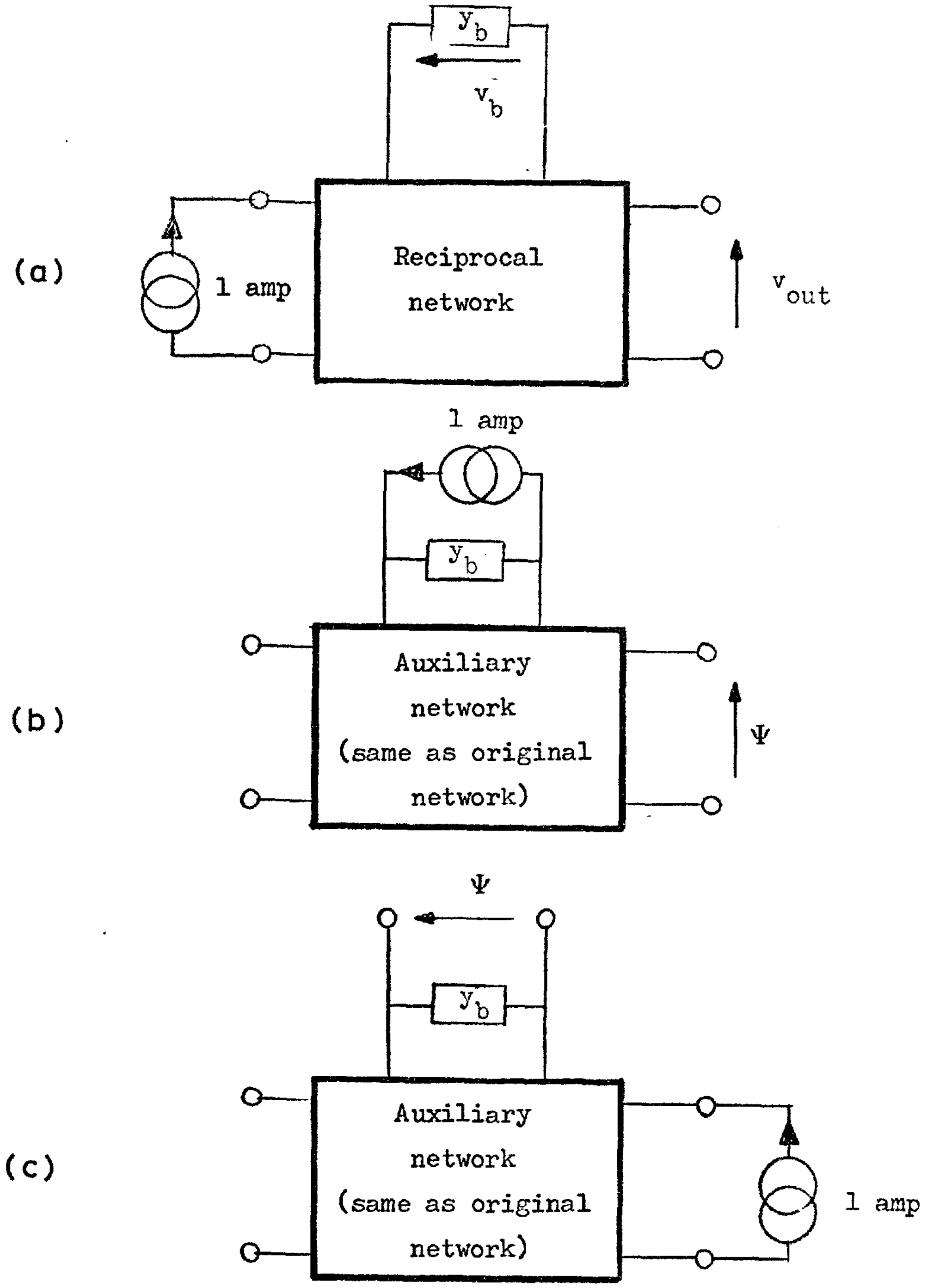
In some situations, a quantitative display of information is needed. Upon giving the appropriate command, the circles are replaced by numerical values as shown in Fig. 2.4.

By exploring the small change sensitivities of the components of a circuit, the following information may be obtained by the designer:

* This figure is related to the size of the circuit.

1. A feel for which are the significant components in various parts of the frequency range of interest.
2. The indication of components that might usefully be adjusted to improve circuit behaviour.
3. The indication of approximate tolerance for components.
4. The identification of any unduly sensitive components.

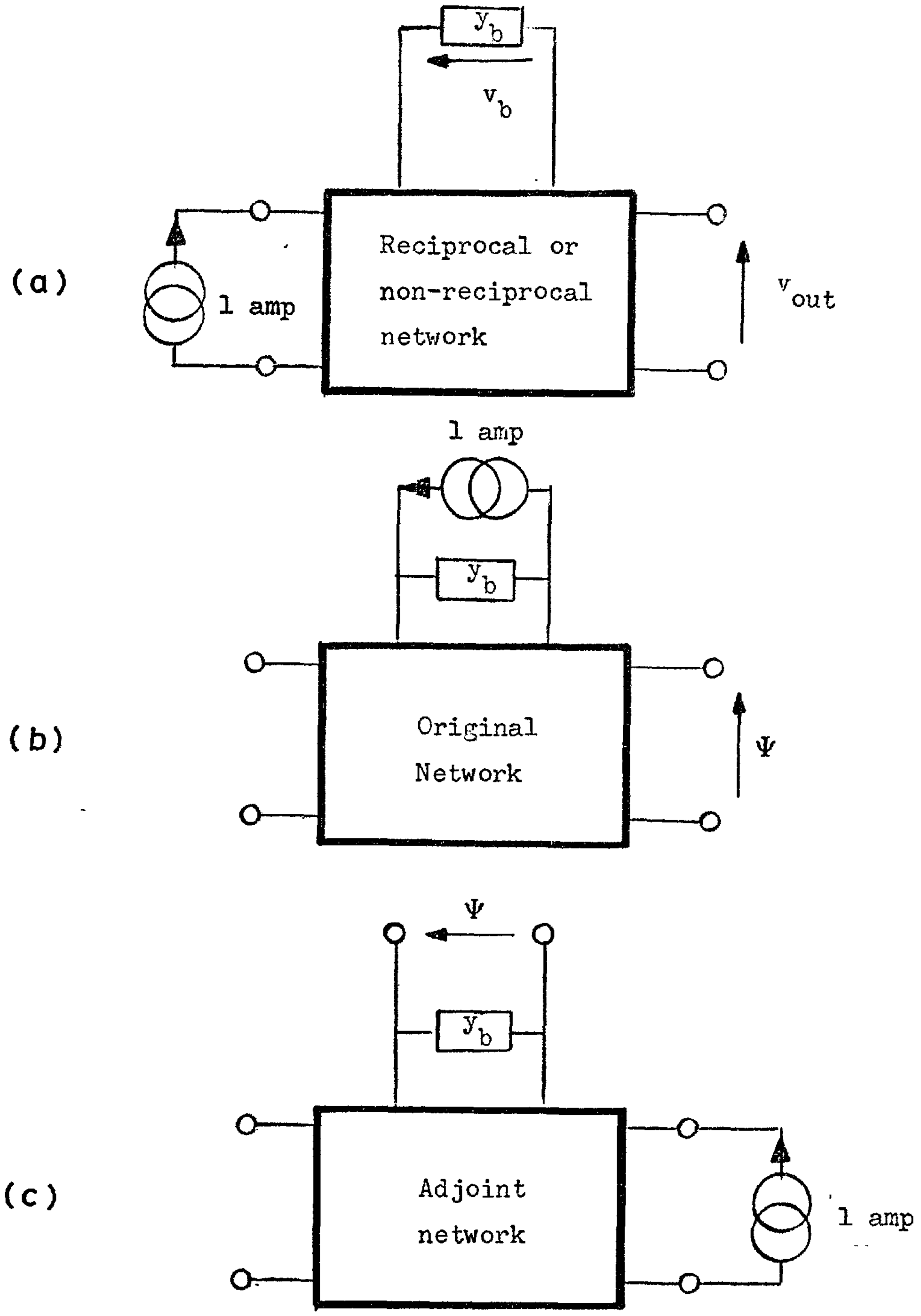
The other uses of small change sensitivity include optimisation and worst case simulation etc. The problem of the use of small change sensitivity for the worst case simulation of a circuit will be treated in detail in the following chapters.



y_b
v_b
(a)
1 amp
Reciprocal network
v_{out}
1 amp
y_b
(b)
Auxiliary network (same as original network)
Ψ
Ψ
y_b
(c)
Auxiliary network (same as original network)
1 amp

$$\frac{\partial v_{out}}{\partial y_b} = - v_b \Psi$$

Fig. 2.1. Computation of small change sensitivity of a reciprocal network using auxiliary network technique.



$$\frac{\partial v_{out}}{\partial y_b} = - v_b \Psi$$

Fig. 2.2. Computation of small change sensitivity of a non-reciprocal network using adjoint network technique.

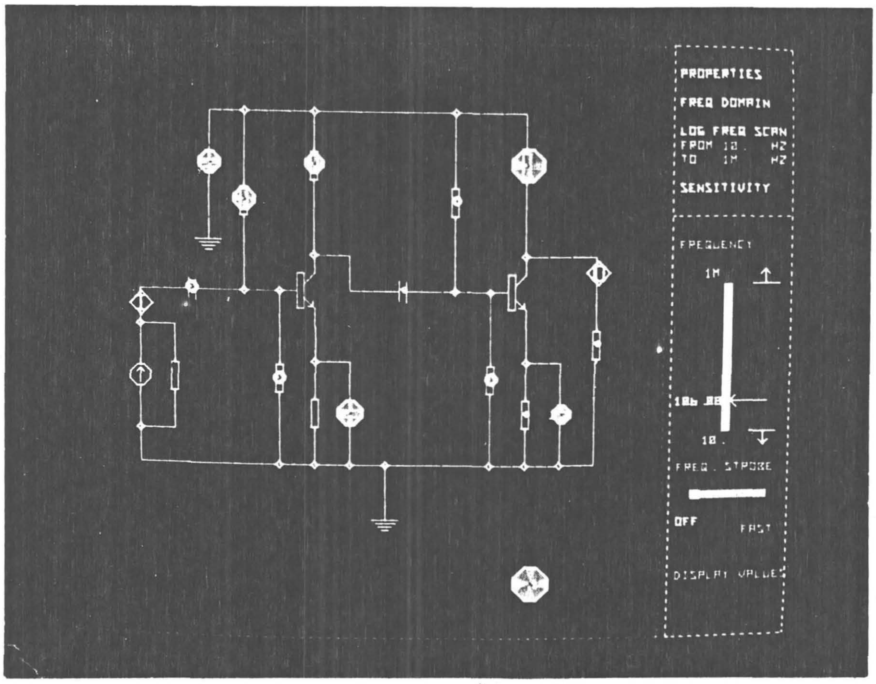


Fig. 2.3. Graphical display of small change sensitivity information. Circles having radii proportional to the magnitude of sensitivity are superimposed on the symbols of the corresponding components. The running parameter - frequency - is indicated by the pointer on the light potentiometer.

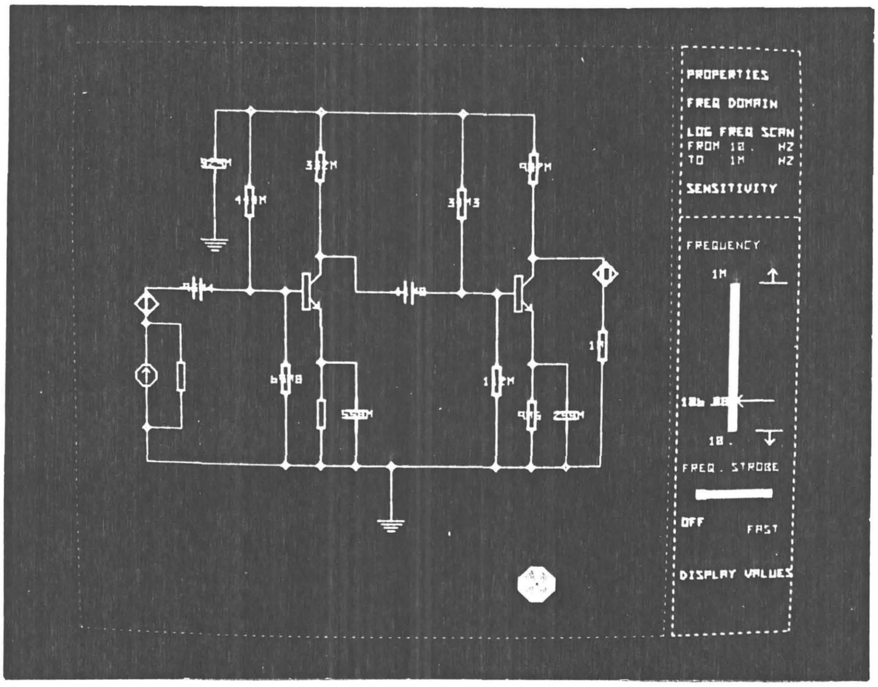


Fig. 2.4. Alternative means of representing sensitivity information. The circles on Fig. 2.3 are substituted by numerical values.

CHAPTER 3

SINGLE PARAMETER LARGE CHANGE SENSITIVITY

3.1 Introduction

In the previous chapter, the efficient means for the computation of small change network sensitivity are described. The effect of a small change in the value of circuit components upon an output voltage has been shown to be amenable to simple description and its computation has been successfully implemented for a number of applications, [8,21,28] . However, in many cases, the changes in component values of a circuit cannot be regarded as infinitesimal. Therefore, efficient means are required to predict the effect of large element value changes. In response to this need, the problem of computation of large change sensitivity came to our attention.

The principal relation of interest in a large change sensitivity analysis is that which exists between a change in the value of a component parameter and the corresponding change in circuit response, where the changes are not necessarily infinitesimally small. Two aspects of this relation are important: either a component change is given and the effect upon circuit response has to be computed, or the tolerance upon response is specified and the permitted change in component value is sought.

A circuit model which is adequate for our initial discussion is shown in Fig. 3.1. A linear 2-port is excited at one port by a unit current source and generates at port 2 a voltage response equal in value to the transfer impedance z_{21} (Fig. 3.1(a)). For simplicity, only one linear 2 or multi-terminal component is assumed to be subject to change. It is convenient to extract from the circuit the change Δy in the admittance of the 2-terminal component, thereby creating a new port [22] , denoted as port 3, in addition to those associated with the excitation and response as shown in Fig. 3.1(b). In Fig. 3.1(c) the change Δi_{cs} of a voltage controlled current source, as well as its controlling ports, are extracted from the 2-port, thereby creating two new ports (port 3 and port 4) of the nominal value circuit N.

The change Δv in voltage response due to the change Δy in component admittance can, of course, always be determined by a fresh analysis of the entire circuit. But such an approach is prohibitively expensive, especially if the circuit is of reasonable size and if a number of values of component change are to be explored. Even if pivotal condensation is employed to obtain the admittance matrix of the 3-port (Fig. 3.1(b)), the additional effort involved in calculating the effect of connecting Δy is quite excessive.

Fortunately, two simple alternative methods for computing Δv are available. In one, called the 'current source substitution method' [23] , the change in admittance Δy is simulated by a current source whose value is the current in Δy. The response to this current at port 2 is then the required value of Δv. In the other, the impedance description of the circuit is permanently modified to take account of the new component value. We give the name 'matrix modification' to this latter method.

In this chapter, we examine these two methods to assess their suitability both for the calculation of response when a component change is known, and for the inverse problem of finding the component change permitted by a specified change in response i.e. for both forward and reverse tolerance analysis.

Exposition of the theoretical basis of the two methods of large-change sensitivity analysis is considerably eased if a port description of the circuit is employed. Such a description will also be adopted in the description of the associated algorithm since, in addition to clarity of discussion, it allows the principal computation effort easily to be identified.

The total cost of a computer calculation is virtually impossible to predict in detail, and we follow the common practice of assigning a cost of one to multiplication and division, and zero to addition and subtraction. Also, since many of the calculations involved are either a Gaussian elimination, or approximate very closely in cost to that of an integer multiple of Gaussian eliminations we shall denote the computational cost of such an elimination by G_x, where the subscript x indicates the dimension of the vectors involved.

3.2 The Substitution Theorem

Consider an arbitrary network which contains a number of independent sources. Suppose that, for these sources and for the initial conditions given, the network has an unique solution for all its branch voltages and branch currents. Consider a branch, say branch α. Let i_α and v_α be the current and voltage of branch α. Suppose that this branch is replaced by either an independent current source with current i_α, or an independent voltage source with voltage

v_α. If the modified network has an unique solution for all its branch currents and branch voltages, then these branch currents and branch voltages are identical with those of the original network.

The above theorem, also known as the 'substitution theorem' [24] , allows us to replace any particular branch of a network by a suitably chosen independent source without changing any branch current or branch voltage. It applies to any network, linear or non-linear, time-varying or time-invariant. In the following section it will be seen how it could be employed to derive an efficient method to simulate large change in circuit component values.

3.3 Current Source Substitution Method

The first of two large-change sensitivity algorithms to be examined can be explained with reference to the circuit model of Fig. 3.1(a). The conceptual basis of the current source substitution method [23] is the replacement of the admittance Δy, or the voltage controlled current source Δi_{cs}, by a current source whose value is identical with the current i_3, initially flowing in Δy (Fig. 3.2(a)) or Δi_{cs} (Fig. 3.2(b)), respectively. According to the Substitution Theorem described in the previous section, no disturbance is thereby occasioned to the voltages and currents in the circuit. With reference to Fig. 3.1(b) and (c), since the only change from the nominal circuit N is the connection of the current source i_3 to port 3, and the circuit is linear, the voltage response at port 2 due to this source acting alone is the required change Δv in output voltage. The presence of this additional current source in N is accommodated in the port matrix equation by the addition of i_3 to the 3rd element of the current vector, so that the port equation has the form

$$V_p' = Z_p I_p' \tag{3.1}$$

The above equation can be written in more detail as

$$\begin{bmatrix} v_1' \\ v_2' \\ v_3' \end{bmatrix} = \begin{bmatrix} z_{11} & z_{12} & z_{13} \\ z_{21} & z_{22} & z_{23} \\ z_{31} & z_{32} & z_{33} \end{bmatrix} \begin{bmatrix} i_1 \\ 0 \\ i_3 \end{bmatrix} \tag{3.2}$$

for the two terminal components, or as

$$\begin{bmatrix} v_1' \\ v_2' \\ v_3' \\ v_4' \end{bmatrix} = \begin{bmatrix} z_{11} & z_{12} & z_{13} & z_{14} \\ z_{21} & z_{22} & z_{23} & z_{24} \\ z_{31} & z_{32} & z_{33} & z_{34} \\ z_{41} & z_{42} & z_{43} & z_{44} \end{bmatrix} \begin{bmatrix} i_1 \\ 0 \\ i_3 \\ 0 \end{bmatrix} \tag{3.3}$$

for the multi-terminal components (Fig. 3.1(c)). Thus the new port voltages can now be computed by the matrix multiplication of the originally computed Z_p matrix with the new column vector I_p' of the current sources. The column vector V_p' yields not only the new value v_2' of the response voltage, but also the voltage v_3' across the simulated admittance Δy or Δi_{cs}. The latter voltage v_3' can be expressed in terms of the original circuit excitation and the added current source i_3 by

$$v_3' = z_{31}\, i_1 - z_{33}\, i_3 \tag{3.4}$$

Thus, for a given circuit and excitation, v_3' varies only with i_3, the added current source simulating the change in y or g_m of the original circuit. For the multi-terminal components, the voltage v_4' is found in the same way as

$$v_4' = z_{41}\, i_1 - z_{43}\, i_3 \tag{3.5}$$

The value of the simulated admittance y' is given by

$$y' = \frac{y\, v_3' + i_3}{v_3'} \tag{3.6}$$

and for the new value of the mutual admittance g_m', we found

$$g_m' = \frac{g_m\, v_4' + i_3}{v_4'} \tag{3.7}$$

Substitution for v_3' from (3.4) into (3.6) allows the value of i_3 required in the simulation of the admittance y' to be expressed as

$$i_3 = \frac{z_{31}}{z_{33} + \frac{1}{\Delta y}} \tag{3.8}$$

Knowledge of the new value of the admittance y', therefore, allows the current i_3 which is simulating the admittance change $y' - y$ to be computed from equation 3.8.

The value of i_3 required in the simulation of the new mutual admittance g_m is given by

$$i_3 = \frac{z_{41}}{z_{43} + \frac{1}{\Delta g_m}} \tag{3.9}$$

The new value of the response voltage v_2' can be computed from (3.2) by simple matrix multiplication, without the need to formulate and invert a matrix describing the new circuit configuration. It may be preferable, in fact, to compute the change in response voltage directly from

$$\Delta v_2 = z_{23}\, i_3 \tag{3.10}$$

or

$$\Delta v_2 = \frac{z_{23}\, z_{31}}{z_{33} + \frac{1}{\Delta y}} \tag{3.11}$$

for the two terminal components, and

$$\Delta v_2 = z_{23}\, i_3 \tag{3.12}$$

or

$$\Delta v_2 = \frac{z_{23}\, z_{41}}{z_{43} + \frac{1}{\Delta g_m}} \tag{3.13}$$

for the multi-terminal components.

In addition, equation (3.11) and (3.13) illustrates the possibility of applying the large-change sensitivity 'in reverse'. For example, if a certain amount of deviation from the nominal is specified for the response voltage, the corresponding allowed change in a single element can be evaluated.

3.4 Matrix Modification Method

The basis of the method to which we have assigned the name 'matrix modification' is the fact that, if a matrix F is modified by the addition of the triple matrix product GHK, then the inverse of the new matrix is related to that of F by the expression [25,56]

$$(F + GHK)^{-1} = F^{-1} - F^{-1}G\,(H^{-1} + KF^{-1}G)^{-1}\,KF^{-1} \tag{3.14}$$

provided the indicated inverses exist and the dimensions are properly matched. Thus F and H are square matrices and G and K may be rectangular. The implication of this relation for our discussion is easily appreciated. With reference to Fig. 3.1(b) or (c), let Y_p be the nominal port admittance matrix of the circuit N (i.e., without Δy connected) and Z_p its inverse. The connection of an admittance Δy to port 3 is described by the addition of a matrix ΔY_p to Y_p,

$$Y_p' = Y_p + \Delta Y_p \tag{3.15}$$

where ΔY_p has the form

$$\Delta Y_p = \begin{array}{cc} \left[\begin{array}{ccc} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \Delta y \end{array}\right] & 3 \\ \begin{array}{ccc} & & 3 \end{array} & \end{array} \tag{3.16}$$

where only nonzero entries are shown explicitly. Thus

$$\Delta Y_p = P\,\Delta y\,Q^t \tag{3.17}$$

where P and Q are column vectors whose only nonzero elements are $P_3 = Q_3 = 1$, and t denotes transposition. Reference to (3.14) enables the new port impedance matrix Z_p' to be expressed in terms of its initial value Z_p by

$$Z_p' = Z_p - \frac{1}{\gamma} Z_p P Q^t Z_p \tag{3.18}$$

where

$$\gamma^{-1} = \frac{1}{\Delta y} + Q^t Z_p P \tag{3.19}$$

$$\left[Z_p P\right]^t = \left[z_{13}, z_{23}, z_{33}\right] \tag{3.20}$$

$$Q^t Z_p = \left[z_{31}, z_{32}, z_{33}\right] \tag{3.21}$$

For the simple case where only one component is subject to variation, equation 3.18 leads to an expression for the new value of the transfer impedance z_{21}

$$z_{21}' = z_{21} - \frac{z_{23}z_{31}}{z_{33} + \frac{1}{\Delta y}} \tag{3.22}$$

The preceding interpretation of the matrix modification procedure shows how it may be modified to include multi-terminal components. Since the discussion presumes the existence of both the port impedance and admittance matrices, multi-terminal components may always be represented in terms of voltage controlled current sources. Consider a port admittance matrix Y_p and assume that a dependent source is added to the network in which the current i_3 in a branch across port 3 is controlled by a voltage v_4 (Fig. 3.1(c)). The modified port admittance matrix has the form of (3.16) with

$$\Delta Y_p = \begin{array}{c c} \left[\begin{array}{cccc} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \Delta g_m & \cdot \end{array}\right] & \begin{array}{c} \\ \\ \\ 4 \end{array} \\ \begin{array}{cccc} & & 3 & \end{array} & \end{array} \tag{3.23}$$

where again only nonzero entries are shown explicitly. Hence

$$\Delta Y_p = P\, g_m\, Q^t \tag{3.24}$$

where P and Q are column matrix with zero entries everywhere except that $P_4 = Q_3 = 1$. This permits the modified port impedance matrix Z_p to be written as

$$Z_p' = Z_p - \frac{1}{\gamma} Z_p\, P\, Q^t\, Z_p \tag{3.25}$$

with

$$\gamma^{-1} = \frac{1}{\Delta g_m} + Q^t\, Z_p\, P \tag{3.26}$$

$$\left[Z_p\, P\right]^t = \left[z_{13},\, z_{23},\, z_{43}\right] \tag{3.27}$$

$$Q^t \cdot Z_p = \left[z_{41},\, z_{42},\, z_{43}\right] \tag{3.28}$$

For the simple case where only one g_m is subject to variation, equation 3.25 leads to an expression for the new value of the transfer impedance z_{21}

$$z_{21}' = z_{21} - \frac{z_{23}\, z_{41}}{z_{43} + \frac{1}{\Delta g_m}} \tag{3.29}$$

It is interesting to note that (3.22) and (3.29) are <u>in agreement</u> with (3.11) and (3.13) derived from consideration of the current source substitution method. Thus, for computing the effect of changes in a single component there is little justification for differentiation between the two methods or their associated (equal) costs. The same conclusion applies to the inverse problem of computing Δy if Δv is specified.

Nevertheless, there is an important conceptual distinction between the basis of the two methods. In the matrix modification method the admittance change can be permanently absorbed into the circuit (Fig. 3.3(a),(b)), and the result is one or all of the port parameters describing the new circuit. By contrast, what the substitution current source method provides is a source which, when applied to the nominal circuit, causes the same change in circuit response as does the admittance change Δy. Here the new circuit description is not directly to hand: the effect of (say) a further excitation current could not be deduced simply by connecting it and the substitution source simultaneously to the nominal circuit.

3.5 Computational cost

The advantage of the method of large-change sensitivity analysis described above is easy to establish. Without it, determination of the effect of large parameter changes could involve an analysis of the nominal circuit followed by as many similar analyses as there are parameter changes. If we follow the practice of assigning a cost of one to multiplication and division, and zero to addition and subtraction then, for each new component value, the cost of computing Δv_2 is two, in addition to the cost of one incurred in the initial calculation of the product $z_{23}z_{31}$ or $z_{23}z_{41}$. The latter cost quickly becomes insignificant if more than a few values of component change are explored.

Base on the efficient methods described above, two schemes for the computation of the effect of large changes in component value are proposed. Fig. 3.4 outlines these two schemes and itemizes the associated computational costs. If only one parameter is subject to change, then, at each frequency, pivotal condensation of the circuit's nodal admittance matrix Y_n to obtain the 3 x 3 port* admittance matrix Y_p, followed by matrix inversion to yield the port impedance matrix Z_p, provides the impedances appearing in (3.11) or (3.22), from which Δv_2 is computed. This approach is indicated by a solid line in Fig. 3.4. The alternative approach indicated by a dashed line may be more economical if it is not known, beforehand, which parameter is liable to change.

The computational costs of the schemes of Fig. 3.4 are compared in Table 3.1 with two straightforward methods. One (column A) refers to a repeated nodal analysis for each new parameter value. The other (column B) employs an initial pivotal condensation to obtain the port description of N (Fig. 3.1(b) or (c)) and then, for each parameter change, a Gaussian elimination (Fig. 3.4 Note 1)): this method, whether used in connection with one or more variable components, is referred to throughout this and the next chapter as the reference method. The three rows of the table indicate the costs incurred once, K times and KL times if the effect of L different values of each of K separate components is to be computed: in each case, a single variable parameter is assumed to return to

* For simplicity, the situation in which ports share one or more nodes are not considered here. Also, in assessing computational cost, we do not exploit any symmetry of the matrices involved.

its nominal value before another parameter is varied*. Progression to the right over the table shows an increasing "once only" or "K only" cost, and a decreasing "per component change" cost, a situation which is necessary if the effect of many large changes in one or more components is to be explored economically. Clearly, on the basis of the costs shown, only one change in one parameter is sufficient for the new methods of large-change sensitivity analysis (columns 3,4) to be substantially more efficient than a repeated analysis of the entire circuit. Choice of one of the two schemes of Fig. 3.4 will depend not only upon a priori knowledge of the number of parameters likely to be varied but also on other considerations such as storage and the means by which the designer interacts with the program.

To illustrate the savings that can be incurred by using the new methods, Fig. 3.5 shows some actual costs predicted from Table 3.1.

3.6 Engineering Applications

An advantage of the efficient approach to large-change sensitivity analysis has been exploited to good effect in an interactive graphic circuit design facility [8] . Identification of a variable component in a linear circuit causes the appropriate impedances (i.e., those appearing in (3.11 or 3.22)) to be generated.

* In the case of the current source,substitution method, if a variable parameter is held at a new value, the admittance matrix Y_n must, of course, be reformulated before the effect of a new variable parameter can be computed. For the matrix modification method, the new value of the variable parameter can be absorbed into the circuit by modifying three additional elements of the matrix Z_p.

The calculation of Δv_2 is then so simple that it is possible for the designer to move his light-pen continuously over a light-potentiometer (Fig. 3.6, right) to indicate component variation and, within about half a second*, to see on the screen the new response curve (Fig. 3.6). Such a rapid display of the effect of component variation allows the designer to gain insight into the significance of that component by a process of dynamic exploration rather than repeated trial-and-error.

* This figure is related to the number of points characterizing the response curve. The circuit of this example is illustrated in Fig. 1.9. The variable component is R11.

	A Repeated nodal analysis of complete circuit	B Reference method (Fig.3.4 Note 1)	C Scheme of Fig. 3.4	D Scheme of Fig. 3.4 (dashed line)
Once-only initial calculation	G_n (to obtain nominal response)	12 (to obtain nominal response)	-	$3G_n$
Preparatory calculation repeated K times	-	G_n-G_6	$G_n-G_6+3G_3+1$	1
Calculation for each component change	G_n	12	2	2

Table 3.1

Approximate costs associated with four different methods of computing the effect, on a circuit's transfer impedance, of single component changes

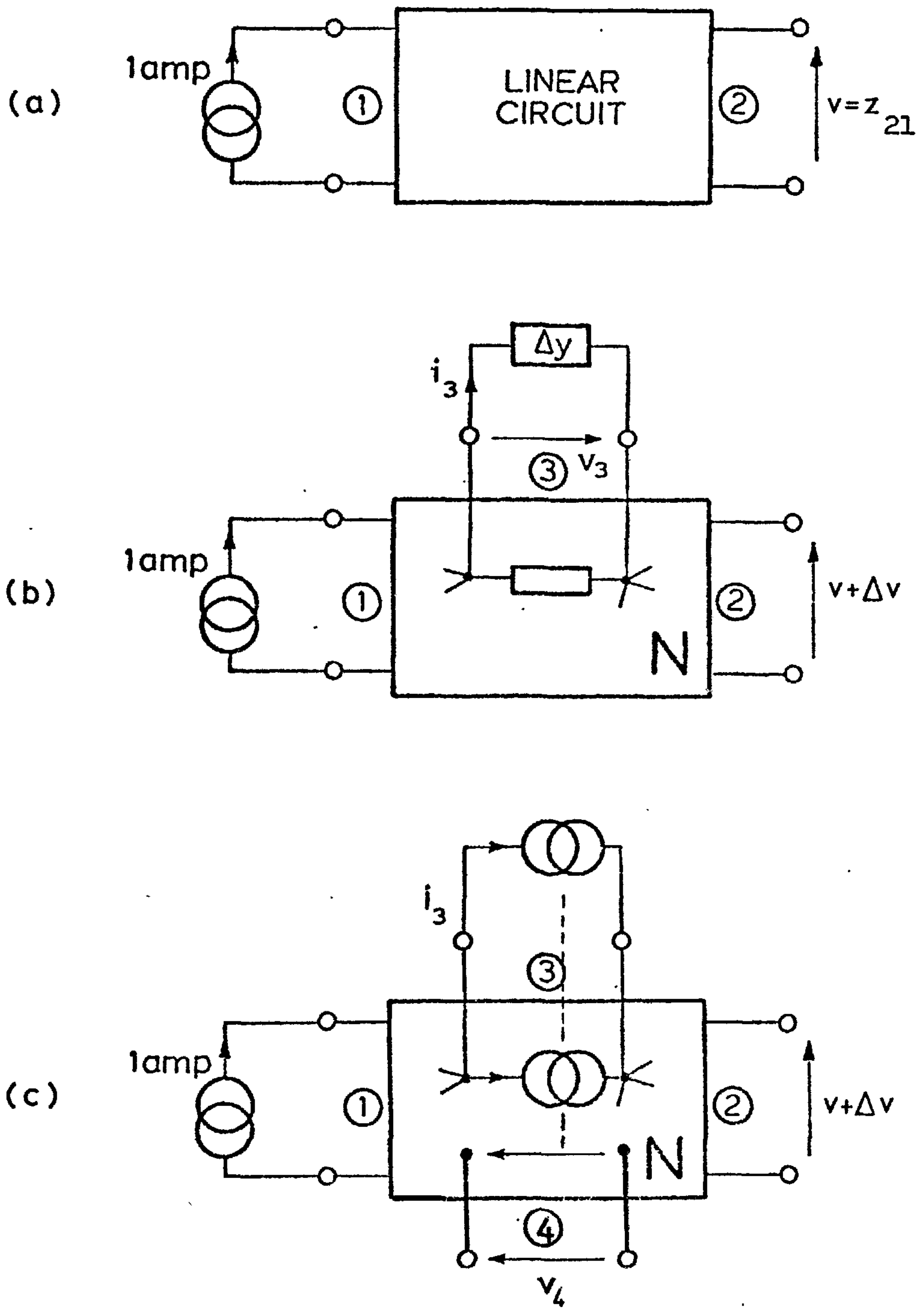
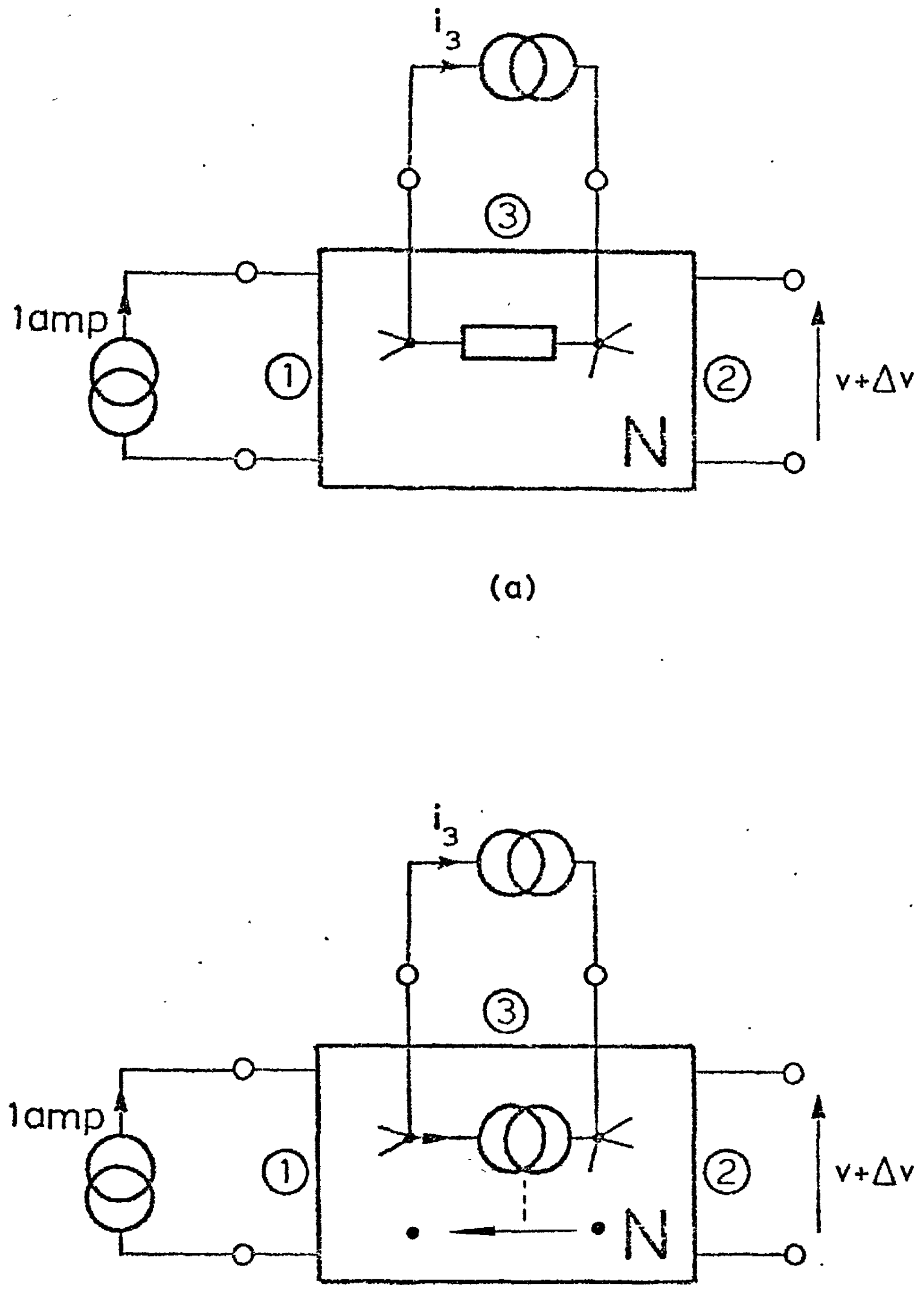


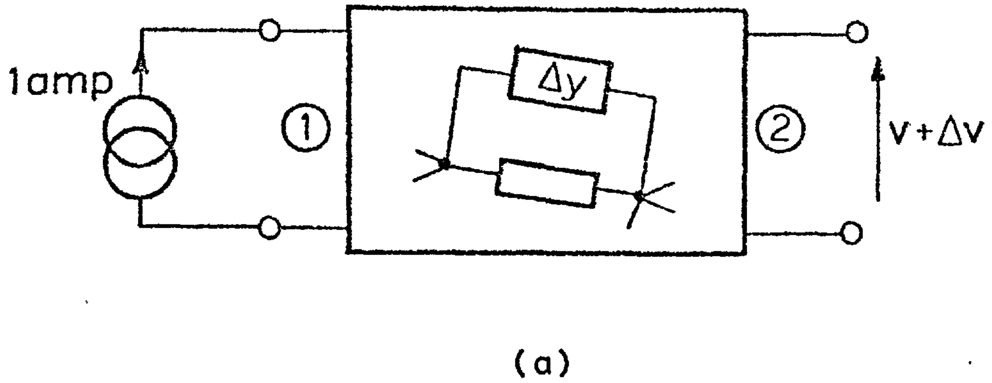
Fig. 3.1. Linear 2-port, one of whose 2-terminal or multi-terminal components is liable to change.



(a)

(b)

Fig. 3.2. Simulation of component change by current source.



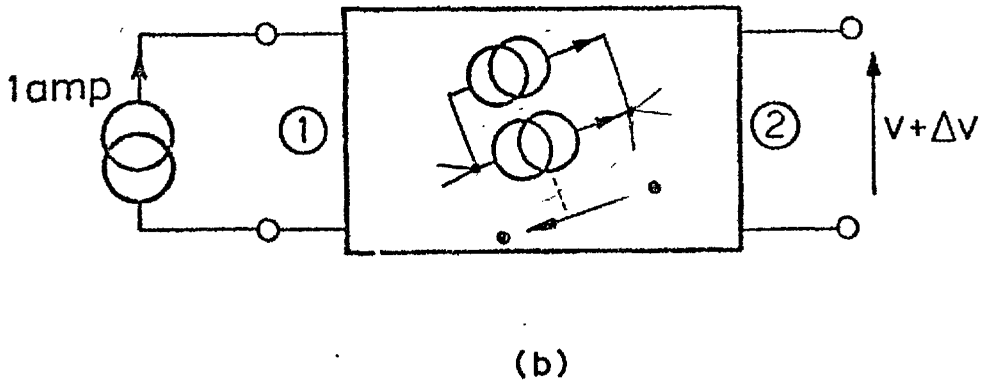


Fig. 3.3 Absorption of component change by matrix-modification method.

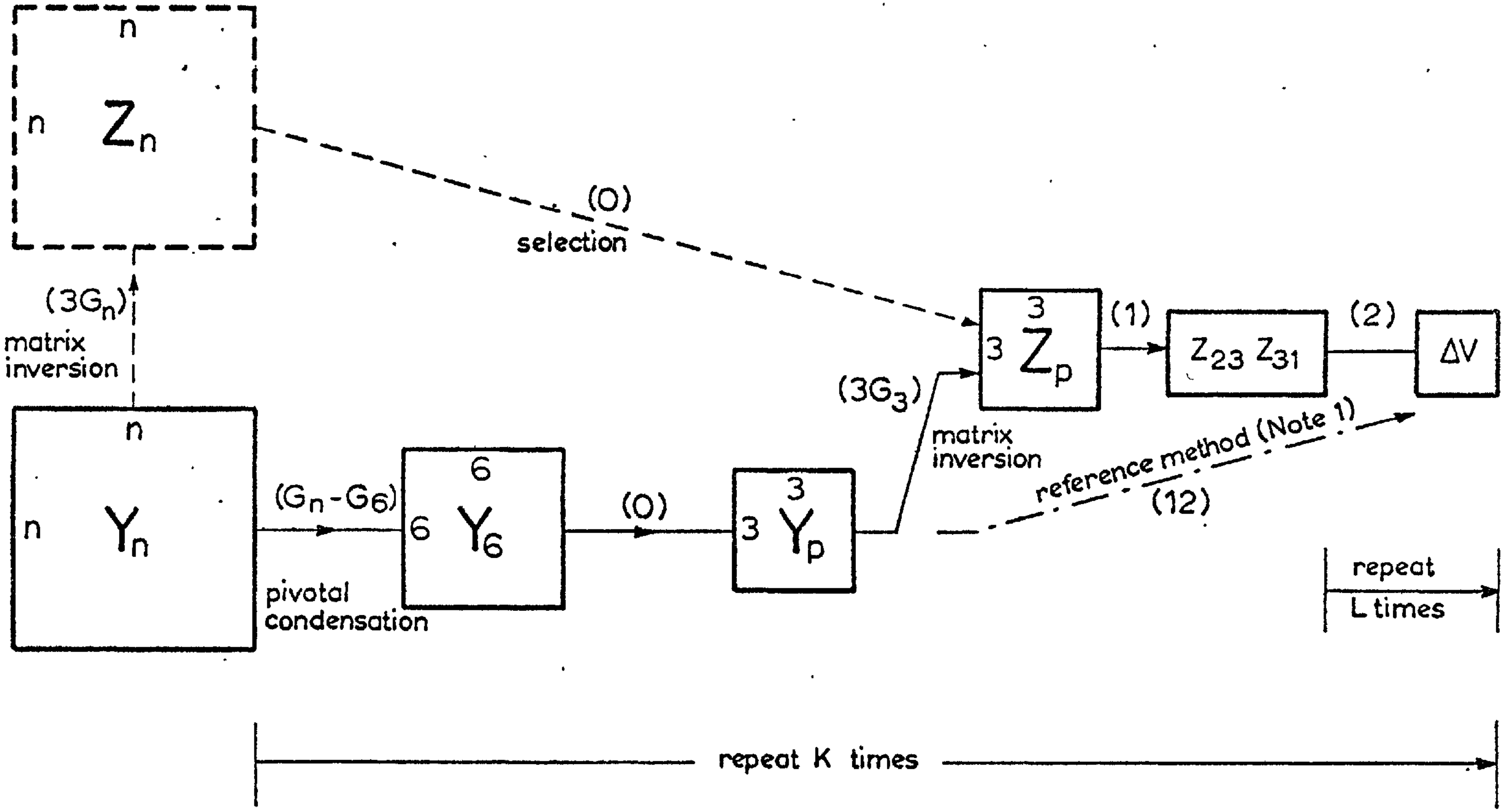


Fig. 3.4. Approximate cost of computing the effect of changes in one component on the transfer impedance of an (n + 1)-terminal circuit. Values in brackets indicate costs. Note 1 - this path is associated with a straightforward analysis of a 3-port by Gaussian elimination. Note 2 - arrows indicating repeat calculations associated with L different values of the same component, and with K different components, do not apply to the reference method.

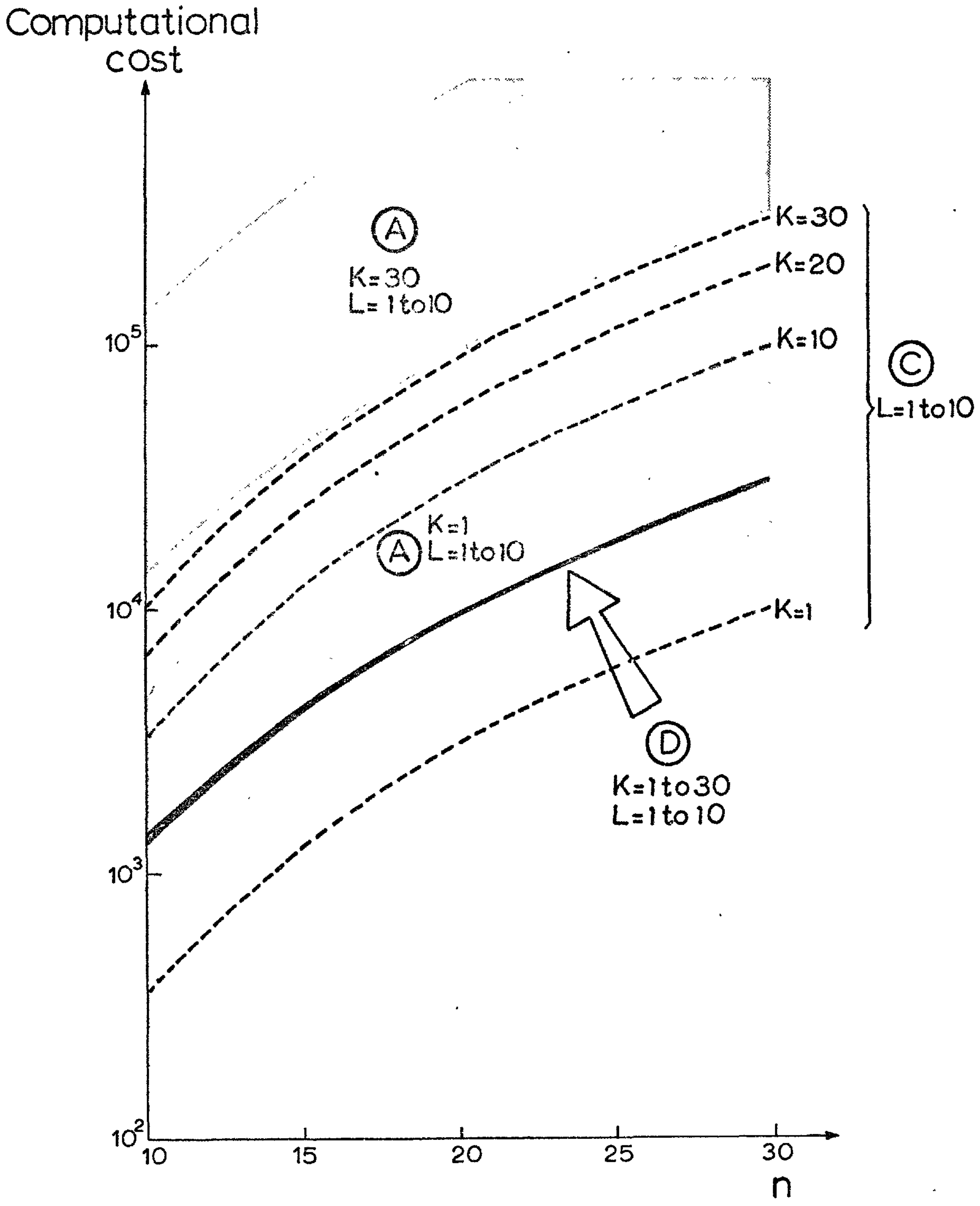


Fig. 3.5. Plots illustrating some actual computational costs predicted from Table 3.1.

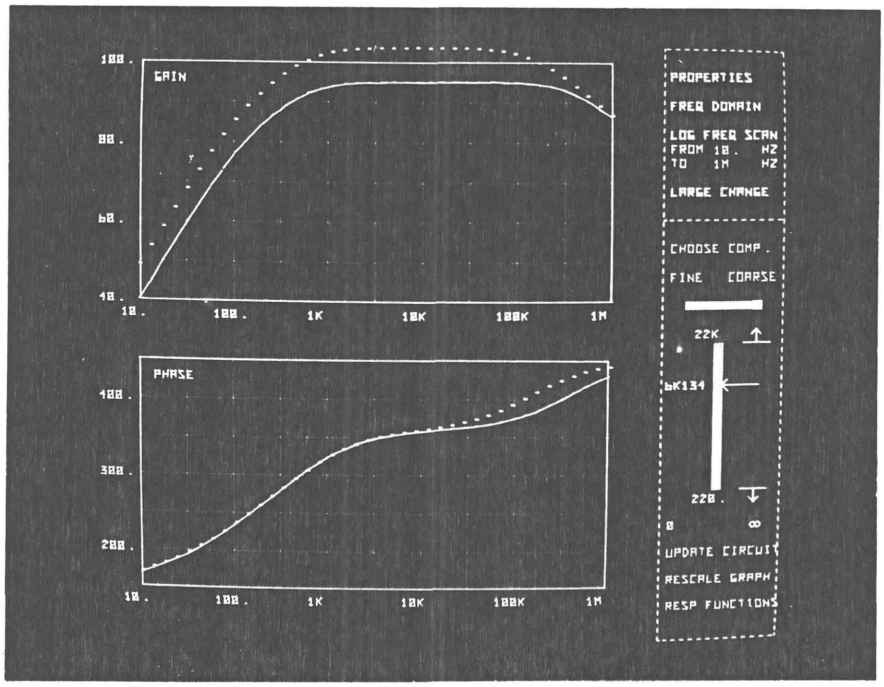


Fig. 3.6(a). Interactive display associated with dynamic exploration of the effect of a component on circuit response. Component change is simulated by moving an activated light-pen along the light-potentiometer shown at right. The new (dotted) response curve is displayed on the same graph as the (full) nominal curve.

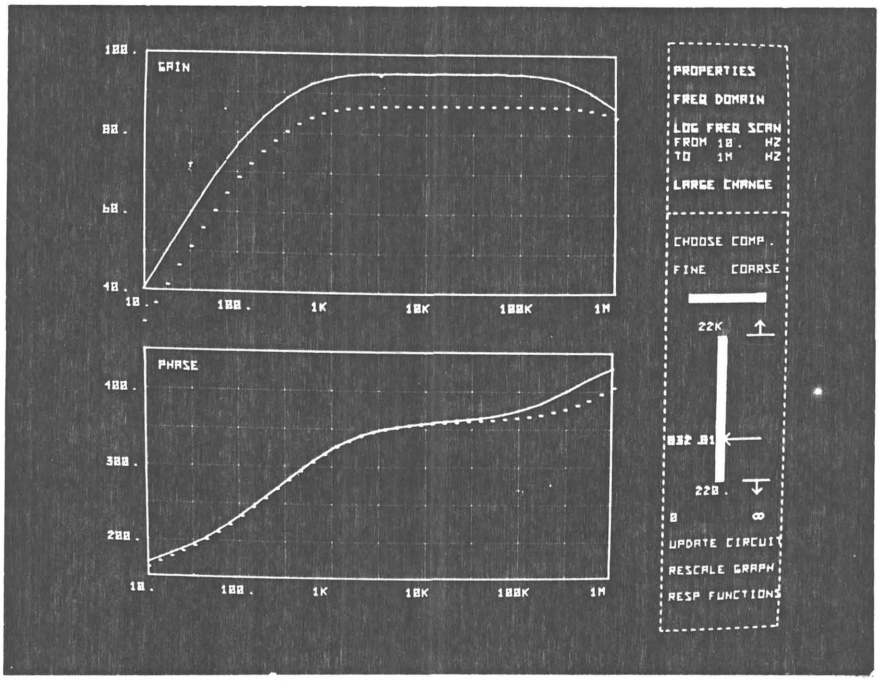


Fig. 3.6(b). Another new response curve is displayed in less than a second after the component has been adjusted to a new value.

CHAPTER 4

MULTI-PARAMETER LARGE-CHANGE SENSITIVITY

4.1 Introduction

The effect of simultaneous changes in more than one component parameter is frequently of interest. The circuit designer may, for example, be adjusting a group of parameters with the object of achieving a desired circuit response. Or the changes may be automatic, as in a Monte Carlo statistical circuit analysis or an optimization algorithm. The need to examine the consequence of change in the physical properties (e.g., temperature or doping level) of a circuit also involves the exploration of simultaneous parameter changes. A final example, and one which will illustrate many of the points discussed in this chapter, is the calculation of performance contours [1] ; for a given tolerance on circuit response, a performance contour describes the permissible simultaneous variation in two or more component parameters, and provides valuable information for the circuit designer.

It is clear that there is a need to examine alternative efficient algorithms for calculating the effect of large change in a number (more than one) of component parameters, and to compare their cost with that of repeated straightforward circuit analysis. Such an examination is the purpose of this chapter. An important conclusion - that systematic exploration of the effect of component change is economically potentially attractive - forms the subject of many illustrative examples which will be described in the following chapters. Finally, the implementation of multiparameter large-change sensitivity analysis within an interactive graphic computer-aided circuit design facility will be illustrated with an example.

4.2 Simultaneous Component Changes

Consider now the situation in which m components in a circuit are subject to simultaneous change. The corresponding changes in admittance at a given frequency are denoted by $\Delta y_1, \ldots\ldots, \Delta y_m$. It is again convenient to extract these admittance changes from the circuit, thereby creating m new ports numbered 1,.....,m (Fig. 4.1(a)) in addition to those denoted by ϕ and ψ and associated, respectively with excitation and response. The effect upon the transfer impedance $z_{\psi\phi}$ of this set of simultaneous component admittance changes is to be computed.

One very straightforward approach, of course, is to update the (m + 2)-port's admittance matrix (at no cost) to account for the component admittance changes, and then carry out a Gaussian elimination to find the new transfer impedance. The cost for each set of m component changes is G_{m+2}, as is the initial cost of finding the circuit's nominal transfer impedance.

It is against this reference that we now examine the substitution current source and matrix modification methods. In the latter case, however, we shall broaden our discussion to consider the sequential connection of the members of a set of admittance changes, for this will later be seen to offer the possibility of greater computational economy.

4.3 Substitution Current Source Method

The apparent simplicity of this method diminishes rapidly as the number of simultaneously variable parameters increases. Basically, this is because the admittance simulated by a current source is the ratio of its current and voltage: whereas its current is defined, its voltage is a function of all the sources (including those simulating component changes) connected to the circuit. It followed that if m component changes are simulated by m current sources, then the voltage across each source, and hence the admittance simulated, is a function of the other substitution current sources as well as the circuit's independent excitation. Thus, broadly speaking, the values of the m substitution current sources can be found only by the solution of m equations in m unknowns [27,28,29] . We shall now proceed to establish the computational cost with more precision.

Let the (m + 2)-port circuit N of Fig. 4.1(a) be described by a port admittance matrix Y_p, and let the changes $\Delta y_1,\ldots\ldots,\Delta y_m$ in the m component admittances be simulated, respectively, by m current sources $i_1,\ldots\ldots,i_m$ (Fig. 4.1(b)). Such a simulation requires that

$$\begin{bmatrix} v_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ v_m \end{bmatrix} = \begin{bmatrix} -\frac{1}{\Delta y_1} & & & 0 \\ & -\frac{1}{\Delta y_2} & & \\ & & \ddots & \\ 0 & & & -\frac{1}{\Delta y_m} \end{bmatrix} \begin{bmatrix} i_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ i_m \end{bmatrix} \qquad (4.1)$$

Denote the nominal circuit's port impedance matrix by Z_p, and that partition of Z_p appropriate to the m newly created ports by Z_m.

When the circuit is excited by the substitution current sources as well as the original excitation at port ∅, the voltages generated across the substitution sources are given by

$$\begin{bmatrix} v_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ v_m \end{bmatrix} = \begin{bmatrix} & & \\ & Z_m & \\ & & \end{bmatrix} \begin{bmatrix} i_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ i_m \end{bmatrix} + \begin{bmatrix} z_{1\emptyset} \\ \\ z_{2\emptyset} \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ z_{m\emptyset} \end{bmatrix} I_{\emptyset} \qquad (4.2)$$

Combination with (4.1) yields

$$- \begin{bmatrix} z_{1\emptyset} \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ z_{m\emptyset} \end{bmatrix} I_{\emptyset} = \left\{ \begin{bmatrix} \frac{1}{\Delta y_1} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{\Delta y_m} \end{bmatrix} + \begin{bmatrix} Z_m \end{bmatrix} \right\} \begin{bmatrix} i_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ i_m \end{bmatrix} \triangleq Z' I_S \qquad (4.3)$$

where I_s is the vector of substitution current sources. Thus, once Z_m has been found (by inversion of Y_p) at a cost of about $3G_{m+2}$, the values of the m substitution current sources can be found in two steps: 1. calculation of the reciprocal of each admittance change at a total cost of m, and 2. Gaussian elimination at a cost of G_m. The matrix product $Z_m I_s$ then yields the change in port voltages. If only $\Delta v_{\emptyset}$ is of interest, the cost is m. This approach is illustrated in Fig. 4.2, where it is assumed that the starting point is the nominal circuit's nodal admittance matrix Y_n. If there are L sets of values of the same m components, the calculation of I_s at a cost of G_m + m, and of $\Delta v_{\emptyset}$ at a cost of m, must be repeated L times. If K different groups of components are known beforehand to be of interest - as may well be the case in truly interactive design - then a more economical alternative may be that which is indicated by the dashed lines in Fig. 4.2.

The computational costs are itemized in the first two columns of Table 4.1: The second column is the more appropriate if the variable parameters are initially unidentified or if K is large. The final column refers, for purposes of comparison, to a straightforward method which is identical in form to that employed as a reference in the previous chapter (column B of Table 3.1); it involves updating the (m + 2)-port's admittance matrix (at no cost) to account for component admittance changes, followed by Gaussian elimination at a cost of G_{m+2} to find the transfer impedance (see Fig. 4.2) Table 4.1 shows that no very clear advantage can be associated with the substitution current source approach.

4.4 Matrix Modification Method

Consider a 2-port in which three 2-terminal components are subject to simultaneous change. New ports are created in parallel with these components (Fig. 4.3) and the relevant admittance changes Δy_1, Δy_2, and Δy_3 simulated by the connection of identical admittances externally to these ports. First, however, we consider the connection of Δy_3 alone*, so that ports 1 and 2 are left on open circuit.

* A single direct application of the Householder relation (3.14) to simultaneously account for all component changes is far more costly than the straightforward method used as a reference in Section 4.3 (see Table 4.1, column 3). Therefore, no further reference to the "once only" application of (3.14) will be made.

With reference to (3.17), this connection is described if P and Q are the vector* $(00001)^t$ and $\Delta y = \Delta y_3$. Equation (3.18) then becomes

$$Z_p' = Z_p - \gamma^{-1} Z_p \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 00001 \end{bmatrix} Z_p \tag{4.4}$$

where, from (3.19),

$$\gamma^{-1} = z_{33} + \frac{1}{\Delta y_3} \tag{4.5}$$

(note that z_{33} lies on the fifth row and column of Z_p). A cost of unity is associated with the calculation of γ . The matrix product $Z_p\ P\ Q^t\ Z_p$ on the right of (4.4) is

$$\begin{bmatrix} z_{\phi 3} \\ z_{\psi 3} \\ z_{13} \\ z_{23} \\ z_{33} \end{bmatrix} \begin{bmatrix} z_{3\phi} z_{3\psi} z_{31} z_{32} z_{33} \end{bmatrix} \tag{4.6}$$

Examination of (4.4) shows that if division by γ precedes calculation of the product (4.6) then determination of the new port impedance matrix Z_p' involves a total computational cost of $1 + 5 + 5^2 = 31$. The matrix so obtained describes the result of absorbing the

* The first two elements of P and Q refer to the input and output ports designated ϕ and ψ , respectively. The remaining elements are numbered 1, 2, 3.

admittance Δy_3 within the original circuit and leaving port 3 on open circuit. In the general case of m variable components, the calculation of Z_p' to account for the connection of one component* in the set of m is therefore $1 + (m + 2) + (m + 2)^2$. Starting with the matrix Z_p', the process just described is repeated in turn for the remaining externally connected admittances Δy_1 and Δy_2. The cost for each is the same as for Δy_3. Thus, for m variable components, the cost of computing the new port impedance matrix is easily seen to be $m\left[(m + 1)^2 + m + 3\right]$, which exceeds $3G_m$.

Considerable economy can be achieved if it is acknowledged that all but two of the (m + 2) ports are artificially created, and that it may only be the 2 x 2-port impedance matrix of the original 2-port which is of interest. In this case, when modifying the matrix Z_p' to account for the connection of, say, Δy_3, row 3 and column 3 need not be computed, and can be omitted when updating to account for Δy_1 and Δy_2 (Fig. 4.3(b), 4.4). If this econonomy is practiced successively as the admittance changes are sequentially absorbed within the circuit, the total cost of computing the new 2 x 2-port impedance matrix is about G_{m+1}. Moreoever, if only a single transfer impedance (say $z_{\psi\phi}$) of the original 2-port is of interest, then further economy can be practiced by eliminating row ϕ and column ψ at the outset (Fig. 4.4); the total cost is then approximately G_m.

These economies can also be achieved if the circuit contains voltage-controlled current sources: each requires the creation of two new ports, one across the controlling voltage and the other

* If more than one value of Δy_3 must be explored, the product (4.6) would be calculated before division by γ so that, after a computational cost of $(m + 2)^2$, each value of Δy_3 involves a cost of one (to find γ) plus $(m + 2)^2$.

across the controlled source. Rows associated with controlled ports, and columns with controlling ports, can be eliminated at the outset. Similarly, during matrix modification to account for change in a current source at port G controlled by a voltage at port H, row G and column H need not be computed, and can be omitted from the new port impedance matrix (See (3.23)).

The matrix-modification approach is illustrated in Fig. 4.5 (full lines), where the computational costs are those associated with the economies described in the previous paragraph. If more than one group of components is known beforehand to be of interest, the alternative algorithm (dashed line) may be more economical. To summarize the costs involved, we make use of the summary (Table 4.1) originally prepared for the current source substitution method, because the only change necessary is the omission of the cost 2m in row 3, columns 1 and 2.

4.5 Comparison of Methods

It appears that there is little to choose between the two methods and the reference method if the effect of simultaneous changes in m components is to be computed. Two comments are, however, appropriate. First, one possible advantage of the matrix modification method is that the new circuit description is available after one or more admittance changes have been absorbed: this is not the case with the current source substitution method. The second comment is relevant when each of m potentially variable components is varied sequentially rather than simultaneously, as might well happen during interactive, manual exploration by designer. Following each component change, the new circuit description costs only $m + 3 + (m + 2)^2$ to compute, which is about $3/m$ times the cost of the current source substitution method and the reference method. In other words, the matrix modification method does not exact a penalty if the individual component changes take place one at a time.

An alternative approach to large-change multiparameter sensitivity analysis described by Singhal et al [30] is based on a multilinear expression [31] for transfer impedance in which the numerator and denominator both contain all possible simple multiples of the variable admittances, each multiple being associated with a different coefficient*. However, the author does not identify the computational cost of computing the numerator and denominator coefficients (step 4), a cost which appears to be sufficiently high to render the method unattractive. Even if the cost were negligible, the cost of evaluating the multi-linear expression is greater than for the methods just reviewed, a conclusion which is supported by Gadenz et al [27] .

4.6 Systematic Exploration

For multiple sets of simultaneous changes, a further and often dramatic reduction in computational cost is possible if one condition holds. Namely, if each variable component can only assume one of a limited number of values, and if each set of m component values is

* For three variable parameters, the expression has the form

$$\tilde{z}_{21} = \frac{a_{000} + a_{100}y_1 + a_{010}y^2 + a_{001}y_3 + a_{110}y_1y_2 + a_{101}y_1y_3 + a_{011}y_2y_3 + a_{111}y_1y_2y_3}{b_{000} + b_{100}y_1 + b_{010}y_2 + b_{001}y_3 + b_{110}y_1y_2 + b_{101}y_1y_3 + b_{011}y_2y_3 + b_{111}y_1y_2y_3}$$

some combination of these individual component values. Under these circumstances, systematic exploration [32] of the different sets can be achieved at reduced cost.

This additional saving is made possible by a fact already discussed in Section 4.4: that if a component value is temporarily held constant, there is no further need to consider the port with which it is associated, and the port can be suppressed merely by eliminating the corresponding row and column from the port impedance matrix. In this way, only a smaller impedance matrix need be modified according to (4.4). More importantly, however, the longer a component can be held constant in value, the lower will be the cost of accounting for changes in the remaining components. Thus there is a computational advantage in rearranging the sequence in which the effect of the sets is computed. Firstly, the sets should be so arranged in sequence that one of the components changes as infrequently as possible. Then, within each group of sets throughout which the aforementioned component is constant in value, a fresh rearrangement into groups is effected so that a new component changes as infrequently as possible. This procedure is repeated as many times as there are components. Calculation of the effect of the component sets in the resulting sequence, while taking advantage of the computational savings associated with port suppression, is referred to as systematic exploration.

With reference to Fig. 4.3(a) suppose that, for each of C values of Δy_3, B predetermined values of Δy_2 are to be explored, and that for each value of Δy_2, the effect of A predetermined values of Δy_1 on the transfer impedance $z_{\psi\phi}$ is to be calculated. First, of course, since only $z_{\psi\phi}$ is of interest, row ϕ and column ψ of the port impedance matrix are immediately deleted, yielding a 4 x 4 matrix (the unshaded part of Fig. 4.4) which we shall denote by Z_4. Systematic exploration of the effect of the sets of component combinations then proceeds as follows. Initially, the first value of Δy_3 is absorbed. This involves the calculation of γ, the deletion of row and column 3 of Z_4, and the calculation of a new 3 x 3 impedance matrix Z_3 according to (4.6) and (4.4). Similarly, for the first value of Δy_2, a new 2 x 2-port impedance matrix (row ψ,1; columns ϕ,1) is calculated from Z_3. Finally, the same procedure is applied for each value of Δy_1: first, the produce $z_{\psi 1} z_{1\phi}$ ((4.6)) is computed; then, for each Δy_1, the constant γ is found at a cost of one and the ratio of $z_{\psi 1} z_{1\phi}$ and γ ((4.4)) also at a cost of one to determine the new $z_{\psi\phi}$. After all values of Δy_1 have been considered, the next value of Δy_2 is absorbed, as before, by operating upon the 3 x 3 matrix Z_3, a matrix which would continue to be stored until Δy_3 took on its second value. Thus the lower bound to which the "per-set" cost can asymptote is the remarkably low figure of <u>one</u>*; this is the cost of computing, for each Δy_1, the ratio $(z_{\psi 1} z_{1\phi})/\gamma$.

Some appreciation of the savings offered by systematic exploration can be gained from the computational costs shown in Table 4.2,which refers to the 3 component example of Fig. 4.3; in each case, the starting point is the impedance matrix (Z_p in Fig. 4.5) describing the original circuit with an additional port across each variable component. In case X, each of the 1000 possible combinations of 10 values each of the 3 variable components is independently evaluated by the matrix modification method. In case Y, the effect of the same combinations is explored systematically. The "per-set" cost shows a dramatic reduction.

* This asymptote is obtained if the values of $1/\Delta y$ are computed at the outset and stored for subsequent use.

In the preceding examples, the first admittance considered (Δy_3) was the most expensive to absorb and, for ease of exposition, was associated with the last row and column of the impedance matrix (Fig. 4.4). Similarly, the last admittance absorbed (Δy_1) involved the least recurring "per-component" cost, and was associated with the first of the rows and columns pertinent to the variable components. Thus although it is the order in which ports are suppressed that determines the cost of their suppression, it is convenient to (artificially) associate the cost of suppression with the port's row/column position in the impedance matrix.

As a general rule, the component whose value changes least frequently as the component sets are explored in sequence should be associated with the most expensive port, and vice versa. For the situation, like case Y, in which the number of values which each change can assume is equal, and all possible combinations of those values must be explored, it does not matter, of course, which of the components changes least or most frequently, provided the sets are ordered for systematic exploration as described earlier. But the assignment of components to ports would matter, for example, in the situation in which the number of values which a component can assume varies from one component to another. Case Z of Table 4.2, like cases X and Y, concerns the systematic exploration of 1000 combinations of 3 components, but the combinations are those of 4 values of Δy_3, 10 values of Δy_2, and 25 values of Δy_1. The order in which the changes are accounted for (Δy_3, Δy_2, Δy_1; see Fig. 4.6) is such that the least fluctuating component (with 4 value changes) is associated with the most expensive port, the next least fluctuating component (with 40 value changes) with the next-to-most expensive port, and so on. The cost per set is below that of case Y although it cannot, of course, fall below 1.

The substitution current source method does not appear to lend itself to efficient systematic exploration, since a component value held constant does not imply that the corresponding substitution current source remains constant in value. The method of Singhal et al [30] can be shown to be less efficient than matrix modification for systematic exploration.

4.7 Engineering Applications

The efficient approach to multi-parameter large change sensitivity has proved to be a very useful tool in computer aided circuit design. Two examples can be given of the manner in which circuit design can benefit from the method of systematic exploration. One is the generation of multidimensional performance contours. A two-dimensional performance contour describes how two component values may vary simultaneously - all others retaining their nominal value - without causing the circuit response to move outside a specified tolerance region. Both Butler [1] and Karafin [2] justified the use of such contours in economic circuit design, and Calahan [14] indicated that their display was the best method of communicating large-change sensitivity data to the designer who must decide parameter tolerance tradeoffs. Clearly, it is possible to extend the concept beyond two dimensions, and an efficient way of so doing is the technique of systematic exploration. First, for each frequency of interest, the values of m - 1 of the m variable components are explored systematically by the matrix modification method. Then, for each set of m - 1 changes, the corresponding value of the m^{th} is found by the inverse of (3.11). The calculation and display of three-dimensional contours has already been implemented within an interactive graphic medium [8] and will be treated in detail in the following chapter.

Systematic exploration can also be exploited in a new and efficient approach to statistical circuit analysis [4,5,46] . To obtain the benefit of systematic exploration, the probability distributions of the component values are first quantized and then appropriately discretized. In this way, a reasonable approximation to component value distribution is obtained within the constraint of the limited values set required by systematic exploration. Then, the details of the individual trials are so ordered as to ensure that, in the calculation of response change, each component change can be held constant for as many trials as possible. The problem of the application of systematic exploration to statistical circuit analysis will be one of the main subjects of Chapter 7 of this thesis.

One interesting example is the implementation of the matrix modification method within the interactive graphic circuit design facility to allow one-at-a-time adjustment of up to five parameters. This example will now be described in detail.

In the course of interactive graphic circuit design, once the circuit has been input satisfactorily and displayed on the screen, the designer may activate the appropriate light button for the multi-parameter large sensitivity display. He then indicates a group of up to 5 components whose simultaneous changes in values are to be explored. At the same time, numbers from 1 to 5 are assigned to these components sequentially as they are being indicated.

After a time lapse needed for the formation of the 7 x 7 impedance matrix according to the scheme shown in Fig. 4.5, the circuit gain and phase response curves are now displayed on the screen with solid curves representing the nominal response, and dotted curves representing the deviated response due to the simultaneous variations of the 5 components. A light potentiometer appears on the right hand side of the screen. Above the potentiometer, there are 5 dim light buttons labelled 1, 2, 3, 4 and 5. Each of these light buttons represents one of the 5 components which has been assigned to the same number (See Fig. 4.9(a)).

The designer may now start the exploration by varying the value of one of the 5 components. For instance, if variable component number 3 is going to be first explored, the light button labelled 3 is activated by the designer with the light pen and becomes bright. The arrowed position of the potentiometer now indicates the value of component number 3 under which the deviated response curves are evaluated. By moving the pointer of the potentiometer with the light pen, the value of component 3 can be adjusted to any desired value. With reference to Fig. 4.7(a), it can be noted that for each new value of component number 3, only two operations are needed for the evaluation of the new value of $z_{\psi\phi}$[+]. This enables the new dotted response curves to be displayed roughly half a second[*] after the pointer has been moved to a new position.

After the exploration of component number 3, the designer may keep component number 3 at the last value of the exploration and start the exploration of another component, say, component number 5. To do these, the light button labelled 5 is activated by the designer (Fig. 4.7(b)). On the receipt of this command, all elements of the

[*] This figure is related to the number of points characterizing the response curve. In this example the number of points is 20.

[+] To compute the circuit gain, $z_{\phi\phi}$ should also be updated.

7 x 7 matrix are now modified to absorb the change of component number 3 into the matrix (or, the circuit). As indicated in section 4.5, this process would require 50 operations*. After the time lapse of approximately 7 seconds+, light button labelled 5 becomes fully lit and light button labelled 3 becomes dim indicating that the arrowed position of the potentiometer is now the value of component number 5. Component number 5 is now ready for exploration (Fig. 4.7(c)). For each new value of component number 5, roughly half a second time lapse is needed before the new curve is displayed on the screen. At this stage, the computational cost for each value of component number 5 is identical to that of the preceding exploration of component number 3. Having finished the exploration of component number 5, the exploration of another component may be attempted by repeating the above procedure.

The circuit example of a filter shown in Fig. 4.8 has been chosen for illustration. The effect on the pass band response of the simultaneous changes of the five components (indicated by superimposed arrows) is to be studied. By using the above mentioned interactive graphic facility, the designer is able to adjust the 5 components dynamically to obtain an optimum response, as if he was adjusting the actual circuit and watching the response curve displayed on the screen of an oscilloscope. Fig. 4.9(a) and Fig. 4.9(b) show the output response curves of this circuit as two different components are being explored.

* This figure is obtained by eliminating column 4 of the matrix.

+ This figure is related to the number of points characterizing the response curve. In this example, the number of points is 20. The computer used is a PDP-15 without floating point processor.

	Current Source Substitution Method		Reference Method
	Fig. 4.2	Fig. 4.2 (dashed line)	
Once-only calculation	-	$3G_n$	G_{m+2}
Preparatory calculation repeated K times	$G_n - G_{2(m+2)} + 3G_{m+2}$	-	$G_n - G_{2(m+2)}$
Calculation for each set of component changes	$G_m + 2m$	$G_m + 2m$	G_{m+2}

Table 4.1

Approximate costs associated with different methods of computing the effect, on a circuit's transfer impedance, of simultaneous changes in m components. The table is equally valid for the matrix modification method (Fig. 4.5) if the entries 2m are omitted from row 3, columns 1 and 2.

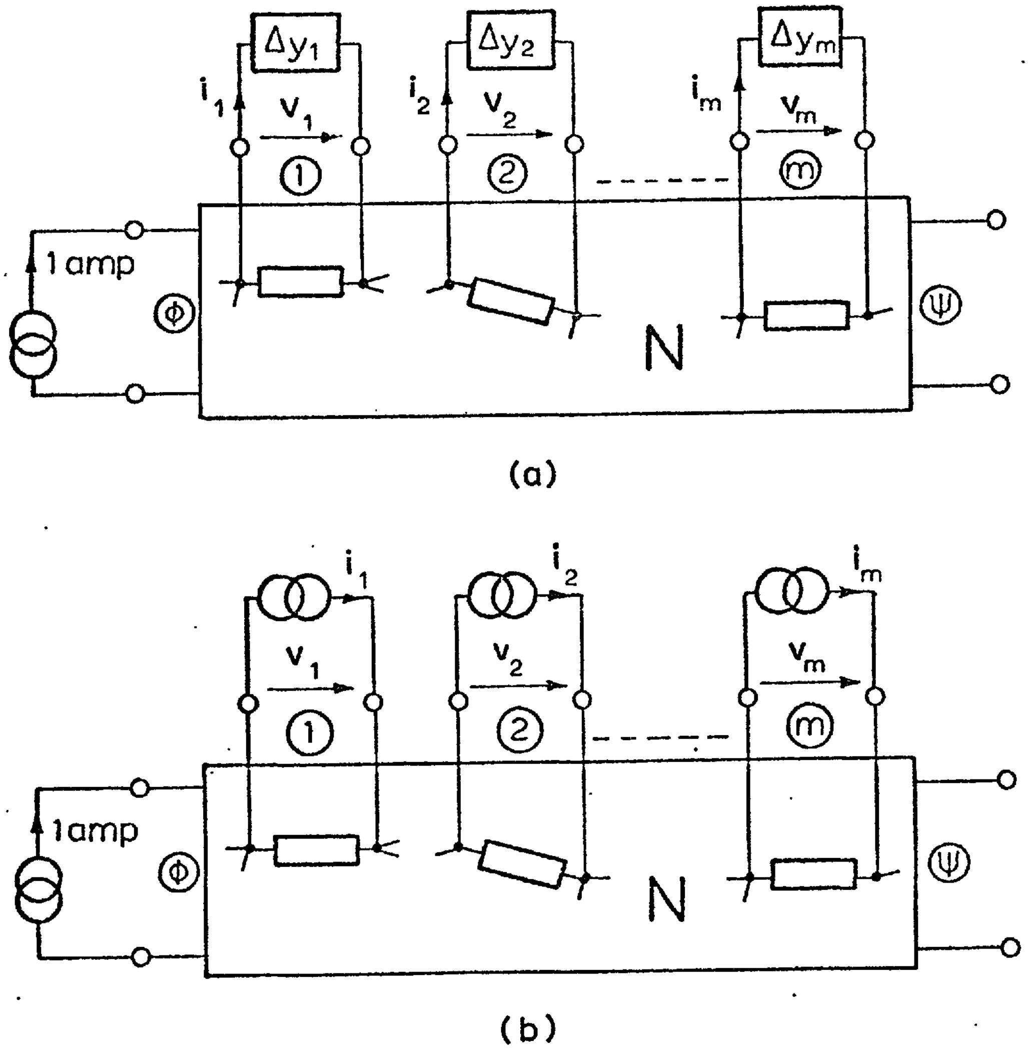
		number of values of			COMPUTATIONAL COST	
		Δy_3	Δy_2	Δy_1	1000 combinations	per set
X	Matrix modification (Fig. 4.5)	10	10	10	23,000	23
Y	Systematic exploration	10	10	10	1,669	1.67
Z	Systematic exploration with ordering	4	10	25	1,300	1.30

Table 4.2

Computational costs associated with systematic and non-systematic exploration of the effect of simultaneous changes in three components. The cost of case Z can be expressed as

$$(3^2) + \left\{4 + 4 \text{ x } 3^2\right\} + 4(2^2) + \left\{10 + 40 \text{ x } 2^2\right\} + 40(1^2) + \left\{25 + 1000 \text{ x } 1^2\right\}$$

where normal brackets correspond to the calculation of equation (4.6) and curly brackets to the calculation of γ and the modification expressed by (4.4).



(a)

(b)

Fig. 4.1. The use of substitution current sources to simulate simultaneous changes in a number of components.

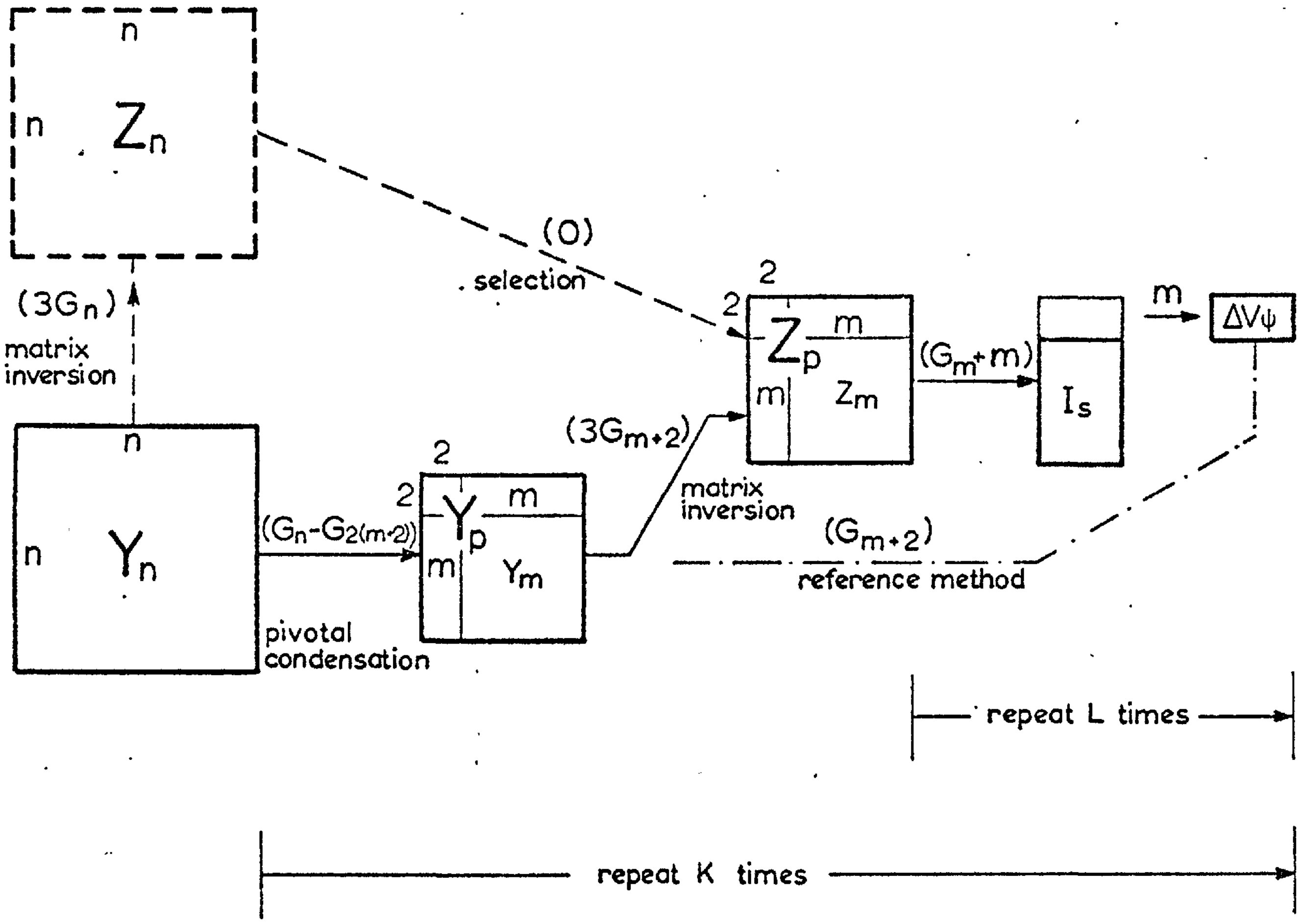
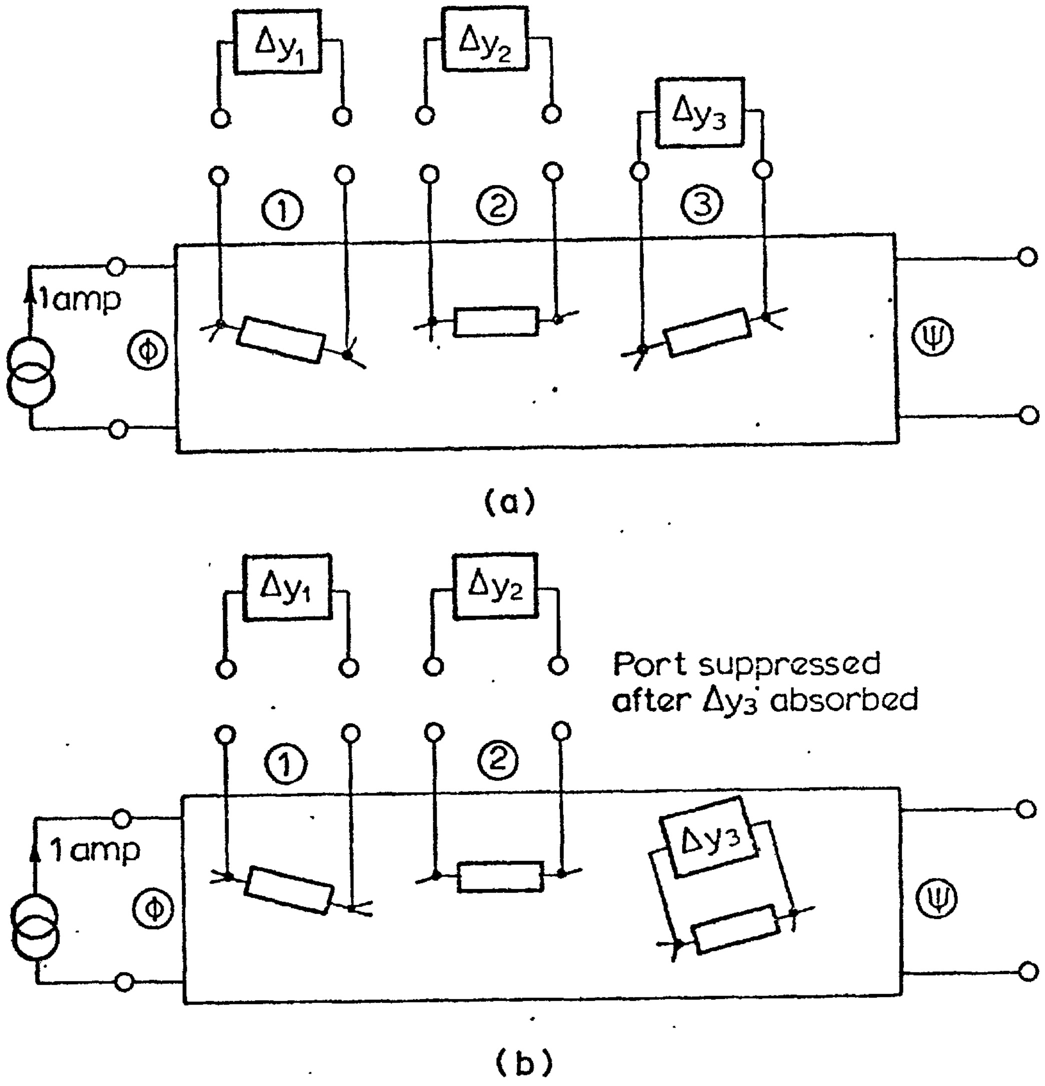


Fig. 4.2. Use of substitution current source method to compute effect, on circuit's transfer impedance, of simultaneous changes in m components. Arrows indicating repeat calculations associated with L different values of same set of components, and K different groups of components, do not apply to the reference method. Computational costs, some approximate, are indicated in parentheses.



(a)

(b)

Fig. 4.3. Relevant to the matrix-modification method of predicting effect of simultaneous component changes.

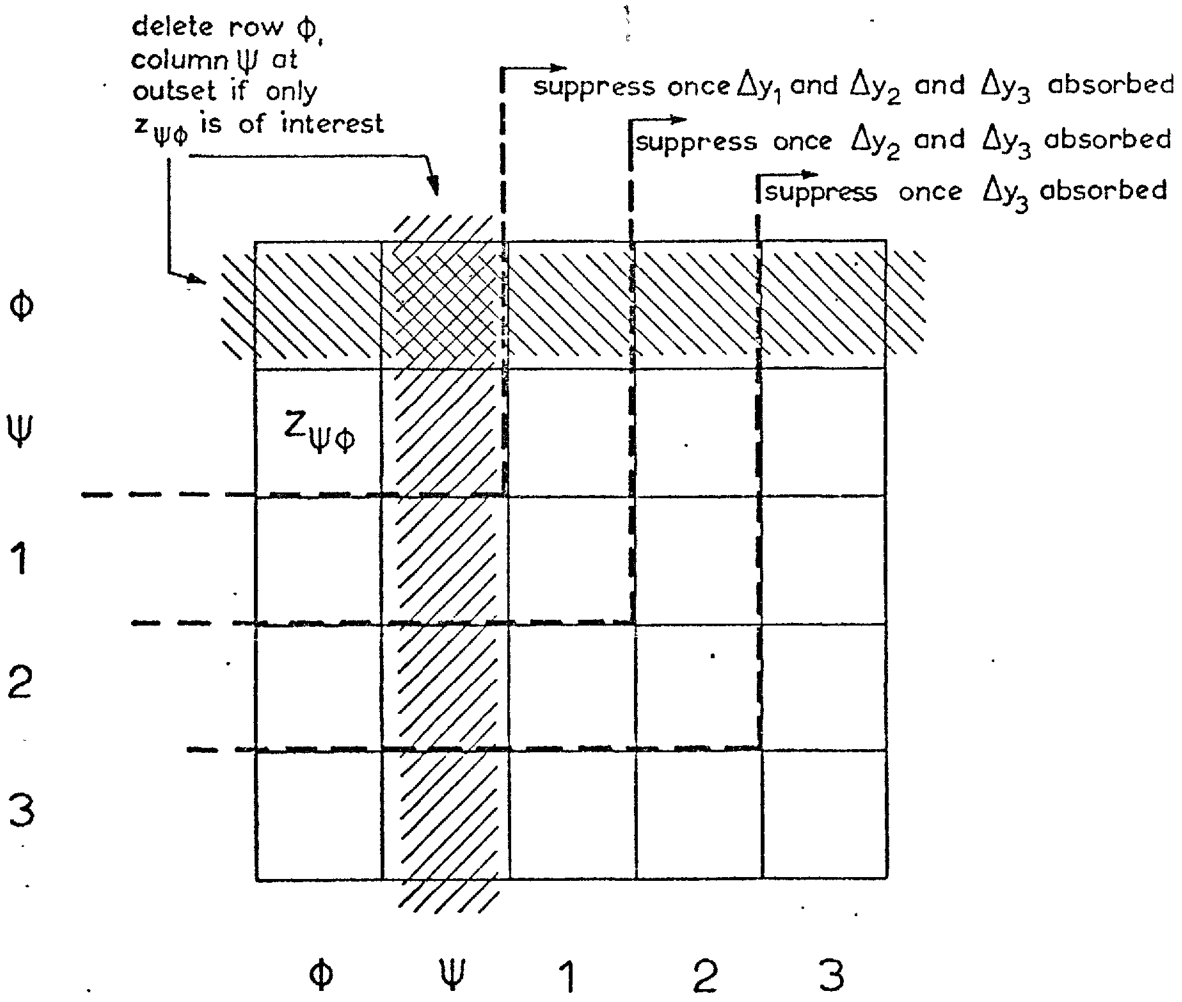

delete row ϕ, column ψ at outset if only $z_{\psi\phi}$ is of interest
suppress once Δy_1 and Δy_2 and Δy_3 absorbed
suppress once Δy_2 and Δy_3 absorbed
suppress once Δy_3 absorbed
ϕ
ψ
1
2
3
$z_{\psi\phi}$
ϕ
ψ
1
2
3

Fig. 4.4. Identification of rows and columns of port impedance matrix which can be discarded during sequential calculation of effect of component changes (see Fig. 4.3).

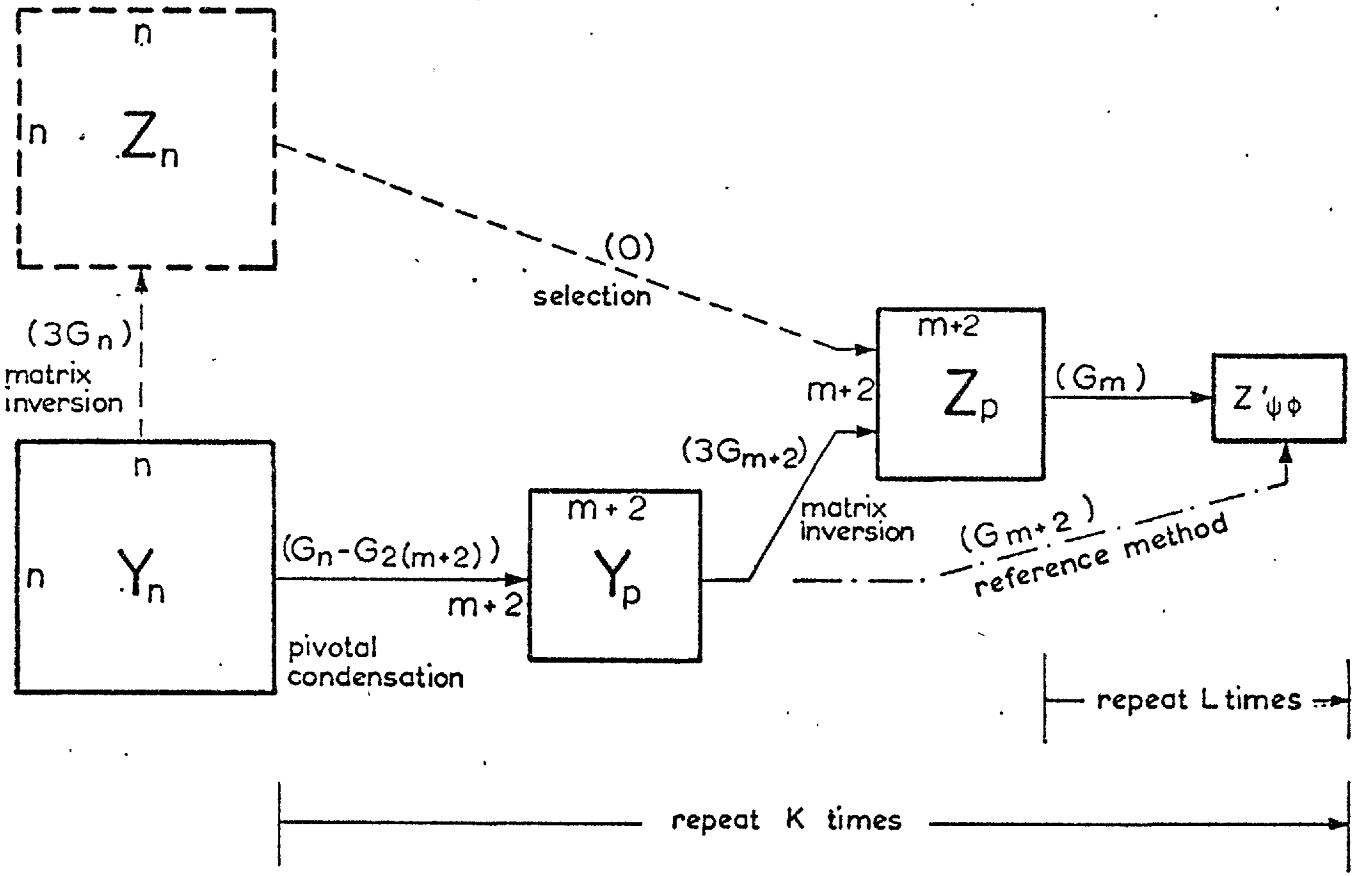


Fig. 4.5. Matrix-modification method of computing effect on circuit's transfer impedance of simultaneous changes in m components. Arrows indicating repeat calculations associated with L different values of same set of components, and K different groups of components, do not apply to reference method. Computational costs, some approximate, are indicated in parentheses.

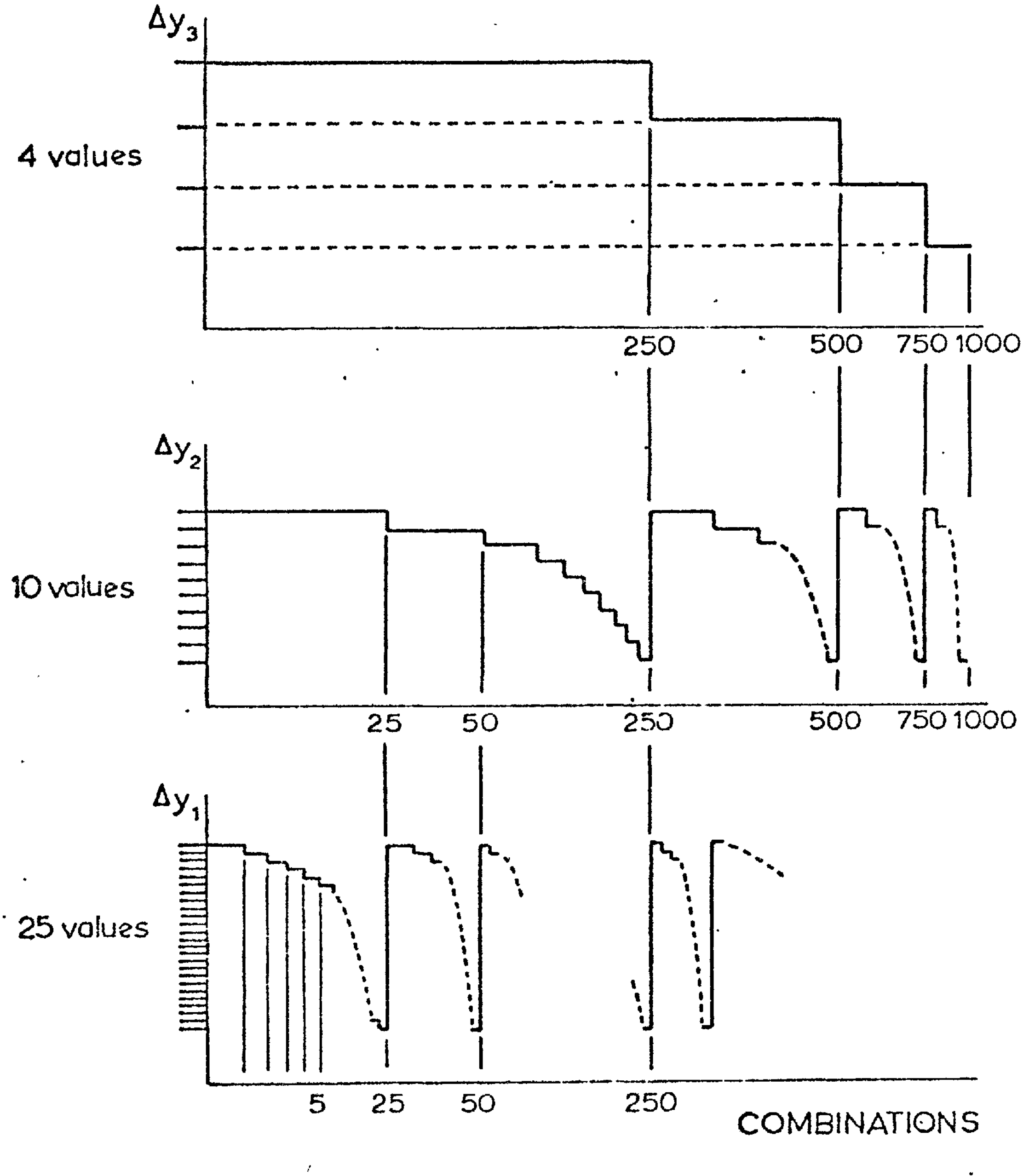


Fig. 4.6. Ordering of combinations of 4 values of Δy_3, 10 values of Δy_2, and 25 values of Δy_1 for systematic exploration of all possible combinations at lowest cost.

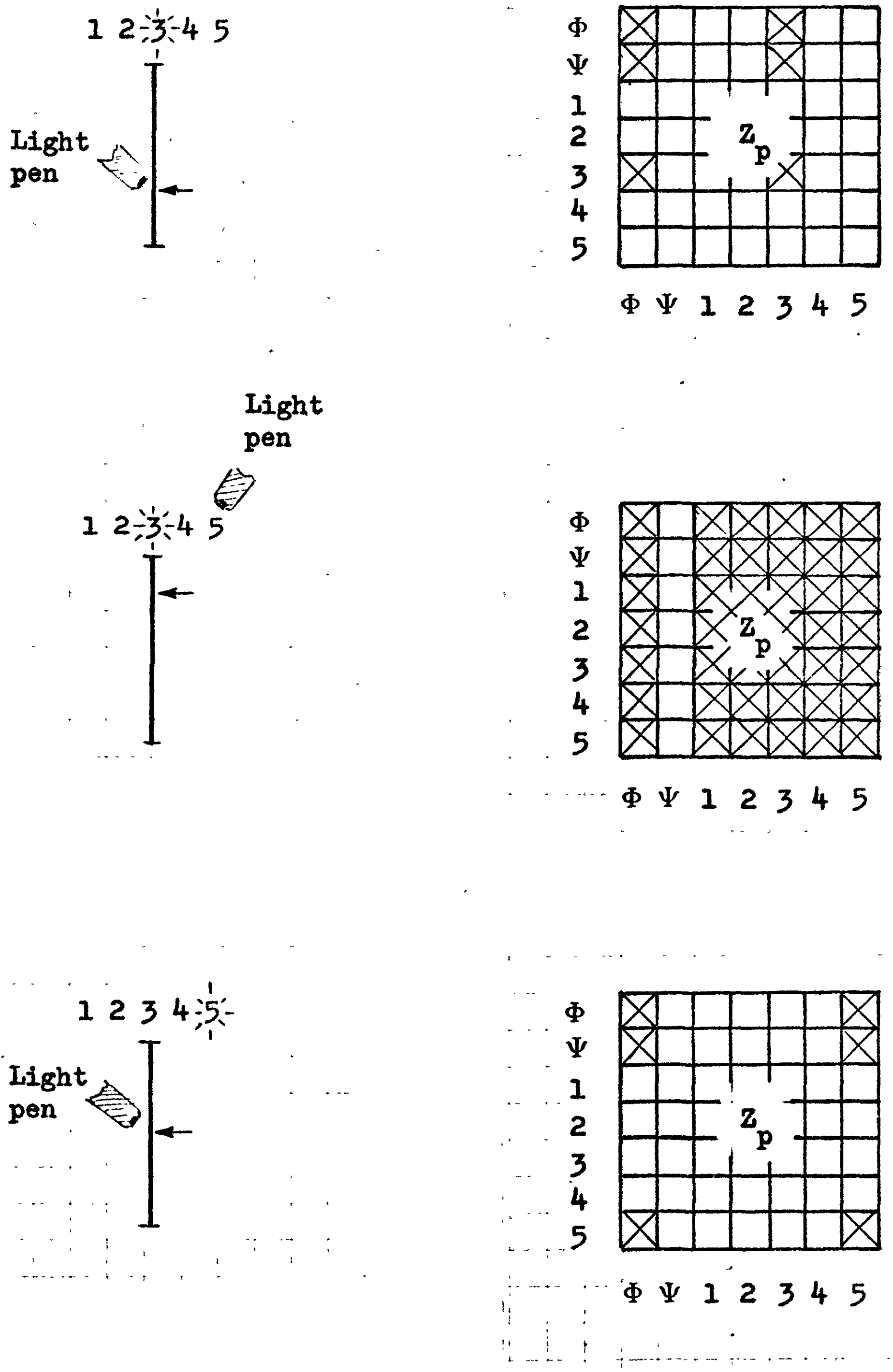


Fig. 4.7. Dynamic exploration of the effect of the simultaneous changes of a group of 5 components. Shaded regions indicate the matrix elements relevant to the computations at different stages. (a) Component No. 3 is being explored. (b) Switching from component No. 3 to No. 5. The current value of component No. 3 is absorbed into the circuit. (c) Component No. 5 is ready for exploration.

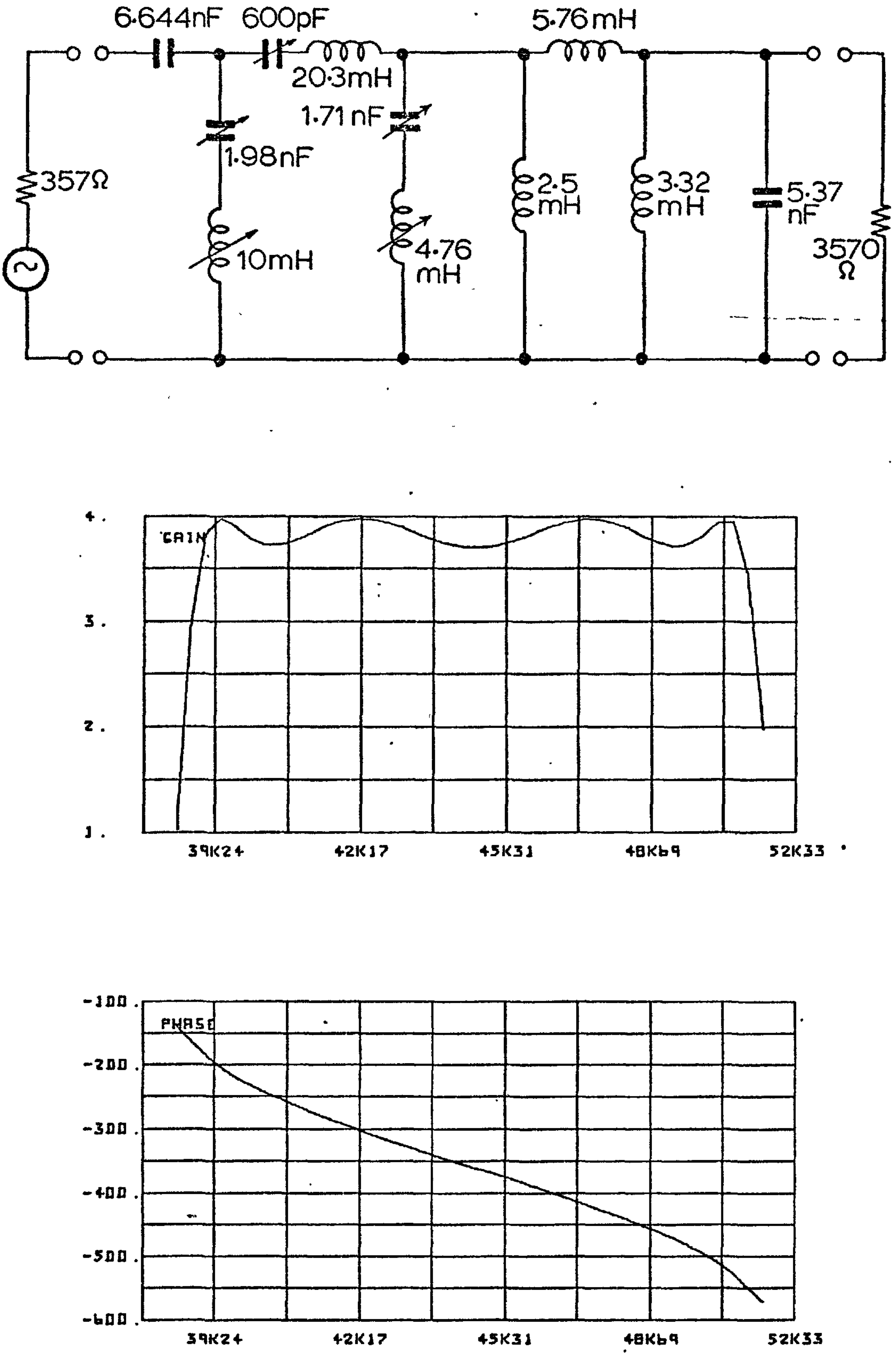


Fig. 4.8. A bandpass filter containing variable (arrowed) components and its nominal response curves (pass-band).

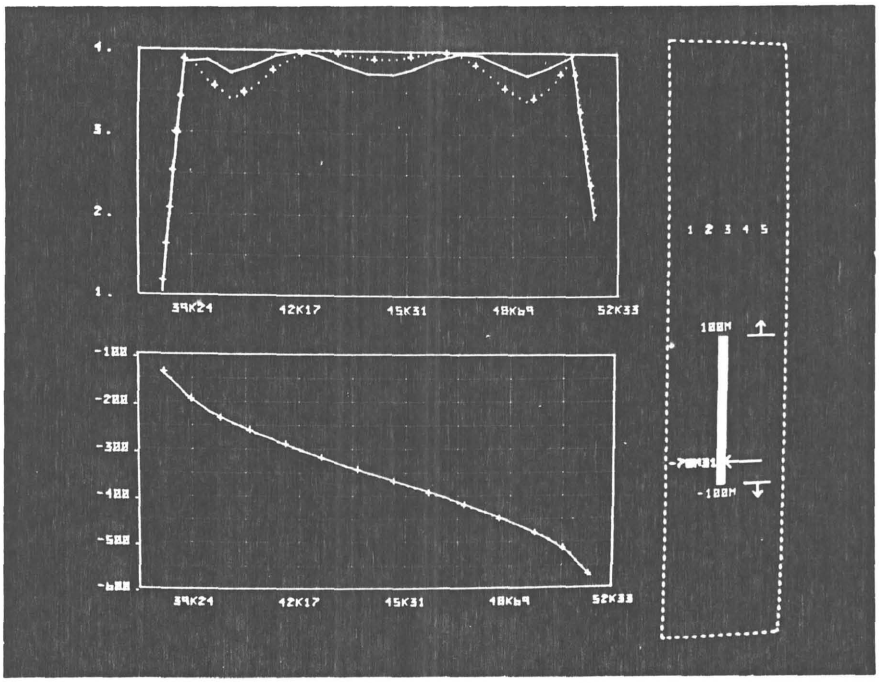


Fig. 4.9(a). Interactive display associated with dynamic exploration of the effect of simultaneous change of a group of 5 components. The one-at-a-time component change is simulated by activating the light-button which represents the corresponding component, and moving an activated light-pen along the light-potentiometer shown at right.

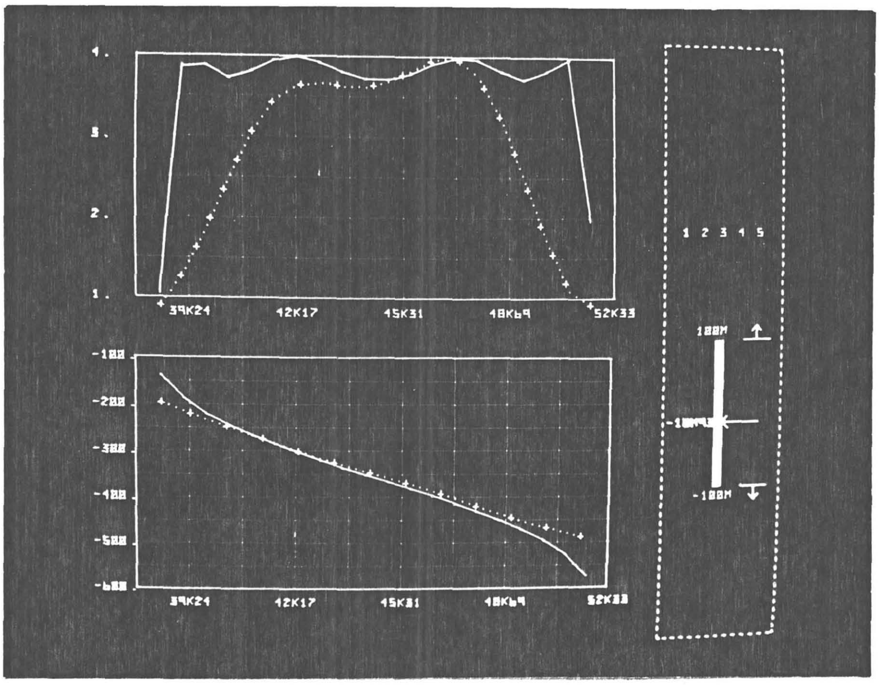


Fig. 4.9(b). Another new (dotted) response curve is displayed on the same graph as the (full) nominal curve, after the 5 components have been adjusted to a new set of values.

CHAPTER 5

TOLERANCED CIRCUIT DESIGN

5.1 Introduction

The fact that all components exhibit a spread in value requires the specification of circuit performance to be phrased in terms of an allowed tolerance region about a nominal value. In recent years, interest has been shown in numerical techniques capable of assigning maximum tolerances to the components of a circuit, subject to the performance remaining within specified limits. The reasons for doing this can be seen from a typical cost tolerance relationship of a manufactured component (Fig. 5.1). For a given specification, the wider the tolerance on each component of a circuit, the smaller is its unit cost. As more and more circuits are being designed for increasingly large production runs, the need for toleranced circuit design to become part of a computer aided circuit design package has been recognized.

At present, successful tolerance assignment algorithms are based on the technique by which the permitted tolerance on circuit output performance can be projected into the component space. If the changes in a component value are small, the technique described in Chapter 2 which permits the efficient computation of small change sensitivity may be used. With this method, either component change or performance change may be specified and the other computed; therefore the permitted tolerance on the circuit performance can easily be projected into the component space at a very low computation cost. However, the simplifying assumptions underlying this technique do not extend to the situation in which changes in component value are large.

If only one component undergoes a large change, the projection of permitted tolerance on circuit performance into the admittance plane of a particular variable component is a straightforward task, since bilinear relationship between the output voltage and the variable admittance may be used. However, if the number of components which undergo large change is greater than one, the projection becomes increasingly costly as the number of variable components increases [33] .

In this chapter, an efficient method for the projection of permitted tolerance on circuit performance into a multi-dimensional component space will be proposed. As a result of the projection, a multi-dimensional performance contour is generated in the variable component space. The implementation of this efficient method within an interactive graphic circuit design facility will be illustrated with an example.

5.2 Bilinear Relation

In order that the tolerance on circuit output voltage can be efficiently projected into the component space, an explicit expression between a change in output voltage and the resulting change in the admittance is needed. Referring to the circuit model in Fig. 3.1(a)-(c), consider equations (3.4), (3.6) and (3.10) again. We get

$$y' = \frac{yv_3' + i_3}{v_3'} \qquad (5.1)$$

$$\Delta v_2 = z_{23}\, i_3 \qquad (5.2)$$

$$v_3' = z_{31} - z_{33}\, i_3 \qquad (5.3)$$

Rewriting (5.2) so that

$$i_3 = \Delta v_2 / z_{23} \qquad (5.4)$$

and substituting for i_3 in (5.1) and (5.3) and then for v_3' in (5.1) leads to the following bi-linear relationship between the new admittance value y' and the corresponding change in response voltage $\Delta\, v_2$:

$$y' = \frac{a + b\, \Delta\, v_2}{c + d\, \Delta\, v_2} \qquad (5.5)$$

where $a = y\, z_{31}$, $b = -\left(y \frac{z_{33}}{z_{23}} + \frac{1}{z_{23}}\right)$

$c = z_{31}$, $d = - \frac{z_{33}}{z_{23}}$

Using equations (3.5), (3.7) and (3.12) a similar expression can be derived for a new mutual admittance value g_m'.

$$g_m' = \frac{a + b\, \Delta\, v_2}{c + d\, \Delta\, v_2} \qquad (5.6)$$

where

$$a = g_m z_{41} \quad , \quad b = -(g_m \frac{z_{43}}{z_{23}} + \frac{1}{z_{23}})$$

$$c = z_{41} \quad , \quad d = -\frac{z_{43}}{z_{23}}$$

Equations (5.5) and (5.6) represent a special case of the conformal transformations [34,35] . A transformation is said to be conformal if it preserves the sense as well as the magnitude of angles.

To ensure that the bilinear transformation is conformal, the condition $bc - ad \neq 0$ must hold (theorem on the conformal corresponence of two domains). If $ad = bc$, y' is a constant independent of Δv_2 and thus the entire Δv_2-plane is mapped onto the same point in the admittance plane. If $b = -c$, the transformation is involutionary, i.e., y' and Δv_2 may be interchanged. Critical points, i.e., points where the transformation is not conformal, are $\Delta v_2 = -c/d$ and $\Delta v_2 = \infty$. The inverse of the bilinear transformation is also a bilinear transformation.

If a region in the Δv_2-plane with a circumference C is mapped on to the y-plane to give another region with circumference $\tilde{C}$, then the interior of C is mapped either onto the interior or the exterior of $\tilde{C}$. The latter situation will occur if, and only if, the pole of the function (5.5) or (5.6) is situated in the interior of C.

The bilinear transformation maps circles into circles. Straight lines are included here since they represent circles with infinite radii, but it is not implied that straight lines are necessarily transformed into straight lines. Those circles in the y-plane which correspond to circles passing through the point $\Delta v_2 = - c/d$ (whose image is $y' = \infty$) are straight lines.

If a circle in the Δv_2-plane is given by its centre point p and its radius r, then the centre point $\tilde{p}$ and the radius $\tilde{r}$ of the corresponding circle in the y- plane can be computed by the following expressions

$$\tilde{p} = \frac{b\,p + a - d^* r^2 S}{d\,p + c} \quad \text{and} \quad \tilde{r} = r\,|S| \qquad (5.7)$$

where

$$S = \frac{bc - ad}{|d\,p + c|^2 - |d\,r|^2} \quad \text{and} \quad \left|p + \frac{c}{d}\right| \neq r \quad (5.8)$$

The inequality condition means that the pole of the bilinear function (5.5) or (5.6) lies inside or outside the circle in the Δv_2-plane, but not on it. d* is the complex conjugate value of d.

5.3 Simple Tolerance Region

The projection of permitted tolerance on circuit output voltage into a one dimensional component space using the bilinear relationship can be illustrated by a simple example. It was assumed that a circuit and its single circular tolerance region on the output voltage plane had been proposed and that it was required to find the allowed tolerance on one of the network components. Assume that all other components remain fixed at their nominal values.

In view of the bilinear nature of the relation between the change in response voltage Δv_2 and the resulting change in admittance of the variable component, the corresponding region in the admittance planewill then also be circular. Therefore, in order to transform the circular tolerance region from the v_2-plane* onto the y-plane, only three points on the circumference of the circular region need to be calculated. In general, the centre ($\tilde{p}$) of the resulting circle will not be coincident with the nominal value of the admittance. The inside of the tolerance region in the v_2 plane will be transformed into either the inside or outside of the circular region in the admittance plane. The region on the admittance plane resulting from this projection indicates how far the admittance of a component is allowed to deviate from the nominal, subject to the output voltage remaining within the specified tolerance region. Fig. 5.2 shows the transformation of such a simple tolerance region from the v_2-plane onto the y-plane. This transformation constitutes the basic technique for the computation of performance contour [32,8] and the scheme for the elimination of redundant model complexity [10] .

5.4 Complex Tolerance Region

1. Multiply defined specifications

In many of the practical cases, the circuits with which we are dealing have a scalar performance criterion which is usually represented by a circle C on the complex output voltage plane. For example, the scalar performance criterion could be the magnitude or phase angle of the output voltage of a circuit.

If the performance of a circuit only has a single criterion, one performance measure C_0 and an allowable degradation Δ C from nominal are specified. Two circles ($C_0 \pm \Delta C$) mark the boundaries of a tolerance region separating acceptable circuits from the unacceptable

* Note that the v_2-plane is the same as the Δ v_2-plane except for a displaced origin.

ones. Therefore, in order to establish the tolerance region on the admittance plane, more than one circle on the output voltage plane has to be transformed onto the admittance plane.

Fig. 5.3 shows the two circles representing the upper and lower tolerance limits of the magnitude of an output voltage on the complex voltage plane. The two circles $|v_0| \pm \Delta |v|$ are then transformed onto the admittance plane of the variable component. The area between these two transformed circles (shaded area) constitutes the tolerance region on the admittance plane.

In many cases, the performance of a circuit is evaluated by a number of performance criteria. For example, in the case of an amplifier, the phase and magnitude of the output voltage may be the desired performance criteria, in the case of a filter, the insertion loss at several frequencies will be specified. The tolerances associated with all these criteria must be considered. By projecting the tolerance regions associated with the different criteria onto the same admittance plane, we obtain an area shared by all the resultant individual tolerance regions. This is the tolerance region for all criteria, and it will possess a character such that within this region, the admittance of a component is allowed to deviate from the nominal without forcing the output voltage to move outside the specified tolerance region for all criteria. Fig. 5.4 shows that, as a result of projecting the tolerance regions of phase and magnitude onto the admittance plane, the tolerance region for both of these two criteria (shaded area) is established.

2. Region of acceptability

Theoretically, within the tolerance region, any complex admittance value of a variable component would result in satisfactory circuits. However, on practical considerations, when a component is assumed variable, its value can change, and not its type. Taking this consideration into account, for the tolerance region on the admittance plane, only the intersection on the corresponding real or imaginary axis indicates how far a component value is allowed to deviate from the nominal without producing unsatisfactory circuits. For example, if the tolerance region cuts both the real and imaginary axes on the complex admittance plane, and the component is a resistor, only the section of the positive real axis within the tolerance region is the actual permitted tolerance region of the particular component. The actual permitted tolerance region in a component space may also be termed the region of acceptability, the contour of which is denoted the performance contour [1] . In Fig. 5.3 and Fig. 5.4, assuming that the component is an inductor, the heavy line within the tolerance region is the region of acceptability of the variable component.

From the above discussion, it can be noted that the dimension of a region of acceptability is equal to the number of variable components. Therefore, the projection of the tolerance region (single or multiple criteria) from the output voltage plane into a multi-dimensional component space to form a region of acceptability now becomes practical. A multi-dimensional region of acceptability describes how a number of components of a circuit can vary simultaneously around their nominal values, all the other components being kept constant, without the circuit response being forced outside its tolerance. The concept of a multi-dimensional region of acceptability is best illustrated by a two dimensional, two criteria example. Consider the circuit shown in Fig. 5.5. If the designer's specification calls for the deviation of the magnitude of the output voltage to be limited to within ± 10% at 30 Hz and 300 Hz, and L and C are assumed variable, then, the region of acceptability shown in Fig. 5.6 is bounded by four lines. Each of these is the locus of all points producing circuits exactly satisfying one of the specifications.

5.5 Performance Contours

1. Computational method

In order to characterize the region of acceptability in a variable component space, the contour of the region (i.e., the performance contour) must be computed. So far, different methods for the computation of performance contours have been proposed [1,33] . However, due to the fact that all of these methods are based on either the repetitive analysis of a circuit or the substitution current source technique, the computational cost has been very high indeed. This is especially true when a multidimensional performance contour has to be computed.

As a result of the discussion of the previous chapter, an efficient approach to the computation of m-dimensional performance contours can be formulated. Briefly, it can be stated as follows; first, for each frequency of interest, the values of m - 1 of the m variable components are explored systematically by the matrix modification method. Then, for each set of these m - 1 component changes, the corresponding values of the mth component are found by the inverse calculation expressed by (5.5) or (5.6). A more detailed description of the algorithm will now be presented.

In the following description of the computational procedure of a 3-dimensional performance contour we refer to Fig. 4.3(a); the three admittance changes Δy_1, Δy_2 and Δy_3 will correspond, respectively, to the x-axis, y-axis and running parameters (z-axis) of the family of 2-dimensional performance contours.

The steps in the computational procedure are:

a) For the 5-port of Fig. 4.3(a) the port impedance matrix at each frequency is obtained by one of the schemes indicated in Fig. 4.5

b) A value is assigned to the component whose change Δy_3 from nominal may be regarded as temporarily fixed, and the port impedance matrix appropriately modified. If the transfer impedance of the original 2-port is the only circuit response for which tolerances have been specified, then only a 3 by 3 matrix need be computed and retained (see Fig. 4.4).

c) The first of a sequence of values is assigned to the component whose admittance change is denoted by Δy_2, and the port impedance matrices modified. If the situation is as stated in b) above, then only a 2 by 2 matrix need be computed and retained for each frequency.

d) For each value of Δy_2, equation (5.5) allows the range of Δy_1 permitted by the tolerance on transfer impedance to be determined. If the tolerances of the magnitude of the complex transfer impedance at a number of frequencies are considered,for each frequency, three values of Δv_2 are selected and transformed according to (5.5) in order to establish the upper or lower boundary of the tolerance region in the Δy_1 plane. By a simple construction the corresponding maximum and minimum component values can be found, thereby yielding two points on the performance contour relevant to each frequency. Fig. 5.3 illustrates this part of the computational procedure for the situation in which the component is an inductor.

e) Step b) through d) are repeated for a new frequency.

f) By appropriate comparisons, too detailed to be described here, two most significant points are selected to characterize the boundary of the performance contour for all frequencies.

g) Steps b) through f) are repeated for a new value of Δy_3.

2. Properties

In a computer aided design, 2 or 3-dimensional performance contours has found their uses in many places. This is because a 2 or 3-dimensional performance contour can be visualized when it is displayed graphically, and the shape of a performance contour can reveal some important properties of a circuit. Here are some properties associated with 2-dimensional performance contours: obviously they can easily be extended to 3-dimensional cases.

a) If the performance criterion is singularly sensitive to one of the two components constituting the performance contour, the performance contour will contain the line (dotted) shown in Fig. 5.7(a). This line, in fact, is the x or y axis and is unbounded at either end.

b) If the performance criterion is singularly sensitive to the product of the two components consituting the performance contour, the performance contour will contain the curve shown in Fig. 5.7(b). This curve is not bounded at either end.

c) If the performance criterion is singularly sensitive to the ratio of the two components consituting the performance contour, the performance contour will contain the 45^{o} line shown in Fig. 5.7(c). This line is not bounded at either end.

The proof of the above properties and the others may be found in [1] .

5.6 Engineering Applications

In view of their reported value in circuit design, and their relative ease of interpretation when presented graphically, it was decided that the provision of 3-dimensional performance contours should be added to an existing interactive graphic circuit design facility [8] .

The above procedure was programmed for PDP-15 computer with VT-15 interactive display, and is illustrated here by an example. The circuit of interest (Fig. 5.5) and its nominal component values are easily defined by a light-pen and displayed on the screen. Similarly, light-pen activation of suitable command buttons and parts of the displayed circuit diagram permit the easy definition of the variable components (L, C and R), the response of interest (v), and the tolerance upon this response at various frequencies (± 10% of magnitude at 30 and 300 Hz). The resulting display is shown in Fig. 5.8. On cue, a new performance contour appropriate to the next value of the running parameter (R) is displayed. Fig. 5.9 shows a sequence of performance contours corresponding to five values of the resistance R. Clearly, a number of different methods could be devised to display the information embodied in Fig. 5.9.

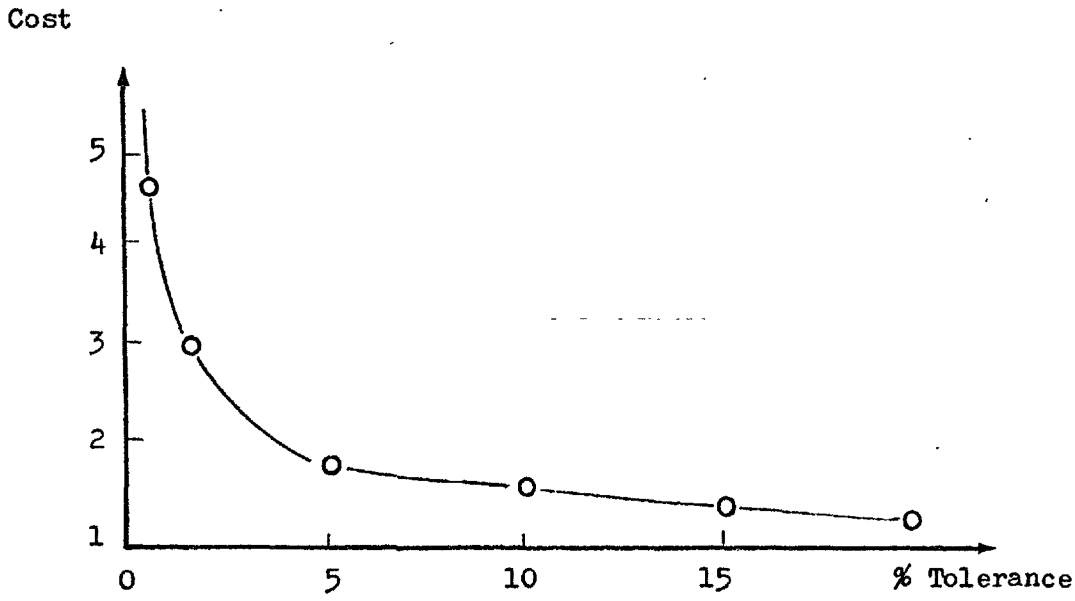


Fig. 5.1. Typical cost vs. tolerance relationship of a manufactured component.

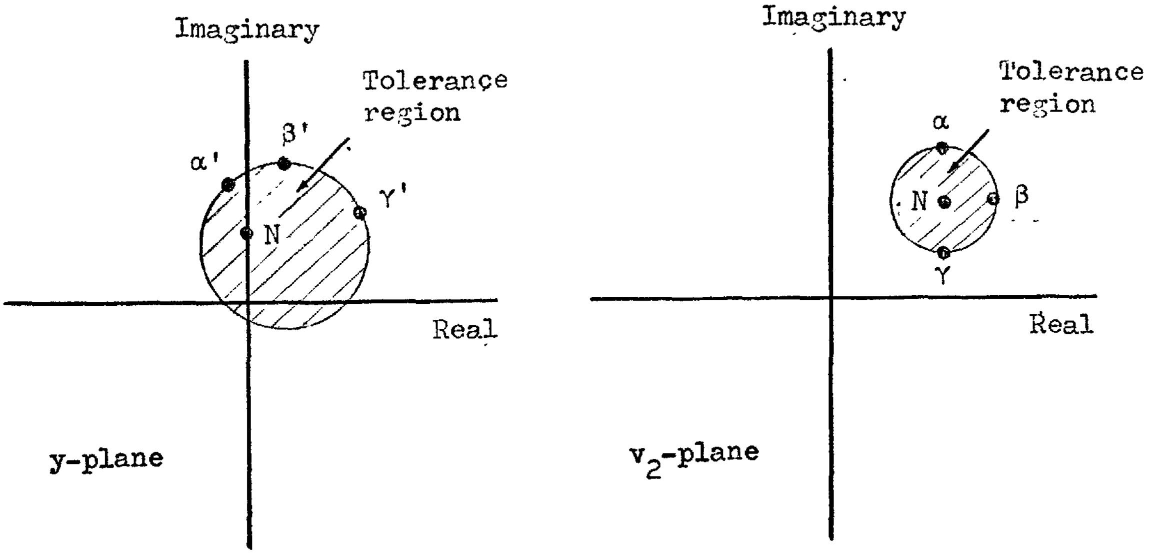


Fig. 5.2. Transformation of a simple tolerance region in the v_2-plane on to the y-plane.

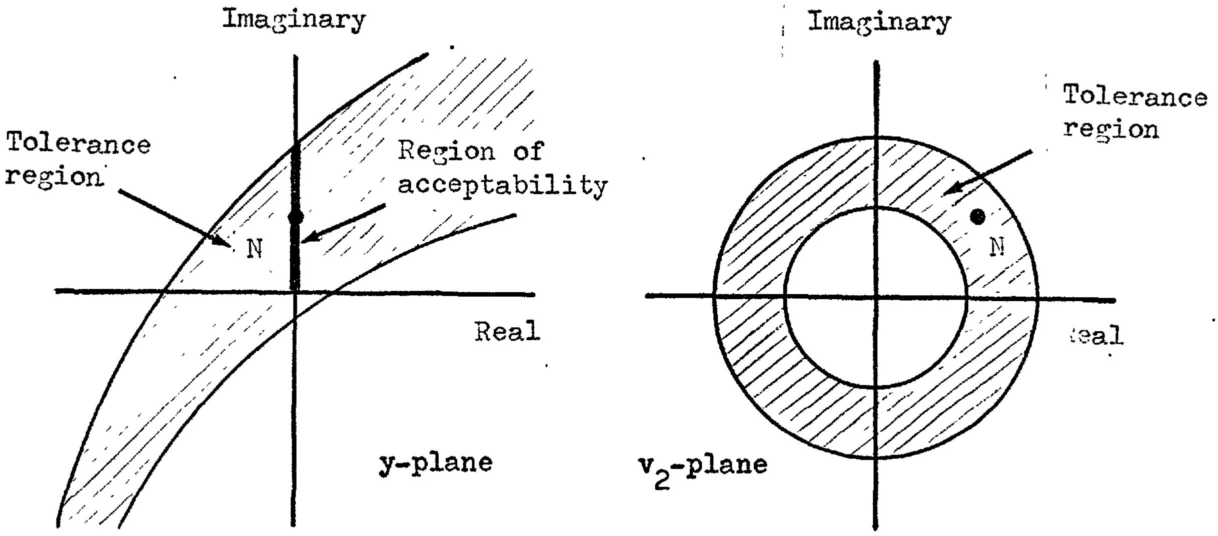


Fig. 5.3. Transformation of a complex tolerance region in the v_2-plane on to the y-plane. Heavy line represents the region of acceptability.

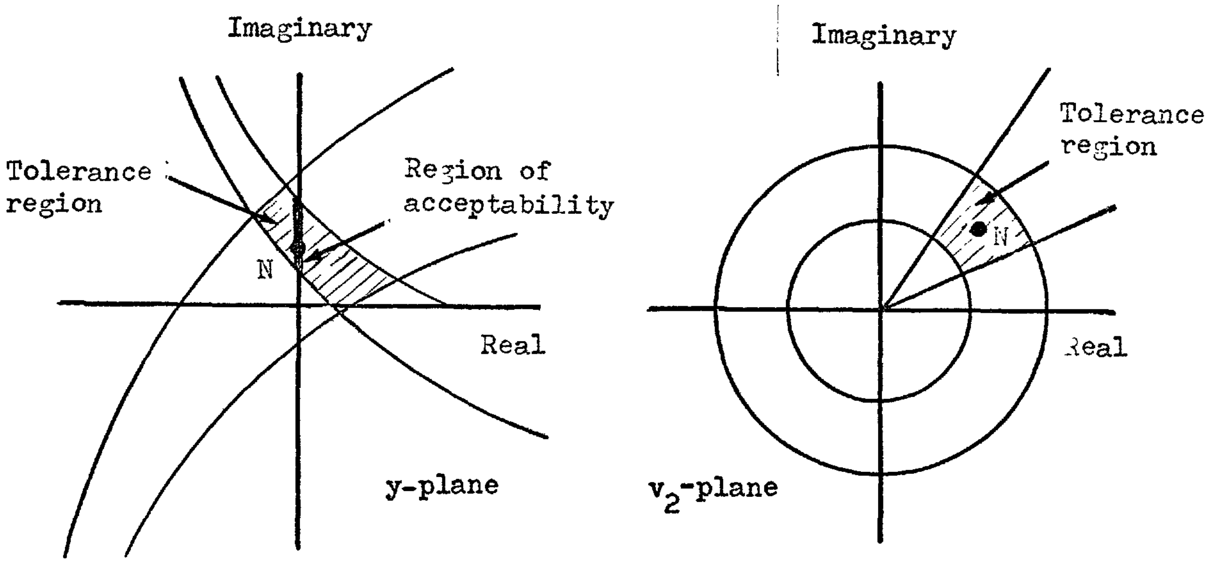


Fig. 5.4. Transformation of a two criteria tolerance region in the v_2-plane on to the y-plane. Shaded area indicates the resultant tolerance region on the admittance plane. Heavy line represents the region of acceptability.

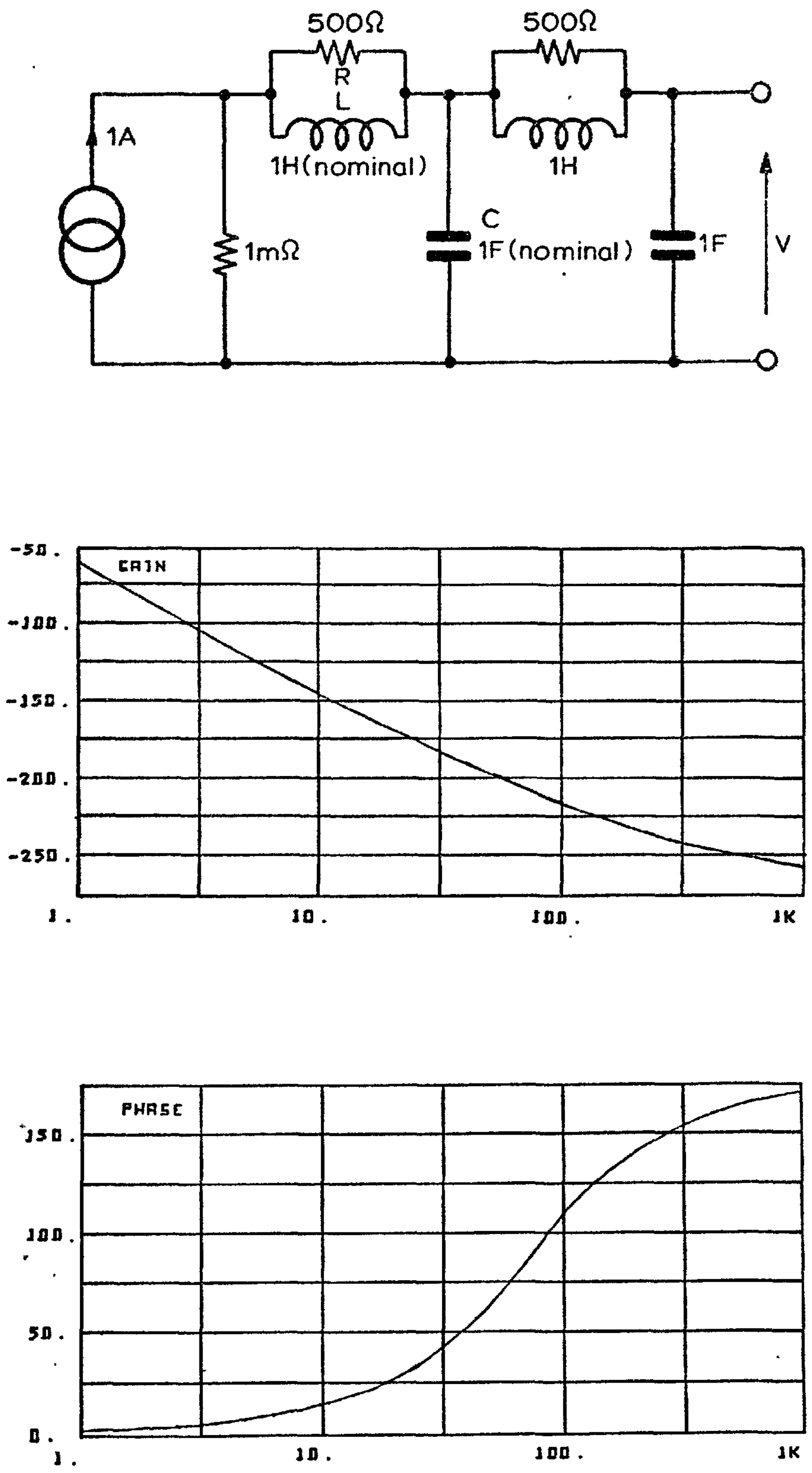


Fig. 5.5. A circuit and its nominal response curves relevant to the illustration of performance contours.

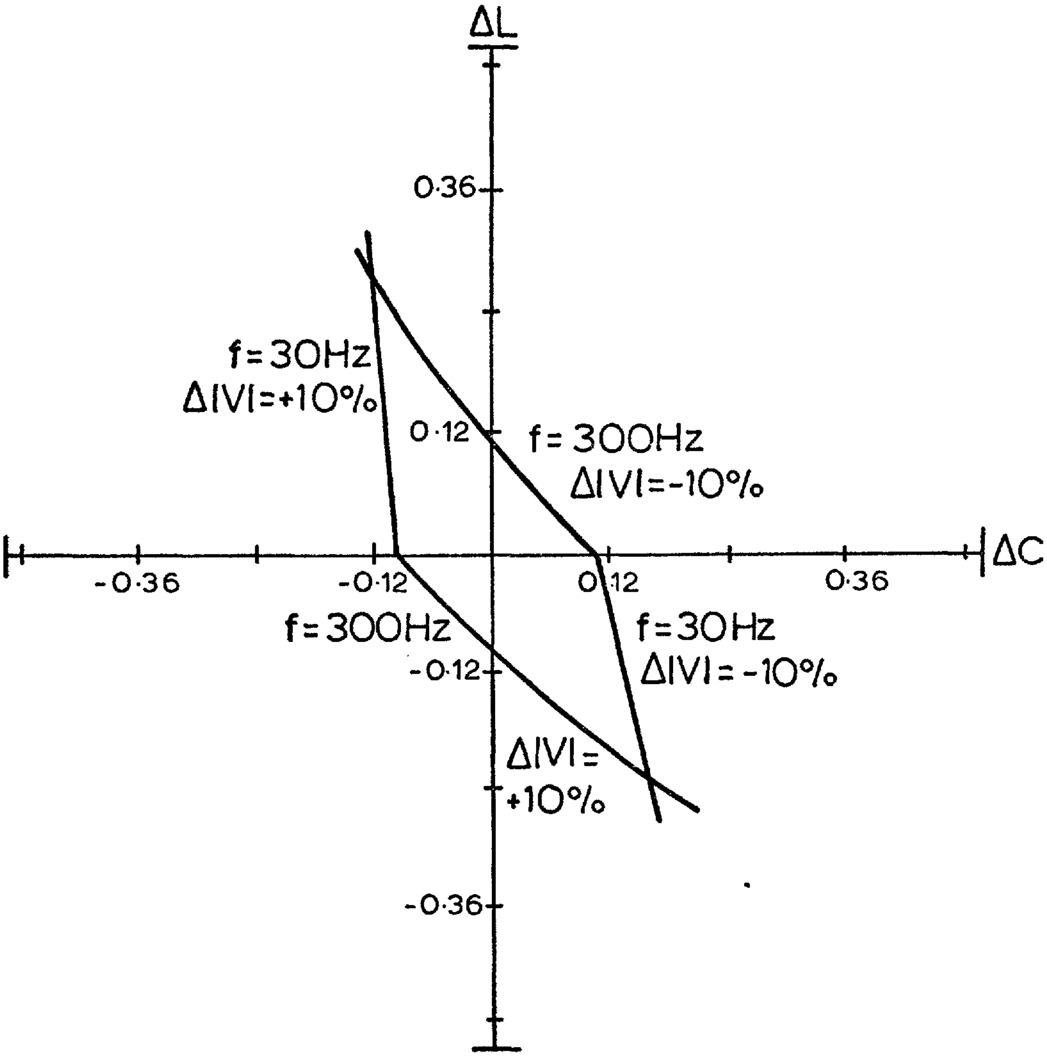


Fig. 5.6. Performance contour appropriate to the variable components L and C of the circuit of Fig. 5.5.

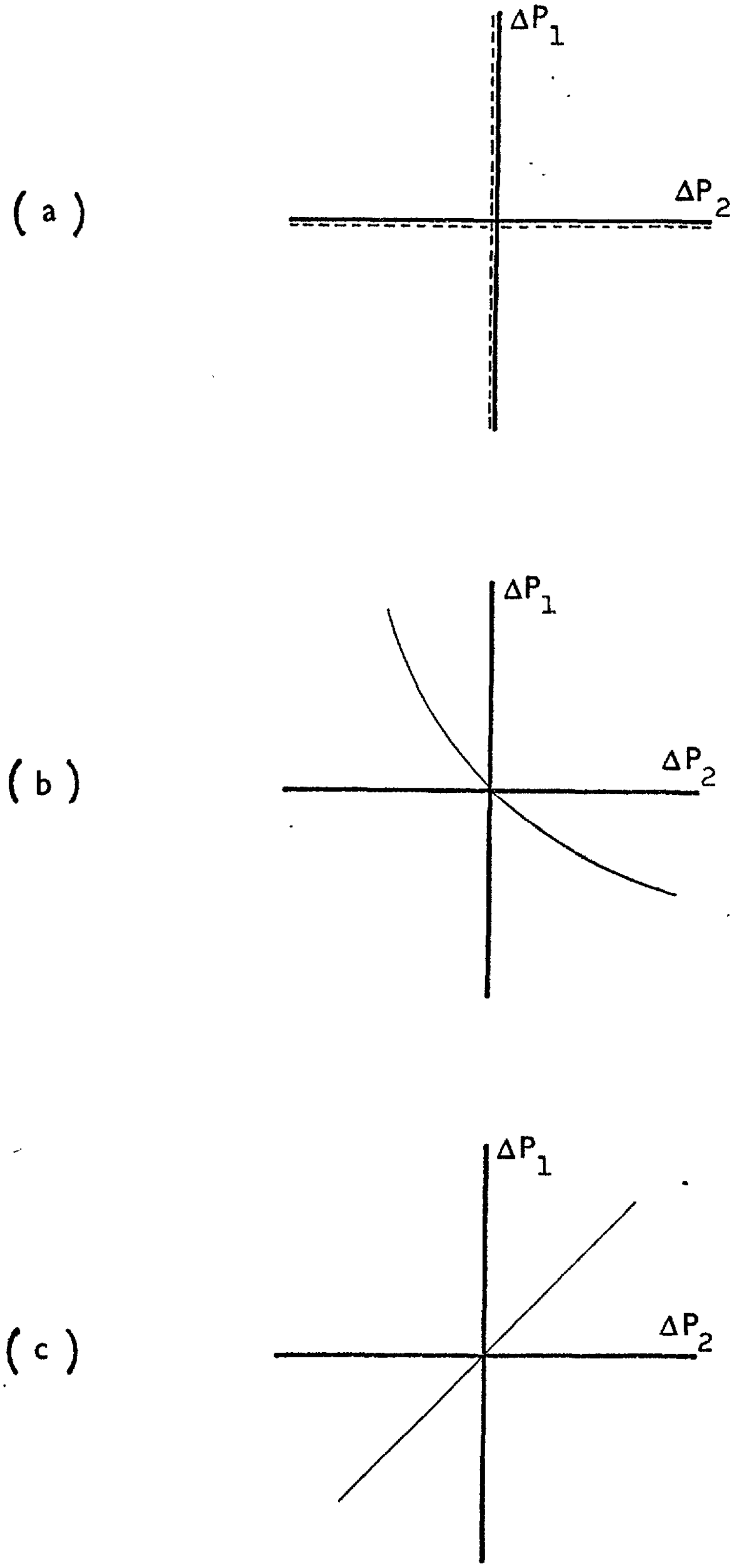


Fig. 5.7. Graphs illustrating some important properties of performance contours. P_1 and P_2 are the two variable components.

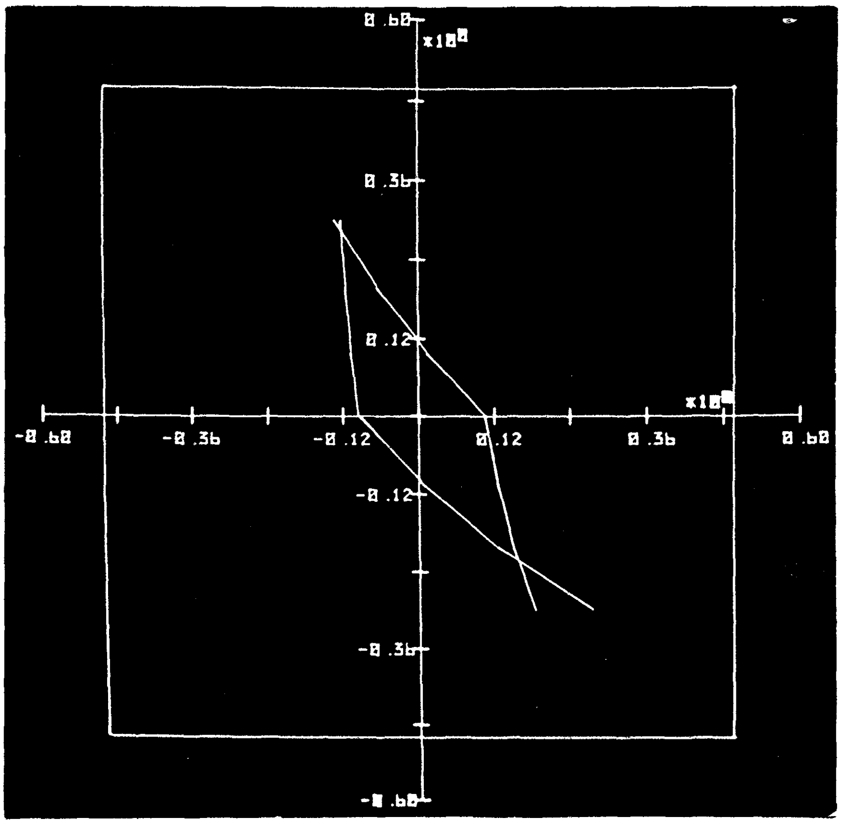


Fig. 5.8. Displayed performance contour appropriate to the variable components L and C of the circuit of Fig. 5.5 (see Fig. 5.6).

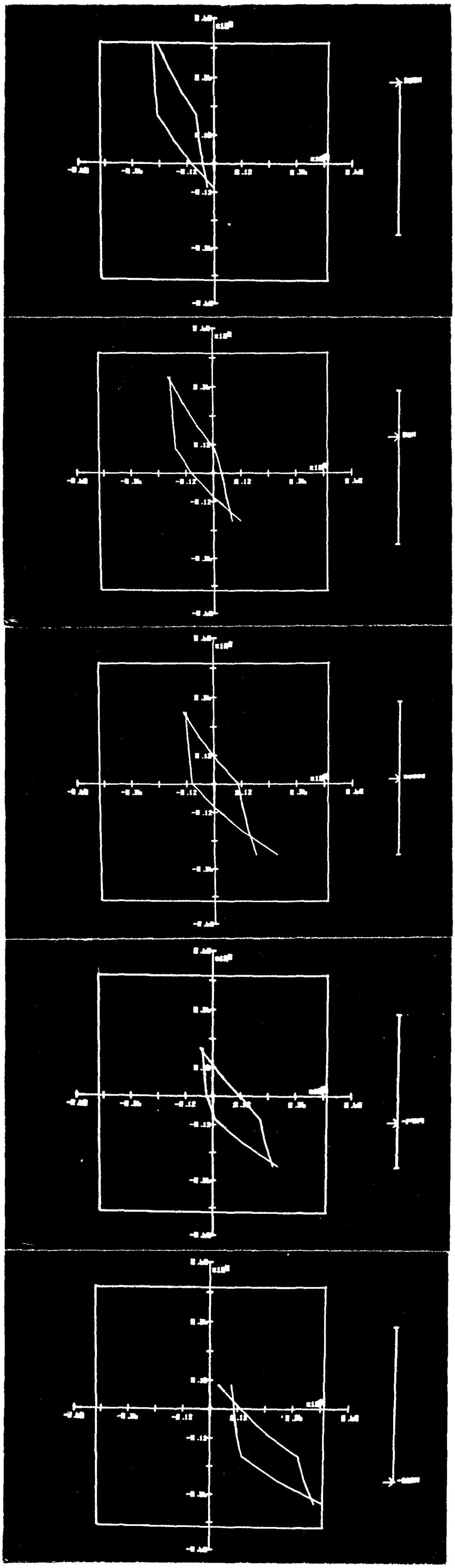


Fig. 5.9. Sequence of performance contours for the circuit of Fig. 5.5, each associated with a given value of R. The square outline defines ± 50% changes in the components. Upper and lower extremities of the vertical potentiometer define + 20% and - 20% variations in R respectively.

CHAPTER 6

TRACKING SENSITIVITY

6.1 Introduction

The design of integrated circuits, as well as many discrete circuits, must accommodate simultaneous variations in component values as a result of change in some environmental (global) variables such as temperature, time (ageing) or radiation. In such cases, a number of component values exhibit the same form of functional dependence upon the global variable and for this reason the term 'tracking sensitivity' [36] is employed. In investigating and attempting to compensate for this dependence, the designer would clearly be assisted by some efficient means of computing its effect upon circuit behaviour. It is additionally useful, particularly within an interactive graphic circuit design facility, if the computation is efficient enough to permit dynamic exploration of the effect, so that its compensation may more easily be achieved.

So far, several efficient methods for the evaluation of differential and large change sensitivity have been discussed. In order to compute the effect of the global variable on the circuit response, large change sensitivity algorithms may be used. However, due to the fact that the global variable usually has an effect on nearly all the components of a circuit, the number of variable components is proportionally very large. In addition, the nature of this particular kind of components variation does not make itself suitable for systematic exploration*. The large change sensitivity algorithms so far developed are not efficient enough for such a purpose. In the method proposed in this chapter, the bulk of the calculation is independent of the global variable and is carried out initially; the additional calculation for each value of the global variable then involves only a small number of operations. The application of this approach, within the context of highly interactive computer aided design, is illustrated by an example.

6.2 Parametric Circuit Description

Consider a linear nonreciprocal two-port containing resistors, capacitors, inductors and voltage-controlled current sources, whose property of interest is its transfer impedance $z_{\psi\phi}$. To simulate the effect of component variation induced by change in the global variable, the single-frequency admittance of each component is described by the following expression:

$$y_i = y_{iN}\,(1 + K\,\xi_i) \qquad (6.1)$$

Of the two coefficients ξ_i and K, the former is 'local' to the ith component, whereas the latter, K, is 'global' and common to all

* If the effect on circuit performance of a number of values of the global variable is to be explored sequentially, for each new value of the global variable, the variable components will have a set of new values which is totally different from the preceding set. Therefore, no advantage can be taken by holding any one of the variable components temporarily constant during the exploration.

components. For example, the effect of ambient temperature (characterized by K) upon components having a range of temperature coefficients (characterized by ξ_i) is capable of simple description. y_{iN} denotes the nominal value of admittance.

If all components in a circuit are so described, the entire circuit can be described by the sum of two nodal admittance matrices, viz:

$$Y = Y_N + K\,Y_V \tag{6.2}$$

Here, the subscripts N and V denote the nominal and variable parts of the admittance matrix. The single dimensionless scalar constant K can be extracted as shown because it is common to all components. A value of zero for K corresponds to the nominal circuit.

6.3 Algorithm

The problem of computing the tracking sensitivity of a circuit can be stated as follows: Given a single-frequency nodal admittance description of a circuit in the form of (6.2), determine the transfer impedance $z_{\psi\phi}$ for each of a number of values of K.

The transfer impedance $z_{\psi\phi}$ of interest is an element of the circuit's impedance matrix Z which is the inverse of Y:

$$\begin{aligned} Z &= (Y_N + K\,Y_V)^{-1} \\ &= \left[(I + K\,Y_V\,Y_N^{-1})\,Y_N\right]^{-1} \\ &= Y_N^{-1}\,(I + K\,Y_V\,Y_N^{-1})^{-1} \end{aligned} \tag{6.3}$$

where I is the unit matrix. Writing $Y_N^{-1} = Z$, equation 6.3 becomes

$$Z = Z_N \, (I + K \, Y_V \, Z_N)^{-1} \tag{6.4}$$

We now factorize the matrix $Y_V \, Z_N$ of (6.4) into a product [37] of three matrices, $V \, \Lambda \, V^{-1}$, where V is the eigenvector matrix of $Y_V \, Z_N$ and Λ is a diagonal matrix whose diagonal elements are the eigenvalues $(\lambda_1, \lambda_2, \ldots\ldots, \lambda_n)$ of the matrix $Y_V \, Z_N$. Thus, equation 6.4 can be written as

$$\begin{aligned}
Z &= Z_N \, (I + K \, V \, \Lambda \, V^{-1})^{-1} \\
&= Z_N \left[V \, (I + K \, \Lambda) \, V^{-1} \right]^{-1} \\
&= Z_N \, V \, (I + K \, \Lambda)^{-1} \, V^{-1} \\
&= P \begin{bmatrix} \frac{1}{1 + K\lambda_1} & & & O \\ & \frac{1}{1 + K\lambda_2} & & \\ & & \ddots & \\ O & & & \frac{1}{1 + K\lambda_n} \end{bmatrix} V^{-1}
\end{aligned} \tag{6.5}$$

where $P = Z_N \, V$. If $z_{\psi\phi}$ is the only transfer impedance of interest, the relation appropriate to its calculation is seen from (6.5) to be

$$z_{\psi\phi} = \frac{\alpha_1}{1 + K\lambda_1} + \frac{\alpha_2}{1 + K\lambda_2} + \cdots\cdots + \frac{\alpha_n}{1 + K\lambda_n} \tag{6.6}$$

where

$$\alpha_r \triangleq P_{\psi r} (V^{-1})_{r\phi} \qquad r = (1, 2, \ldots, n)$$

and n indicates the size of the nodal impedance matrix. Fig. 6.1 shows the scheme of this algorithm.

6.4 Eigenvalues and Eigenvectors

We know that what are spoken of as the eigenvalues of a matrix are the zeros of its characteristic polynomial; i.e., the roots of the equation

$$|A - \lambda I| = \begin{vmatrix} a_{11} - \lambda & a_{12} & & a_{1n} \\ a_{21} & a_{22} - \lambda & & a_{2n} \\ & & \cdot & \\ & & \cdot & \\ & & \cdot & \\ a_{n1} & & & a_{nn} - \lambda \end{vmatrix}$$

$$= (-1)^n \lambda^n - p_1 \lambda^{n-1} - \ldots - p_n = 0 \qquad (6.7)$$

In fact, the coefficients p_i are, but for sign, the sume of all principal i^{th} order minors of the determinant of the matrix A. The direct computation of the coefficients p_i is extremely awkward and requires a huge number of operations, particularly when the size of the matrix A is large.

The determination of the components of an eigenvector requires the solution of a system of n homogeneous equations in n unknowns;

in order to compute all the eigenvectors of a matrix one must solve, generally speaking, n systems of the form

$$(A - \lambda_i I) X_i = 0 \tag{6.8}$$

where $X_i^t = (x_{1i}, x_{2i}, \ldots, x_{ni})$ is the i^{th} eigenvector of the matrix A.

It is thus perfectly natural that special computational artifices that simplify the numerical solution of both problems before us should have made their appearance. In response to this need, an efficient iterative method which makes possible the direct determination of the eigenvalues of the matrix without resorting to the characteristic polynomial has been developed. By using it, the eigenvalues and eigenvectors can be obtained at a low cost.

The steps of the proposed iterative method [39,40,41] for the finding of eigenvalues and eigenvectors of a general complex matrix are as follows*:

1. Reduce the matrix to Hessenberg form by similarity transform or Householder method.
2. Find eigenvalues of the Hessenberg form matrix by QR iterations.
3. Find eigenvectors by Wielandt inverse iteration.
4. Recover eigenvectors of input matrix.

The scheme of the above procedure has been shown in Table 6.1. The time chart shown in Fig. 6.2 shows the real time expenditure of this method as it is implemented on a CDC 6800 computer. It is important to note that, for symmetrical matrix (i.e. passive circuits), by using a more efficient algorithm, the time expenditure can be greatly reduced if the eigenvalues and eigenvectors are to be found.

* See Appendix for details.

Since the timing graphs are made with log scale against log scale, the graphs have a tendency to straighten out for large matrix orders and tend towards an asymptote as the cost term involving the highest power of the matrix order becomes dominant. In most of the cases, this term is a cube, giving a terminal slope of three if both log scales are to the same base.

6.5 Computational efficiency

At a single frequency, and following the formation of Y_N and Y_V, one matrix inversion is required for the generation of Z_N. Then, the major computational effort can be divided into two parts.

The initial calculation (which must be repeated if a new choice of the coefficients ξ_i is made) comprises:

1. One n by n matrix inversion (of V) which requires n^3 multiplications.
2. Two n by n matrix products ($Y_V Z_N$ and $Z_N V$) which requires $2n^3$ multiplications.
3. Determination of the eigenvectors and eigenvalues of an n by n matrix ($Y_V Z_N$). The computational cost of this step is illustrated in Table 6.1. For large matrix orders, the cost approximates to $11n^3$.
4. n multiplications to generate the coefficients α_r ($r = 1,\ldots,n$)
5. 2n divisions to arrange (6.6) into a form such that least computational effort is required for the computation of $z_{\psi\phi}$ for a new value of K.

Following the initial computation, the calculation for each value of K comprises only n multiplications to calculate $z_{\psi\phi}$ according to (6.6).

The computational effort (n) associated with each value of K is small enough to permit dynamic exploration of the effect of K on transfer impedance. That is, it is possible to calculate and display the new frequency response curve essentially instantaneously as K is varied manually.

6.6 Exact Method

Basically, the initial computations of the tracking sensitivity of a circuit involve the computation of the coefficients of the characteristic polynomial of a matrix. In the algorithm proposed above, this problem has been transformed into that of finding eigenvalues and eigenvectors of a matrix whereby the well developed efficient iterative computer routines can be used. In some rare occasions, the diagonalization of a matrix may not be possible. As it happens, an exact method was later proposed [38] . In this method, Leverriers method in D.K. Faddeevs modification [42] has been used for the direct computation of the coefficients of the characteristic polynomial and the adjoint of a matrix simultaneously. The algorithm of this method will now be described.

Instead of expressing Z in the form of equation 6.4, Z may be written as

$$Z = (1 / K)\, Z_N \left[(1 / K)\, I - (- Y_V\, Z_N) \right]^{-1} \qquad (6.9)$$

Define $(1 / K) \overset{\Delta}{=} \lambda$ and $- Y_V\, Z_N \overset{\Delta}{=} A$, so that we have

$$Z = \lambda\, Z_N\, (\lambda\, I - A)^{-1} \qquad (6.10)$$

Use Leverriers' algorithm to evaluate, simultaneously, Adj $(\lambda\, I - A)$ and Det $(\lambda\, I - A)$ as follows:

$$\begin{array}{lll}
A_1 \triangleq A & , \; q_1 \triangleq - \text{tr } A_1 & , \; B_1 \triangleq A + q_1 I \\
A_2 \triangleq A \; B_1 & , \; q_2 \triangleq - \frac{1}{2} \text{ tr } A_2 & , \; B_2 \triangleq A_2 + q_2 I \\
\cdot\;\cdot & \cdot\;\cdot\;\cdot & \cdot\;\cdot\;\cdot \\
A_{n-1} \triangleq A \; B_{n-2} & , \; q_{n-1} \triangleq \frac{-1}{(n-1)} \text{ tr } A_{n-1} & , \; B_{n-1} \triangleq A_{n-1} + q_{n-1} I \\
A_n \triangleq A \; B_{n-1} & , \; q_n \triangleq - \frac{1}{n} \text{ tr } A_n & , \; B_n \triangleq A_n + q_n \; I = 0
\end{array}$$

The quantity $a_{11} + a_{22} + \dots + a_{nn}$ is called the trace of the matrix A, and is denoted by tr A. The last identity, $B_n = 0$, provides a numerical check on the algorithm. Then it is well known [56] that

$$\text{Adj}(\lambda \; I - A) = \lambda^{n-1} \; I + \lambda^{n-2} \; B_1 + \dots + \lambda \; B_{n-2} + B_{n-1} \qquad (6.11)$$

and $$\text{Det}(\lambda \; I - A) = \lambda^n + q_1 \; \lambda^{n-1} + \dots + q_{n-1} \; \lambda + q_n \qquad (6.12)$$

Hence

$$Z = \frac{Z_N \left[I + B_1 \; K + B_2 \; K^2 + \dots + B_{n-1} \; K^{n-1} \right]}{1 + q_1 \; K + q_2 \; K^2 + \dots + q_n \; K^n} \qquad (6.13)$$

and

$$z_{\psi\phi} = \frac{\sum_{j=1}^{n} \left[Z_N \right]_{\psi j} \left[I_{j\phi} + \left[B_1 \right]_{j\phi} K + \dots + \left[B_{n-1} \right]_{j\phi} K^{n-1} \right]}{1 + q_1 + q_2 \; K^2 + \dots\dots + q_n \; K^n}$$

$$= \frac{\alpha_0 + \alpha_1 \; K + \alpha_2 \; K^2 + \dots + \alpha_{n-1} \; K^{n-1}}{1 + q_1 \; K + q_2 \; K^2 + \dots + q_n \; K^n} \qquad (6.14)$$

where

$$\alpha_r \triangleq \sum_{j=1}^{n} \left[Z_N\right]_{\psi j} \left[B_r\right]_{j\phi} \quad (r = 0, 1, \ldots, n-1); \quad (B_0 \triangleq I).$$

Here, and throughout, n denotes the size of the nodal admittance matrix Y. Note that α_r and q_r may be evaluated before substituting values of K.

At a single frequency, and following the formation of Y_N and Y_V, one n by n matrix inversion is required for the generation of Z_N. Thereafter, the major computational effort of this method can also be divided into two parts. The initial calculation (which must be repeated if a new choice of the coefficients ξ_i is made) comprises

1. One n by n matrix multiplication (of $Y_V Z_N$) which requires n^3 multiplications.
2. Computation of the coefficients of the characteristic polynomial of an n by n matrix $(- Y_V Z_N)$ by Leverriers algorithm which involves (n - 1) matrix multiplications, i.e., $(n - 1) n^3$ multiplications. This figure tends towards an asymptote of n^4 for large matrix orders.
3. (n - 1) inner products, i.e. $(n^2 - n)$ multiplications to generate the coefficients α_r (r = 1, 2, ..., n-1).

Following this, the calculation for each value of K requires a total of (2n - 1) multiplications to evaluate the numerator and denominator polynomials, followed by one division - a total of 2n operations. This takes twice as much computational effort as that required by the previous method. The scheme for the computation of tracking sensitivity using the exact method is shown in Fig. 6.3.

6.7 Engineering applications

The advantage of the efficient approach to tracking sensitivity analysis has been exploited to good effect in the interactive graphic circuit design facility [8] with the global variable designated as ambient temperature of the components.

In the process of the interactive graphic circuit design, once the circuit has been input satisfactorily and displayed on the screen, the designer may activate the appropriate light button for the tracking sensitivity display. He then indicates the components which are susceptible to temperature variations and, at the same time, specifies the corresponding temperature coefficients of these components. After a time lapse needed for the computation of the eigenvalues and the eigenvectors and relevant coefficients for a number of discreted frequencies composing the response curve, the computer is ready to display the effect on the voltage response of the temperature variations.

The magnitude and phase response curves are now displayed on the screen with solid curves representing the nominal response, and dotted curves representing the deviated response due to temperature variations. A light potentiometer appears on the right hand side of the screen. The arrowed position indicates the temperature under which the deviated response curves are evaluated. Due to the small number of operations associated with each value of the global variable (temperature), following the initial computation, new dotted response curves are displayed roughly half a second after the pointer on the potentiometer has moved to a new position, manually or automatically.

The circuit example of a transistor amplifier shown in Fig. 6.4 has been chosen for illustration. The effect on the output response of ambient temperature variation of this circuit is to be investigated. All components of this circuit are assumed susceptible to temperature variations and assigned with the appropriate temperature coefficients. Fig. 6.5(a) and Fib. 6.5(b) show the output voltage response curves of this circuit at two different ambient temperatures. The rapid display of the new response curves under the new temperature enables the designer to explore the temperature effect dynamically.

Method	Work per iteration	Convergence properties	Cost	Achievement
Householder			5/3 n^3	matrix → Hessenberg
QR-iteration	4 n^2	cubic	approx. 8 n^3	find all eigenvalues of Hessenberg matrix
Wielandt inverse iteration	$\frac{1}{2}$ n^2	in two passes	n^2	finds all eigenvector given an eigenvalue
Eigenvector recovery			n^3	recover all vectors after Householder reduction to tridiagonal

Table 6.1.

Approximate costs associated with the four steps of computing eigenvalues and eigenvectors by iterative method.

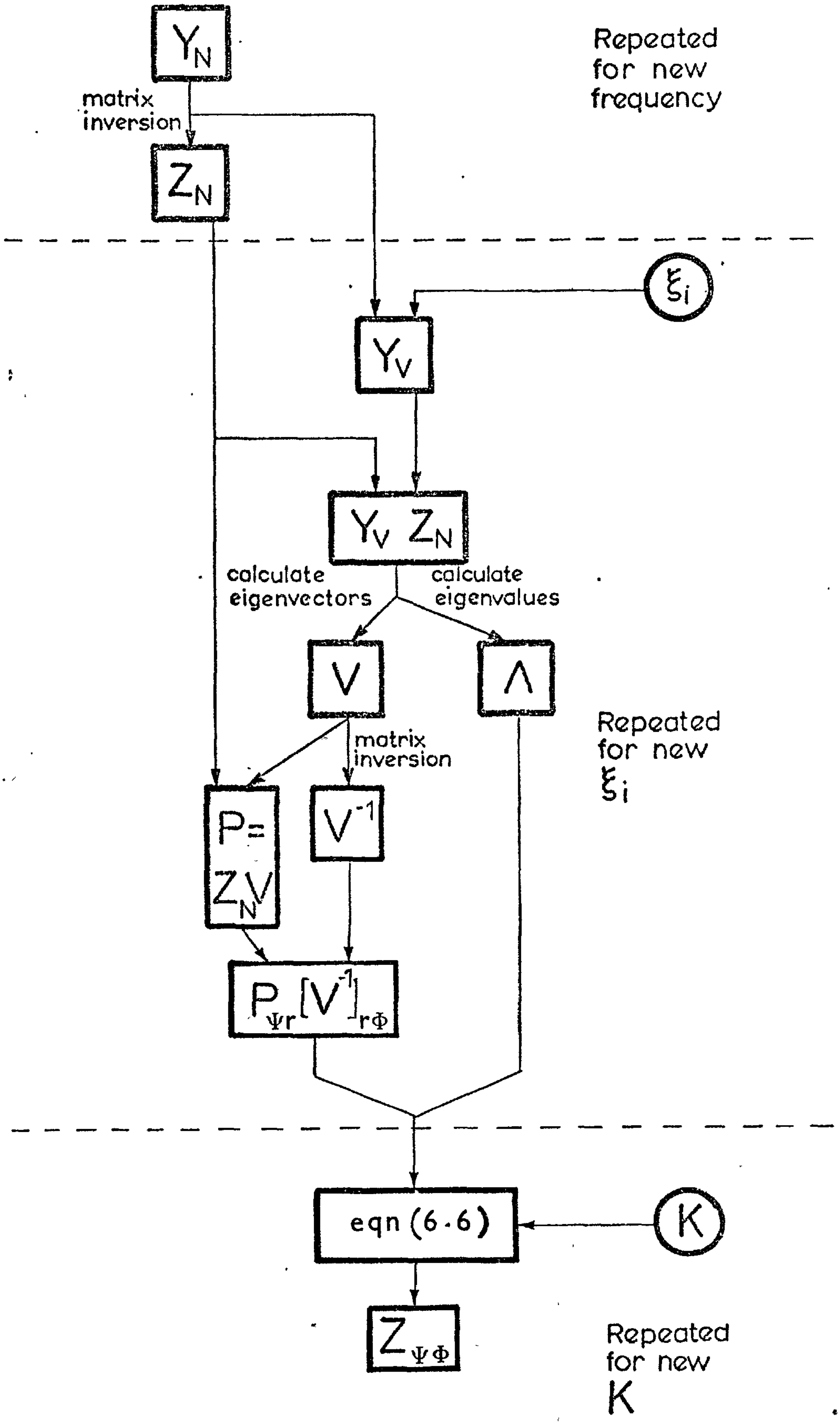


Fig. 6.1. Scheme for the computation of tracking sensitivity

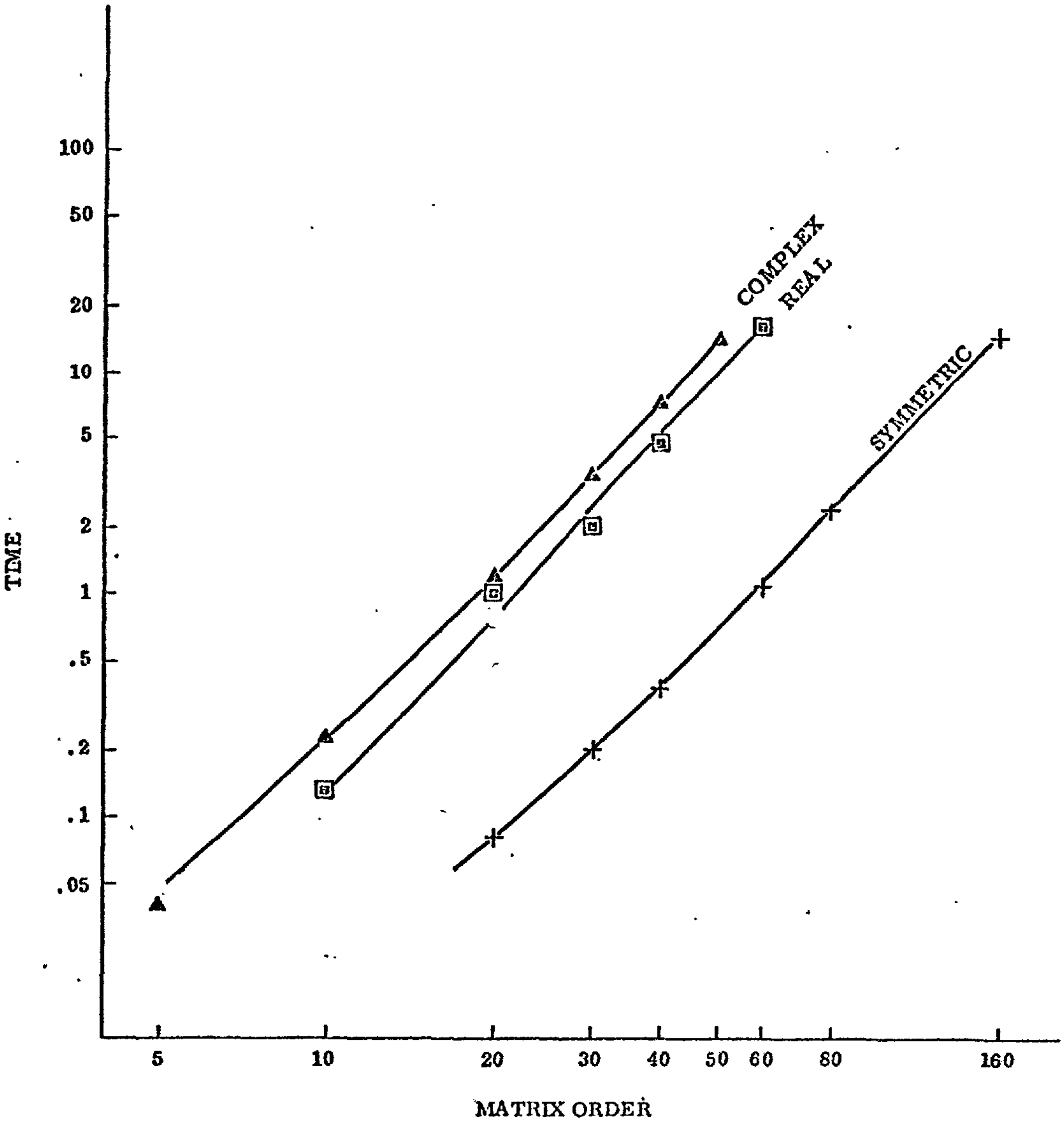


Fig. 6.2. Timing chart showing the real time expenditure of the iterative method for the computation of all eigenvalues and eigenvectors of a complex, real or symmetric matrix.

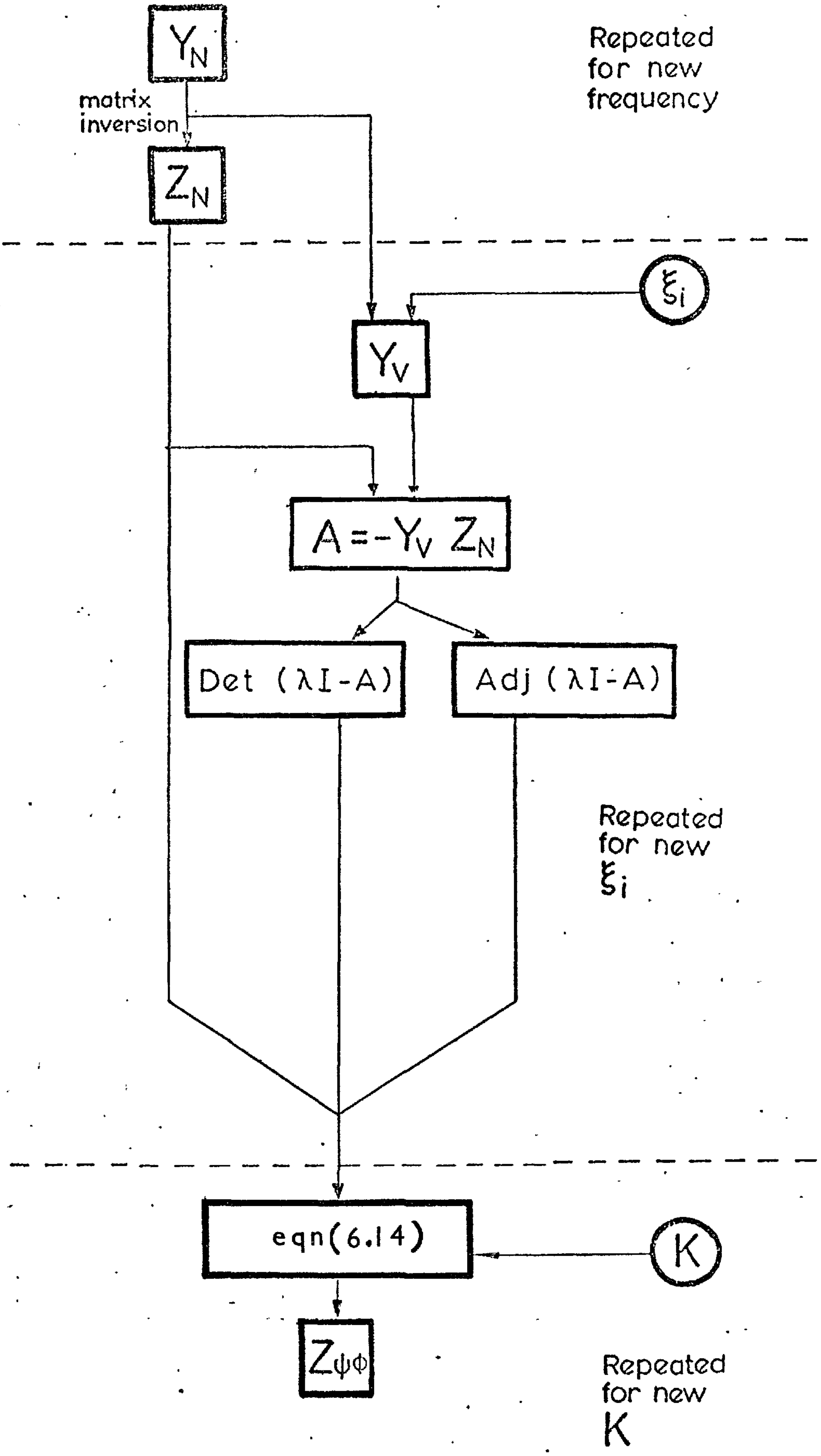


Fig. 6.3. Scheme of the exact method.

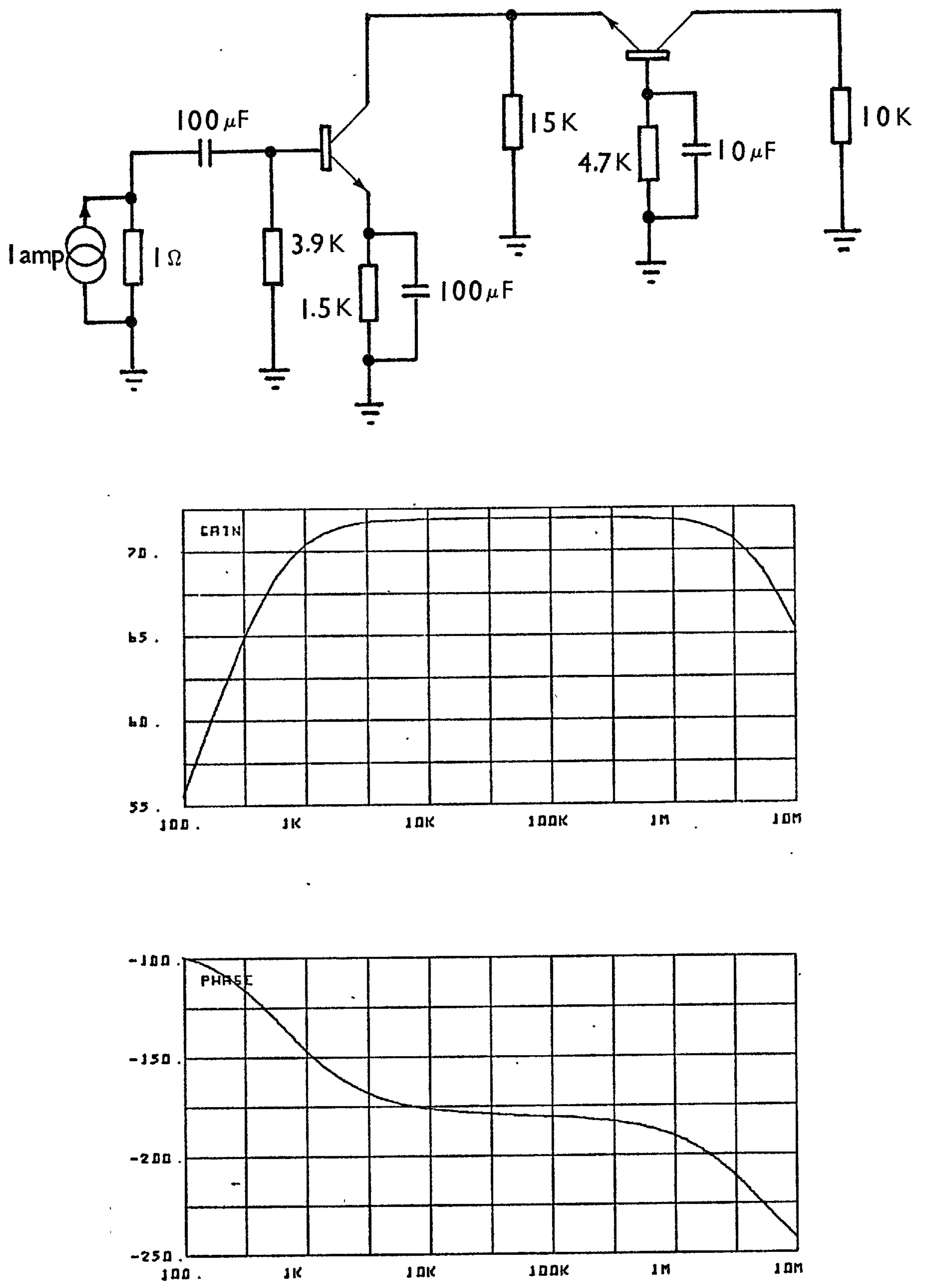


Fig. 6.4. A transistor amplifier subjected to temperature variation and its nominal response curves.

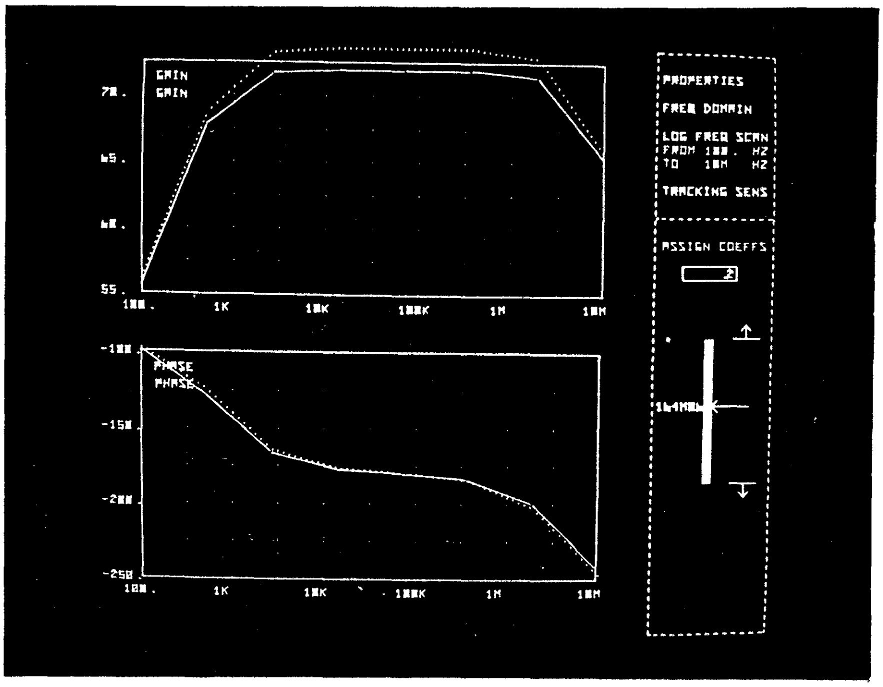


Fig. 6.5(a). Interactive display associated with dynamic exploration of the effect of temperature variation on circuit response. Temperature variation is simulated by moving an activated light-pen along the light-potentiometer shown at right. The new (dotted) response curve is displayed on the same graph as the (full) nominal curve.

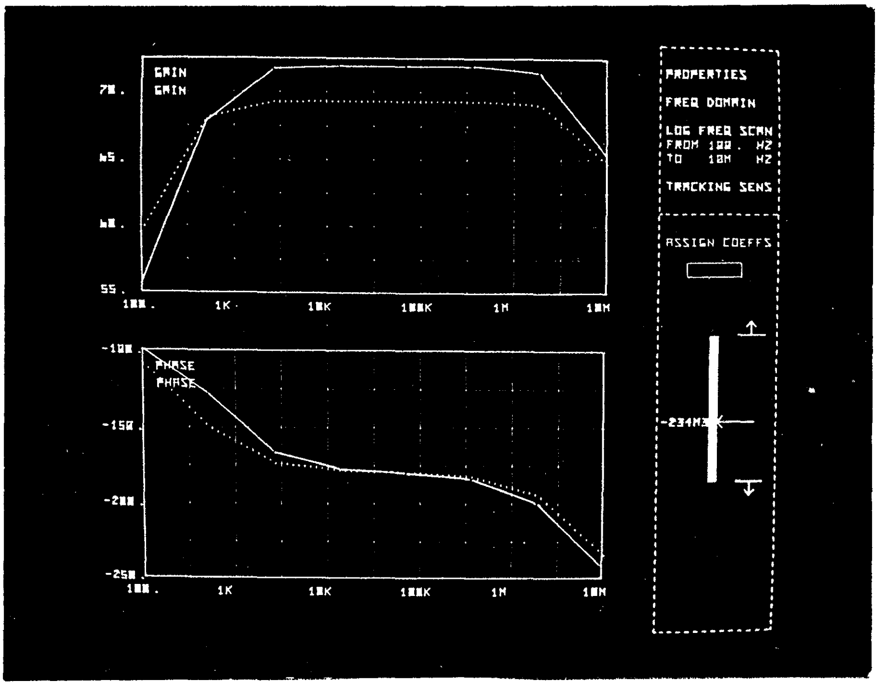


Fig. 6.5(b). Another new response curve is displayed in less than a second after the temperature has been varied to a new value.

CHAPTER 7

STATISTICAL CIRCUIT ANALYSIS

7.1 Introduction

Circuits are manufactured from components of specified tolerances. Therefore, in the final stages of circuit design, it is necessary to test the 'realistic' circuits to ensure that the design requirements are met. A popular technique of testing the 'realistic' circuit is the computation of worst-case circuit performance. Although worst-case analysis has been widely used in the past, it is known to be unnecessarily conservative in setting component tolerance limits, and its use can result in increased production costs. On the other hand, a statistical analysis [43,44] sometimes known as Monte Carlo analysis, helps to assure both acceptable manufacturing yield and minimum production costs.

Basically, the computational approach used in a Monte Carlo analysis is to evaluate the performance of a simulated circuit many times, introducing at each iteration random variation of a specific probability distribution to the component values. The performance values are collected to form histograms of the frequency and cumulative distributions of the data. The process is continued until the distribution is an adequate representation of the anticipated statistical behaviour of the circuit.

In a Monte Carlo analysis, the statistical performance is directly measured by applying random variations with known distributions to the circuit components. For this reason, the estimates of performance are more realistic than those obtained by worst case analysis in which only the extreme values are used in evaluating performance*. Although Monte Carlo analysis has all the advantages mentioned previously, they are usually outweighed by the computational cost of acquiring sufficient data to determine distributions that are representative of the entire population. Therefore, alternatives which are far less costly, even if approximate are being sought.

In this chapter, we examine one statistical technique known as 'regionalization' [4,45] , and show how a further and often very considerable increase in efficiency can be secured by invoking the concept of an idealized statistical model [45] and exploiting the technique of systematic exploration [32] . The feasibility of the new approach, as well as its relevance to highly interactive computer-aided design, is illustrated by an example.

7.2 Monte Carlo Analysis

The structure of a Monte Carlo analysis is given in Fig. 7.1. The function of the blocks will now be described.

1. Input of circuit data.

The circuit and initial data are supplied to the computer through a suitable computer dialogue and subsequently may be modified by the designer. The variable components of the circuit to be investigated are characterized by their nominal values and probability distributions, the latter including tolerance limits. Components without tolerances may be considered as constants or as components with zero tolerances.

* It should be noted that although a worst case may generally be assumed to occur at a set of extreme values, a set of extreme values will not necessarily result in a worst case.

2. Random variable sources.

Random number generators are generally used in a Monte Carlo analysis as the sources of random numbers. Initially, the distribution of a random number generator in a computer is uniform in shape. Gaussian and other distributions may be obtained by reshaping the uniform distribution into the shape required.

The random numbers initially generated in a computer are so scaled that the variates are in the range -1 to +1. For Gaussian or other distributions, it is necessary to rescale the range so that unity corresponds to a selected number on the abscissa of the distribution and the distribution may be truncated outside a specified range. Fig. 7.2(a) shows the initial uniform distribution of the random number generator of a PDP-15 computer and Fig. 7.2(b) shows the Gaussian distribution obtained by reshaping the initial distribution. In this example, the Gaussian distribution has been truncated and normalized so that 2σ corresponds to unity.

3. Perturbation of component values.

The variations in component values are a function of tolerances and the random distribution. The perturbed value η for each component is computed according to

$$\eta = \eta_0 (1 + kx)$$

where η_0 is the nominal component value, k is the specified tolerance and x is the normalized random number. The set of perturbed values for all variable components is then used directly in the computation of the circuit performance. Fig. 7.2(b) also shows the relationship between the component deviation and the normalized deviation of the Gaussian distribution.

4. Evaluation of circuit performance.

The effect of the component tolerances on the circuit performance is evaluated by analysing the circuit for each set of variable component values. As it happens, the analysis algorithm has to be performed repeatedly as many times as the number of trials in a Monte Carlo analysis. Therefore, this block is the major computational time consumer of the whole process and should be made as efficient as possible in order that the computational cost of a Monte Carlo analysis is kept to a minimum.

5. Data process and display.

The various values of circuit performance obtained from the repeated analysis are accumulated and processed to form frequency and cumulative distribution histograms. The frequency distribution contains the relative occurrence of samples in each class interval of the selected range, while the cumulative distribution contains the proportion of the samples that are found below each value of the performance deviation.

The frequency or cumulative distribution histogram is displayed graphically or printed on paper. The abscissa of the display represents the performance deviation, and is labelled with its minimum and maximum values. The ordinate of the frequency distribution is normalized with respect to the maximum class interval count. This normalization was chosen so that the maximum amplitude is always unity and not a function of the total number of samples. Therefore, the variance of distributions obtained for different component tolerances can be compared without renormalization of the peak values. The cumulative distribution is normalized with respect to the total count and it enables one to determine rapidly what percentage of the designs produce acceptable values of the performance deviation.

6. Man-machine interaction.

In a Monte Carlo analysis, the end product is the frequency or cumulative distribution histogram whose use in the prediction of manufacturing yield can only be justified when the distribution is an adequate representation of the anticipated statistical behaviour of the circuit. Therefore, it is crucial for the designer to terminate the analysis at the right time in order that a reasonably good representation of the circuit's statistical behaviour can be obtained at low computational cost. If interactive facilities are provided for a computer aided circuit design, the distribution histogram may be monitored and updated as the Monte Carlo analysis is in progress. By so doing, the designer is able to stop the computation immediately when he is satisfied with the result [3] .

7.3 Regionalization

In the Monte Carlo analysis described in the preceding section, the simulated circuit is re-analysed for each set of perturbed component values. Usually, the number of trial vectors for a Monte Carlo analysis is very large, so that the computational cost of this method is very high indeed. For a linear circuit with m variable components, even if the large change sensitivity algorithms described in Chapter 4 are used, each trial would normally involve approximately the cost G_{m+2} for Gaussian elimination, or G_m for the current substitution method. In view of the excessive cost, alternatives which are far less costly, even if approximate, are being sought.

In view of its proven high efficiency in sensitivity analysis, it was decided that systematic exploration should be used to increase the efficiency of a statistical circuit analysis. In order for systematic exploration to be applied to statistical circuit analysis, each variable component should assume one of a limited number of values and each set of m variable component values should be a combination of these. To this end, the technique termed 'regionalization' [4,45] is invoked.

Briefly, the concept of regionalization can be illustrated by a simple example. Consider a circuit in which two toleranced components of values x_1 and x_2 are subject to spreads defined by continuous probability distribution functions (PDF) P_{x_1} and P_{x_2} respectively: the two PDF's and the 2-dimensional variable component space may appear as in Fig. 7.3(a) whose Monte Carlo samples are represented by points. By regionalization, the two distributions of the two variable components are quantised into 5 class intervals, and the data within each of the 5 class intervals are discretized to the centre of their own class interval. As a result, each variable component can now assume one of a limited number (5) of values and each set of 2 variable component values is a combination of these. This fact may be geometrically interpreted as: The variable component space is regionalized into 25 regions and all points that may exist in a region are modelled by a single point, as illustrated in Fig. 7.3(b).

In general, regionalization has the following two important features:

1. The variable component space is partitioned into a finite number of non-overlapping regions. In order to perform the partition, the individual frequency distributions of the variable components must be quantised into a finite number of class intervals. As a result of quantisation, the variable component space is now partitioned into M regions where

$$M = Q_1 \cdot Q_2 \cdot \cdots \cdot Q_m \tag{7.1}$$

In the above expression, m is the number of the variable components of the circuit and Q_i $(i = 1,..,m)$ is the number of class intervals into which the individual frequency distribution of the i^{th} variable component is quantised. If the Q_i's of the variable components are identical, the number of regions can be computed by the equation

$$M = Q^m \tag{7.2}$$

2. All points (i.e., trial vectors) that may exist in a region are modelled by a single point. In order to do so, the random data within each class interval of a quantized distribution are discretized to an unique value within the corresponding class interval. As a result, each region is characterized by an unique set of m values denoted x_i' $(i = 1,...,m)$. This set of values may also be called the coordinates of the point which represents the region. The point, denoted p, can be expressed as

$$p = (x_1',x_2',...,x_m') \tag{7.3}$$

7.4 Linear Circuits and Systematic Exploration

1. Processing of trial vectors.

Systematic exploration requires that, for statistical circuit analysis, the trial vectors should be generated and processed before the circuit analysis takes place. The procedure for processing trial vectors is as follows:

a) According to the probability distributions specified for the variable components, the random number generators generate a number, t, of trial vectors before the analyses of the circuit take place. The trial vectors, denoted T_j, can be expressed as

$$T_j = (x_{1j}, x_{2j}, \ldots\ldots, x_{mj}) \qquad (7.4)$$

where $j = 1, \ldots, t$, and where X_{ij} are perturbed normalized variable component values. After all trial vectors have been generated, an m by t matrix is formulated (Fig. 7.4(a)). This matrix is denoted X_{ij} and each column represents a trial vector in a statistical analysis.

b) By regionalization, the randomly generated trial vectors T_j are represented by a single point p_j within their associated regions. The points p_j can be expressed as

$$p_j = (x'_{1j}, x'_{2j}, \ldots\ldots, x'_{mj}) \qquad (7.5)$$

where $j = 1, \ldots, t$, and where x'_{ij} are the coordinates of the point p_j. As a result, the trial vectors in matrix X_{ij} were replaced by p_j; therefore, a new matrix, denoted X'_{ij} is formulated* (Fig. 7.4(b)).

c) The columns of the matrix X'_{ij} are then ordered into a sequence which is best suited for systematic exploration. Briefly, the columns of the matrix are so ordered that row 1 undergoes the least number of changes and, for each value in row 1, row 2 undergoes the least number of changes, and so on to the last row.

* In fact, steps a) and b) can be combined in a statistical analysis process. Therefore, only X'_{ij} is actually generated and stored.

As a result of this operation, matrix X'_{ij} is transformed into another matrix denoted X''_{ij} as shown in Fig. 7.4(c) and this matrix is the starting point of the systematic exploration.

It is important to note that, by regionalization, a certain amount of repetition in p_j (trial vectors) would exist in the matrix X'_{ij}. The repetition will increase as the number of trials grows bigger or as the variable component space is partitioned into fewer regions. By ordering the trial vectors for systematic exploration, the identical trial vectors would automatically be grouped together (an unique trial vector is a group of its own). After one of the trial vectors of each group has been investigated, the rest of the trial vectors of this group can be bypassed without the need for any computational effort. Consequently, the number of trial vectors is effectively reduced. To account for the repetitive trial vectors, the number of trial vectors of each group is counted and used to multiply the statistics obtained from the investigation of a single trial vector of the same group. Since each group of identical trial vectors belongs to an unique region in the variable component space, the number of trial vectors of each group may also be termed the weight of a region.

2. Computational efficiency.

Consider a linear 2-port containing a number of components of which m variable components are subject to change. Let a port be created across each variable component (Fig. 4.1(a)), thereby defining an (m + 2)-port circuit described, at a single frequency, by a port admittance matrix Y_p or its inverse, the port impedance matrix Z_p. The transfer impedance $z_{\psi\phi}$ between the original ports ϕ (input) and ψ (output) is the circuit property of interest. Each set of m component values constituting a trial can be simulated by the connection of appropriate admittances Δy to the newly created ports. The calculation of $z_{\psi\phi}$ for each trial can therefore be obtained by the appropriate modification of the (m + 2) x (m + 2) port admittance matrix Y_p at negligible cost, followed by Gaussian alimination at a cost of G_{m+2}. If the number of variable components is sufficiently less than the number of circuit nodes, the cost G_{m+2} can be less than G_n. This method of analysis will be adopted as a reference for purposes of comparison with the more efficient approach employing systematic exploration now to be exploited.

For a given number m of variable components, each of which can assume any of Q values, the circuit data describing t trials were generated randomly. Thus, for 400 trials in which five variable components could each assume any of seven values, a 5-row 400-column matrix containing a random distribution of the integers from 1 to 7 (X''_{ij}) was generated after the trial vectors had been processed. The number of value changes in each row then determines the total cost C of computing $z_{\psi\phi}$ for all trials (See Section 4.6). For some values of m and t, the full curves of Fig. 7.5 show the saving t G_{m+2}/C in computational cost over the reference method as a function of Q, and the broken curves indicate the saving t G_m/C solely by the discretisation of component values and the ordering of trials. Particularly for low Q, it is clearly possible to obtain a useful economy in computation.

3. Example.

The effect of regionalization was tested by means of a Monte Carlo analysis of a bandpass filter (Fig. 7.6) at a single frequency

(800 Hz). The five (most sensitive) components so indicated were assumed variable according to a truncated Gaussian probability distribution (σ = 2.5%, limit = 5%). For 1000 trials, Fig. 7.7 shows the frequency distribution and cumulative distribution of the insertion loss for no regionalization, other than that inherent to the computer, and with Q equal to ten and six. For the regions where the insertion loss deviation is less than -3 db, the maximum discrepancy between the three cumulative distributions is 3% at a horizontal resolution of 100. In this example, the sample data have been discretized to the centre of the class intervals of each individual distribution. These results appear to indicate that regionalization of component space need not lead to a misleading result. The example also shows that an acceptable prediction of trials can be obtained at low cost.

7.5 Idealized Statistical Model

1. Definition.

In a conventional statistical circuit analysis, the performance of a simulated circuit is evaluated repeatedly introducing at each iteration random variation of a specific probability distribution to the variable components. Repetition continues a) within the limits of available computational resources or b) until what is deemed to be satisfactory approximation to the anticipated statistical behaviour of the circuit has been obtained. Special attention should be given to conditions a) and b), since they are fundamental to most approaches to statistical analysis; typically, a) is concrete, whereas b) is somewhat ill-defined. If the number of analyses is limited, an evenly regionalized variable component space with the number of samples set equal to the permitted number of analyses may constitute a satisfactory sampling model for a statistical circuit analysis. Referring to the example shown in Fig. 7.3(b), the evenly regionalized two dimensional variable component space may be considered as a satisfactory sampling model for a statistical circuit analysis with a limited number of analyses of 25 if the regions are properly weighted.

Two method are available for the assignment of weights, and are conveniently described in the context of this 2-variable example. In one, points in the (x_1,x_2)-plane are generated by a random process according to the probability distributions of the individual variable components, denoted P_{x_1} and P_{x_2} respectively, and any stated interdependence, until what is judged to be a sufficient number has been obtained (Fig. 7.3(a)). The points in each region are then counted to find the weight to be associated with the point representing the region (Fig. 7.3(b)). In the second method proposed, the weights are computed directly from the component probability distributions without recourse to a random number generator. Referring to Fig. 7.8(a), if the two components are independent, the weight of point $(x_1[2], x_2[3])$ is set equal to the product $P_{x_1}[2]\Delta x_1 \times P_{x_2}[3]\Delta x_2$. The resulting representation of the variable component space by points whose weights are so generated will be called the <u>idealized statistical model</u> [5,46] of the circuit (Fig. 7.8(b)).

2. Properties.

In general, the idealized statistical model appears to have the following features when compared with the conventional Monte Carlo model of Fig. 7.3(a).

a) With proper choice of regions, the statistical information content of each analysis can be increased, so that the number of analyses can be reduced. In conventional Monte Carlo analysis, points tend to cluster around the nominal, which is not usually the area of greatest interest. With the idealized statistical model, confidence in the prediction of the statistical behaviour of a circuit can to some extent be controlled. For example, the prediction of manufacturing yield can be made pessimistic by locating the representative point at the 'worst' vertex of its region.

b) The preceding example and other experiments suggest that a relatively even sampling of the variable component space is likely to yield a satisfactory approximation to the anticipated probability density function of the performance of the circuit, even when Q is at a low value. The higher the value of Q, the better is the approximation.

c) Analysis associated with the regionalized variable component space permits a mapping of the relation between circuit failure and component combination [45] . The mapping of failures into an idealized statistical model also permits identification of an approximation to the region of acceptability [1] . Such results are valuable for the insight they provide, and for the assistance they can render in tolerance assignment and design centering.

d) As the number (m) of variable components and the number (Q) of quantization levels increase, the number of regions (Q^m) - and hence the computational cost - increases quite rapidly. Nevertheless, by using systematic exploration, the cost can be remarkably low for linear circuits.

3. Example.

To examine the computational effort and the effect of the quantization inherent in an idealized statistical model, an algorithm based on systematic exploration was implemented within an interactive graphic circuit design facility.

For the circuit example of Fig. 7.9, the components indicated as being subject to spread were each assumed to be described by a truncated Gaussian PDP ($\sigma = 0.5\%$, limit = ± 1%). At a frequency of 36.16 KHz conventional Monte Carlo analyses* were carried out for reference purposes. At the same frequency an idealized statistical circuit model was analysed for a number of values of Q.

The frequency distribution and cumulative distribution of output voltage, as predicted by Monte Carlo analyses of different sample sizes, is illustrated in Fig. 7.10 to 7.13. The frequency distribution and cumulative distribution of output voltage, as predicted by the analysis of idealized statistical models of different Q values, is illustrated in Fig. 7.14 to 7.17. The computing time associated with the analysis and the construction of the histogram is indicated in each case.

* Each sample of the Monte Carlo analysis involved a conventional analysis of the complete circuit of Fig. 7.9. No advantage was taken of the fact that, since only four components were variable, a smaller circuit could be subjected to repeated analysis (see reference [32]). Nevertheless, this advantage is not substantial for the illustrative example: it can be calculated that the ratio of the 'per-trial' computational cost for the straightforward Monte Carlo analysis, the repeated analysis of a smaller circuit, and the analysis based on systematic exploration is, approximately, 161:65:1.

In order to make a comparison between the results obtained from the Monte Carlo analysis and the results obtained from the analysis of idealized statistical models, the cumulative distribution curve obtained from the 5000-sample Monte Carlo analysis (requiring 4600 seconds*) is superimposed on all other cumulative distribution curves for reference purpose. It is interesting to note that for Monte Carlo analysis, even for the 1000-sample analysis (requiring 920 seconds), the resultant cumulative distribution curve is not yet a good approximation to the reference curve. On the other hand, the cumulative distribution curves of the idealized statistical models exhibit good general agreement with regard to shape, and involve much less computing-time: from 3 seconds for Q = 3 to 304 second for Q = 10. A similar conclusion can be drawn if a comparison between frequency distribution curves is made. In Fig. 7.18, frequency distributions predicted by Monte Carlo analyses of 3 different sample sizes and by idealized statistical models of 3 different Q values are brought together for ease of comparison.

For the analysis of an idealized statistical model, any discrepancy+ resulting from a low Q value must, of course, be weighed against the remarkable difference in computing times. For Q = 3, the result can be presented essentially immediately following its request, can provide the designer with some feel for the general nature of the cumulative or frequency distribution curve. In highly interactive circuit design, the availability of such a curve on demand could be of great value.

The above results provide very strong evidence that the concept

* These times refer to a PDP-15 without floating point processor.

+ This discrepancy, in fact, can be smoothed to a certain extent by interpolation [53] .

of an idealized statistical model based on regionalization, combined with the technique of systematic exploration, offers considerable potential for the low-cost yield analysis of linear circuits.

7.6 Worst Case Simulation

As a result of regionalization, a certain amount of discrepancy is introduced into the resultant probability distribution of a statistical analysis. The lower the number of regions into which the variable component space is regionalized, the higher the discrepancy could be. If the variable component space is separated into a reasonably large number of regions then, by using the central point to represent a region, a relatively unsophisticated interpolation would lead to a good approximation to the anticipated performance probability density distribution.

However, if the variable component space is separated into a very small number of regions, interpolation may not produce a highly accurate result. In such a case, it may well be preferable to locate p_j within a region in such a way that the representation of the region by p_j would lead to a pessimistic rather than optimistic effect on the circuit statistics. The reason is that a pessimistic prediction would produce an unnecessary conservative and expensive design. However, such a design is reliable, whereas an over optimistic result would produce unreliable circuits.

A scheme which positions point p_j to a vertex of a region to produce a pessimistic effect on the statistics of circuit performance will now be described. The approach is based on the assumption that for each variable component, within each class interval of its quantised distribution, the sign of small change sensitivity remains unchanged for the whole interval and is not affected by the large changes of the other variable components. The steps of the procedure are as follows:

1. By quantisation, each individual frequency distribution of the variable components is quantised into Q_i class intervals (Fig. 7.19(a)).

2. The nominal Z matrix is updated to absorb the effect of the large change (mid-point of a class interval) of one variable component alone, and then the sign of small change sensitivity of this component at the corresponding mid-point is computed. The procedure repeats until the signs of small change sensitivity at all mid-points of all individual frequency distribution have been computed (Fig. 7.19(b)).

3. The sign of small change sensitivity is used as an indicator for the discretization. As a rule, the random data are discretized to the right hand side edge of a class interval if the sign of the small change sensitivity at the mid-point is positive, or to the left hand side edge of a class interval if the sign of the small change sensitivity at the mid-point is negative (Fig. 7.19(c)).

The procedure is performed only once at the outset before discretization. It should be noted that in the above procedure, when the sign of small change sensitivity of a component is computed, only the large change of this component itself is taken into consideration and this might not be realistic enough to produce a pronounced pessimistic effect on the statistics of the circuit performance. If a more definite pessimistic effect has to be guaranteed when the sign of small change sensitivity of a component is computed, not only the large change of the component itself but also the large changes of the rest of the variable components have to be accounted for. In such a case, a higher computational cost will be incurred*.

7.7 Mapping of Failures

By regionalization, the variable component space is separated into a limited number of regions, the mapping of failures into it becomes an easier task. With p_j denoting a trial vector, $z_{\psi\phi_j}$ is the response. It is assumed that $z_{\psi\phi_j}$ can be classified as a success or failure by some rule. If we let $\tilde{z}_{\psi\phi_j}$ denote a circuit failure, any region whose trial vector p_j generates a $\tilde{z}_{\psi\phi_j}$ can be marked. As it happens, the failures have been mapped into the variable component space.

By developing failure regions in a regionalized variable component spaoe, the following information is obtainable by a file retrieval procedure:

1. The trial vector that produced the corresponding failure.
2. The total number of failures produced after all trial vectors have been analysed.

For an idealized statistical model, every region with nonzero probability density is assigned a nonzero weight. Therefore, after its trial vectors have been analysed, the variable component space

* If the worst vertex is determined by direct comparison of circuit response at different vertices, using systematic exploration, the maximum computational cost is 8 per vertex for each region [54] .

can be considered to have been thoroughly investigated. As a result, the mapping of failures into an idealized statistical model can provide not only numerical information but also geometrical information - the generation of a regionalized region of acceptability.

There are four significant points about this approach for the generation of the region of acceptability.

1. The approach does not rely on first or second-order approximations. No attempt is made to approximate measures of performance.

2. The only approximation made for the approach is the regionalization of the variable component space. The larger the number of regions into which the variable component space is separated, the better is the approximation.

3. The regions which constitute the region of acceptability are weighted.

4. By superimposing failures of different specifications into one area, the approach can accommodate multiple specifications.

The regionalized region of acceptability and its ability to accommodate multiple specifications can be illustrated with a simple example. Consider the two component voltage divider shown in Fig. 7.20(a) where $R_1 = R_2 = 1$. The transfer function T is given by $T = 1/(R_1/R_2 + 1)$, $T = 0.5$. The input resistance R is given by $R = R_1 + R_2$, $R = 2$. The two resistors are assumed variable according to two truncated Gaussian probability distribution ($\sigma = \pm 10\%$, limit $= \pm 20\%$). Suppose the design specifications call for $0.46 \leq T \geq 0.53$ and $1.85 \leq R \geq 2.15$. Then the regions in the idealized statistical model where the first specification is not met are shaded and the regions where the first specification is met but the second specification is not are cross-hatched. It is important to note that in order to mark the regions where the second specification is not met, only the unshaded regions generated by the analysis of the first specification need to be investigated. In this way, a great deal of computational cost can be saved if multiple specifications have to be handled. Fig. 7.20(b) shows the marked idealized statistical model of this example.

So far, statistical circuit analysis is an open-loop structure in the sense that for specified probability distributions of variable components, only the probability distributions of the performance of the circuit is found. The inverse problem - that of using the result of a statistical circuit analysis for the desensitization of a circuit - has not been given sufficient attention. By modelling the variable component space with an idealized statistical model and mapping the failures into it, a weighted region of acceptance is established in the variable component space after a statistical analysis has been performed. A region of acceptability can be used for desensitizing a nominal design, specifying tolerances, recognizing the need for tu ning etc.... In this way, the loop around statistical circuit analysis can be closed.

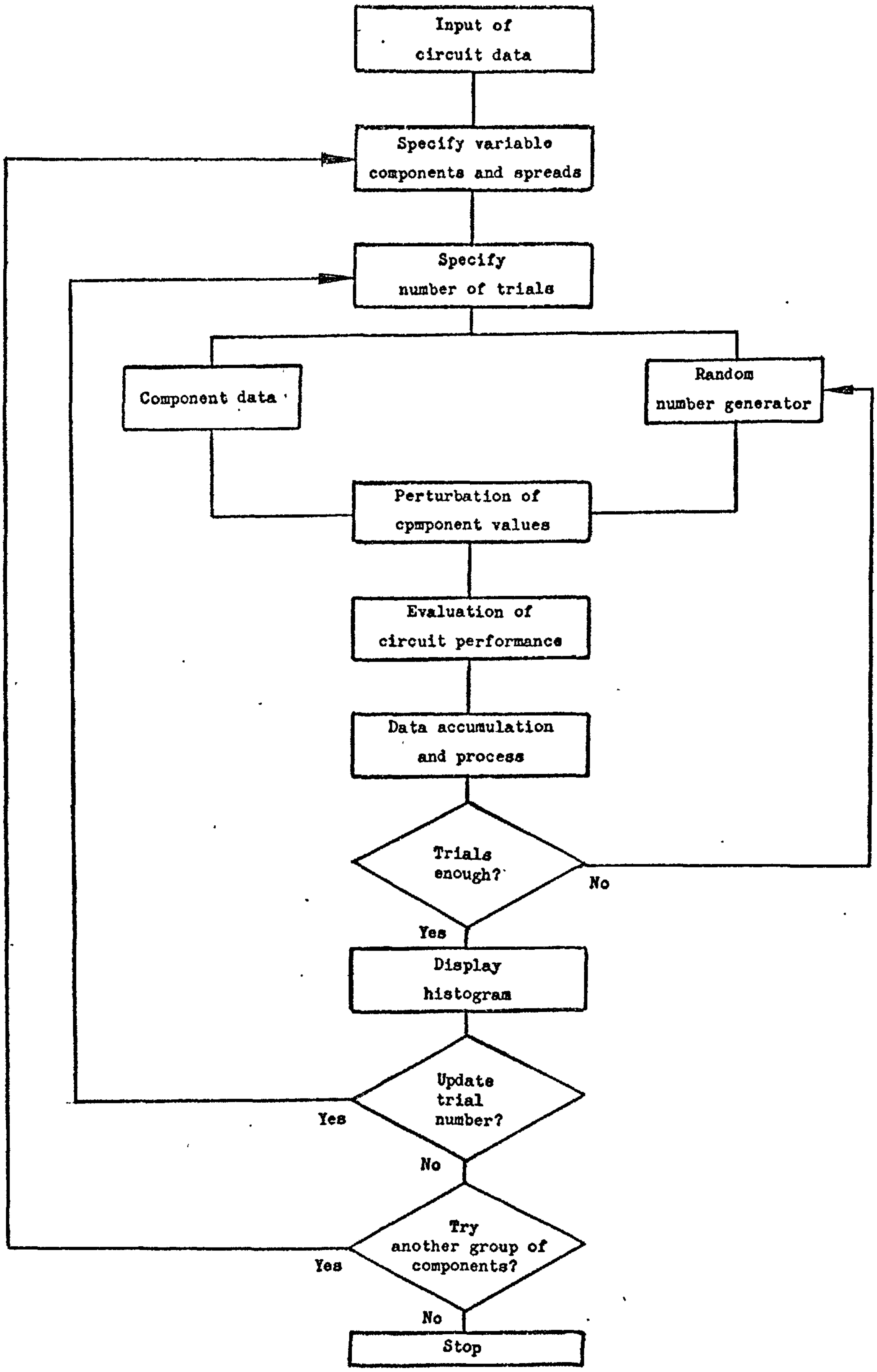

Input of circuit data
Specify variable components and spreads
Specify number of trials
Component data
Random number generator
Perturbation of cpmponent values
Evaluation of circuit performance
Data accumulation and process
Trials enough?
No
Yes
Display histogram
Update trial number?
Yes
No
Try another group of components?
Yes
No
Stop

Fig. 7.1. Interactive Monte Carlo analysis scheme.

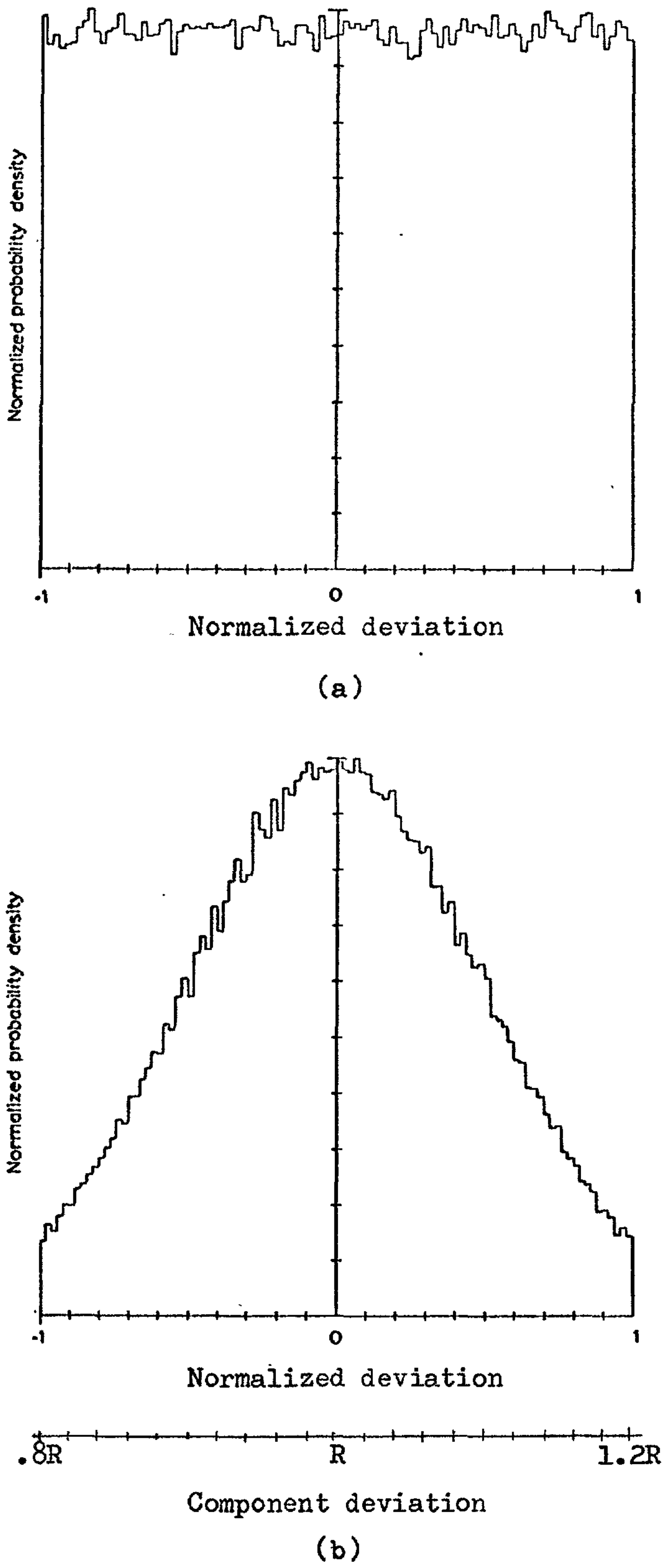


Fig. 7.2. Random distributions measured with 50,000 samples. (a) Uniform distribution of the generator. (b) Gaussian distribution obtained by reshaping the uniform distribution in (a).

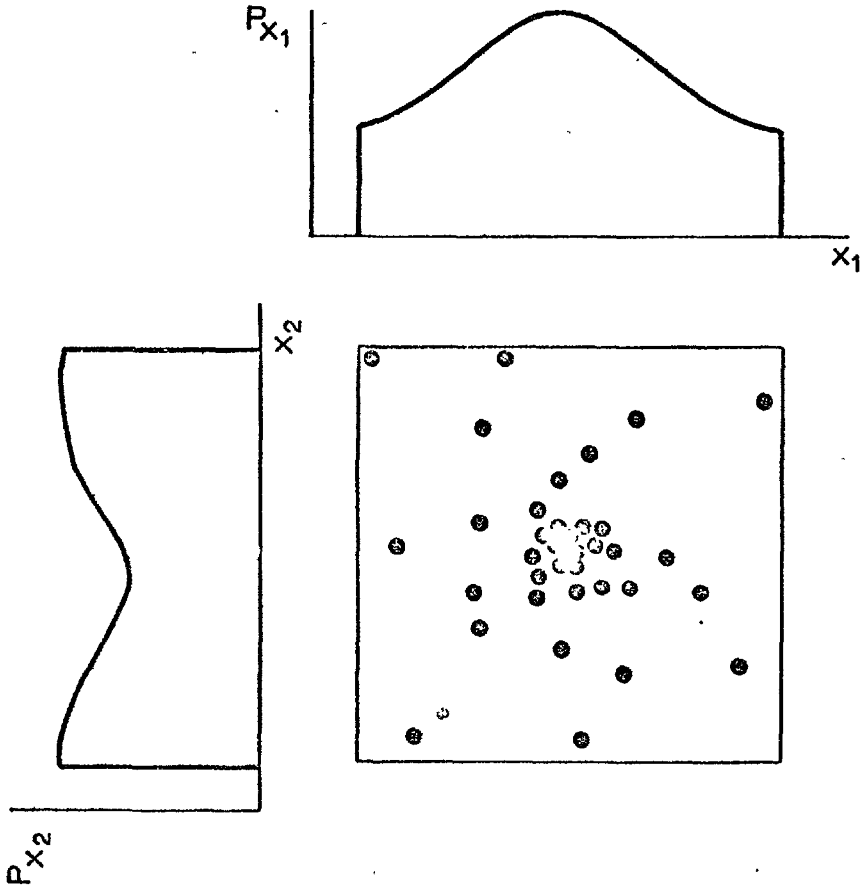


Fig. 7.3(a). Monte Carlo samples (points) are generated by random process according to P_{x_1} and P_{x_2}.

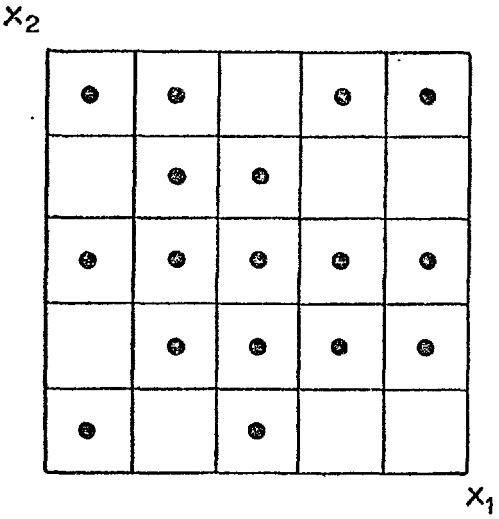
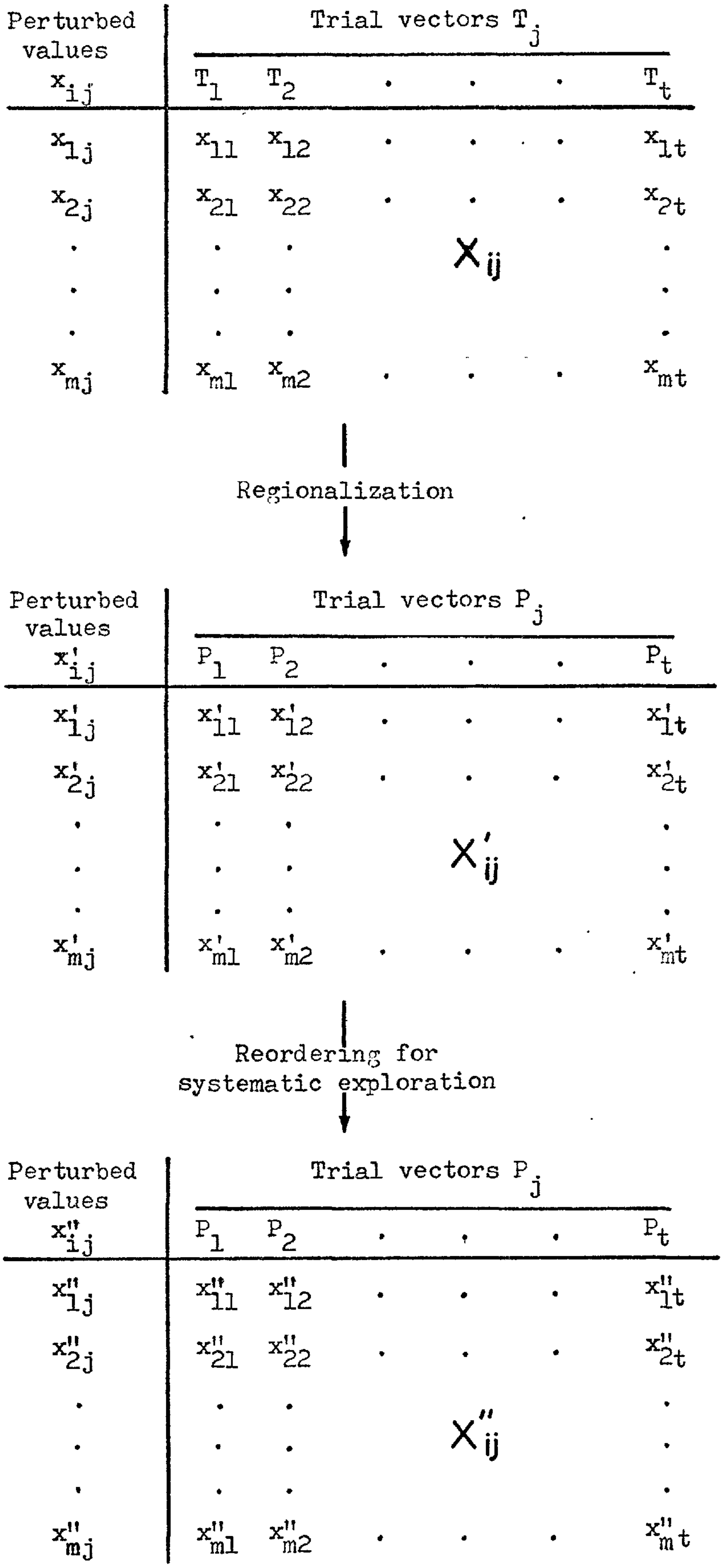


Fig. 7.3(b). Regionalized variable component space.



Perturbed values	Trial vectors T_j					
x_{ij}	T_1	T_2	.	.	.	T_t
x_{1j}	x_{11}	x_{12}	.	.	.	x_{1t}
x_{2j}	x_{21}	x_{22}	.	.	.	x_{2t}
.	.	.		X_{ij}		.
.	.	.				.
.	.	.				.
x_{mj}	x_{m1}	x_{m2}	.	.	.	x_{mt}

Regionalization

Perturbed values	Trial vectors P_j					
x'_{ij}	P_1	P_2	.	.	.	P_t
x'_{1j}	x'_{11}	x'_{12}	.	.	.	x'_{1t}
x'_{2j}	x'_{21}	x'_{22}	.	.	.	x'_{2t}
.	.	.		X'_{ij}		.
.	.	.				.
.	.	.				.
x'_{mj}	x'_{m1}	x'_{m2}	.	.	.	x'_{mt}

Reordering for systematic exploration

Perturbed values	Trial vectors P_j					
x''_{ij}	P_1	P_2	.	.	.	P_t
x''_{1j}	x''_{11}	x''_{12}	.	.	.	x''_{1t}
x''_{2j}	x''_{21}	x''_{22}	.	.	.	x''_{2t}
.	.	.		X''_{ij}		.
.	.	.				.
.	.	.				.
x''_{mj}	x''_{m1}	x''_{m2}	.	.	.	x''_{mt}

Fig. 7.4. Processing of trial vectors for systematic exploration.

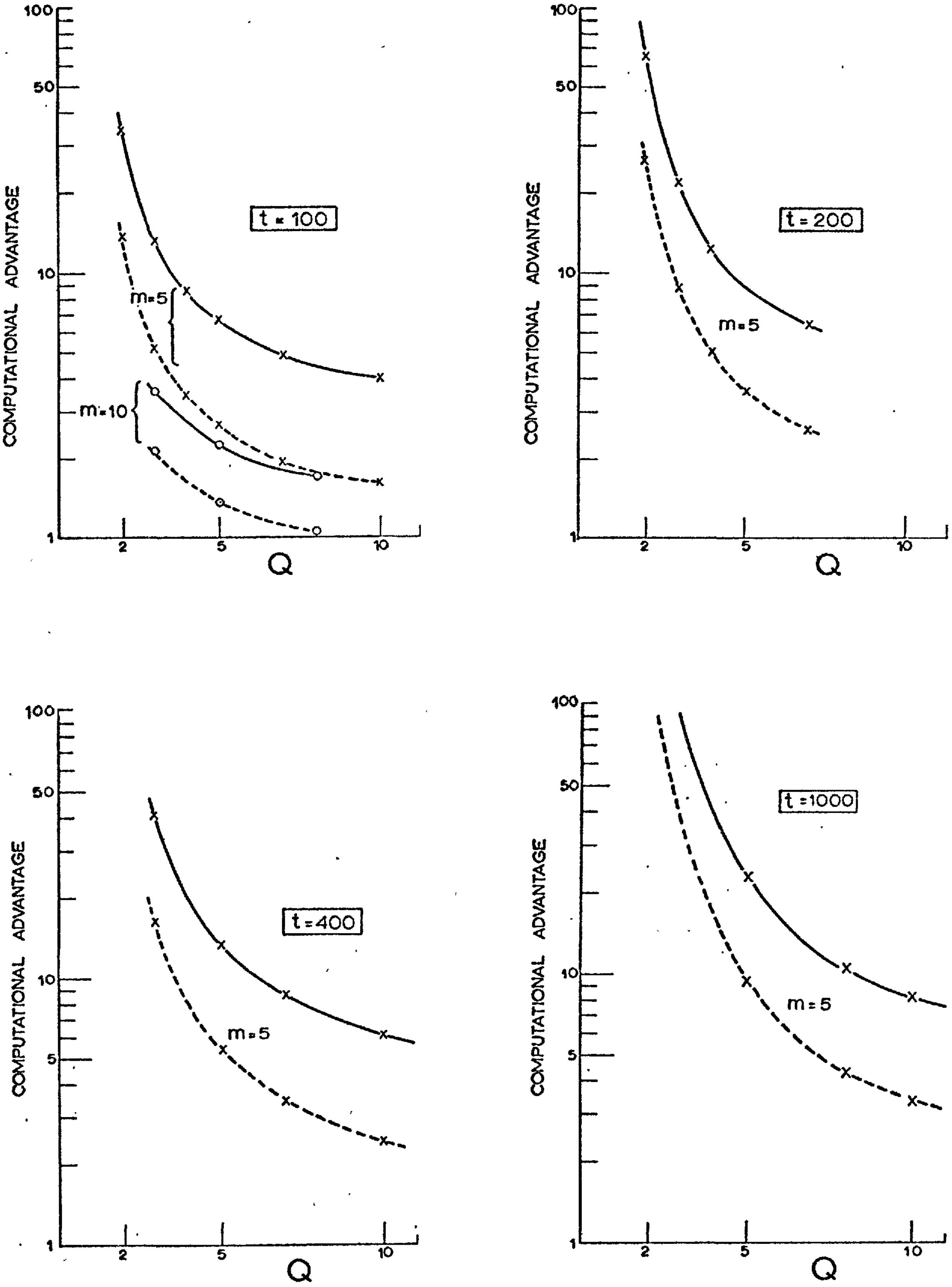


Fig. 7.5. Cost advantage obtained by component discretization and ordering of trials.

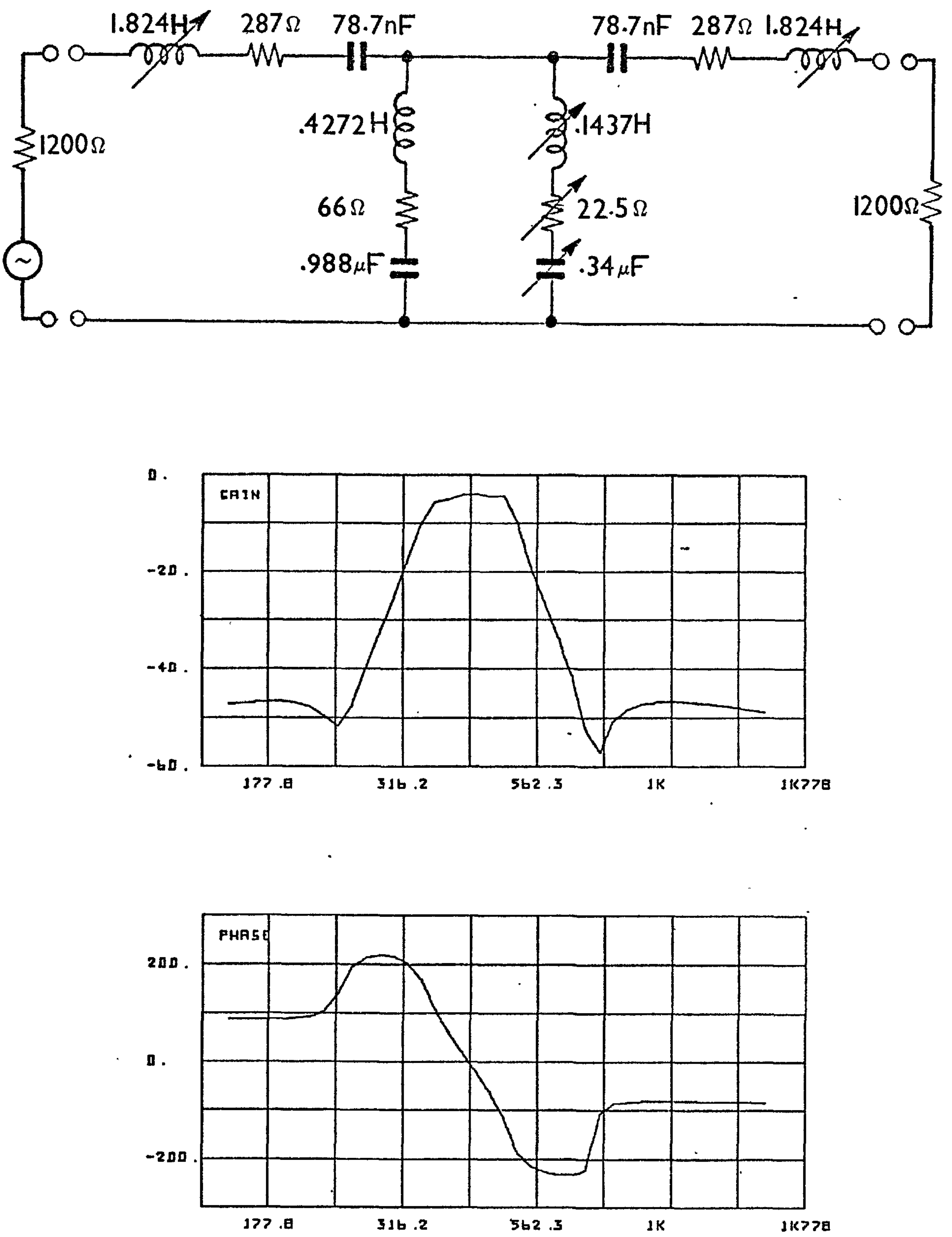


Fig. 7.6. A bandpass filter containing variable (arrowed) components and its nominal response curves.

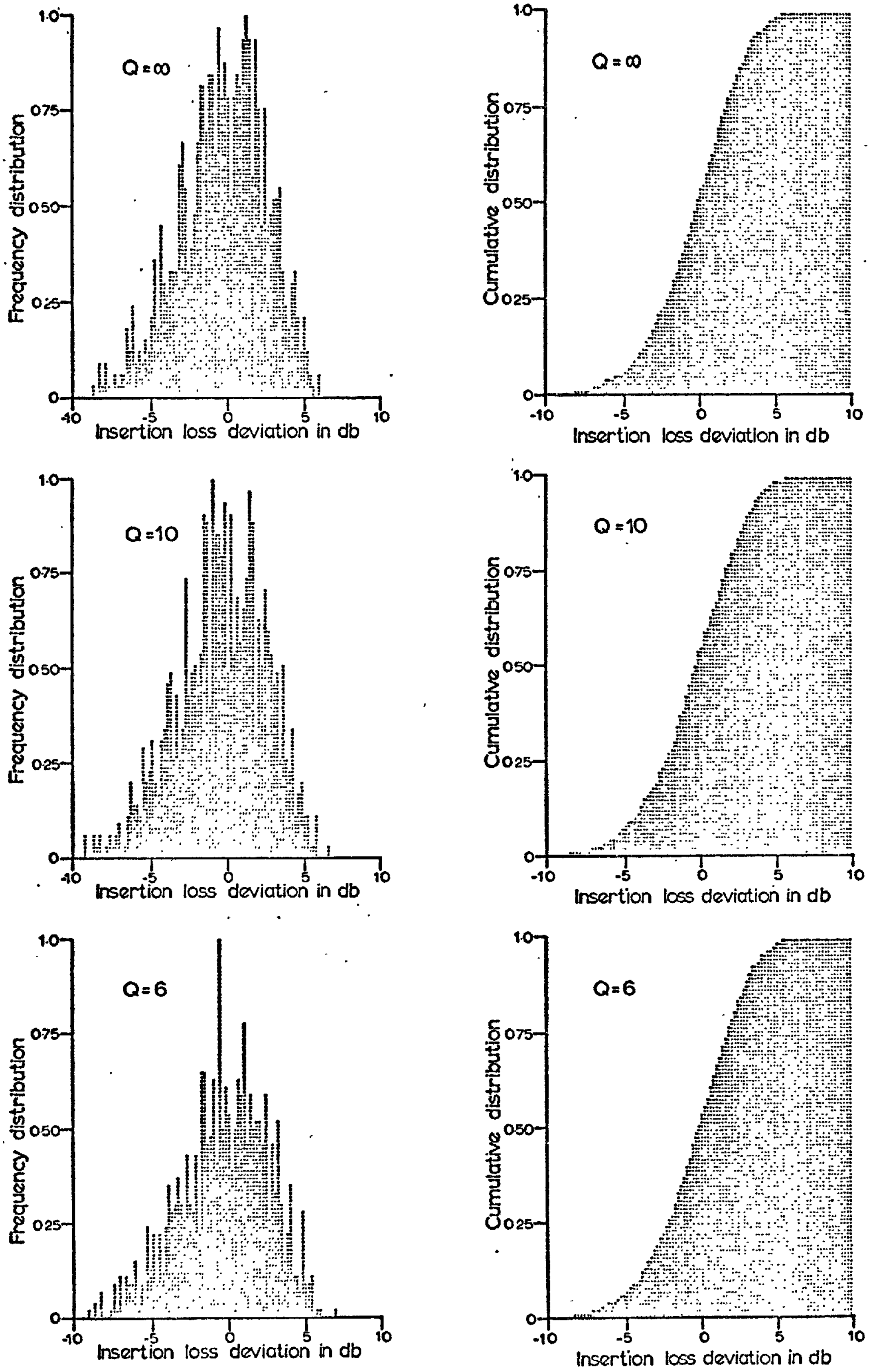


Fig. 7.7. Frequency and cumulative distribution of insertion loss deviation of filter in Fig. 7.6.

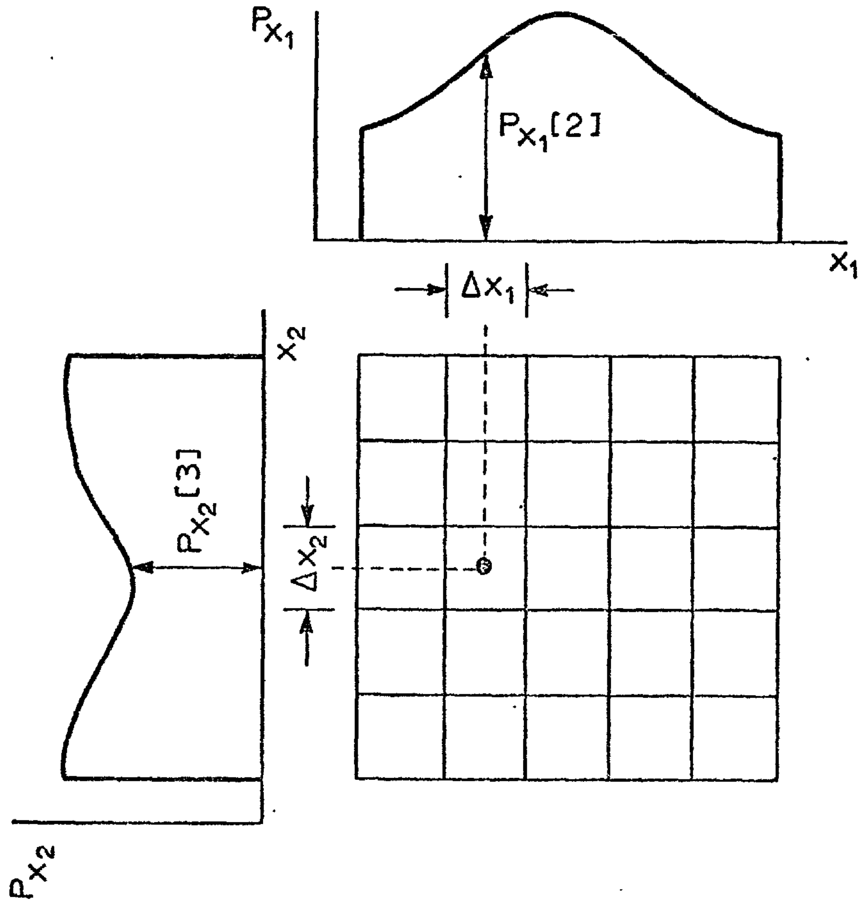


Fig. 7.8(a). Weights of samples are computed directly from P_{x_1} and P_{x_2}.

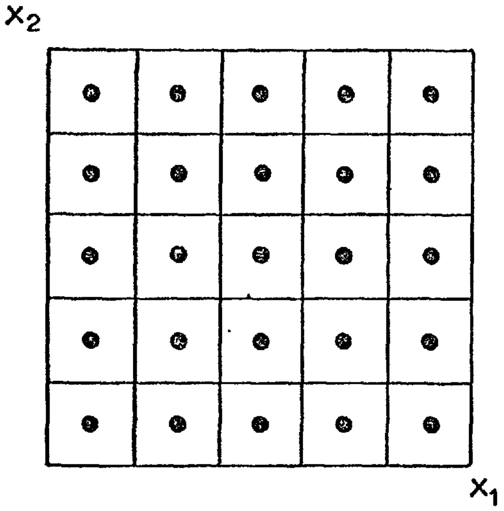


Fig. 7.8(b). Idealized statistical model.

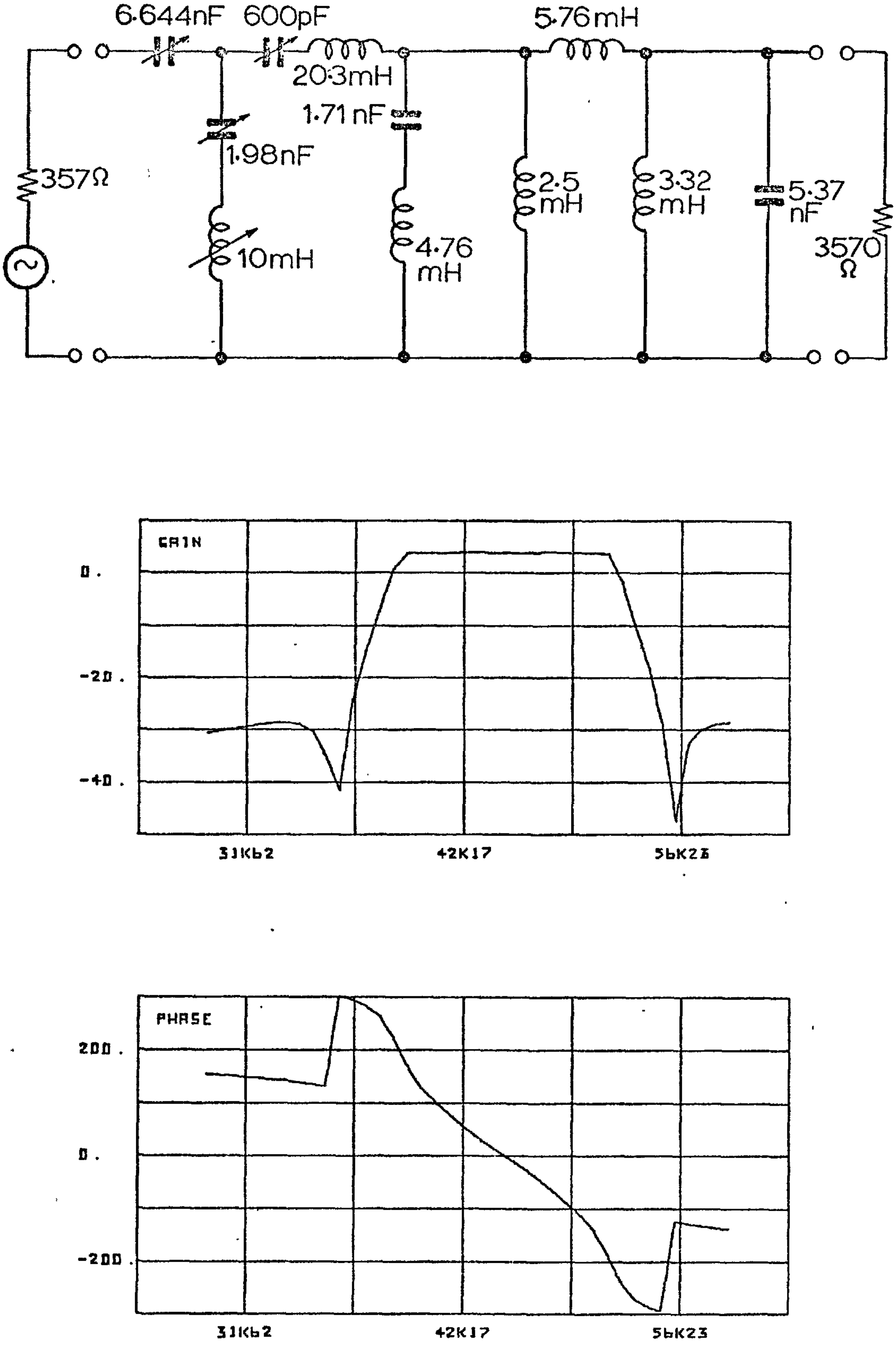
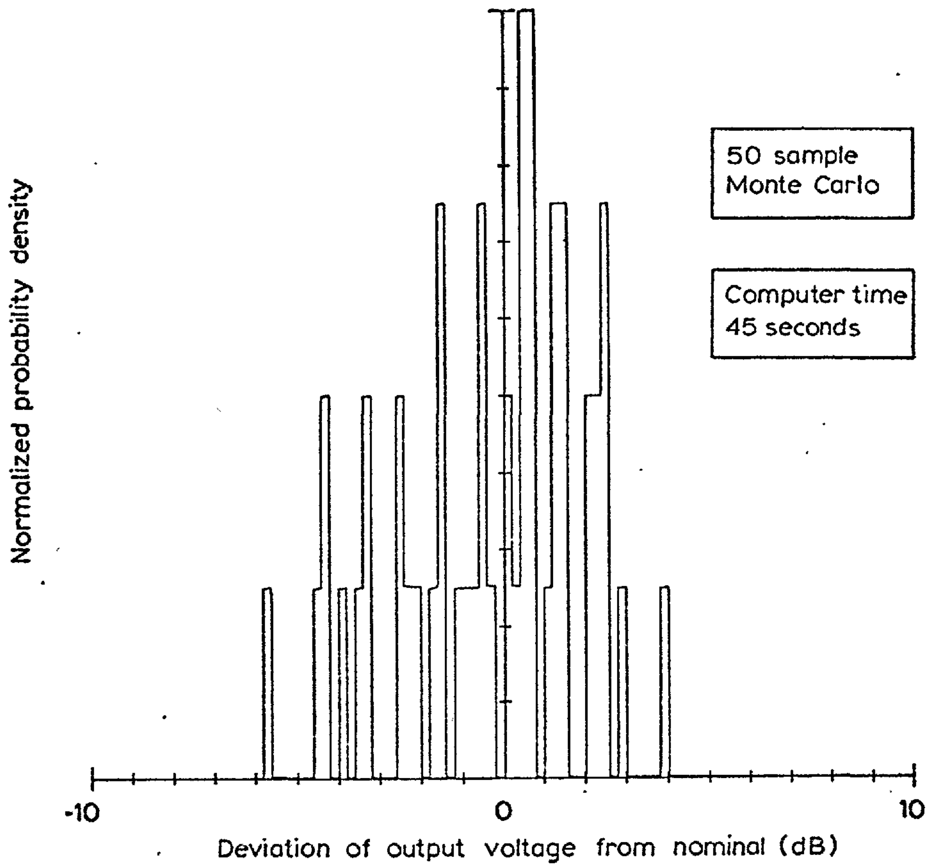


Fig. 7.9. A bandpass filter containing variable (arrowed) components and its nominal reponse curves.



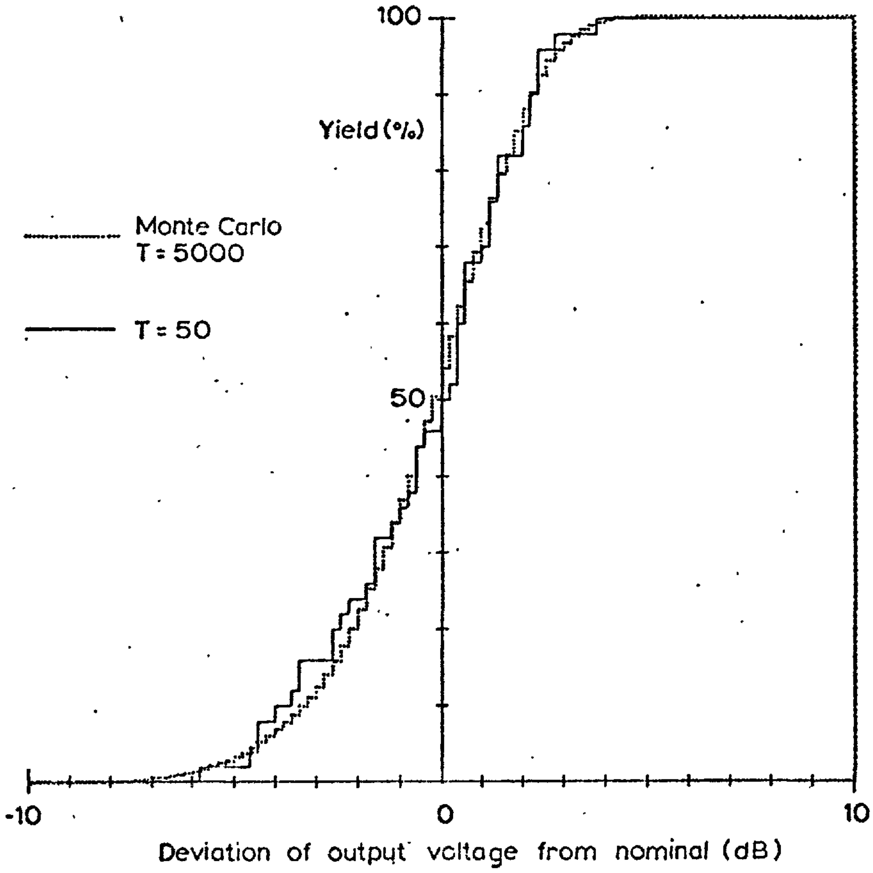


Fig. 7.10. Frequency and cumulative distribution predicted by Monte Carlo analysis.

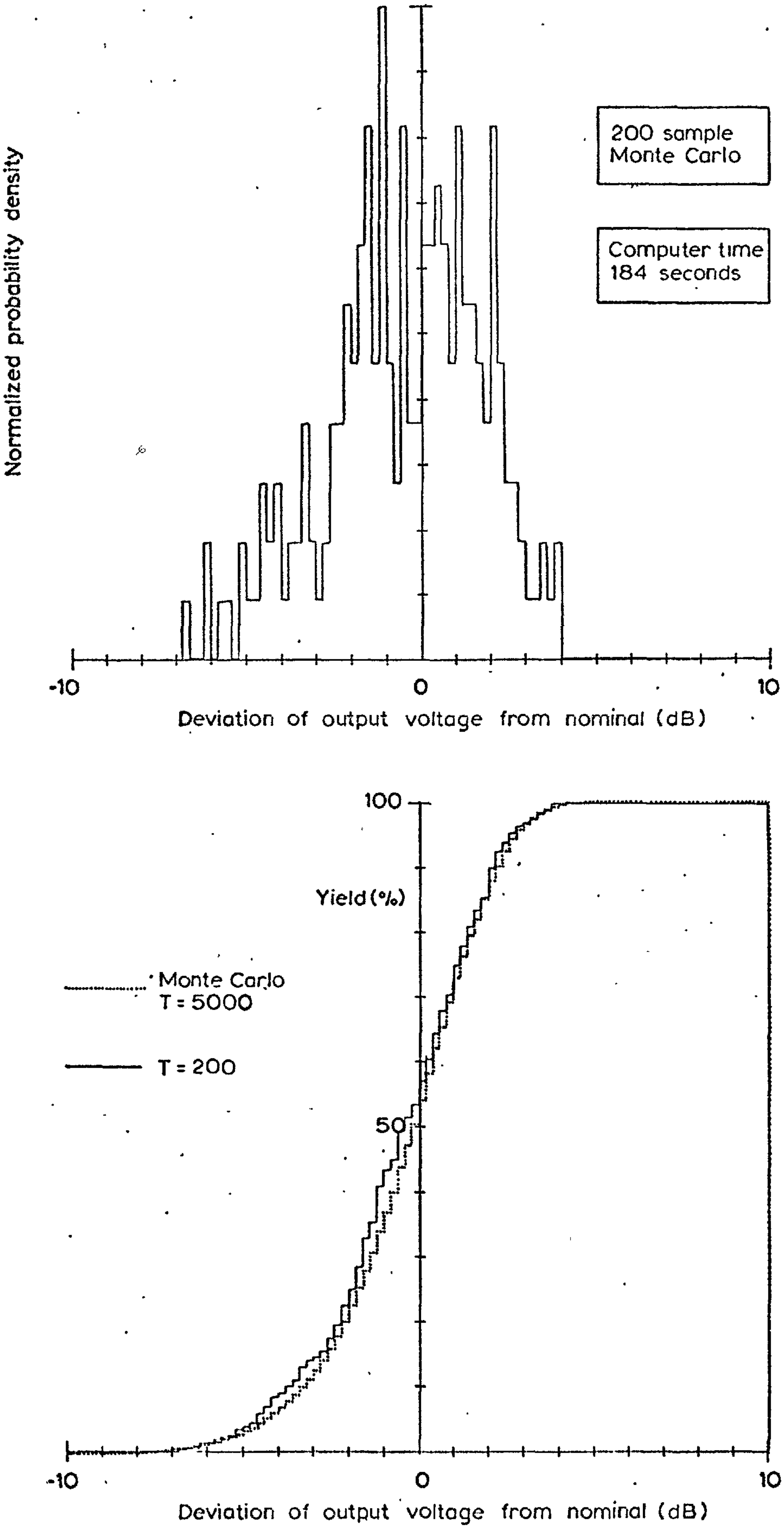
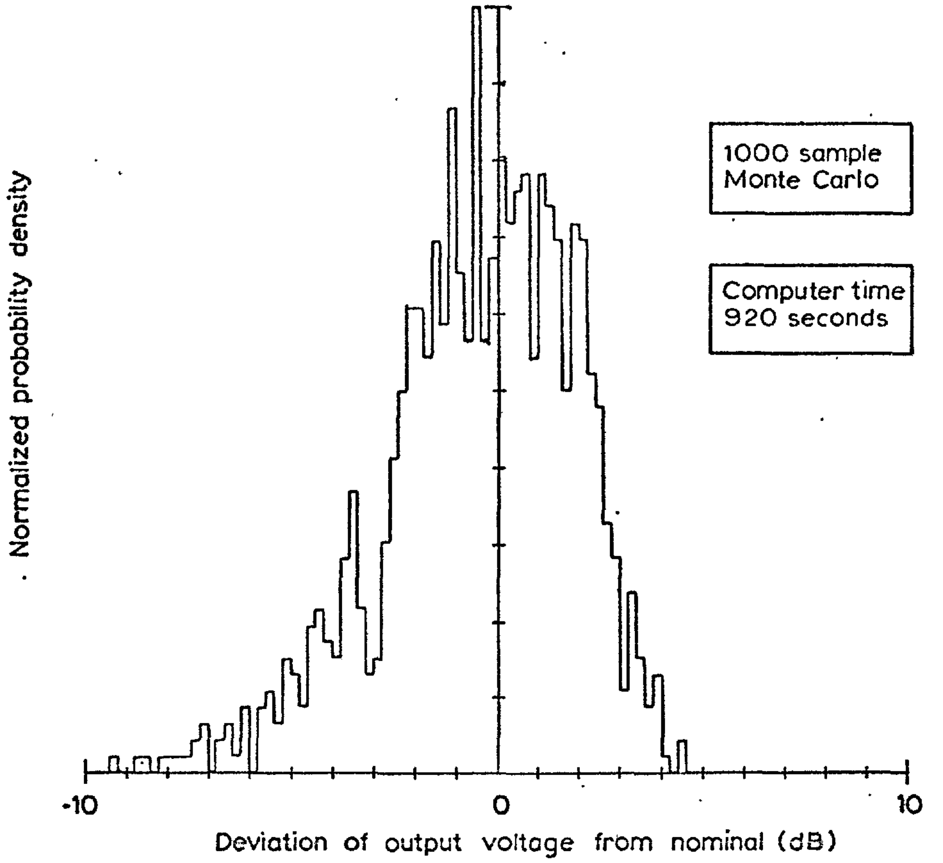


Fig. 7.11. Frequency and cumulative distribution predicted by Monte Carlo analysis.



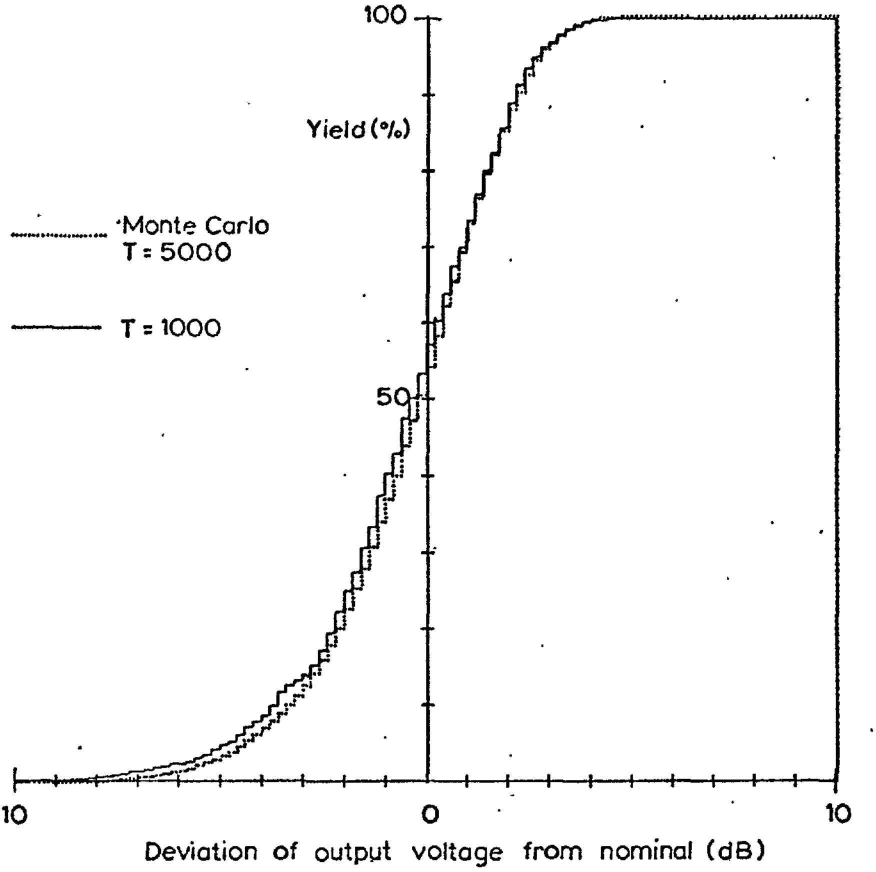


Fig. 7.12. Frequency and cumulative distribution predicted by Monte Carlo analysis.

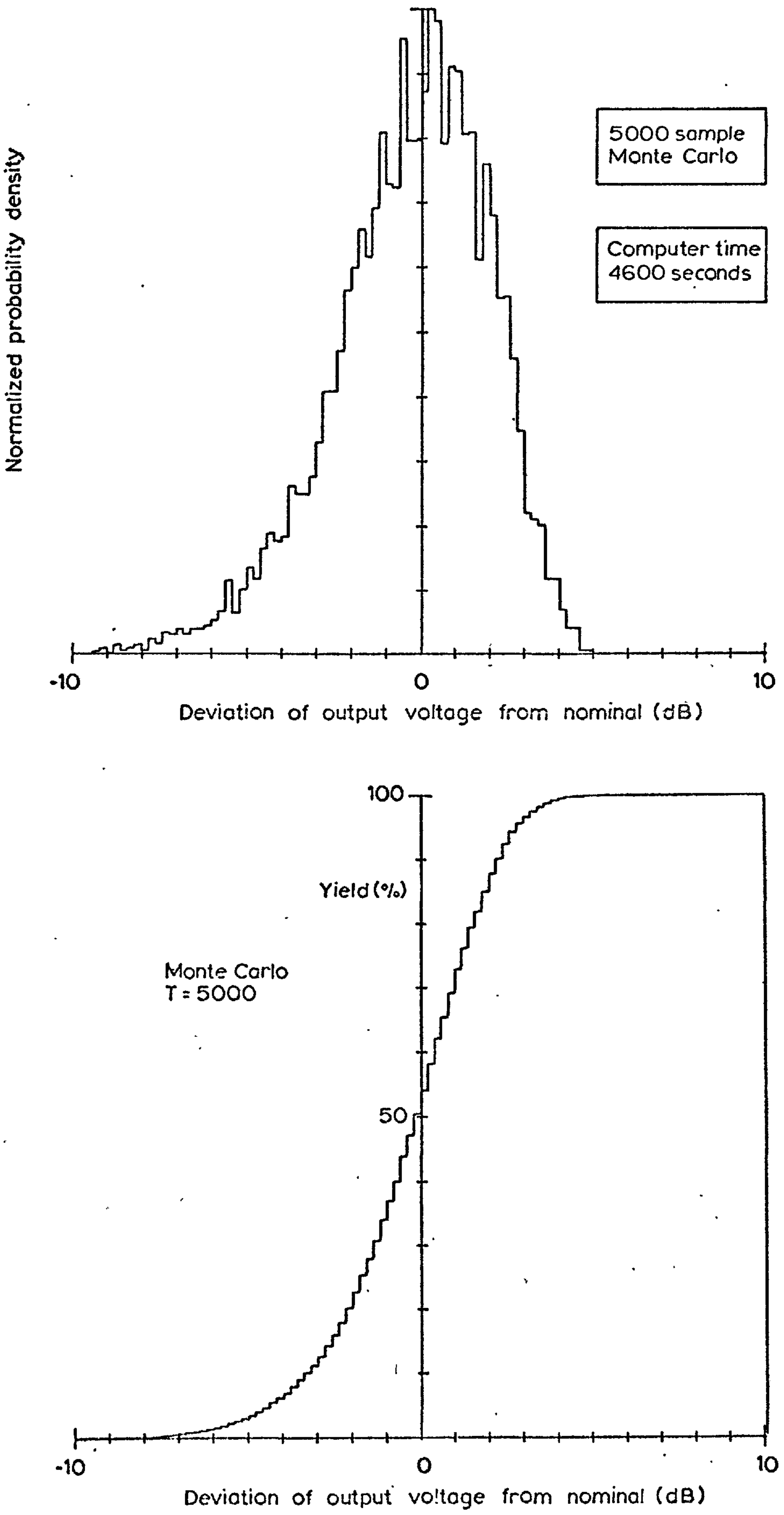
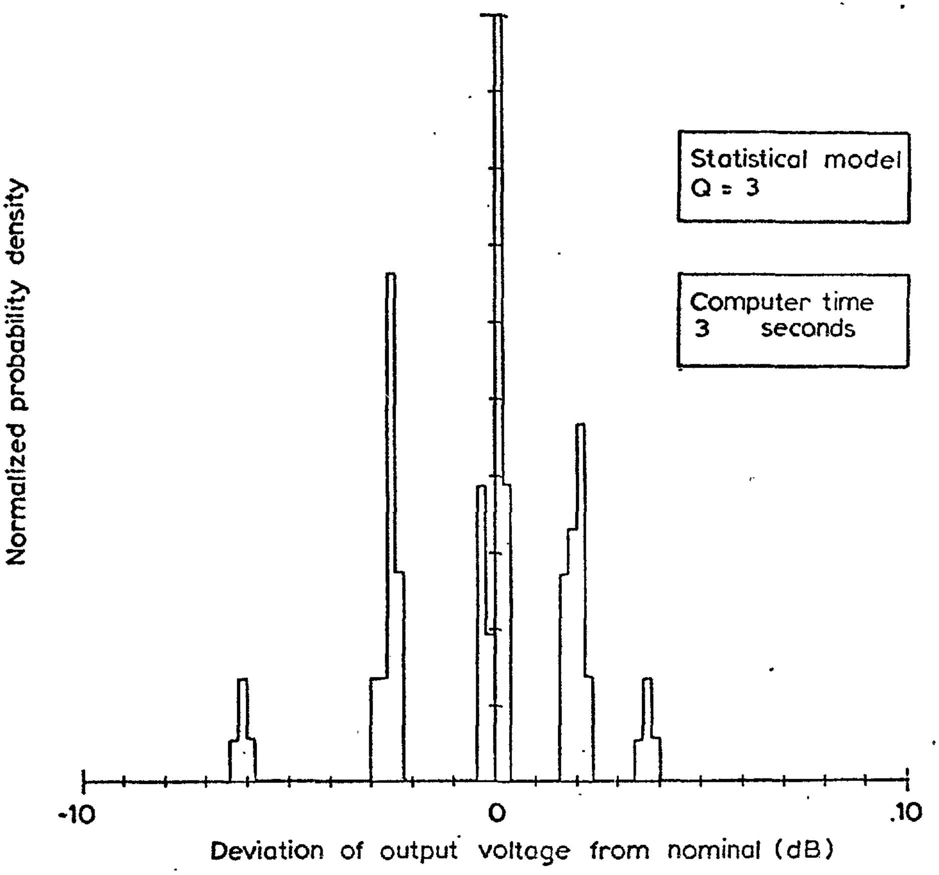


Fig. 7.13. Frequency and cumulative distribution predicted by Monte Carlo analysis.



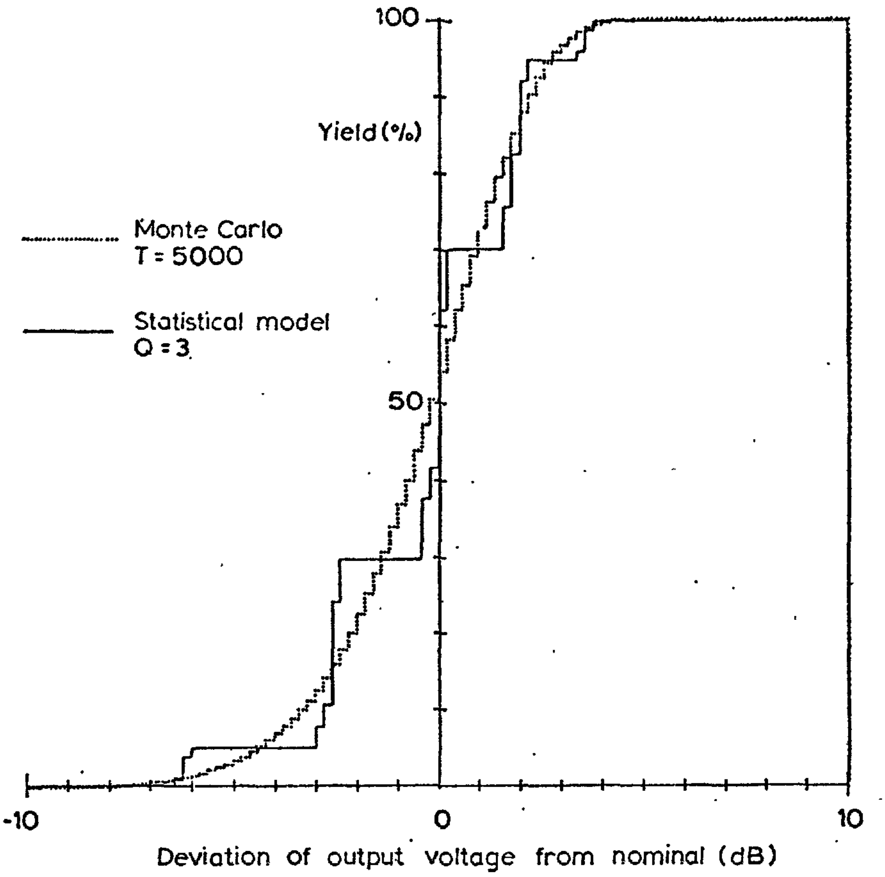


Fig. 7.14. Frequency and cumulative distribution predicted by the analysis of idealized statistical model.

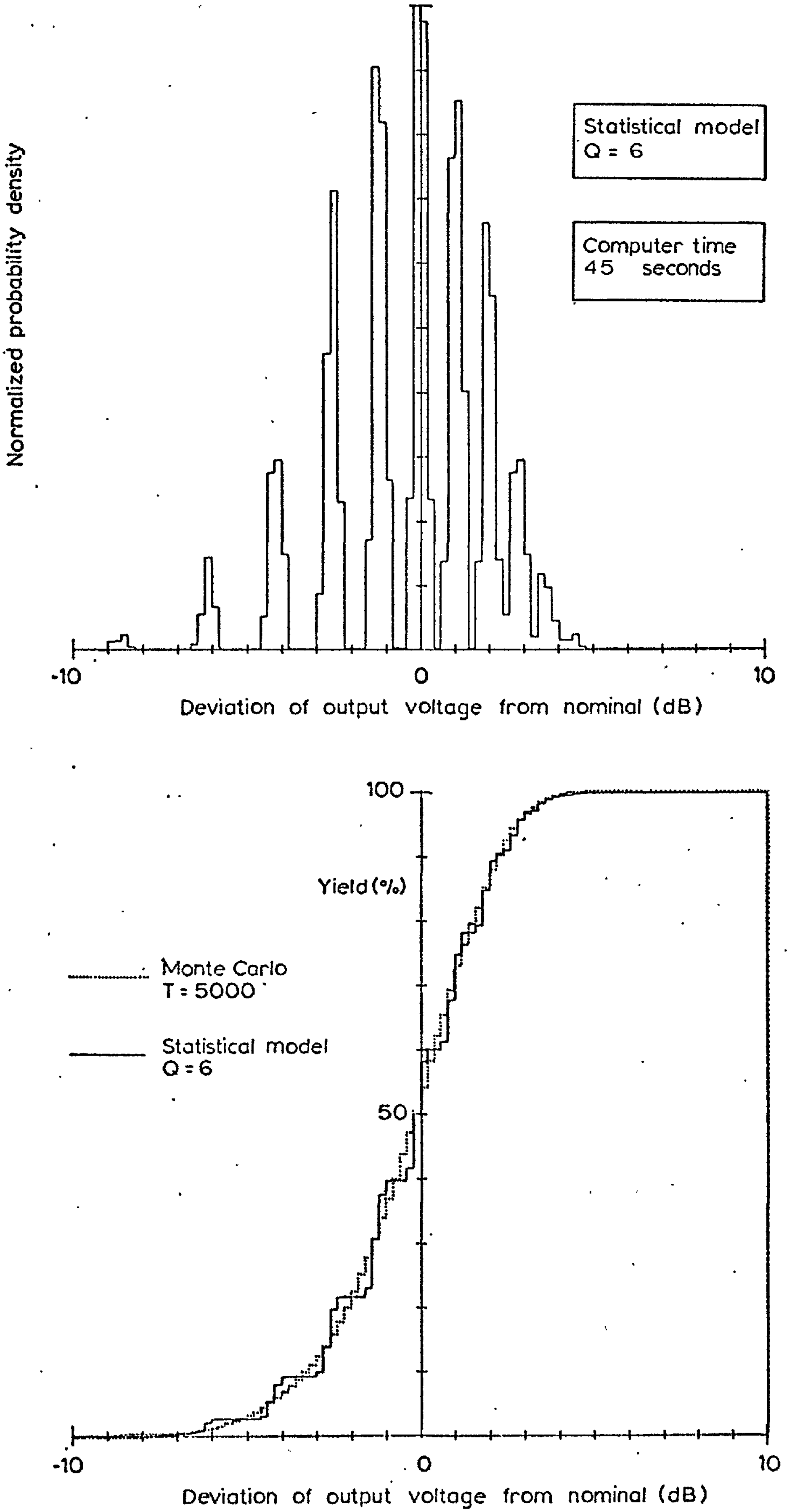
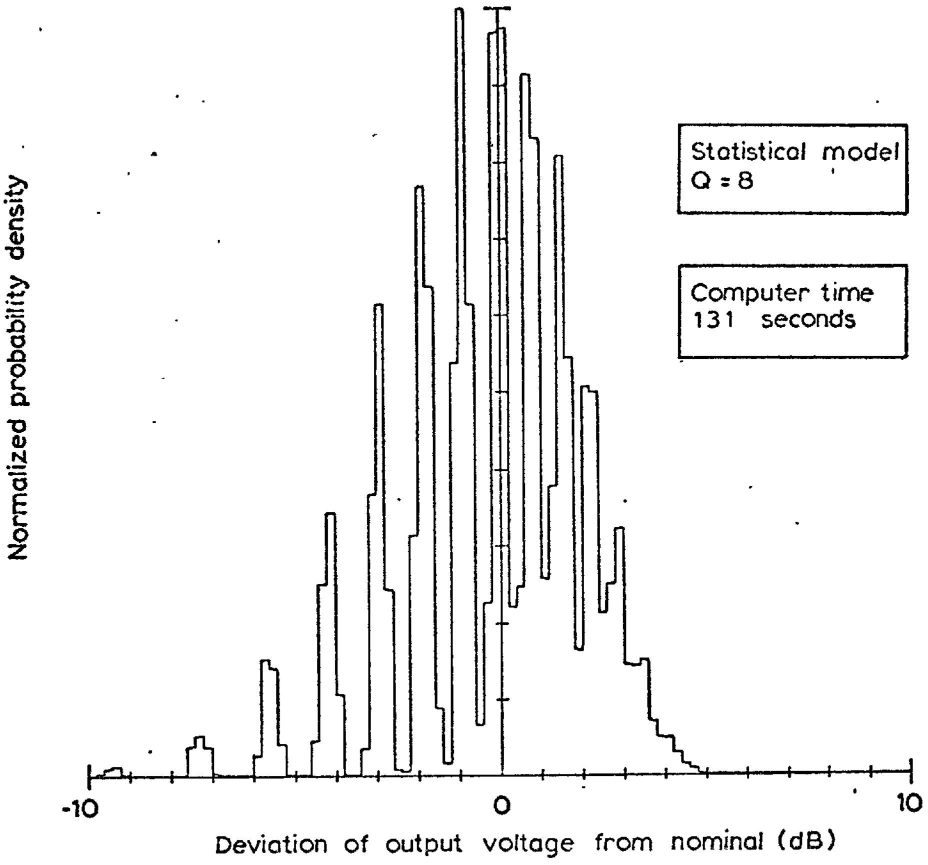


Fig. 7.15 Frequency and cumulative distribution predicted by the analysis of idealized statistical model.



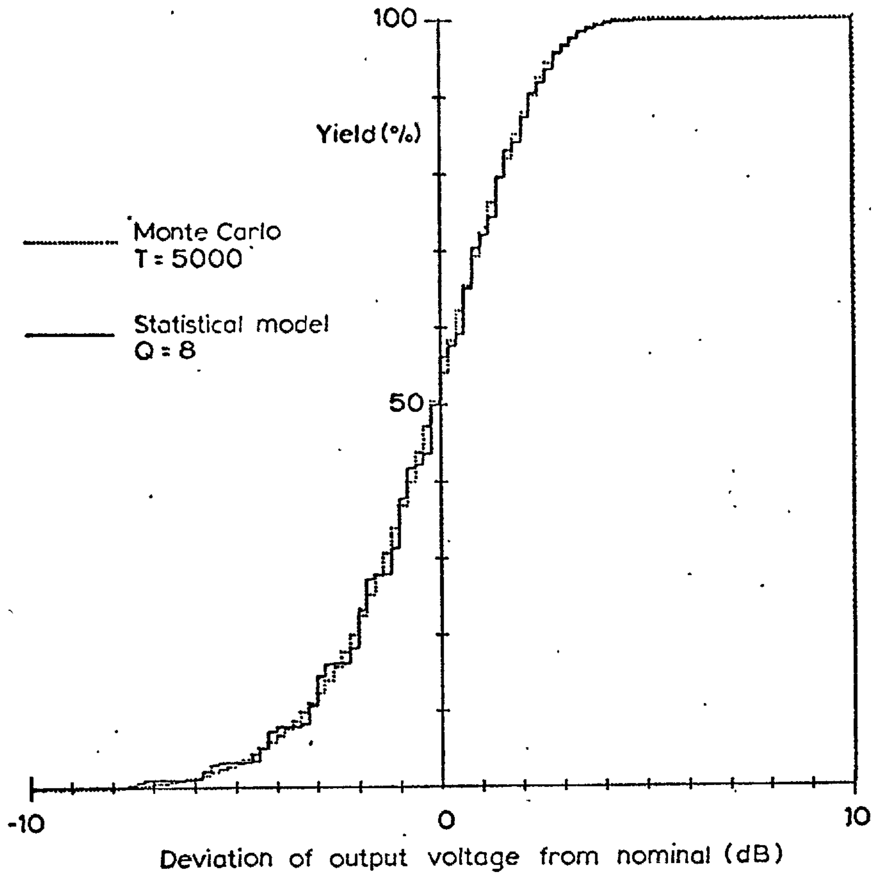
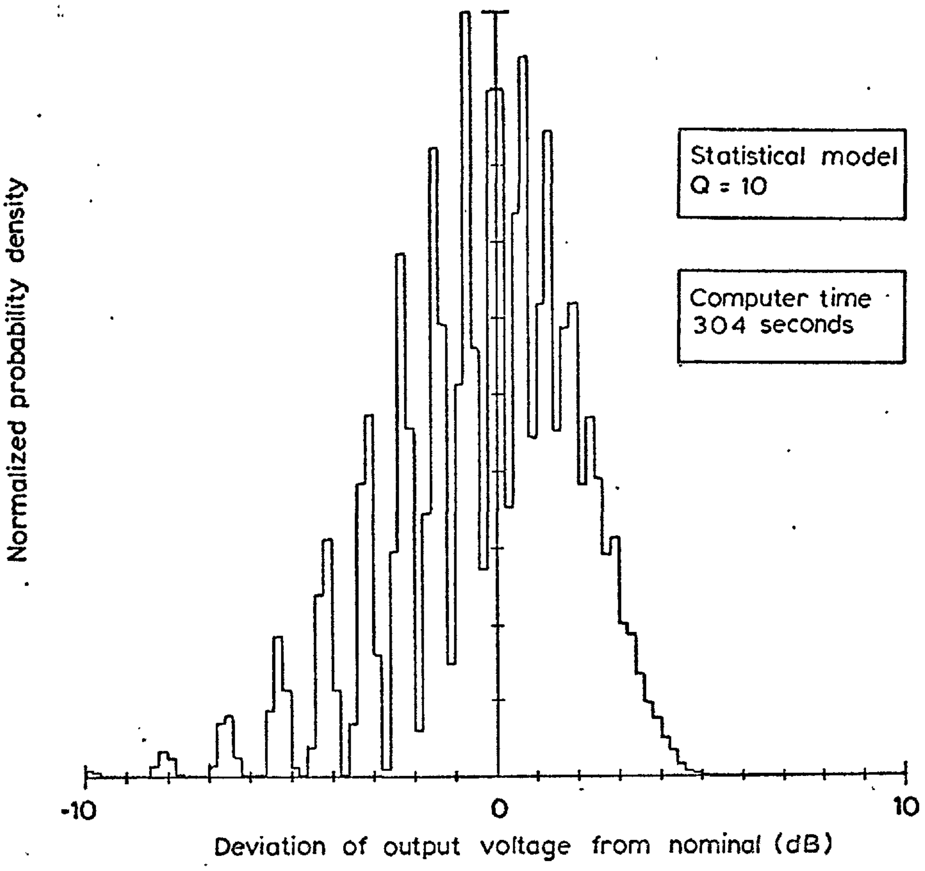


Fig. 7.16. Frequency and cumulative distribution predicted by the analysis of idealized statistical model.



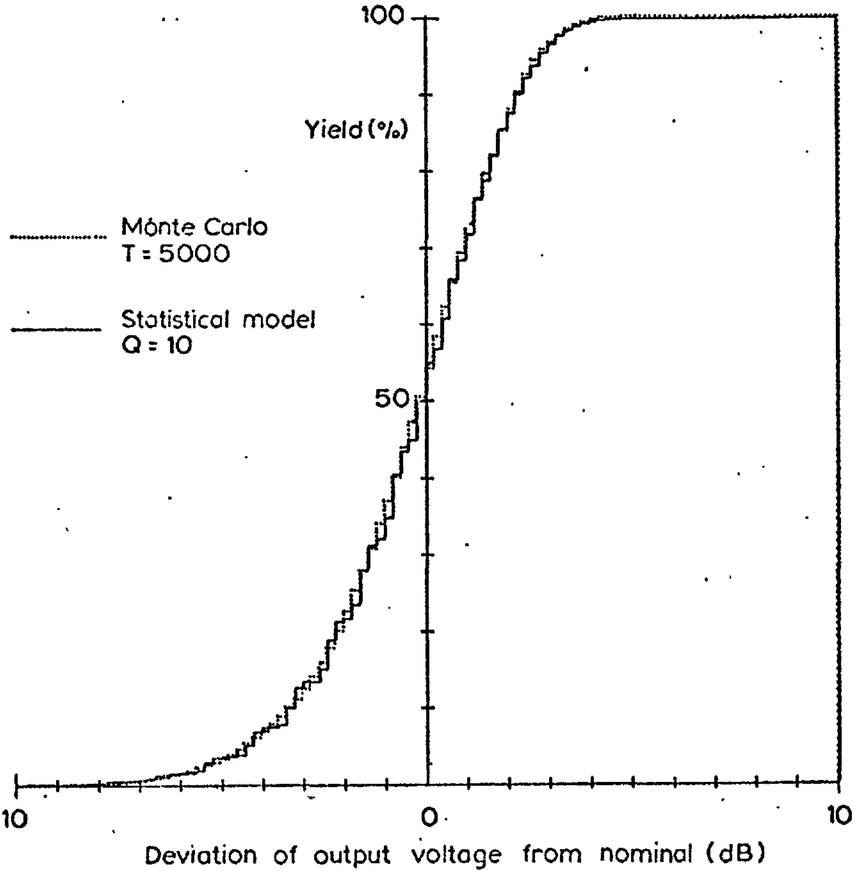


Fig. 7.17 Frequency and cumulative distribution predicted by the analysis of idealized statistical model.

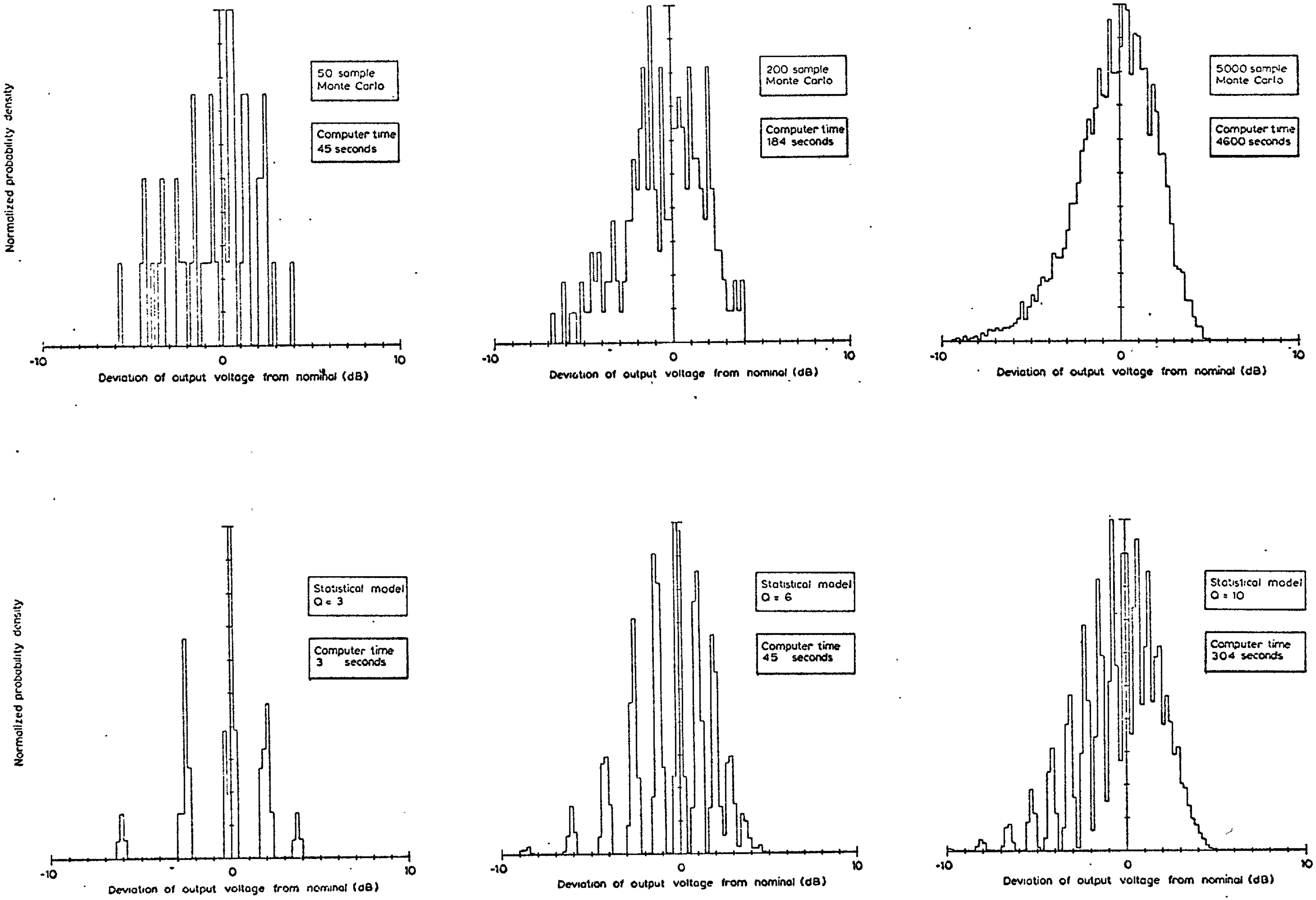


Fig. 7.18. Frequency distribution of output voltage for the circuit of Fig. 7.9, predicted by 50-, 200- and 5000-sample Monte Carlo analyses and by 3-, 6- and 10-quantization level idealized statistical models.

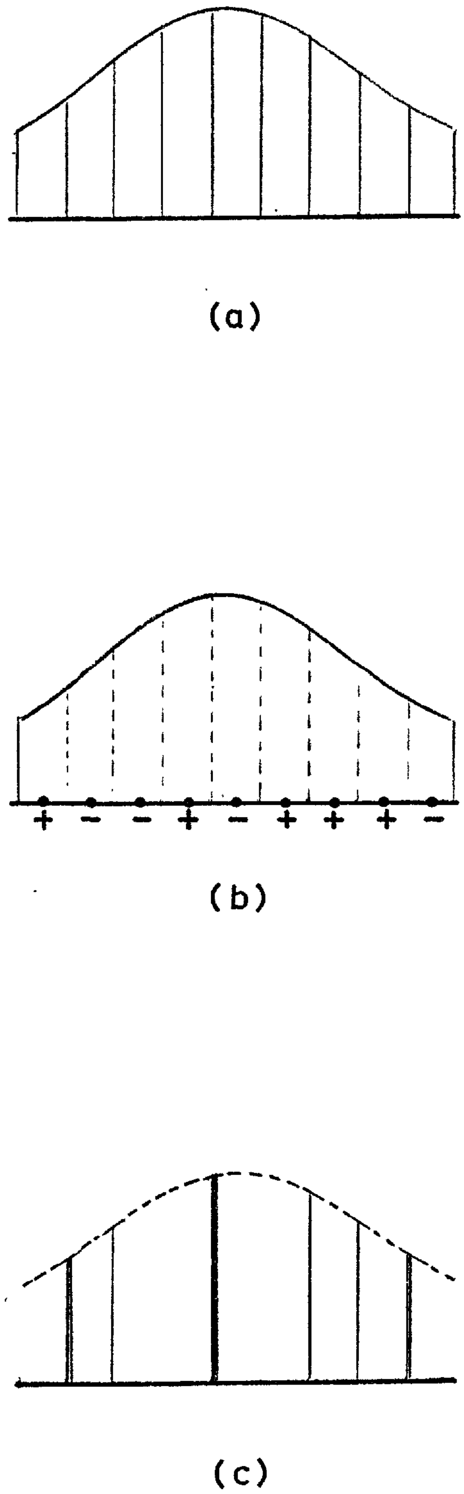


Fig 7.19. Scheme for worst case simulation.

(a) Quantisation of distribution.

(b) Computation of sign of small change sensitivity at the midpoints.

(c) Discretization according to the signs in (b).

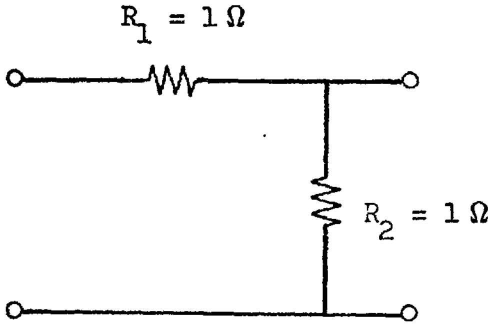


Fig. 7.20(a). A simple circuit.

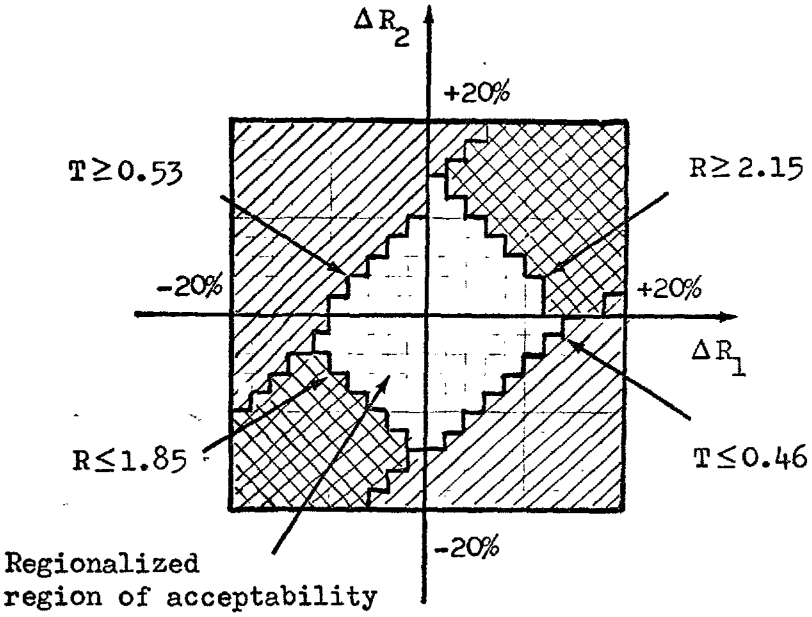


Fig. 7.20(b). Regionalized regions of acceptability of the circuit in (a).

CHAPTER 8

EPILOGUE

In the previous chapters, different methods have been proposed for the efficient computation of sensitivities of electronic circuits. According to the nature of a particular problem, an appropriate method may be chosen for maximum computational efficiency. Out of the sensitivity analysis methods proposed in this thesis, the systematic exploration combined with the concept of regionalization has been shown to be able to solve a number of different problems in computer aided circuit design with remarkable efficiency and, therefore, deserves more attention in future research projects. To conclude the discussion, some problems encountered in the course of the development and application of the sensitivity analysis methods in this thesis are reviewed here, and some possible solutions are also outlined.

1. The computation of small change sensitivity after large changes in circuit component values

In chapter 7, the problem of computing small change sensitivity following large changes in circuit component values arose when the sign of small change sensitivity is used as an indicator in the worst case simulation process. The small change sensitivity associated with the variable components can, of course, be computed by making a completely new analysis of the circuit. In view of the formidable computational cost required by such an approach, a more efficient method must be found. One comparatively efficient method was proposed by Sud [47] . However, in this method, if there are m components subjected to large changes then, in order to compute the set of small change sensitivities associated with the m variable components, a Gaussian elimination of up to a maximum (m x m) matrix is needed for each set of large component changes, and the computational cost is still quite high. Fortunately, if the Z matrix of a circuit is updated to accommodate the effect of large changes in component values, such as in the case of systematic exploration, the small change sensitivities associated with the m variable components are the by-products of the process. Therefore, they can be obtained at a very low computational cost, as will now be demonstrated.

Consider the circuit model of Fig. 4.3(a). If the circuit is excited by current sources which are equal to unity, according to Chapter 2, the small change sensitivity of a variable component, say y_3, is $-z_{3\phi}\, z_{\psi 3}$ where $z_{3\phi}$ and $z_{\psi 3}$ are two elements in the reduced port matrix shown in Fig. 4.4. As we modify the matrix to account for the effect on $z_{\psi\phi}$ of a new change in y_3, very little additional computational effort is required to modify $z_{3\phi}$ and $z_{\psi 3}$ to account for the effect of a new change in y_3 as well. After $z_{3\phi}$ and $z_{\psi 3}$ have been modified, the new small change sensitivity is available.

It should be noted that in the worst case simulation method described in chapter 7, the effect on small change sensitivity of large changes in component values was limited to the case where only one component is assumed to have large change as the sign of small change sensitivity is computed. Therefore, this process can be performed at the outset*. In some cases, for a more realistic worst

* In this case, the number of signs to be stored is $Q_1 + Q_2 + \dots + Q_m$.

case simulation, the effect of simultaneous large changes on a small change sensitivity has to be accounted for. In such a case, each set of changes has its corresponding set of small change sensitivities; therefore, they are best computed as the sets of changes are being connected to the circuit. This simulation of a worst case is more realistic, but at the same time, higher computational cost will be incurred. A low cost realistic worst case simulation has to be found.

2. Tracking sensitivity and nonlinear coefficiency

In the algorithm proposed for the computation of tracking sensitivity, a parametric circuit description $y_i = y_{iN} (1 + K \xi_i)$ was invoked. Of the two coefficients ξ_i and K, the former is 'local' to the i^{th} component, whereas the latter K is 'global' and common to all components. For example, ξ_i represents the linearized temperature coefficients of the i^{th} component and K is the temperature. Usually, the temperature coefficient can be approximated by a constant. However, it may be useful if the algorithm can be developed to accommodate nonlinear ξ_i without undue reduction in efficiency so that the simulation would be more realistic.

3. Statistical circuit analysis and worst case simulation

Statistical circuit analysis has become an important part of computer aided circuit design because of its ability to predict the manufacturing yield. However, the prediction is accurate only when the number of circuits produced and the number of simulated circuit analyses are both very large. A large number of simulated circuit analyses can be achieved by a computational effort which is under the control of the designer whereas the number of circuits produced by the manufacturer is fixed by the production scheme. One problem needs to be considered; if the number of circuits going to be produced is moderate, how could a statistical analysis be applied in such a case?

Basically, if the number of circuits to be produced is small, an accurate prediction of manufacturing yield is almost impossible, even if the prediction is obtained from the analysis of a very large number of simulated circuits or a statistical model. On the other hand, a purely worst case analysis may be unnecessarily conservative in setting component tolerance limits and its use result in increased production costs. In such a case, a hybrid approach - a statistical analysis mixed with a certain degree of worst case simulation - could be the solution.

By modelling the variable component space with an idealized statistical model, and carrying out worst case simulation within each region, a mixing of worst case simulation into the result of a statistical analysis is now feasible. By controlling the sizes of the regions into which the variable component space is partitioned, the degree of worst case simulation can be controlled. A question now exists: How can we establish the relationship, in terms of relative quantity, between the statistical analysis and worst case simulation according to the number of the circuit produced? Some research work in this respect remains to be done.

4. Yield track

An example of mapping failures into a two dimensional variable component space has been illustrated in chapter 7. The result of the mapping is the generation of a regionalized region of acceptability in the variable component space. However, if the number of variable components is greater than two, the mapping of failures into the variable component space is no longer a straightforward task. This is because a variable component space with a high dimensionality is practically impossible for the designer to visualize, let alone to map failures into it. It is for this reason that an alternative method must be devised.

An alternative method of mapping failures resulting from a number of analyses of simulated circuits is now described. In this method, the failures are mapped on a segmented track - termed 'yield track'. The mapping procedure is as follows:

a) A set of m segmented tracks are used to represent the value fluctuations of the m variable components. Each segment of a track is registered to represent a particular perturbed value of a variable component. An m = 2 example is illustrated in Fig. 8.2. In this example, each variable component is assumed to have eight different values.

b) The tracks are so aligned such that in the second track, the eight segments which represent the eight values of the second component repeat within each duration of segment of the first track. This procedure is repeated as many times as there are segments for the rest of the tracks. As a result, the number of segments on the last track is equal to the number of all combinations of the changes of the variable components.

c) Then, another segmented track whose number of segments is equal to the last track is laid alongside and aligned with the last track. This track, termed yield track, is used for the recording of failures. If a set of changes (the segments in line with a segment on the yield track) generates a failure, the corresponding segment on the yield track is marked. In Fig. 8.2, the yield track associated with the two variable components is also shown.

The computation of yield tracks for linear circuit could be highly efficient since systematic exploration can be applied. Yield track has found its applications in the following two important aspects of computer aided circuit design.

5. Isolation of uncorrelated variable components

From the information recorded on the yield track, the regionalized region of acceptability of any combination of the variable components (from 2 to m) can be generated without any arithmetical operation. It is because of this property that all possible two dimensional regions of acceptability of any combination can be generated and examined at high speed. By observing the shapes of these regions, the uncorrelated pairs of variable components can be identified*, the isolation of uncorrelated variable components may now be a feasible task. As a result of this, the number of variable components m, whose effect of simultaneous change must be considered, can be reduced to a minimum.

6. Design centering and tolerance assignment

The aim of design centering is to fit the biggest possible hypercube (i.e., region of possibility) into the region of acceptability. This presents no particular difficulties for a two dimensional problem, since a two dimensional region of acceptability can be displayed graphically and the finding of the largest square inside it can be carried out automatically by a computer or manually by the designer with reasonable accuracy. However, the problem of

* The identification could be based on the properties of performance contours described in Section 5.5. Further research works in this respect may be needed.

design centering becomes increasingly difficult if a high dimensional problem has to be tackled. This is due to the fact that depiction of a high dimensional region of acceptability is an impractical task - we are trapped by "the curse of dimensionality".

So far, different methods have been proposed for tackling the dimensionality problem [1,2,49] . However, in view of the costs and approximations associated with those methods, better methods are still being sought. The use of a yield track to record failures, combined with the concept of the regionalized region of acceptability may enable a solution to be found.

The method proposed here is based on the fact that in a regionalized variable component space, all edges leading to a common vertex are orthogonal. Therefore, the hypercube within a region of acceptability can be characterized by observing its orthogonal silhouettes in m orthogonal directions, where m is the number of dimensions of the variable component space. This concept is best illustrated by a two dimensional example; in Fig. 8.1, assuming that a 3 x 3 square is to be fitted into the depicted region of acceptability. By observing the region of acceptability in the directions of the A axis and the B axis, the feasible centres of the square can be determined.

Based on the same concept, the feasible centres can also be determined by examining the yield track alone without the need of depicting the region of acceptability as illustrated in Fig. 8.1. In this approach, the observation of the region of acceptability in different directions is simulated by transposing track A and track B to generate a new yield track. The ability to find feasible centres without depiction of the region of acceptability is the most important property of this method. It is because of this property that this method

- termed orthogonal silhouette method - can be extended to handle problems of any number of dimensions.

The procedure of fitting the biggest hypercube into a region of acceptability is as follows, illustrated by the two-dimensional example shown in Fig. 8.2(a) and (b).

a) Each segment of the yield track in Fig. 8.2(a) is examined automatically by the computer for success and failure, successes being noted. Each successful segment corresponds to a region inside the region of acceptability. The size of this smaller region is the initial size of the square which can be fitted.

b) The 2 segments adjacent to each success segment are examined. If they are both successes, then the centre segment is noted as a protocentre for a larger square, whose side length is three times that of the initial square.

c) Tracks 1 and 2 are transposed and a new yield track is generated (Fig. 8.2(b)). Those segments already identified as procentres will be in a different position along the new yield track and will have different pairs of adjacent segments. If both segments of each pair is a protocentre, then, the centre segment is identified as a feasible centre of the larger square.

d) If there are more than one of these centres, the above process must be continued by expanding the range of checking for successes to 4 segments (2 adjacent segments for each side) and so on, until a minimum number of centres has been achieved.

The advantages of the above method are as follows:

a) For the above process, no arithmetical operation was required, and therefore, the computational cost has been kept to a minimum.

b) The method does not assume any particular shape for the region of acceptability, the only approximation being the regionalization of the variable component space.

c) If the hypercubic is regionalized and weighted (i.e., an idealized statistical model), a controllable trading between yield and tolerance is feasible.

d) It can be adapted to find centres of hyper-rectangle of various aspect ratios so that the problem of tolerance assignment [50,51,52] can be tackled.

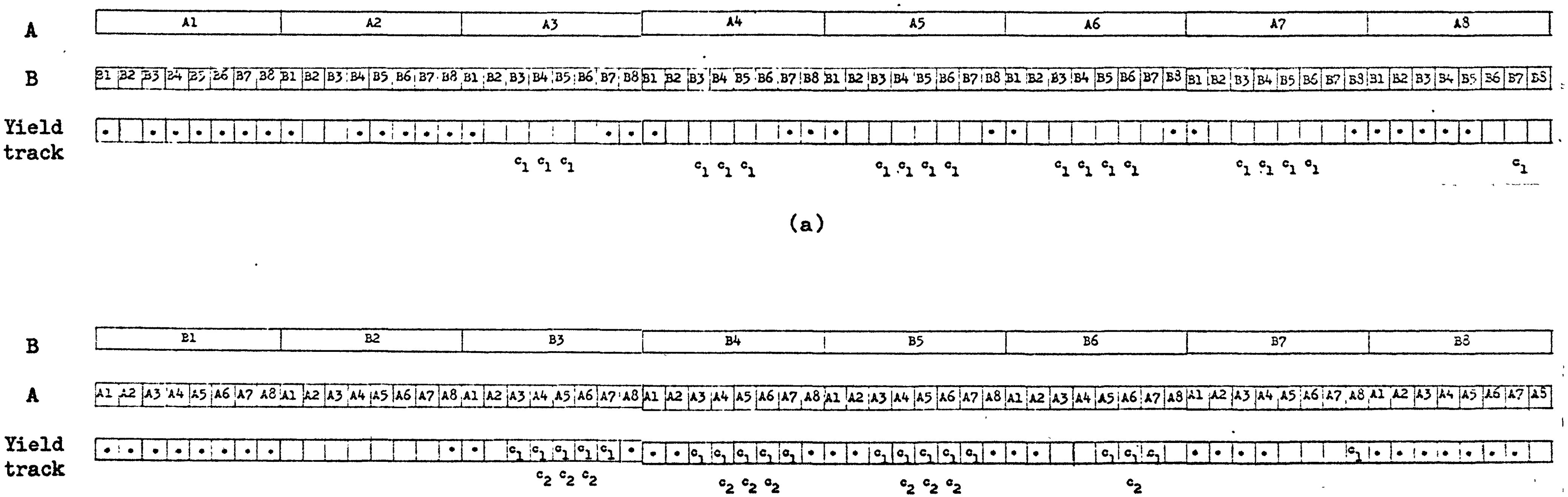
e) Most important of all, the method is easily expanded to handle the multi-dimensional problem.

A									
1	*		*	*	*	*	*	*	
2	*			*	*	*	*	*	
3	*		c_1	c_1	c_1		*	*	
4	*		c_1	c_1	c_1		*	*	
5	*		c_1	c_1	c_1	c_1		*	
6	*		c_1	c_1	c_1	c_1		*	
7	*		c_1	c_1	c_1	c_1		*	
8	*	*	*	*	*		c_1		
	1	2	3	4	5	6	7	8	B

Fig. 8.1(a). Fitting of a 3 x 3 square into the region of acceptability (unmarked area) is attempted. The regions marked c_1 indicate the protocentres obtained by observing the region in the direction of A axis.

A									
1	*		*	*	*	*	*	*	
2	*			*	*	*	*	*	
3	*						*	*	
4	*		c_2	c_2	c_2		*	*	
5	*		c_2	c_2	c_2			*	
6	*		c_2	c_2	c_2	c_2		*	
7	*							*	
8	*	*	*	*	*				
	1	2	3	4	5	6	7	8	B

Fig. 8.1(b). Feasible centres, marked c_2, are confirmed by observing the region in the direction of B axis.



(a)

(b)

Fig. 8.2. Finding feasible centres without depiction of the region of acceptability. (a) Observing the region in the direction of A axis. (b) Observing the region in the direction of B axis by transposing track A and track B.

APPENDIX

THE CALCULATION OF EIGENVALUES AND EIGENVECTORS OF MATRICES

1. The Fundamental Equations

Let A be a square matrix of order n. Its eigenvalues $\lambda_1, \ldots\ldots, \lambda_n$ are the solutions of the determinantal equations, called the characteristic equation

$$| A - \lambda I | = 0 \tag{1}$$

Corresponding to each distinct eigenvalue λ_i, there exists at least one solution of the system of linear equations

$$(A - \lambda_i I) X_i = 0 \tag{2}$$

This solution $X_i^t = (x_{1i}, x_{2i}, \ldots\ldots, x_{ni})$ is an eigenvector of A. Here we shall be interested in the most widely used process for the calculation of all of the eigenvalues and eigenvectors of A.

2. Some Useful Matrix Results

It will be useful to present briefly some parts of the matrix theory which form the back-ground to this appendix. The following results are of interest.

1. A matrix is said to be orthogonal if

$$\underline{a}_i^t \underline{a}_j = 0 \qquad i \neq j$$

$$\underline{a}_i^t \underline{a}_i = 1 \tag{3}$$

where the vector $\underline{a}_i$ is the ith column of A. It should be noted that inverse of an orthogonal matrix can be obtained without calculation since $A^tA = I$ according to (3) and hence $A^{-1} = A^t$.

2. The eigenvalues of a real symmetric matrix are real. The eigenvectors corresponding to distinct eigenvalues are orthogonal.

3. If a matrix of order n has n distinct eigenvalues then there are n linearly independent eigenvectors which can form a base for the space of vectors. An arbitrary vector can then be expressed in terms of the eigenvectors,

$$\underline{y} = \sum_{i=1}^{n} a_i X_i \tag{4}$$

where X_r $(r = 1, 2, \ldots, n)$ are the linearly independent eigenvectors.

4. If X_i is an eigenvector corresponding to the eigenvalue λ_i then $A X_i = \lambda_i X_i$ and $A^k X_i = \lambda_i^k X_i$. Thus, the effect of successive multiplication of an eigenvector by the matrix A is to successively multiply the vector by the scalar λ_i.

5. Two matrices, A and B are said to be similar if a non-singular matrix P exists such that $B = P^{-1} A P$. It is easy to see that similar matrices have the same eigenvalues, since if

$$A X = \lambda X$$

Then

$$P^{-1} A X = \lambda P^{-1} X \tag{5}$$

and, if

$$X = P Y \tag{6}$$

then

$$P^{-1} A P Y = \lambda Y \tag{7}$$

The eigenvectors of A can be found from the eigenvectors of B by the relation X = P Y.

6. If X_i is an eigenvector of a matrix, then any scalar multiple of this also is an eigenvector. It will sometimes be convenient to normalize the eigenvector. This can be done in two ways. One method of normalization is to divide all the elements of a vector by the largest element so that vectors have unity as the largest element. Alternatively, each element could be divided by the sum of the squares of the elements of the vector in which case vectors have unit length.

3. Householder's Method

This method uses orthogonal transformations to reduce a symmetric matrix A to a form in which the only nonzero elements are on the main diagonal and the two diagonals directly above and below it as shown in Fig. 1. This form of matrix is known as tridiagonal matrix.

$$\begin{matrix} a_{11} & a_{12} & 0 & \cdots & 0 \\ a_{12} & a_{22} & a_{23} & & \vdots \\ 0 & a_{23} & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & a_{n-1,n} \\ 0 & \cdots & 0 & a_{n-1,n} & a_{nn} \end{matrix}$$

Fig. 1. A symmetric tridiagonal matrix.

There are (n - 2) steps in this reduction. In the kth step zeros are introduced in the kth row and kth column without destroying the tridiagonal elements and the zeros introduced in the previous steps.

Let V be a vector such that

$$V^t V = 1 \tag{8}$$

Then it is easy to show that the matrix

$$P = I - 2 V V^t \tag{9}$$

is orthongonal and symmetric. In particular we choose V_k to be a vector whose first k - 1 components are zero so that

$$V_k^t = \left[0, 0, \ldots, 0, v_k^{(k)}, v_k^{(k+1)}, \ldots, v_k^{(n)} \right] \tag{10}$$

Then with

$$P_k = I - 2 V_k V_k^t \tag{11}$$

we define

$$A_k = P_k^t A_{k-1} P_k \qquad k = 2, \ldots, n - 1 \qquad A_1 = A \tag{12}$$

Now suppose that the current symmetric matrix $A_{k-1} = \left[a_{ij} \right]$ has zeros in its first k - 2 rows and columns except for the tridiagonal elements:

$$A_{k-1} = \begin{array}{r} \\ \\ \\ \\ \text{row} \\ k-1 \\ \\ \\ \end{array} \left[\begin{array}{ccccccc}
a_{11} & a_{12} & 0 & \cdots & \cdots & \cdots & 0\\
a_{12} & a_{22} & a_{23} & 0 & \cdots & \cdots & 0\\
0 & a_{23} & \ddots & \ddots & \ddots & & \vdots\\
\vdots & 0 & \ddots & \ddots & \ddots & & \vdots\\
\vdots & \vdots & \ddots & a_{k-2,k-2} & a_{k-2,k-1} & 0 \cdots & 0\\
\vdots & \vdots & \ddots & a_{k-2,k-1} & a_{k-1,k-1} & \cdots & a_{k-1,n}\\
\vdots & \vdots & & 0 & \vdots & \ddots & \vdots\\
0 & 0 & \cdots & 0 & a_{k-1,n} & \cdots & a_{nn}
\end{array}\right] \qquad (13)$$

The matrix P_k has the form

$$I - 2\, V_k\, V_k^t = \begin{array}{r} \\ \\ \\ \\ \text{row} \\ k \\ \\ \end{array} \left[\begin{array}{ccccc}
1 & 0 & \cdots & \cdots & 0\\
0 & \ddots & \ddots & & \vdots\\
\vdots & \ddots & \ddots & \ddots & \vdots\\
\vdots & \ddots & 1 & 0 \cdots & 0\\
\vdots & \cdots & 0 & 1-2\left[v_k^{(k)}\right]^2 \cdots & -2v_k^{(n)}\, v_k^{(k)}\\
\vdots & & \vdots & \vdots \quad \ddots & \vdots\\
0 & \cdots & 0 & -2v_k^{(n)}\, v_k^{(k)} \cdots & 1-2\left[v_k^{(n)}\right]^2
\end{array}\right] \qquad (14)$$

Using (12) - (14) we may verify that A_k has zeros in the positions shown as zero for A_{k-1} in (13). Our object is to choose the $n - k + 1$ numbers $v_k^{(k)}$, ..., $v_k^{(n)}$ to satisfy (8), so that the $n - k$ off-tridiagonal elements in row (column) $k - 1$ of A_k are zero.

We define

$$S = \sum_{j=k}^{n} a_{k-1,j}^{2} \qquad (15)$$

and then let

$$\left[v_k^{(k)} \right]^2 = {}^1/_2 \left[1 \pm (a_{k-1,k} / \sqrt{S}) \right] \qquad (16)$$

and

$$v_k^{(j)} = \pm a_{k-1,j}/(2 v_k^{(k)} \sqrt{S}) \quad j = k + 1, \ldots, n \qquad (17)$$

where the plus or minus sign will be chosen below. The motivation for (16) and (17) may be found in the algebra leading to the proof that the desired n - k elements in the k - 1 row (column) of A_k are zero and that (8) is satisfied. Proceeding as above at each step we arrive at a tridiagonal matrix A_{n-1}.

The accuracy of this method depends naturally on the accuracy of the matrices P_k, and these in turn depend upon the accuracy of the components of (10). The key to making this accuracy as great as possible is to make the magnitude of $v_k^{(k)}$ as given by (16) as great as possible because it is a divisor in (17). Therefore we choose the sign in (16) so as to maximize the magnitude of $v_k^{(k)}$ and then use the same sign in (17).

At each stage it would appear that two square roots are required - one for $\sqrt{S}$ and one for $\left\{ \left[v_k^{(k)} \right]^2 \right\}^{\frac{1}{2}}$. However, by arranging the calculation properly, the latter of these two need not be calculated.

The approach to nonsymmetric matrices will be similar in the sense that we shall perform a series of transformations on the matrix A in order to reduce A to a matrix B with the same eigenvalues as A but whose eigenvalues are more easily calculable. If Householder's method is applied to a nonsymmetric matrix A, the result is a Hessenberg matrix as shown in Fig. 2.

$$\begin{matrix} b_{11} & b_{12} & 0 & \cdots & 0 \\ b_{21} & b_{22} & b_{23} & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & \ddots & b_{n-1,n} \\ b_{n1} & \cdots & \cdots & \cdots & b_{nn} \end{matrix}$$

Fig. 2. A matrix in lower Hessenberg form.

Such a matrix is said to be in supertriangular or, more commonly, in lower Hessenberg form whereas the transpose of such a matrix is said to be in upper Hessenberg form, whose eigenvalues are more easily calculable than that of a full matrix.

4. The QR Iteration

The QR-algorithm is today recommended for the solution of eigenvalue problems in general, real, or complex matrices if more than very few eigenvalues are desired. This method making use of orthogonal transformations tends to be numerically stable.

The basis of this method is to decompose an arbitrary matrix A into a product QU where Q is orthogonal and U is upper triangular*.

* Francis (1961), the originator of this method used R as a mnemonic for right triangular.

The assurance that this can be accomplished is contained in a theorem which states that for an arbitrary real matrix A, there exists an orthogonal matrix Q and upper triangular matrix U such that A = QU.

Suppose now we form the similarity transform $Q^t A Q$ of the matrix A. We have

$$Q^t A Q = Q^t (Q U) Q = U Q \tag{18}$$

Hence, if we decompose A and then multiply the factors in the reverse order, we obtain a matrix similar to A. In the QR algorithm this process is repeated indefinitely. If we rename the original matrix A_1, then the algorithm is defined by the equations

$$A_{s-1} = Q_{s-1} U_{s-1} \qquad\qquad U_{s-1} Q_{s-1} = A_s \tag{19}$$

clearly A_s is similar to A_{s-1} and hence by induction, to A_1. It can be shown that under certain restrictions

$$Q_s \longrightarrow I \quad \text{and} \quad U_s \longrightarrow A_s \longrightarrow \begin{bmatrix} \lambda_1 & & & X \\ & \lambda_2 & & \\ & & \ddots & \\ O & & & \lambda_n \end{bmatrix}$$

$$\text{as } s \longrightarrow \infty \tag{20}$$

It is, of course, true that the decomposition of A into QU is in general quite time consuming $\left[\text{requiring } O(n^3) \text{ operations}\right]$. But if A is tridiagonal, the QR transformation can be done quite rapidly. Moreover, we may show that, if A is symmetric and tridiagonal, then so is every A_s. Of more importance is the result that, if A is in upper Hessenberg form, in which case the number of operations required to decompose A into QU is $O(n^2)$, then every A_s is in the upper Hessenberg form.

With the best devices to speed up convergence, the QR algorithm has cubic convergence even in the presence of multiple eigenvalues.

5. Wielandt inverse iteration

The universally accepted method to find eigenvectors when the eigenvalue is already known is Wielandt inverse iteration. We perform the following iteration for each eigenvalue approximation λ :

$$(A - \lambda I)\underline{y}_s = \underline{y}_{s-1} \quad , \quad s = 1, 2, \ldots, \qquad (21)$$

with $\underline{y}_0$ some arbitrary vector. Generally two iterations will be sufficient. The iteration is performed by forming LU decomposition of $(A - \lambda I)$ using Gaussian elimination. It should be noted that the triangular decomposition of a Hessenberg matrix is particularly simple since, speaking in terms appropriate to Gaussian elimination, there is only one non-zero multiplier at each stage of the reduction. There are $\frac{1}{2} n^2$ multiplications in all compared with $\frac{1}{3} n^3$ for a full matrix.

Generally, Wielandt inverse iteration is done on the reduced matrix (tridiagonal or Hessenberg) and the true eigenvectors are recovered subsequently. This recovery scheme depends on the reduction employed, but costs generally much more than the usage of Wielandt iteration, and less than usage of Wielandt iteration would have done on the original matrix.

If the approximation to the eigenvalue λ of (21) is too good, the determination of $\underline{y}_s$ can lead to a singular problem. This is detected by the necessity to pivot on a zero in the solution of the linear equations (21). The problem is resolved by introducing a deliberate "round error" at this point, and proceeding. This does no harm to the solution, it only results in a scaling of the eigenvector in question, which may subsequently be removed.

Varah [55] describes a test to determine if the choice of right-hand sides in the inverse iteration was appropriate or not. He also describes a strategy for constructing alternate right-hand sides, in the event that a particular choice proved to be a poor one. These procedures might be useful for obtaining independent vectors of multiple roots.

REFERENCES

[1] BULTER, E.M. "Realistic design using large-change sensitivities and performance contours", IEEE Trans. Circuit Theory, vol. CT-18, pp. 58-66, Jan. 1971.

[2] KARAFIN, B.J. "The optimal assignment of component tolerances for electrical networks", Bell Syst. Tech. J., vol. 50, no. 4 pp. 1225-1242, Apr. 1971.

[3] BOHLING, D.M. and O'NEILL, L.A. "An interactive computer approach to tolerance analysis", IEEE Trans. Comput., vol. C-19, pp. 10-16, Jan. 1970.

[4] LEUNG, K.H. and SPENCE, R. "Efficient statistical circuit analysis", Electron. Lett., vol. 10, no. 17, pp. 360-362, Aug. 1974.

[5] LEUNG, K.H. and SPENCE, R. "Efficient frequency-domain statistical circuit analysis", Proc. 1976 IEEE Int. Symp. Circuits and Systems, Munich.

[6] SO, H.C. "Analysis and interactive design of networks using on-line simulation", System Analysis by Digital Computer, F.F. Kuo and J.F. Kaiser, Eds., New York, Wiley 1966, Chapter 2.

[7] MAGNUSON, W.G. Jr., KUO, F.F. and WALSH, W.J. "GINA - A computer graphics analysis system", IEEE Trans. Circuit Theory, vol. CT-16, pp. 389-391, Aug. 1969.

[8] SPENCE, R. and APPERLEY, M.D. "On the use of interactive graphics in circuit design", Proc. 1974 IEEE Int. Symp. Circuits and Systems, pp. 558-563.

[9] APPERLEY, M.D. "User-oriented interactive graphics", Proc. 1974 DECUS Europe Conference, Zurich.

[10] VILLALAZ P.A. and SPENCE, R. "Scheme for the elimination of redundant model complexity", Electron. Lett., Vol. 8, no. 2, pp. 38-40, Jan. 1972.

[11] KUO, F.F. Network Analysis and Synthesis, 2nd Ed., New York: Wiley, 1966, pp. 111-126.

[12] SESHU, S. and REED, M.B. Linear Graphs and Electrical Networks Reading, Mass.: Addison Wesley, 1961, pp. 315-330.

[13] WEINBERG, L. Network Analysis and Synthesis, New York: McGraw-Hill, 1962.

[14] CALAHAN, D.A. Computer-Aided Network Design, Rev. Ed., New York: McGraw-Hill, 1972, pp. 13, 1974.

[15] SPENCE, R. and APPERLEY, M.D. "The interactive-graphic man-computer dialogue in computer-aided circuit design", Proc. 1976 IEEE Int. Symp. Circuits and Systems, Munich.

[16] LEEDS, J.V. and UGRON, G.I. "Simplified multiple parameter sensitivity calculation and continuously equivalent networks", IEEE Trans. Circuit Theory, vol. CT-14, pp. 188-191, June 1967.

[17] BORDEWIJK, J.L. "Inter-reciprocity applied to electrical networks", Appl. Sci. Res., sect. B, vol. 6, pp. 1-74, 1956.

[18] DIRECTOR, S.W. and ROHRER, R. "The generalized adjoint network and network sensitivities", IEEE Trans. Circuit Theory, vol. CT-16, pp. 318-323, Aug. 1969.

[19] DIRECTOR, S.W. and ROHRER, R. "Automated network design - The frequency domain case", IEEE Trans. Circuit Theory, vol. CT-16, pp. 330-337, Aug. 1969.

[20] TELLEGEN, B.D.H. "A general network theorem with applications", Philips Res. Rept., vol. 7, pp. 259-269, Aug. 1952.

[21] PENFIELD, P., SPENCE, R. and DUINKER, S. "A general form of Tellegen's Theorem", IEEE Trans. Circuit Theory, vol. CT-17, pp. 302-305, Aug. 1970.

[22] DESOER, C.A. and KUH, E.G. Basic Circuit Theory, New York: McGraw-Hill, 1969, p. 634.

[23] GODDARD, P.J., VILLALAZ, P.A. and SPENCE, R. "Method for the efficient computation of the large-change sensitivity of linear non-reciprocal networks", Electron. Lett., vol. 7, no. 4, pp. 112-113, Feb. 1971.

[24] DESOER, C.A. and KUH, E.S. Basic Circuit Theory, New York: McGraw-Hill, 1969, p. 654.

[25] HOUSEHOLDER, A.S. "A survey of some closed method for inverting matrices", J. SIAM, vol. 5, pp. 155-169, 1957.

[26] BRANIN, F.H. Jr. "The relation between Kron's method and the classical methods of network analysis", Matrix Tensor Quart., vol. 12, no. 3, pp. 69-105, Mar. 1962.

[27] GADENZ, R.N., REZAI-FAKHR, M.G. and TEMES, G.C. "A method for the computation of large tolerance effects", IEEE Trans. Circuit Theory, vol. CT-20, pp. 704-708, Nov. 1973.

[28] GODDARD, P.J. "The calculation and reduction of electrical network sensitivity", Ph.D. Thesis, London University, England, Dec. 1971.

[29] VILLALAZ, P.A. "Theory and techniques of network sensitivity applied to electrical device modelling and circuit design", Ph.D. Thesis, London University, England, Sept. 1972.

[30] SINGHAL, K., VLACH, J. and BRYANT, P.R. "Efficient computation of large-change multiparameter sensitivity", Circuit Theory Appl., vol. 1, pp. 237-247, 1973.

[31] TROOP, W.J. and PESKIN, E. "The transfer function and sensitivity of a network with n variable elements", IEEE Trans. Circuit Theory (Corresp.),vol. CT-16, pp. 242-244, May 1969.

[32] LEUNG, K.H. and SPENCE, R. "Multiparameter large-change sensitivity analysis, and systematic exploration", IEEE Trans. Circuits and Systems, vol. CAS-22, pp. 796-804, Oct. 1975.

[33] VILLALAZ, P.A. "Method for the efficient computation of performance contours", Electron. Lett., vol. 8, no. 23, pp. 575-577, Nov. 1972.

[34] WYLIE, C.R. Jr. Advanced Engineering Mathematics, New York: McGraw-Hill, 1966, p. 735.

[35] NEHARI, Z. Conformal Mapping, New York: McGraw-Hil, 1952, pp. 154-164.

[36] LEUNG, K.H. and SPENCE, R. "Tracking sensitivity: an efficient algorithm for linear nonreciprocal circuits", Electron. Lett. vol. 10, no. 18, pp. 377-378, Sept. 1974.

[37] FADDEEVA, V.N. Computational Method of Linear Algebra, New York: Dover Publications Inc., 1959, ch. 1.

[38] BRYANT, P.R. "Tracking senstivity: an alternative algorithm for linear nonreciprocal circuits", Electron. Lett., vol. 11, no. 5, pp. 114-116, Mar. 1975.

[39] RALSTON, A. A First Course in Numerical Analysis, New York: McGraw-Hill, 1965, p. 512.

[40] WILKINSON, J.H. The Algebraic Eigenvalue Problem, Oxford University Press, 1965, p. 486.

[41] PARLETT, B.N. The LU and QR Algorithms in Mathematical Methods for Digital Computers, A. Ralston and H.S. Wilf, Eds., New York: Wiley, 1967, vol. 2, pp. 116-130.

[42] GANTIMACHER, F.R. The Theory of Matrices, Chelsea Publishing Co., 1959, vol. 1, p. 87.

[43] Special Issue on "Statistical Circuit Design" Bell Syst. Tech. J., vol. 50, Apr. 1971.

[44] HAMMERSLEY,J.M. and HANDSCOMB,D.C. Monte Carlo Methods, New York: John Wiley, 1964.

[45] SCOTT, T.R. and WALKER, T.P. "Regionalization: a method for generating joint density estimates", Proc. 1975 IEEE Int. Symp. Circuits and Systems, San Francisco.

[46] LEUNG, K.H. and SPENCE, R. "Idealized statistical models for low-cost linear circuit yield analysis", IEEE Trans. Circuits and Systems, (Submitted for publication).

[47] SUD, D. "Sensitivity analysis and optimization techniques in the tolerancing and design of electrical networks", Ph.D. Thesis, London University, England, Jan. 1975.

[48] BANDLER, J.W., LIU, P.C. and TROMP, H. "Practical design centering, tolerancing and turning", Proc. 1975 IEEE Int. Symp. Circuits and Systems, pp. 206-209.

[49] DIRECTOR, S.W. and HACHTEL, G.D. "The simplicial approximation approach to design centering and tolerance assignment", IBM Internal paper, Mathematics, RC 5595 (24192), Aug. 1975.

[50] PINEL, J.F. and ROBERTS, K.A. "Tolerance assignment in linear networks using nonlinear programming", IEEE Trans. Circuit Theory, vol. CT-19, pp. 475-479, Sept. 1972.

[51] THORBJORNSEN, A.R. and DIRECTOR, S.W. "Computer-aided tolerance assignment for linear circuit with correlated elements", IEEE Trans. Circuit Theory, vol. CT-20, pp. 518-524, Sept. 1973.

[52] ELIAS, N.J. "New statistical methods for assigning device tolerances", Proc. 1975 IEEE Int. Symp. Circuits and Systems, pp. 329-332.

[53] BRAYTON, R.K. A talk presented in a seminar on circuit sensitivities at the Elect. Eng. Dept., Imperial College, Dec. 1975.

[54] AGNEW, D. A talk presented in a seminar on circuit sensitivities at the Elect. Eng. Dept., Imperial College, Feb. 1976.

[55] VARAH, J.M. "The calculation of the eigenvectors of a general complex matrix by inverse iteration", Mathematics of Computation, vol. 22, no. 104, pp. 785-791, 1968.

[56] GANTMACHER, F.R. "The theory of matrices", Chelsea Publishing Company, vol. 1, p. 87, 1959.