Digital Computer Calculation of Power System Short Circuit and Load Flow Utilising Diakoptics and Sparsity Techniques

by

W.A. Prebble, B.Sc., B.E.

submitted in fulfilment of

the requirements for the degree of

Master of Engineering

UNIVERSITY OF TASMANIA

HOBART

3 October 1975

The thesis contains no material which has been accepted for the award of any other degree or diploma in any university, and that, to the best of my knowledge and belief, the thesis contains no copy or paraphrase of material previously published or written by another person, except when due reference is made in the text of the thesis.

W. Arzebbe _______________

SUMMARY

In part A a non-singular connection matrix is used to combine the self and mutual impedance matrix of a group of mutually coupled elements with a network bus impedance matrix; the resulting impedance matrix is then reduced by eliminating rows and columns if necessary, to give the bus impedance matrix of the interconnected network. The self impedances of the mutually coupled group of elements are added to the network bus impedance matrix in the same way as uncoupled elements, then the mutual impedances are added followed by matrix reduction. By considering examples of the connection matrix applied to adding a single element, then to adding groups of uncoupled and coupled elements to a network, rules are devised for combining the self impedances of branch and loop elements and group mutual impedances with the network bus impedance matrix.

From the bus impedance matrix of power system sequence networks fault parameters are derived by simple arithmetic operations. It is shown that rules for adding a group of mutually coupled loop elements can be applied to modify a bus impedance matrix when element self and mutual impedances are changed. The derivation of an equivalent network from the bus impedance matrix is noted; the addition of two network bus impedance matrixes is considered and shown to be a special case of the more general problem of adding a self and mutual impedance matrix to a bus impedance matrix. A numerical example involving the calculation and modification of the bus impedance matrix, deriving an equivalent circuit and adding bus impedance matrixes is included.

An outline of a digital computer power system short circuit programmme which calculates fault parameters from the bus impedance matrix derived from randomly ordered lists of network element self and mutual impedances is given.

The inverse of the connection matrix discussed in part A is used in part B to combine a network bus admittance matrix with the self and mutual admittance matrix of a group of mutually coupled elements. From this the well known method of forming the bus admittance matrix from uncoupled element self admittances follows and is

extended to cover self and mutual admittances of coupled elements. For a group of mutually coupled elements, the diagonal terms of the group admittance matrix are added to the bus admittance matrix in the same way as self admittances of uncoupled elements while the off-diagonal terms are added in a matrix operation either before or after the diagonal terms. A relationship is indicated between the admittance connection matrix and the group element bus incidence matrix.

Although the presence of mutual coupling results in some loss of sparsity, it is shown that for power systems the bus admittance matrix still has a large proportion of zero terms. By eliminating terms below the main diagonal in an optimal order, a "factored inverse" of the admittance matrix is derived which has considerably fewer non-zero terms than the corresponding bus impedance matrix. Terms of the impedance matrix can be obtained from the inverse as required. The numerical calculation of the bus admittance matrix of a power system zero sequence network is set out and derivation of fault impedance and current distribution factors included.

A digital computer programme using the bus admittance matrix and factored inverse method for power system short circuit studies is described and a tabulation indicates the effect on computer storage requirements of the optimal factoring procedure.

In part C Newton's method of power system load flow calculation using Gaussian elimination to solve the voltage correction equations is discussed. The network and problem parameters are specified in rectangular cartesian co-ordinates. As the voltage correction equation matrix has the same form as the bus admittance matrix, a preferred order for the Gaussian elimination which preserves sparsity is devised by analogy with network reduction.

A digital computer load flow programme is outlined and a tabulation included which shows that, for typical power system networks, the preferred elimination order retains sparsity in the matrix.

Algol listings of the digital computer short circuit and load flow programmes are included in the supplement with data and corresponding calculated results for power system studies.

CONTENTS

SUPPLEMENT

S.1. The Digital Calculation of Sequence Networks Including Mutual Impedances. Proc.I.E.E.,Vol.112,April 1965,p.711.

S.2. The Calculation of the Transfer and Driving Point Impedance Matrix by Digital Computer. Matrix and Tensor Quarterly, Vol.15, June 1965, p.126.

S.3. Digital Calculation of Short-Circuit Networks. Elect.Engg.Trans. I.E.Aust., Vol.EE2, No.2, 1966, p.62.

S.4. The Digital Solution of the Load Flow Problem by Elimination. Elect.Engg.Trans.I.E.Aust., Vol.EE4, No.1, 1968, p.23.

S.5. Discussion on paper by Tinney,W.F. and Hart,C.E. - Power Flow Solution by Newton's Method. IEEE Trans. on Power Apparatus and Systems, Vol.PAS-86, Nov. 1967, p.1449.

S.6. Letter from W.F.Tinney of Bonneville Power Administration dated October 3, 1967.

S.7. Discussion on paper by Freris,L.L. and Sasson,A.M. - Investigation of the Load Flow Problem. Proc.IEE, Vol.117, 1970, p.397.

S.8. Digital Computer Programme HEI 2a Short Circuit - Algol Listing and Example of Network Calculation.

S.9. Digital Computer Programme HEI 13 Short Circuit - Algol Listing and Example of Network Calculation.

S.10. Digital Computer Programme HEI 8g Complex Load Flow - Algol Listing and Examples of Network Calculations.

GENERAL INTRODUCTION

The paper, concerning the digital computer solution of power system short circuit and load flow problems using a nodal representation of the revelent network,is divided into three parts. In part A a method, based on Kron's diakoptics (Ref.1 and 2), of deriving the nodal or bus impedance matrix from randomly ordered lists of element self and mutual impedances is discussed. The method which has particular relevance to the inclusion of mutual coupling between network elements may be described as a "geometric" approach in contrast to the algorithms derived by El-Abiad (Ref.23). Subsequent amendments to the digital computer programme HEI 2, which has been used since 1964 for power system short circuit studies, have been to make use of increased computer storage, removal of the option of excluding element resistance and replacing the procedure for grouping mutual impedances by a simplified version. Publications relevent to part A are :-

1. Prebble,W.A. - The Digital Calculation of Sequence Networks Including Mutual Impedances. Proc.I.E.E.,Vol.112,April 1965, p.711. (Awarded the Institution of Electrical Engineers Overseas Premium for 1965).
2. Prebble,W.A. - The Calculation of the Transfer and Driving Point Impedance Matrix by Digital Computer. Matrix and Tensor Quarterly, Vol.15, June 1965, p.126.
3. Prebble,W.A. - Digital Calculation of Short-Circuit Networks. Elect.Engg.Trans.I.E.Aust., Vol.EE2, No.2, 1966, p.62. (Presented at the Institutions Annual Conference, Newcastle, March 1966).

In part B the formation of the network nodal or bus admittance matrix (the reciprocal of the bus impedance matrix) including mutual coupling between network elements is discussed. By applying diakoptics it is shown that in the formation of the admittance matrix from randomly ordered lists of element self and mutual admittances, processing of the mutual can be separated from that of the self admittances with each group of mutual admittances being added at any time during or after the addition of element self admittances to the matrix. The reciprocal relationship between the connectionmatrixes used in the derivation of the bus impedance and admittance matrixes and their relation to the bus incidence matrix is noted. This method of constructing the network bus admittance matrix is used in the digital computer programme HEI 13 which has been in use since 1968 for power system short circuit calculations. Sparsity techniques are utilised in this programme together with triangularisation of the admittance matrix, as outlined by Tinney and Walker (Ref.46), to derive one row at a time of the bus impedance matrix and thus solve problems involving networks too large for the programme HEI 2.

Part C is concerned with the elimination (or Newton's) method of solving power system load flow problems with particular reference to exploiting sparsity of the bus admittance matrix. The order for Gaussian elimination of the voltage correction equations is derived from a consideration of network connections, i.e. the "geometry" of the network, and is suited to the rectangular co-ordinate form in which the equations are written. This procedure differs from that of Tinney et al (Ref.45, 46 and 47) who use polar co-ordinates and matrix triangularisation, but the optimal processing order derived is similar to Method 2 of Tinney and Hart (Ref.47). The digital computer programme HEI 8 at present in use for solving power system load flow problems by ordered elimination is, with the exception of minor amendments, the same as that first used in 1966. Publications relevent to part C are :-

1. Prebble,W.A. - The Digital Solution of the Load Flow Problem by Elimination. Elect.Engg.Trans.I.E.Aust., Vol.EE4, No.1, 1968, p.23. (Presented at the Institution's Power Systems Conference, Melbourne, August 1967).
2. Prebble,W.A. - Discussion on paper by Tinney,W.F. and Hart,C.E. - Power Flow Solution by Newton's Method. IEEE Trans. on Power Apparatus and Systems, Vol.PAS-86, Nov. 1967, p.1449.
3. Prebble,W.A.- Discussion on paper by Freris,L.L. and Sasson,A.M. - Investigation of the Load Flow Problem. Proc.IEE, Vol.117, Feb. 1970, p.397.

Included in the list of references are books and papers relevant to the subject matter of parts A, B or C which have been published subsequent to the developement of the computer programmes HEI 2, HEI 13 and HEI 8 and the following comments refer to these publications.

(a) The book by Stagg and El-Abiad (Ref.5, 1968) has become a stan-reference for digital computer application to short circuit, load flow and transient stability problems. It contains an extensive discussion of network mesh and nodal matrixes including the application to three phase networks, also the derivation of the bus impedance matrix from randomly ordered lists of element self and mutual impedances by the algorithms given in previously published works (Ref.23). The Gauss-Seidel and elimination (Newton's) methods of solving the load flow problem are considered in detail but the important techniques for preserving matrix sparsity are not discussed. Brameller, John and Scott (Ref.6, 1969) in their book apply diakoptics to mesh and nodal analysis of power system networks including assembly of the bus impedance and admittance matrixes from network element lists but mutual coupling between elements is not considered.

(b) Tarsi (Ref.28, 1970) and Dy Liacco and Ramarao (Ref.29, 1970) extend El-Abiad's algorithms for deriving the bus impedance matrix to include the end-fault case. Storry and Brown (Ref.27, 1970) and Daniels and Chen (Ref.30, 1971) suggest first forming the bus impedance matrix of element self impedances, then in a separate operation adding the mutual impedances. In Storry and Brown's method mutual couplings are dealt with one at a time in a two step process associated with network reduction, while Daniels and Chen add all mutual impedances in one operation followed by matrix reduction. The method developed in Part A for separate processing of self and mutual impedances has advantages over both of these procedures as the mutual impedances are added directly into the appropriate matrix terms and no extra matrix reductions are necessary.

(c) Although the formation of the bus admittance matrix from lists of network element self admittances is well known from the load flow problem (e.g. Ref.36), the inclusion of mutual coupling between network elements has not been covered in earlier literature. Nagappan (Ref.32, 1970) develops a step-by-step procedure in which as each element is added allowance is made for mutual coupling with previously processed elements. This is analogous to El-Abiad's method for forming the bus impedance matrix, but is unnecessarily complicated and, as shown in Part B, the concept of branch and loop elements has no relevance when applied to the bus admittance matrix. Anderson, Bowen and Shah (Ref.33, 1970) derive simple rules for adding mutual admittances, calculated from the inverse of group mutual impedance matrixes, directly to the bus admittance matrix thereby avoiding any multiplication with a bus incidence matrix. Tinney (Ref.34, 1972) extends the use of the factorised inverse formed from the admittance matrix by showing how to incorporate network modifications, including that of varying mutual coupling between network elements, without changing the factorisation.

(d) Many different methods of solving power system load flow problems have been devised and a large number of papers on this subject published; Gupta and Davies (Ref.39, 1961) giving a comprehensive list of those appearing prior to 1961. Although new procedures are still being suggested, it is now recognised that the elimination (Newton's) method method combined with an optimum order of processing the equations such as that developed in Part C or that of Tinney and Hart (Ref.47, 1967) is the best way of solving power system load flow prolems.

LIST OF SYMBOLS

$\boxed{}$	= denotes a matrix.
$\boxed{Z}$ $\boxed{Z_A}$	= network nodal or bus impedance matrix.
$\boxed{Y}$ $\boxed{Y_A}$	= network nodal or bus admittance matrix.
Z_{ij} Y_{ij}	= term i-j of $\boxed{Z}$, $\boxed{Y}$.
$\boxed{G}$ $\boxed{B}$	= real, imaginary components of $\boxed{Y}$.
G_{ij} B_{ij}	= term i-j of $\boxed{G}$, $\boxed{B}$.
$\boxed{z_{qp}}$ $\boxed{y_{qp}}$	= group impedance, admittance matrix.
$\boxed{z_s}$ $\boxed{z_m}$	= self, mutual impedance component of $\boxed{z_{qp}}$.
$\boxed{r_s}$ $\boxed{x_s}$	= resistive, reactive components of $\boxed{z_s}$.
$\boxed{y_s}$ $\boxed{y_m}$	= equivalent self, mutual admittance components of $\boxed{y_{qp}}$.
$\boxed{Z'}$ $\boxed{Y'}$	= impedance, admittance matrix of partial network combined with group self impedances, admittances.
$\boxed{Z''}$ $\boxed{Y''}$	= impedance, admittance matrix of partial network combined with group self and mutual impedances, admittances.
$\boxed{V}$ $\boxed{V_A}$	= nodal voltage matrix.
$\boxed{I}$ $\boxed{I_A}$	= nodal current matrix.
$\boxed{V'}$ $\boxed{I'}$	= nodal voltage, current matrix for partial network.
V_k I_k	= nodal voltage, current at node "k".
e_k f_k	= real, imaginary components of V_k.
a_k b_k	= real, imaginary components of I_k.
I_{ed}	= current in element e-d (e>d).
$\boxed{S}$	= nodal power matrix.
S_k	= nodal power at node "k".
$\boxed{P}$ $\boxed{Q}$	= real, reactive components of $\boxed{S}$.
P_k Q_k	= real, reactive components of S_k.
S_k' P_k' Q_k'	= specified power and components at node "k".
V_k'	= specified voltage magnitude at node "k".
$\boxed{C_a}$ $\boxed{C_b}$	= sub-matrixes of the impedance connection matrix.
$\boxed{A_a}$ $\boxed{A_b}$	= sub-matrixes of the admittance connection matrix.
$\boxed{A_b'}$	= matrix formed by discarding columns of $\boxed{A_b}$.
$\boxed{A}$	= bus incidence matrix.
$\boxed{Z_{eqiv}}$	= bus impedance matrix of equivalent network.
$\boxed{Y_{eqiv}}$	= bus admittance matrix of equivalent network.
$\boxed{Z_{diag}}$	= lower diagonal matrix of $\boxed{Z''}$ corresponding to group elements.

$\boxed{Z_{factor}}$		= factored inverse form of $\boxed{Z}$ derived from $\boxed{Y}$.
$\boxed{M}$		= voltage correction equation matrix.
$\boxed{U}$	$\boxed{H}$	
$\boxed{W}$	$\boxed{T}$	= sub-matrixes of $\boxed{M}$.
U_{ij}	H_{ij}	
W_{ij}	T_{ij}	= term i-j of the sub-matrixes of $\boxed{M}$.
t		= subscript denoting matrix transpose.
*		= superscript denoting complex conjugate.
Δ		= denotes incremental value.
Σ		= denotes summation.
$\mid\ \mid$		= denotes modulus of a complex quantity.
$\boxed{I_{qp}}$		= matrix of currents in group elements (q>p).
$\boxed{V_q - V_p}$		= matrix of voltages applied to group elements.

A. THE BUS IMPEDANCE MATRIX FOR POWER SYSTEM FAULT CALCULATIONS

A.1. INTRODUCTION

Before the advent of the digital computer, short circuit calculations on small power system networks were done by hand using familiar network reduction techniques and neglecting element resistance. Disadvantages of this method are :-

(a) separate network reductions are required for each fault;

(b) inclusion of resistance and mutual coupling between elements makes the calculatin too laborious for any but the simplest networks;

(c) derivation of element fault currents and node voltages involves "back-tracking" through the network reduction. These disadvantages were overcome by representing the network on a plugging board which allowed simulation of faults at all nodes.

Early digital computer methods (Ref.3) used a loop or mesh representation of the network (Ref.18-21) which requires matrix inversion or, alternatively iterative procedures were used (Ref.14-17). However, it was recognised that the nodal or bus impedance matrix provided the best approach (Ref.3) as all fault calculations can be done by simple arithmetic operations on the matrix terms thus avoiding inversion or iterative procedures and network coding is simpler than for the mesh approach. Brown and Person (Ref.24) have shown that the bus impedance matrix can be formed automatically from randomly ordered element impedance lists and this has been extended to include mutual coupling between elements by El-Abiad (Ref.23). The method of including mutual coupling involves inverting matrixes and applying corrections to the impedance matrix row and column corresponding to the coupled element to allow for its direct and indirect coupling to network elements already included in the matrix.

By processing all elements directly and indirectly coupled together, it is shown that mutual impedances can be added into the impedance matrix, thus avoiding matrix inversion and calculating term corrections. The self impedances of the coupled elements are added, applying the rules for processing uncoupled elements, to the impedance matrix followed by the addition of the group mutual impedances. The rules are extended to the modification of a given bus impedance matrix when network element self and mutual impedances are changed. It is shown that an equivalent circuit for a given network can be derived from the network bus impedance matrix and also, that adding the bus impedance matrix of one network to that of a second network (forming the bus impedance matrix of the combined network) is a special case of adding the self and mutual impedance matrix of a group of mutually coupled elements to the bus impedance matrix of a partial network.

A digital computer programme for calculating the bus impedance matrix of sequence networks by this method is outlined.

A.2. COMBINATION OF GROUP IMPEDANCE AND BUS IMPEDANCE MATRIXES.

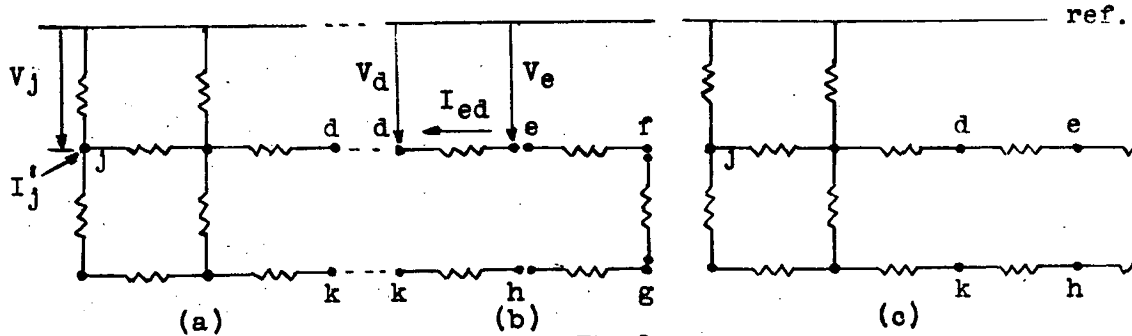


Fig.1

Fig. 1(a) represents a network with n+1 nodes, one of which is chosen as reference or base to which the voltages V' at the other n nodes are referred. The relation between these n voltages and the corresponding n currents I' flowing into the network at the nodes is given by the equation written in matrix notation :-

$$\boxed{V'_A} = \boxed{Z_A}\ \boxed{I'_A} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (1)$$

where $\boxed{V'_A}$, $\boxed{I'_A}$ are column matrixes of the n node voltages and currents and $\boxed{Z_A}$ is the square n×n transfer and driving point impedance matrix of the network. In the following it is shown how $\boxed{Z_A}$, which for simplicity is referred to as the bus impedance matrix, is built up in a step by step procedure from lists of self and mutual impedances of the network elements.

Fig. 1(b) represents a group of m lines or elements, at this stage not physically connected, between any pair of which there may be mutual coupling. For this group the matrix equation

$$\boxed{V_q - V_p} = \boxed{z_{qp}}\ \boxed{I_{qp}} \quad . \quad . \quad . \quad . \quad . \quad . \qquad (2)$$

holds where $\boxed{V_q - V_p}$ and $\boxed{I_{qp}}$ are column matrixes of the m voltages across and m currents flowing in each element, and $\boxed{z_{qp}}$ the m×m matrix with the self impedances of the elements as diagonal terms and the mutual impedances between the elements as the off-diagonal terms. V_q, V_p are the voltages, referred to a common reference, at the nodes of the element q-p and I_{qp} is the current in this element assuming the positive direction of flow from q to p, where q is numerically greater than p.

Combining eqns (1) and (2) into one matrix equation :-

$$\begin{array}{|c|}\hline V'_A \\ \hline V_q - V_p \\ \hline \end{array} = \begin{array}{|c|c|}\hline Z_A & 0 \\ \hline 0 & z_{qp} \\ \hline \end{array}\ \begin{array}{|c|}\hline I'_A \\ \hline I_{qp} \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \qquad (3)$$

The group of elements (b) are connected together and and connected to the network (a) to form the interconnected network shown in Fig. 1(c). The bus impedance matrix of the interconnected network (c) will be derived from eqn (3).

The group of elements (b) are connected together and to the network (a) in such a way that the power flowing into the nodes before and after interconnection is unchanged, i.e. the references for (a) and (b) are connected together, the node voltages are the same and the current flowing in each element is the same before and after interconnection.

The equations relating the currents before and after interconnection can be written :-

$$\begin{bmatrix} I'_A \\ I_{qp} \end{bmatrix} = \begin{bmatrix} 1 & C_a \\ 0 & C_b \end{bmatrix} \begin{bmatrix} I_A \\ I_B \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad (4)$$

where $\begin{bmatrix} I'_A \\ I_{qp} \end{bmatrix}$ and $\begin{bmatrix} I_A \\ I_B \end{bmatrix}$ are column matrixes with n+m rows and

$\begin{bmatrix} 1 & C_a \\ 0 & C_b \end{bmatrix}$ is the square connection matrix with n+m rows and columns.

In eqn.(4) the connection matrix is sub-divided into four sub -matrixes: a unit matrix of order n, a zero matrix with m rows and n columns, $\boxed{C_a}$ with n rows and m columns and $\boxed{C_b}$ a square matrix with m rows and columns. The terms of $\boxed{C_a}$ and $\boxed{C_b}$ are +1, -1 or 0. It is shown in eqns.(183 and 204) that $\boxed{C_b}\boxed{C_a}$ are related to the incidence matrix for the group of lines (b).

The connecting of the elements of (b) to network (a) forming the interconnected network (c) can be considered as a step-by-step procedure - firstly element d-e is connected establishing a new node e in the interconnected network, then elements e-f, k-h and h-g in that order establishing new nodes f, h and g respectively and finally element f-g which connects established nodes f and g. In eqn. (4) the n terms $\boxed{I_A}$ are the node currents at nodes in (c) corresponding to nodes in (a) while the m terms $\boxed{I_B}$ are node currents at new nodes established by the elements (b) and element currents for elements of (b) which connect established nodes.

In this discussion of eqn. (4) mutual coupling between the elements (b) has been ignored. It is shown below that the bus impedance matrix of network (c) is formed from that for network (a) by firstly processing the self impedances of the elements (b) and then adding in the mutual impedances.

The discussion also shows that in the step-by-step formation of the bus impedance matrix of the interconnected network (c), an element of (b) cannot be added unless at least one of its nodes is established in the network - this node may be the reference node. Thus in the formation of the interconnected network and its bus impedance matrix from a partial network and a group of elements or, simply from a group of elements, two types of element arise :-

(1) a branch element i.e. one which connects a new node to an established node thus forming a new node in the interconnected network; and

(2) a loop element i.e. one which connects two established nodes in the network.

A.2.1. CURRENT CONNECTION MATRIX.

From eqn.(4) the relation between the element currents of the group of elements (b) and the node and element currents of the interconnected network (c) is given by :-

$$\boxed{I_{qp}} = \boxed{C_b} \boxed{I_B} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (5)$$

If in Fig. 1 it is assumed that the group of elements (b) are added to the network (a) so that the branch elements d-e, e-f, k-h, and h-g form new nodes e, f, h and g respectively in the interconnected network (c) while loop element g-f connects nodes g and f, also if e>d, f>e, k>h, h>g and g>f then the terms of eqn.(5) for the branch elements are :-

$$I_{ed} = I_e + I_f + I_{gf} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (6)$$

$$I_{fe} = I_f + I_{gf} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (7)$$

$$I_{kh} = -(I_h + I_g - I_{gf}) = -I_h - I_g + I_{gf} \quad . \quad . \quad (8)$$

$$I_{hg} = -(I_g - I_{gf}) = -I_g + I_{gf} \quad . \quad . \quad . \quad . \quad (9)$$

and for the loop element g-f is :-

$$I_{gf} = I_{gf} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (10)$$

Combining eqns.(6) to (10) into one matrix equation corresponding to eqn.(5) :-

$$\begin{bmatrix} I_{ed} \\ I_{fe} \\ I_{kh} \\ I_{hg} \\ I_{gf} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & -1 & -1 & 1 \\ 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I_e \\ I_f \\ I_h \\ I_g \\ I_{gf} \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad (11)$$

Because the element currents are listed in the order inwhich the corresponding elements of (b) are added to form the interconnected network (c) and the node currents are listed in the order in which the corresponding nodes are established in (c) with the loop currents last, the terms below the main diagonal of the connection sub-matrix $\boxed{C_b}$ in eqn.(11) are zero. Consider the addition of branch element e-f to the interconnected network :- as node e is established, the element current I_{fe} does not depend on the node current I_e and hence the term to the left of the main diagonal in the second row of the connection sub-matrix in eqn.(11) is zero. Similarly the terms to the left of the main diagonal in all rows corresponding to branch elements are zero, while for loop elements all the terms in the corresponding rows except the main diagonal terms are zero.This is a general property of $\boxed{C_b}$ and is used in deriving this sub-matrix in the calculation of the bus impedance matrix by digital computer (page 9, section A.3.).

From eqn.(4) the relation between the node currents of network (a) and the node and element currents of the interconnected network (c) is given by :-

$$\boxed{I'_A} = \boxed{I_A} + \boxed{C_a}\,\boxed{I_B} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad .(12)$$

For nodes d and k of the network Fig.1(a) to which elements of the group (b) are connected in forming the interconnected network (c), the terms of eqn.(12) are :-

$$I'_d = I_d + I_{ed} = I_d + I_e + I_f + I_{gf} \quad . \ (13)$$

$$I'_k = I_k - I_{kh} = I_k + I_h + I_g - I_{gf} \quad . \ (14)$$

Combining eqns.(13) and (14) into one matrix equation :-

$$\begin{bmatrix} I'_d \\ I'_k \end{bmatrix} = \begin{bmatrix} I_d \\ I_k \end{bmatrix} + \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} I_e \\ I_f \\ I_h \\ I_g \\ I_{gf} \end{bmatrix} \quad . \quad . \quad . \quad (15)$$

Writing eqns.(11) and(15) as one equation which relates the node and branch currents of networks (a) and (b) in Fig.1 to the currents in the interconnected network (c) :-

$$\begin{bmatrix} I'_d \\ I'_k \\ I_{ed} \\ I_{fe} \\ I_{kh} \\ I_{hg} \\ I_{gf} \end{bmatrix} = \left[\begin{array}{cc|ccccc} 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \begin{bmatrix} I_d \\ I_k \\ I_e \\ I_f \\ I_h \\ I_g \\ I_{gf} \end{bmatrix} \quad \ldots\ldots \quad (16)$$

Eqn. (16) is eqn. (4) written out in full omitting the rows and columns corresponding to node currents such as I_j of Fig.1(a) which are the same before and after interconnection of (a) and (b) to form the network Fig.1(c), i.e. $\boxed{C_b}$ is written out in full but the rows of $\boxed{C_a}$ in which all terms are zero are omitted.

A.2.2. VOLTAGE CONNECTION MATRIX.

From eqn. (3) the voltages before interconnection are $\begin{bmatrix} V'_A \\ V_q - V_p \end{bmatrix}$ and if the voltages after connecting networks Fig.1(a) and (b) to form (c) are $\begin{bmatrix} V_A \\ V_B \end{bmatrix}$, then the equation relating these voltages is derived on the basis of constant power before and after interconnection as follows :-

$$\begin{bmatrix} V_{At} & V_{Bt} \end{bmatrix} \begin{bmatrix} I_A^* \\ I_B^* \end{bmatrix} = \begin{bmatrix} V'_{At} & (V_q - V_p)_t \end{bmatrix} \begin{bmatrix} I_A'^* \\ I_{qp}^* \end{bmatrix} \quad \ldots\ldots \quad (17)$$

where * denotes the conjugate of the complex current terms and the subscript t the transpose of the voltage matrix terms. From eqn. (4) :-

$$\begin{bmatrix} V_{At} & V_{Bt} \end{bmatrix} \begin{bmatrix} I_A^* \\ I_B^* \end{bmatrix} = \begin{bmatrix} V'_{At} & (V_q - V_p)_t \end{bmatrix} \begin{bmatrix} 1 & C_a \\ 0 & C_b \end{bmatrix} \begin{bmatrix} I_A^* \\ I_B^* \end{bmatrix} \quad \ldots\ldots \quad (18)$$

as there are no complex terms in the connection matrix. Hence, from eqn. (18) and after transposing the relation between the voltages before and after connecting the group of lines Fig.1(b) to the network (a) to form (c) is :-

$$\begin{bmatrix} V_A \\ V_B \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ C_{at} & C_{bt} \end{bmatrix} \begin{bmatrix} V'_A \\ V_q - V_p \end{bmatrix} \quad \ldots\ldots \quad (19)$$

From eqn. (19) :-

$$\boxed{V_A} = \boxed{V'_A} \quad \ldots\ldots \quad (20)$$

i.e. the voltages at the nodes of the partial network Fig.1(a) remain unchanged when connected with the group of elements (b) to form the network (c) - this follows from the specified condition of constant power before and after interconnection.

From eqns.(19) and (20) :-

$$\boxed{C_{at}}\,\boxed{V_A} + \boxed{C_{bt}}\,\boxed{V_q - V_p} = \boxed{V_B} \quad . \; . \; . \; . \; . \; . \; .(21)$$

From eqn.(11) :-

$$\boxed{C_{bt}} = \begin{array}{|c|c|c|c|c|}\hline 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & -1 & 0 & 0 \\ \hline 0 & 0 & -1 & -1 & 0 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline \end{array} \quad . \; . \; . \; . \; . \; . \; .(22)$$

and from eqn.(15) :-

$$\boxed{C_{at}} = \begin{array}{|c|c|}\hline 1 & 0 \\ \hline 1 & 0 \\ \hline 0 & 1 \\ \hline 0 & 1 \\ \hline 1 & -1 \\ \hline \end{array} \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; .(23)$$

On substituting eqns.(22) and (23) into eqn.(19), writing in the terms of $\boxed{V_q - V_p}$ and using eqn.(20) but omitting the terms of $\boxed{V_A}$ not common to the networks Fig.1(a) and (b) then multiplying out the left hand side gives :-

$$\begin{array}{|c|c||c|c|c|c|c|}\hline 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & -1 & -1 & 0 \\ \hline 1 & -1 & 1 & 1 & 1 & 1 & 1 \\ \hline \end{array} \; \begin{array}{|c|}\hline V_d \\ \hline V_k \\ \hline V_e - V_d \\ \hline V_f - V_e \\ \hline V_k - V_h \\ \hline V_h - V_g \\ \hline V_g - V_f \\ \hline \end{array} = \begin{array}{|c|}\hline V_d \\ \hline V_k \\ \hline V_e \\ \hline V_f \\ \hline V_h \\ \hline V_g \\ \hline 0 \\ \hline \end{array} \quad . \; . \; . \; .(24)$$

From the right hand side of eqn.(24), which gives the voltages in the interconnected network Fig.1(c), it follows that the terms of $\boxed{V_B}$ corresponding to branch elements from the group Fig.1(b) are node voltages of the new nodes eg. V_e for branch element d-e, and zero corresponding to loop elements eg. loop element g-f.

A.2.3. IMPEDANCE MATRIX OF INTERCONNECTED NETWORK.

The matrix equation relating the node voltages and currents of the interconnected network Fig.1(c) is now formed using eqn.(3) which relates voltages and currents before the group of elements Fig.1(b) are connected to the partial network Fig.1(a). Premultiplying both sides of eqn. (3) by $\begin{bmatrix} 1 & 0 \\ C_{at} & C_{bt} \end{bmatrix}$ and using eqns. (19) and (4) gives the matrix equation :-

$$\begin{bmatrix} V_A \\ V_B \end{bmatrix} = \boxed{Z'} \begin{bmatrix} I_A \\ I_B \end{bmatrix} \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \qquad (25)$$

where

$$\boxed{Z'} = \begin{bmatrix} 1 & 0 \\ C_{at} & C_{bt} \end{bmatrix} \begin{bmatrix} Z_A & 0 \\ 0 & z_{qp} \end{bmatrix} \begin{bmatrix} 1 & C_a \\ 0 & C_b \end{bmatrix} \quad . \; . \; . \; . \; . \qquad (26)$$

$$= \begin{bmatrix} Z_A & Z_A C_a \\ C_{at} Z_A & C_{at} Z_A C_a + C_{bt} z_{qp} C_b \end{bmatrix} \quad . \; . \; . \; . \; . \qquad (27)$$

From eqn. (27) it follows that $\boxed{Z'}$ is symmetrical provided that $\boxed{Z_A}$ and $\boxed{z_{qp}}$ are symmetrical, because the product $\boxed{C_{at} Z_A}$ is the transpose of the product $\boxed{Z_A C_a}$ and the products $\boxed{C_{at} Z_A C_a}$ and $\boxed{C_{bt} z_{qp} C_b}$ are symmetrical. In general $\boxed{z_{qp}}$ is symmetrical as it is a matrix of self and mutual couplings of a group of elements – when there are no mutual couplings it is a diagonal matrix – and it can be shown that $\boxed{Z_A}$ is symmetrical by applying the above argument to building up a network starting with an element connected to the reference node for which $\boxed{Z_A}$ has one term, namely the impedance of the element. Hence $\boxed{Z'}$ is, in general, symmetrical which is important for computation as only its upper or lower triangular part is required for calculation and storage.

From eqns. (25) and (27) the equation relating the voltages and currents of the interconnected network Fig.1(c) is :-

$$\begin{bmatrix} V_A \\ V_B \end{bmatrix} = \begin{bmatrix} Z_A & Z_A C_a \\ C_{at} Z_A & C_{at} Z_A C_a + C_{bt} z_{qp} C_b \end{bmatrix} \begin{bmatrix} I_A \\ I_B \end{bmatrix} \quad . \; . \; . \; . \; . \qquad (28)$$

The impedance matrix in eqn. (28) has n+m rows and columns, namely a row and column corresponding to each node of the network Fig.1(a) and to each element of the group Fig.1(b) ie. there is a row and column corresponding to each node of the interconnected network Fig.1(c) with an additional row and column for the element of the group which forms a loop. As the term of $\boxed{V_B}$ corresponding to the loop element is zero, eqn. (24), the row and column of the impedance matrix in eqn. (28) can be eliminated by Kron's rule (ref. 1) leaving a matrix with a row and column for each node, excluding the reference node, of the network Fig.1(c) ie. the bus impedance matrix of this network.

For a group of elements Fig.1(b) the matrix $\boxed{z_{qp}}$ in eqn.(28) can be written as the sum of two matrixes :-

$$\boxed{z_{qp}} = \boxed{z_s} + \boxed{z_m} \quad . . . . (29)$$

where $\boxed{z_s}$ is a diagonal matrix with diagonal terms equal to the self impedances of the elements of Fig.1(b) and $\boxed{z_m}$ is the matrix of the mutual impedances coupling these elements ie. a matrix with zero diagonal. Thus the term $\boxed{C_{bt} z_{qp} C_b}$ may be written :-

$$\boxed{C_{bt} z_{qp} C_b} = \boxed{C_{bt} z_s C_b} + \boxed{C_{bt} z_m C_b} \quad . . . . (30)$$

In the discussion leading to eqns.(16) and (24) which respectively relate the currents and voltages in Figs.1(a) and (b) before inter-connection to those in Fig.1(c) after interconnection, it is assumed that there are no mutual couplings in the group of elements (b), i.e. $\boxed{z_m} = 0$, and on this assumption eqn.(27), the impedance matrix for the interconnected network Fig.1(c), is derived. However, inspection of eqns.(6)-(10) and (13)-(14) relating node and element currents shows that this restriction is unnecessary, i.e. $\boxed{C_b}$ and $\boxed{C_a}$ are the same with and without mutual couplings.Hence, when mutual couplings are present eqn.(30) shows that eqn.(27) may be written :-

$$\boxed{Z''} = \begin{array}{|c|c|}\hline Z_A & Z_A C_a \\ \hline C_{at} Z_A & C_{at} Z_A C_a + C_{bt} z_s C_b + C_{bt} z_m C_b \\ \hline \end{array} \quad . . . . (31)$$

$$= \begin{array}{|c|c|}\hline 0 & 0 \\ \hline 0 & C_{bt} z_m C_b \\ \hline \end{array} + \boxed{Z'} \quad . . . . (32)$$

where $\boxed{Z'}$ is the matrix formed from $\boxed{Z_A}$, the bus impedance matrix of the partial network Fig.1(a), and $\boxed{z_s}$, the self impedances of the group of elements Fig.1(b). Eqn.(32) shows that the impedance matrix $\boxed{Z''}$ can be formed by first calculating $\boxed{Z'}$ and then adding the matrix product $\boxed{C_{bt}}$ $\boxed{z_m}$ $\boxed{C_b}$ formed from the mutual couplings between the elements of the group and the partial connection matrix $\boxed{C_b}$. This is important in the digital calculation of a network bus impedance matrix, as $\boxed{z_s}$ is a diagonal matrix and simple rules can be derived for augmenting the bus impedance matrix of a partial matrix with rows and columns corresponding to the branch and loop elements of a group being added to the network.

A.2.4. RULES FOR FORMING BUS IMPEDANCE MATRIX.

Summarising, the bus impedance matrix $\boxed{Z}$ of an intercon-nected network formed by combining a partial network having a bus impedance matrix $\boxed{Z_A}$ and a group of elements of self impedance $\boxed{z_s}$

and mutual impedance $\boxed{z_m}$ is calculated in three steps :-

(a) the matrix $\boxed{Z_A}$ is augmented by rows and columns corresponding firstly to the branch elements and secondly to the loop elements of the group, i.e. the matrix products $\boxed{Z_A C_a}$, $\boxed{C_{at} Z_A}$ and $\boxed{C_{at} Z_A C_t + C_{bt} z_s C_b}$ of eqn.(31) are formed using the element self impedances only. With the digital computer the group elements are processed one at a time instead of calculating matrix products;

(b) the terms of the matrix product $\boxed{C_{bt} z_m C_b}$ calculated from the group mutual coupling impedances $\boxed{z_m}$ and the partial connection matrix $\boxed{C_b}$ are added into the augmented rows and columns formed in (a) and as indicated in eqn.(31); and

(c) the augmented rows and columns which correspond to loop elements are eliminated by Kron's rule - using the digital computer it is convient to eliminate the rows and columns one at a time.

A.3. STEP-BY-STEP CALCULATION OF BUS IMPEDANCE MATRIX.

A.3.1. SINGLE BRANCH ELEMENT.

Starting with the bus impedance matrix $\boxed{Z_A}$ of a given network simple rules are now derived for calculating the bus impedance matrix of the network formed by :-

(a) adding a branch element to the original network; and

(b) adding a loop element to the original network.

It is shown that these rules apply when adding any number of branch elements, or any number of loop elements, or a group of branch and loop elements to the given network. Also, rules for forming the partial connection matrix $\boxed{C_b}$ of eqn.(32) for a group of mutually coupled elements are derived, and thus the bus impedance matrix of an interconnected network formed by adding a group of mutually coupled elements to the given network can be calculated.

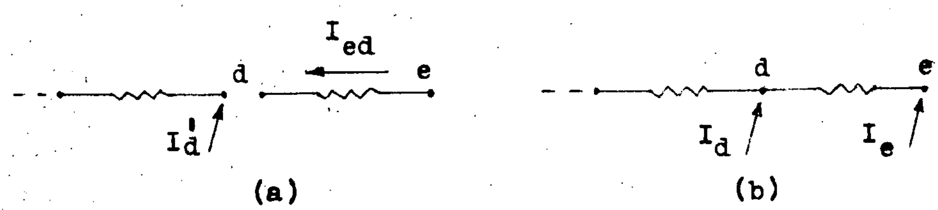


Fig.2

Fig.2(a) and (b) represent respectively conditions before and after the addition of a branch element e-d to node "d" of the network shown in Fig.1(a). If "j" is any node in this network, then the equations corresponding to eqns.(1) and (2) before the branch element is added to the network are :-

$$\begin{bmatrix} \cdot \\ V_j \\ \cdot \\ V_d \\ \cdot \end{bmatrix} = \begin{array}{c} \\ j \\ \\ d \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccc} & j & & d & \end{array} \\ \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{jj} & \cdot & Z_{jd} & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \end{array} \begin{bmatrix} \cdot \\ I'_j \\ \cdot \\ I'_d \\ \cdot \end{bmatrix} \quad \ldots\ldots \quad (33)$$

and $\boxed{V_e - V_d} = \boxed{z_{ed}}\ \boxed{I_{ed}}$ . . . . (34)

The equations corresponding to eqns.(6) and (13) for adding the branch element e-d to the network are :-

$$I_{ed} = I_e \quad \ldots\ldots \quad (35)$$

and $I'_d = I_d + I_{ed} = I_d + I_e$ . . . . (36)

By comparing eqns.(35) and (36) with eqns.(5) and (12) respectively it follows that the connection sub-matrixes for adding the branch element e-d to the network are :-

$$\boxed{C_b} = \boxed{1} \quad \ldots\ldots \quad (37)$$

$$\boxed{C_a} = \begin{array}{c} \\ j \\ \\ d \\ \\ \end{array} \begin{bmatrix} \cdot \\ 0 \\ \cdot \\ 1 \\ \cdot \end{bmatrix} \quad \ldots\ldots \quad (38)$$

i.e. $\boxed{C_b}$ is a unit matrix of order 1 and $\boxed{C_a}$ is a column matrix in which all rows are zero except for row "d" which has the value +1. The impedance matrix for the interconnected network is now found by calculating the matrix products shown in eqn.(27). Using eqn.(38) and $\boxed{Z_A}$ from eqn.(33) it follows that the products $\boxed{Z_A C_a}$ and $\boxed{C_{at} Z_A}$ are respectively column and row matrixes having terms equal to corresponding terms in column and row "d" of $\boxed{Z_A}$. Hence, the product $\boxed{C_{at} Z_A C_a}$ is the diagonal term of $\boxed{Z_A}$ in row "d", i.e.

$$\boxed{C_{at} Z_A C_a} = Z_{dd} \quad \ldots\ldots \quad (39)$$

From eqns.(34) and (37) :-

$$\boxed{C_{bt} z_{qp} C_b} = z_{ed} \quad \ldots\ldots \quad (40)$$

as $\boxed{z_{qp}}$ is a matrix with one term, z_{ed}.

Hence $\boxed{Z'}$ is formed by augmenting $\boxed{Z_A}$ with one row and column :-

$$\boxed{Z'} = \begin{array}{c|cccccc|} & & j & & d & & e \\ \hline & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ j & \cdot & Z_{jj} & \cdot & Z_{jd} & \cdot & Z_{jd} \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ d & \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot & Z_{dd} \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ e & \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot & Z_{dd}+z_{ed} \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \qquad (41)$$

It follows from A.2.3. that $\boxed{Z'}$ is the bus impedance matrix of the interconnected network formed by adding the branch element e-d to the network Fig.1(a).

If d>e then eqns.(35) and (36) become respectively :-

$$I_{de} = -I_e \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (42)$$

and $$I'_d = I_d - I_{de} = I_d + I_e \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (43)$$

i.e. the connection sub-matrix $\boxed{C_a}$ is the same as eqn.(38) and

$$\boxed{C_b} = \boxed{-1} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (44)$$

It follows that the matrix products $\boxed{Z_A C_a}$ and $\boxed{C_{at} Z_A}$ are the same as in the previous case and, as eqns.(39) and (40) also apply, the bus impedance matrix is given by eqn.(41).

If node "d" is the reference node then, as there is no corresponding row and column in the bus impedance matrix $\boxed{Z_A}$, the connection sub-matrix $\boxed{C_a}$ has all terms zero for a branch element e-d and sub-matrix $\boxed{C_b}$ is +1. Hence eqn.(40) applies but, all terms of $\boxed{Z_A C_a}$ and $\boxed{C_{at} Z_A}$ are zero and

$$\boxed{C_{at} Z_A C_a} = 0 \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (45)$$

Therefore the bus impedance matrix after adding a branch element connected from node "e" to the reference node is :-

$$\begin{array}{c} = \; j \\ \boxed{Z'} = \; j \\ \\ e \end{array} \begin{array}{|cccc|} & j & & e \\ \hline \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{jj} & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & 0 & \cdot & z_{ed} \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (46)$$

Note that this follows directly from the relation between current and voltage for an element connected only from node "e" to the reference node, i.e. $V_e = z_{ed} I_{ed} = z_{ed} I_e$ . . . . (47)

Eqn.(41) shows that the bus impedance matrix of the interconnected network formed by adding a branch element e-d to node "d" of a given network is formed by adding a new row and column "e" to the bus impedance matrix of the given network; the off-diagonal terms of row and column "e" being equal to the corresponding terms of row and column "d" and the diagonal term being equal to the diagonal term of row "d" plus the self impedance of the branch element. If the branch element connects node "e" to the reference node, then the new row and column "e" of the bus impedance matrix have all terms zero except the diagonal term which is equal to the self impedance of the element.

A.3.1.1. GROUP OF BRANCH ELEMENTS.

The addition of two branch elements e-d and f-e is now considered where e-d is connected to the existing node "d" of a network and f-e is connected to the newly established node "e".

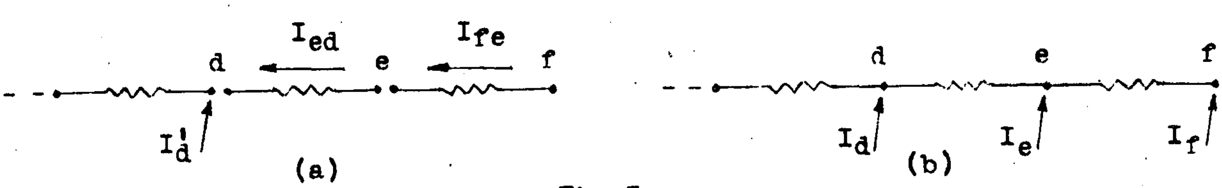


Fig.3

In Fig.3(a) and (b) the two branch elements e-d and f-e are connected to node "d" of the network shown in Fig.1(a); Fig.3(a) showing conditions before and Fig.3(b) conditions after interconnection. Before interconnection eqn.(33) holds for the network while the following equation applies to the branch elements :-

$$\begin{bmatrix} V_e-V_d \\ V_f-V_e \end{bmatrix} = \begin{bmatrix} z_{ed} & 0 \\ 0 & z_{fe} \end{bmatrix} \begin{bmatrix} I_{ed} \\ I_{fe} \end{bmatrix} \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (48)$$

For the connection of the two branch elements to the network the equations corresponding to eqns.(6) and (13) are :-

$$I_{ed} = I_e+I_f \qquad I_{fe} = I_f \quad . \; . \; . \; . \; . \; . \; . \quad (49)$$

and $$I_d' = I_d+I_{ed} = I_d+I_e+I_f \quad . \; . \; . \; . \; . \; . \; . \; . \quad (50)$$

These equations may be written :-

$$\begin{bmatrix} I_{ed} \\ I_{fe} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I_e \\ I_f \end{bmatrix} \quad . \; . \; . \; . \; . \; . \; . \; . \; . \quad (51)$$

and $$\begin{bmatrix} I_d' \end{bmatrix} = \begin{bmatrix} I_d \end{bmatrix} + \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} I_e \\ I_f \end{bmatrix} \quad . \; . \; . \; . \; . \; . \; . \; . \quad (52)$$

Comparing eqns.(51) with (5) and (52) with(12) the connection submatrixes $\boxed{C_b}$ and $\boxed{C_a}$ are :-

$$\begin{bmatrix} C_b \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \quad (53)$$

and $$\boxed{C_a} = \begin{array}{c} \\ j \\ \\ d \\ \\ \end{array} \begin{array}{|c|c|} \hline \cdot & \cdot \\ \hline 0 & 0 \\ \hline \cdot & \cdot \\ \hline 1 & 1 \\ \hline \cdot & \cdot \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \tag{54}$$

From eqn. (54) and $\boxed{Z_A}$ of eqn. (33) it follows that the products $\boxed{Z_A C_a}$ and $\boxed{C_{at} Z_A}$ consist respectively of two identical columns and rows with terms equal to the corresponding terms in column and row "d" of $\boxed{Z_A}$. Hence the product

$$\boxed{C_{at} Z_A C_a} = \begin{array}{|c|c|} \hline Z_{dd} & Z_{dd} \\ \hline Z_{dd} & Z_{dd} \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \tag{55}$$

From eqn. (53) and $\boxed{z_{qp}}$ from eqn. (48) :-

$$\boxed{C_{bt} z_{qp} C_b} = \begin{array}{|c|c|} \hline z_{ed} & z_{ed} \\ \hline z_{ed} & z_{ed}+z_{fe} \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \tag{56}$$

Adding eqns. (55) and (56) gives the diagonal term for the rows and columns "e" and "f" which are added to $\boxed{Z_A}$:-

$$\boxed{Z_{diag}} = \begin{array}{|c|c|} \hline Z_{dd}+z_{ed} & Z_{dd}+z_{ed} \\ \hline Z_{dd}+z_{ed} & Z_{dd}+z_{ed}+z_{fe} \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \tag{57}$$

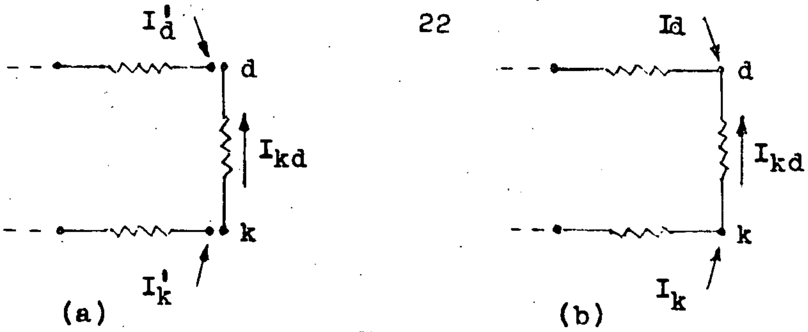
Therefore the matrix $\boxed{Z'}$ is :-

$$\boxed{Z'} = \begin{array}{c|c|c|c|c|c|c|c|} \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{j} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{d} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{e} & \multicolumn{1}{c}{f} \\ \cline{2-8} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cline{2-8} j & \cdot & Z_{jj} & \cdot & Z_{jd} & \cdot & Z_{jd} & Z_{jd} \\ \cline{2-8} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cline{2-8} d & \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot & Z_{dd} & Z_{dd} \\ \cline{2-8} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cline{2-8} e & \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot & \multicolumn{2}{c|}{\multirow{2}{*}{Z_{diag}}} \\ \cline{2-6} f & \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot & \multicolumn{2}{c|}{} \\ \cline{2-8} \end{array} \quad . \quad . \quad . \quad . \quad . \tag{58}$$

which is the bus impedance matrix of the interconnected network.

Eqn. (58) shows that the off-diagonal terms of row and column "f" are equal to the corresponding terms of row and column "e", while the diagonal ~~XXXX~~ term is equal to the diagonal term in row "e" plus the self-impedance of the element f-e. Hence, the rule for adding a single branch element can be extended to a group of branch elements provided that the order in which the individual elements are processed follows the rule before adding an element one of its nodes is established.

A.3.2. <u>SINGLE LOOP ELEMENT</u>.



(a) (b)

Fig.4

Figs.4(a) and (b) represent respectively conditions before and after the addition of a loop element k-d, k>d, connecting nodes "k" and "d" of the network shown in Fig.1(a). If "j" is any node in this network, then the equations corresponding to eqns.(1) and(2) before the loop element is added to the network, i.e. Fig.4(a), are :-

$$\begin{bmatrix} \cdot \\ V_j \\ \cdot \\ V_d \\ \cdot \\ V_k \\ \cdot \end{bmatrix} = \begin{array}{c} \\ \\ j \\ \\ d \\ \\ k \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccccc} & j & & d & & k & \end{array} \\ \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{jj} & \cdot & Z_{jd} & \cdot & Z_{jk} & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot & Z_{dk} & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{kj} & \cdot & Z_{kd} & \cdot & Z_{kk} & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \end{array} \begin{bmatrix} \cdot \\ I_j' \\ \cdot \\ I_d' \\ \cdot \\ I_k' \\ \cdot \end{bmatrix} \quad \ldots\ldots \quad (59)$$

$$\boxed{V_k - V_d} = \boxed{z_{kd}}\ \boxed{I_{kd}} \quad \ldots\ldots\ldots \quad (60)$$

For adding the loop element k-d to the network the equations corresponding to eqns.(10), (13) and (14) are :-

$$I_{kd} = I_{kd} \quad \ldots\ldots\ldots \quad (61)$$

$$I_d' = I_d + I_{kd} \quad \ldots\ldots\ldots \quad (62)$$

$$I_k' = I_k - I_{kd} \quad \ldots\ldots\ldots \quad (63)$$

Writing eqns.(62) and (63) as a matrix equation :-

$$\begin{bmatrix} I_d' \\ I_k' \end{bmatrix} = \begin{bmatrix} I_d \\ I_k \end{bmatrix} + \begin{bmatrix} 1 \\ -1 \end{bmatrix} \boxed{I_{kd}} \quad \ldots\ldots\ldots \quad (64)$$

Comparing eqns.(61) and (64) with eqns.(5) and (12) respectively :-

$$\boxed{C_b} = \boxed{1} \quad \ldots\ldots\ldots \quad (65)$$

and

$$\boxed{C_a} = \begin{array}{c} \\ j \\ \\ d \\ \\ k \\ \\ \end{array} \begin{bmatrix} \cdot \\ 0 \\ \cdot \\ 1 \\ \cdot \\ -1 \\ \cdot \end{bmatrix} \quad \ldots\ldots\ldots \quad (66)$$

i.e. $\boxed{C_a}$ is a column matrix with zero terms except rows "d" and "k" which are +1 and -1 respectively. It follows from eqn.(65) and $\boxed{Z_A}$ from eqn.(59) that $\boxed{Z_A C_a}$ and $\boxed{C_{at} Z_A}$ are column and row matrixes with terms equal to the difference between corresponding terms of columns and rows "d" and "k" respectively. Hence $\boxed{Z'}$ for the interconnected network formed by adding loop element k-d to the network Fig.1(a) is $\boxed{Z_A}$ augmented by a row and column :-

$$\boxed{Z'} = \begin{array}{c} \\ j \\ \\ d \\ \\ k \\ \\ \\ \end{array} \begin{array}{cccccccc} & j & & d & & k & & \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{jj} & \cdot & Z_{jd} & \cdot & Z_{jk} & \cdot & Z_{jd}-Z_{jk} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot & Z_{dk} & \cdot & Z_{dd}-Z_{dk} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{kj} & \cdot & Z_{kd} & \cdot & Z_{kk} & \cdot & Z_{kd}-Z_{kk} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{dj}-Z_{kj} & \cdot & Z_{dd}-Z_{kd} & \cdot & Z_{dk}-Z_{kk} & \cdot & Z_{diag} \end{array} \quad . \quad (67)$$

The product $\boxed{C_{at} Z_A C_a}$ is the difference between terms in rows "d" and "k" of $\boxed{Z_A C_a}$, i.e.

$$\boxed{C_{at} Z_A C_a} = Z_{dd}-Z_{dk}-Z_{kd}+Z_{kk} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (68)$$

while from eqns.(60) and (65) :-

$$\boxed{C_{bt} z_{qp} C_b} = z_{kd} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (69)$$

From eqn.(27) Z_{diag} is the sum of eqns.(68) and (69), i.e.

$$Z_{diag} = Z_{dd}-Z_{dk}-Z_{kd}+Z_{kk}+z_{kd} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (70)$$

From A.2.3. the bus impedance matrix of the interconnected network is found by eliminating the last row and column from $\boxed{Z'}$ using Kron's rule, i.e. Z_{jd} in eqn.(67) is replaced by Z'_{jd} etc. where

$$Z'_{jd} = Z_{jd}-(Z_{jd}-Z_{jk})(Z_{dd}-Z_{kd})/Z_{diag} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (71)$$

If "d" is the reference node, then eqn.(66) becomes :-

$$\boxed{C_a} = \begin{array}{c} \\ j \\ \\ k \\ \\ \end{array} \begin{array}{|c|} \hline \cdot \\ \hline 0 \\ \hline \cdot \\ \hline -1 \\ \hline \cdot \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (72)$$

i.e. all terms are zero except for the term in row "k" which is -1. Therefore the off-diagonal terms of the augmenting column and row formed by the products $\boxed{Z_A C_a}$ and $\boxed{C_{at} Z_A}$ are the negative of the corresponding terms in column and row "k" respectively of $\boxed{Z_A}$ and from eqn.(70) the diagonal term is :-

$$Z_{diag} = Z_{kk} + z_{kd} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (73)$$

Therefore in this case the matrix $\boxed{Z'}$ is :-

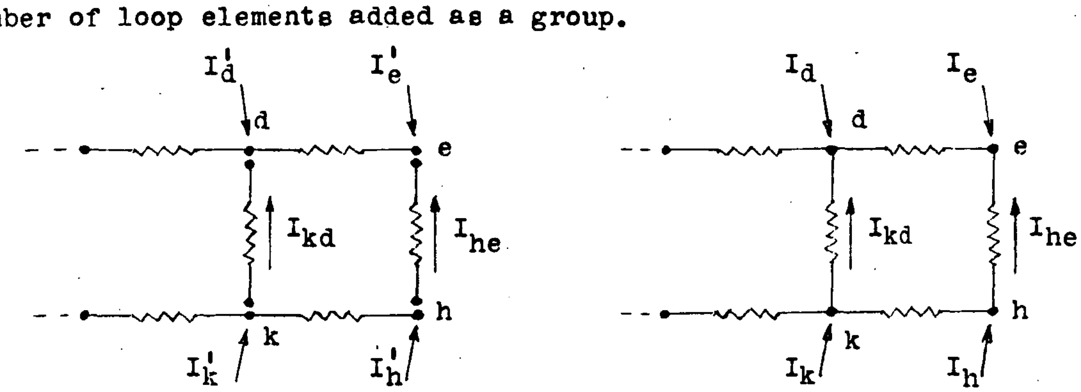
$$\boxed{Z'} = \begin{array}{c} \\ j \\ \\ k \\ \\ \\ \end{array} \begin{array}{|c|c|c|c|c|c|} \multicolumn{1}{c}{} & \multicolumn{1}{c}{j} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{k} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} \\ \hline \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \hline \cdot & Z_{jj} & \cdot & Z_{jk} & \cdot & -Z_{jk} \\ \hline \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \hline \cdot & Z_{kj} & \cdot & Z_{kk} & \cdot & -Z_{kk} \\ \hline \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \hline \cdot & -Z_{kj} & \cdot & -Z_{kk} & \cdot & Z_{diag} \\ \hline \end{array} \quad . \quad . \quad . \quad . \qquad (74)$$

Finally the bus impedance matrix $\boxed{Z_A}$ is formed by eliminating the last row and column of $\boxed{Z'}$.

Summarising, when adding a loop element to a network its bus impedance matrix is augmented by a row and column with terms equal to the difference between corresponding terms of the rows and columns corresponding to the nodes of the loop element and diagonal term equal to the self impedance of the element plus the difference between terms of the augmenting row (or column) corresponding to the nodes of the loop element. If one node of the loop element is the reference node, then the terms of the augmenting row and column are the negative of the corresponding terms in the row and column corresponding to the other node of the element and the diagonal term is equal to the element self impedance plus the negative of the term of the augmenting row (or column) corresponding to the other node of the loop element. Finally the augmenting row and column are eliminated by Kron's rule to give the bus impedance matrix of the new network.

A.3.2.1. GROUP OF LOOP ELEMENTS.

The rule for adding a single loop element to a network is/to a number of loop elements added as a group. extended



(a) (b)

Fig.5

Figs.5(a) and (b) show the relevant parts of a network in which d, e, k and h are established nodes and shows loop elements k-d (k>d) and h-e (h>e) before and after adding to the network. If $\boxed{Z_A}$ is the bus impedance matrix of the network and $\boxed{z_{qp}}$ the impedance matrix of the loop elements, then :-

$$\boxed{Z_A} = \begin{array}{c|ccccccccccc|}
 & & j & & d & & e & & h & & k & \\ \hline
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
j & \cdot & Z_{jj} & \cdot & Z_{jd} & \cdot & Z_{je} & \cdot & Z_{jh} & \cdot & Z_{jk} & \cdot \\
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
d & \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot & Z_{de} & \cdot & Z_{dh} & \cdot & Z_{dk} & \cdot \\
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
e & \cdot & Z_{ej} & \cdot & Z_{ed} & \cdot & Z_{ee} & \cdot & Z_{eh} & \cdot & Z_{ek} & \cdot \\
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
h & \cdot & Z_{hj} & \cdot & Z_{hd} & \cdot & Z_{he} & \cdot & Z_{hh} & \cdot & Z_{hk} & \cdot \\
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
k & \cdot & Z_{kj} & \cdot & Z_{kd} & \cdot & Z_{ke} & \cdot & Z_{kh} & \cdot & Z_{kk} & \cdot \\
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \hline
\end{array} \quad . \quad . \qquad (75)$$

where "j" is any other node in the network; and

$$\boxed{z_{qp}} = \begin{bmatrix} z_{kd} & 0 \\ 0 & z_{he} \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (76)$$

For the loop elements k-d and h-e the equations corresponding to eqns.(10), (13) and (14) relating currents before and after connecting to the network are :-

$$I_{kd} = I_{kd} \qquad I_{he} = I_{he} \quad . \quad . \quad . \quad . \quad . \qquad (77)$$

$$I_d' = I_d + I_{kd} \qquad I_e' = I_e + I_{he} \quad . \quad . \quad . \quad . \quad . \qquad (78)$$

$$I_k' = I_k - I_{kd} \qquad I_h' = I_h - I_{he} \quad . \quad . \quad . \quad . \quad . \qquad (79)$$

Writing eqns.(77) in matrix form :-

$$\begin{bmatrix} I_{kd} \\ I_{he} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I_{kd} \\ I_{he} \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (80)$$

and combining eqns.(78) and (79) into one matrix equation :-

$$\begin{bmatrix} I_d' \\ I_e' \\ I_h' \\ I_k' \end{bmatrix} = \begin{bmatrix} I_d \\ I_e \\ I_h \\ I_k \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & -1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} I_{kd} \\ I_{he} \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (81)$$

then comparing eqns.(5) and (80) gives :-

$$\boxed{C_b} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (82)$$

and comparing eqns.(12) and (81) gives :-

$$\boxed{C_a} = \begin{array}{c} \\ j \\ \\ d \\ \\ e \\ \\ h \\ \\ k \\ \\ \end{array} \begin{bmatrix} \cdot & \cdot \\ 0 & 0 \\ \cdot & \cdot \\ 1 & 0 \\ \cdot & \cdot \\ 0 & 1 \\ \cdot & \cdot \\ 0 & -1 \\ \cdot & \cdot \\ -1 & 0 \\ \cdot & \cdot \end{bmatrix} , \quad . \; . \; . \; . \; . \; . \; . \; . \; . \quad (83)$$

From eqns.(75) and (83) it follows that the products $\boxed{Z_A C_a}$ and $\boxed{C_{at} Z_A}$ are matrixes of two columns and two rows respectively, with terms in the first column and row equal to the difference between the corresponding terms of columns and rows "d" and "k" of $\boxed{Z_A}$ and in the second column and row equal to the difference between the corresponding terms of columns and rows "e" and "h". The terms of the column matrix are indicated below :-

$$\boxed{Z_A C_a} = \begin{array}{c} \\ j \\ \\ d \\ \\ e \\ \\ h \\ \\ k \\ \\ \end{array} \begin{bmatrix} \cdot & \cdot \\ Z_{jd}-Z_{jk} & Z_{je}-Z_{jh} \\ \cdot & \cdot \\ Z_{dd}-Z_{dk} & Z_{de}-Z_{dh} \\ \cdot & \cdot \\ Z_{ed}-Z_{ek} & Z_{ee}-Z_{eh} \\ \cdot & \cdot \\ Z_{hd}-Z_{hk} & Z_{he}-Z_{hh} \\ \cdot & \cdot \\ Z_{kd}-Z_{kk} & Z_{ke}-Z_{kh} \\ \cdot & \cdot \end{bmatrix} \quad . \; . \; . \; . \; . \; . \quad (84)$$

From eqns.(83), (84), (82) and (76) the sum of the matrix products $\boxed{C_{at} Z_A C_a}$ and $\boxed{C_{bt} z_{qp} C_b}$ is a square matrix of order 2 as given below :-

$$\boxed{Z_{diag}} = \boxed{C_{at} Z_A C_a + C_{bt} z_{qp} C_b}$$

$$\begin{bmatrix} Z_{dd}-Z_{dk}-Z_{kd}+Z_{kk}+z_{kd} & Z_{de}-Z_{dh}-Z_{ke}+Z_{kh} \\ Z_{ed}-Z_{ek}-Z_{hd}+Z_{hk} & Z_{ee}-Z_{eh}-Z_{he}+Z_{hh}+z_{he} \end{bmatrix} \quad . \; . \quad (85)$$

The matrix $\boxed{Z'}$ for the interconnected network is formed by augmenting $\boxed{Z_A}$ of eqn.(75) with two columns and rows given by $\boxed{Z_A C_a}$, $\boxed{C_{at} Z_A}$ and $\boxed{Z_{diag}}$.

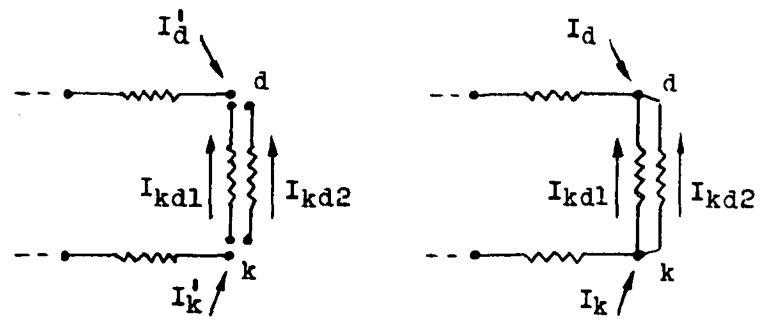
$$\boxed{Z'} = \begin{array}{c|c|c|c|c|c|c|c|c|c|c|c|}
 & & d & & e & & h & & k & & & \\
\hline
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\hline
d & \cdot & Z_{dd} & \cdot & Z_{de} & \cdot & Z_{dh} & \cdot & Z_{dk} & \cdot & Z_{dd}-Z_{dk} & Z_{de}-Z_{dh} \\
\hline
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\hline
e & \cdot & Z_{ed} & \cdot & Z_{ee} & \cdot & Z_{eh} & \cdot & Z_{ek} & \cdot & Z_{ed}-Z_{ek} & Z_{ee}-Z_{eh} \\
\hline
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\hline
h & \cdot & Z_{hd} & \cdot & Z_{he} & \cdot & Z_{hh} & \cdot & Z_{hk} & \cdot & Z_{hd}-Z_{hk} & Z_{he}-Z_{hh} \\
\hline
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\hline
k & \cdot & Z_{kd} & \cdot & Z_{ke} & \cdot & Z_{kh} & \cdot & Z_{kk} & \cdot & Z_{kd}-Z_{kk} & Z_{ke}-Z_{kh} \\
\hline
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\hline
 & \multicolumn{9}{c|}{C_{at} Z_A} & \multicolumn{2}{c|}{Z_{diag}} \\
\hline
\end{array} \quad . \qquad (86)$$

The bus impedance matrix of the interconnected network is then calculated by eliminating the last two rows and columns of $\boxed{Z'}$.

From eqn.(85) it follows that the off-diagonal terms of $\boxed{Z_{diag}}$ can be determined by applying the rule stated in section A.3.2. for adding a loop element to the matrix $\boxed{Z_A}$, i.e. by repeated application of the rule the matrix $\boxed{Z_A}$ is augmented by a row and column corresponding to loop element k-d, then the matrix so formed is augmented by a further row and column corresponding to loop element h-e.

Two special cases involving loop elements are now considered :-

(a) two parallel loop elements, i.e. elements having two common nodes.



(a) (b)

Fig.6

Figs.6(a) and (b) represent the addition of two parallel loop elements to a network of bus impedance $\boxed{Z_A}$ before and after interconnection respectively. In this case the two columns of $\boxed{Z_A C_a}$ and the two rows of $\boxed{C_{at} Z_A}$ are identical having terms equal to the difference between the corresponding terms in columns and rows "d" and "k" of $\boxed{Z_A}$. The terms of $\boxed{Z_{diag}}$ are found by replacing the subscripts "e" and "h" by "d"and "k" respectively in eqn.(85) :-

$$\boxed{Z_{diag}} = \begin{array}{|c|c|} \hline Z_{dd}-Z_{dk}-Z_{kd}+Z_{kk}+z_{kd1} & Z_{dd}-Z_{dk}-Z_{kd}+Z_{kk} \\ \hline Z_{dd}-Z_{dk}-Z_{kd}+Z_{kk} & Z_{dd}-Z_{dk}-Z_{kd}+Z_{kk}+z_{kd2} \\ \hline \end{array} \quad . \quad . \quad (87)$$

The four terms of $\boxed{Z_{diag}}$ formed by the product $\boxed{C_{at} Z_A C_a}$ are identicalwith the self-impedances of the two loop elements being added to the diagonal terms.

(b) two loop elements with one common node.

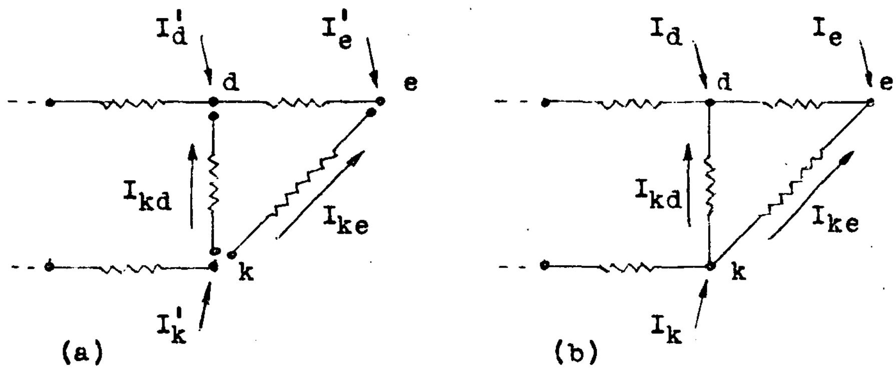


Fig.7

Figs.7(a) and (b) represent the addition of two loop elements k-e and k-d to a network having bus impedance $\boxed{Z_A}$ before and after interconnection respectively. If element k-d is added first the terms of the first augmenting row and column are the same as in eqn.(86) and the terms of the last row and column of $\boxed{Z'}$ are found by replacing subscript "h" by "k" in this row and column in eqn.(86). The terms of $\boxed{Z_{diag}}$ for this case are as indicated :-

$$\boxed{Z_{diag}} = \begin{array}{|c|c|} \hline Z_{dd}-Z_{dk}-Z_{kd}+Z_{kk}+z_{kd} & Z_{de}-Z_{ke}-Z_{dk}+Z_{kk} \\ \hline Z_{ed}-Z_{ek}-Z_{kd}+Z_{kk} & Z_{ee}-Z_{ek}-Z_{ke}+Z_{kk}+z_{ke} \\ \hline \end{array} \quad . \quad . \quad . \quad (88)$$

It is seen that the off-diagonal terms in eqn.(88) are the difference between the "k" and "e" terms of the corresponding row and column.

Summarising the rules for modifying the bus impedance matrix $\boxed{Z_A}$ of a network when adding a number of loop elements are :-

(a) take the elements one at a time augmenting $\boxed{Z_A}$ with a new row and column for each element, the off-diagonal terms being equal to differences between existing row and column terms (as determined by element node numbers) and with a diagonal term equal to the difference between the two terms which correspond to the element node numbers in the new column (or row) plus the element self-impedance; and

(b) eliminate the augmenting rows and columns by Kron's rule eqn.(71).

A.3.3. GROUP OF BRANCH AND LOOP ELEMENTS.

The rules for adding separate groups of branch or loop elements can be applied to adding a mixed group of branch and loop elements to a given network.

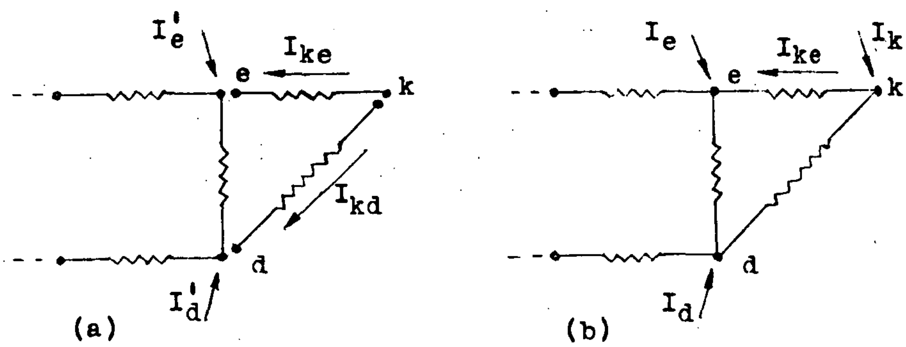


Fig.8

Figs.8(a) and (b) show, before and after interconnection respectively, a branch element k-d (k>d) and a loop element k-e (k>e) which are joined to nodes "d" and "e" of a network having bus impedance matrix $\boxed{Z_A}$ given by :-

$$\boxed{Z_A} = \begin{array}{c|ccccccc|} & & j & & d & & e & \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ j & \cdot & Z_{jj} & \cdot & Z_{jd} & \cdot & Z_{je} & \cdot \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ d & \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot & Z_{de} & \cdot \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ e & \cdot & Z_{ej} & \cdot & Z_{ed} & \cdot & Z_{ee} & \cdot \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \qquad (89)$$

The impedance matrix of the elements is :-

$$\boxed{z_{qp}} = \begin{array}{|c|c|} \hline z_{kd} & 0 \\ \hline 0 & z_{ke} \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (90)$$

The equations corresponding to eqns.(6), (10), (13) and (14) for connecting k-d and k-e to the network are :-

$$I_{kd} = I_k - I_{ke} \qquad I_{ke} = I_{ke} \quad . \quad . \quad . \qquad (91)$$

$$I_d' = I_d + I_{kd} = I_d + I_k - I_{ke} \qquad I_e' = I_e + I_{ke} \quad . \quad . \qquad (92)$$

Writing eqns.(91) in matrix form :-

$$\boxed{\begin{array}{c} I_{kd} \\ \hline I_{ke} \end{array}} = \boxed{\begin{array}{c|c} 1 & -1 \\ \hline 0 & 1 \end{array}} \; \boxed{\begin{array}{c} I_{k} \\ \hline I_{ke} \end{array}} \quad \ldots \ldots \quad (93)$$

and comparing this with eqn.(5) gives :-

$$\boxed{C_b} = \boxed{\begin{array}{c|c} 1 & -1 \\ \hline 0 & 1 \end{array}} \quad \ldots \ldots \quad (94)$$

Writing eqns.(92) as a matrix equation :-

$$\boxed{\begin{array}{c} I'_{d} \\ \hline I'_{e} \end{array}} = \boxed{\begin{array}{c} I_{d} \\ \hline I_{e} \end{array}} + \boxed{\begin{array}{c|c} 1 & -1 \\ \hline 0 & 1 \end{array}} \; \boxed{\begin{array}{c} I_{k} \\ \hline I_{ke} \end{array}} \quad \ldots \ldots \quad (95)$$

and comparing with eqn.(12) gives :-

$$\boxed{C_a} = \begin{array}{c|c|c|} \hline & \cdot & \cdot \\ \hline j & 0 & 0 \\ \hline & \cdot & \cdot \\ \hline d & 1 & -1 \\ \hline & \cdot & \cdot \\ \hline e & 0 & 1 \\ \hline & \cdot & \cdot \\ \hline \end{array} \quad \ldots \ldots \quad (96)$$

From eqns.(89) and (96) it is seen that the products $\boxed{Z_A C_a}$ and $\boxed{C_{at} Z_A}$ are matrixes of two columns and two rows respectively having terms in the first column and row equal to the corresponding terms of $\boxed{Z_A}$ in column and row "d" and in the second column and row equal to the difference between corresponding terms of columns and rows "d" and "e". These products are indicated in $\boxed{Z'}$ below :-

$$\boxed{Z'} = \begin{array}{c|c|c|c|c|c|c|c|c|c|}
 & & j & & d & & e & & & \\ \hline
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \hline
j & \cdot & Z_{jj} & \cdot & Z_{jd} & \cdot & Z_{je} & \cdot & Z_{jd} & Z_{je}-Z_{jd} \\ \hline
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \hline
d & \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot & Z_{de} & \cdot & Z_{dd} & Z_{de}-Z_{dd} \\ \hline
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \hline
e & \cdot & Z_{ej} & \cdot & Z_{ed} & \cdot & Z_{ee} & \cdot & Z_{ed} & Z_{ee}-Z_{ed} \\ \hline
 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \hline
 & \cdot & Z_{dj} & \cdot & Z_{dd} & \cdot & Z_{de} & \cdot & \multicolumn{2}{c|}{} \\ \cline{2-8}
 & \cdot & Z_{ej}-Z_{dj} & \cdot & Z_{ed}-Z_{dd} & \cdot & Z_{ee}-Z_{de} & \cdot & \multicolumn{2}{c|}{Z_{diag}} \\ \hline
\end{array} \quad (97)$$

From eqns.(90), (94), (96) and (97) the square sub-matrix $\boxed{Z_{diag}}$ is :-

$$\boxed{Z_{diag}} = \boxed{C_{at} Z_A C_a + C_{bt} z_{qp} C_b}$$

$$= \boxed{\begin{array}{l|l} Z_{dd}+z_{kd} & Z_{de}-Z_{dd}-z_{kd} \\ \hline Z_{ed}-Z_{dd}-z_{kd} & Z_{ee}-Z_{ed}-Z_{de}+Z_{dd}+z_{kd}+z_{ke} \end{array}} \quad \ldots \ldots \quad (98)$$

Inspection of eqns.(97) and (98) shows that for element k-d a new row and column "k", with terms derived by applying the rule for adding a branch element, is added to $\boxed{Z_A}$ and for element k-e this matrix is then augmented by a row and column with terms calculated by the rule for adding a loop element. The augmenting row and column are then eliminated to give the bus impedance matrix of the interconnected network.

In the examples discussed above of adding branch, loop or groups of branch and loop elements to a given network, it is shown that the bus impedance matrix for the interconnected network is derived from $\boxed{Z_A}$ of the original network by the application of simple rules. These rules which involve repeating, adding or subtracting matrix terms apply generally to any combination of branch and loop elements, because

(a) the terms of the connection sub-matrixes $\boxed{C_a}$ and $\boxed{C_b}$ are 0, +1 or -1 depending on the type of element, its node numbers and mode of connection to the existing network, i.e. direct or indirect through other group elements; and

(b) the group impedance matrix $\boxed{z_{qp}}$ is a diagonal matrix having terms equal to the element self- impedance values.

It is also necessary to process the group elements in a specified order with branch elements first and loop elements last.

A.3.4. GROUP OF MUTUALLY COUPLED ELEMENTS.

For an interconnected network formed by adding branch and loop elements to an existing network having bus impedance matrix $\boxed{Z_A}$, it has been shown that the matrix $\boxed{Z'}$ may be derived from $\boxed{Z_A}$ by simple rules without formally setting up the connection sub-matrixes $\boxed{C_a}$ and $\boxed{C_b}$. However, for adding a group of mutually coupled elements the connection sub-matrix $\boxed{C_b}$ is required in the calculation of the product $\boxed{C_{bt} z_m C_b}$ basically because $\boxed{z_m}$ consists of off-diagonal terms whereas $\boxed{z_s}$ has diagonal terms only and hence the simple rules for adding the self-impedance values cannot be extended to adding mutual impedances.

From consideration of a number of cases in which, for convience, nodes are identified by integers instead of alphabetical letters as previously, rules depending on the relationships between integer magnitudes, types and interconnections of elements and having a general application for setting up $\boxed{C_b}$ are derived. As the purpose is to derive rules for a computer programme applicable to any network having nodes numbered in any regular or irregular pattern and elements listed in any order, these cases of which some may seem artificial can arise during processing of power system networks.

In the following diagrams the relevant part of the established network and the group of mutually coupled elements being added are shown for various combinations of branch and loop elements. For each case the associated equations- corresponding to eqn.(5) - relating element currents before to network currents after interconnection, are written in full showing $\boxed{C_b}$ for the correct processing order of the elements. The reference node is denoted by "0" in the diagrams.

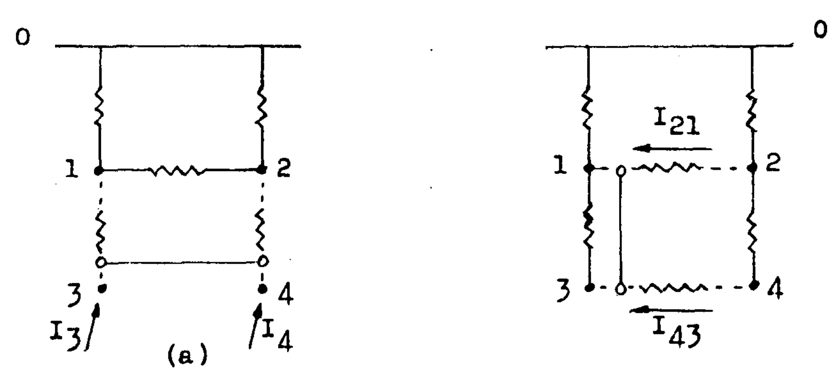


Fig.9

(a) Two mutually coupled branch elements 3-1 and 4-2 connected to established network nodes 1 and 2 respectively.

$$\begin{bmatrix} I_{31} \\ I_{42} \end{bmatrix} = \begin{array}{c} \\ 3\text{-}1 \\ 4\text{-}2 \end{array} \begin{array}{c} \begin{array}{cc} 3 & 4 \end{array} \\ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{array} \begin{bmatrix} I_3 \\ I_4 \end{bmatrix} \quad \ldots\ldots\ldots \quad (99)$$

(b) Two mutually coupled loop elements 2-1 and 4-3 connected to the network at nodes 1,2 and 3,4 respectively.

$$\begin{bmatrix} I_{21} \\ I_{43} \end{bmatrix} = \begin{array}{c} \\ 2\text{-}1 \\ 4\text{-}3 \end{array} \begin{array}{c} \begin{array}{cc} 2\text{-}1 & 4\text{-}3 \end{array} \\ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{array} \begin{bmatrix} I_{21} \\ I_{43} \end{bmatrix} \quad \ldots\ldots\ldots \quad (100)$$

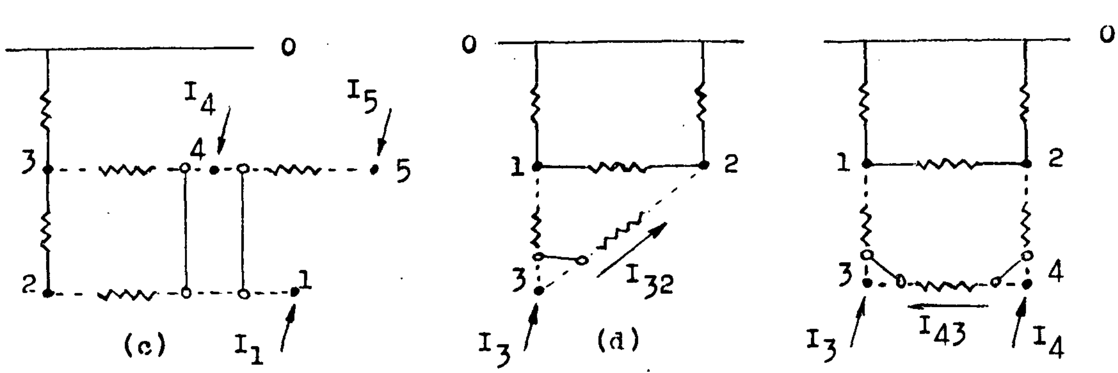


Fig.9

(e)

Fig.9(c). Two branch elements 2-1 and 4-3 connected to the network at nodes 2 and 3 respectively with a third branch element 5-4 connected indirectly to node 3 through element 4-3.

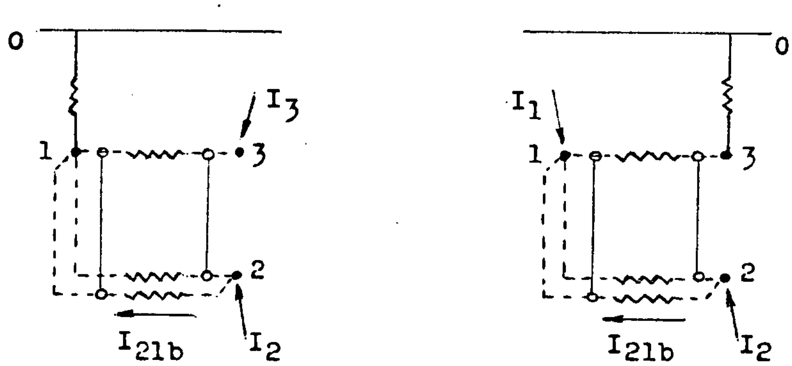
$$\begin{bmatrix} I_{21}\\ I_{43}\\ I_{54}\end{bmatrix} = \begin{array}{cc} & \begin{array}{ccc}1 & 4 & 5\end{array}\\ \begin{array}{c}2\text{-}1\\ 4\text{-}3\\ 5\text{-}4\end{array} & \begin{bmatrix}-1 & 0 & 0\\ 0 & 1 & 1\\ 0 & 0 & 1\end{bmatrix}\end{array} \begin{bmatrix} I_1\\ I_4\\ I_5\end{bmatrix} \quad \ldots\ldots\ldots (101)$$

Fig.9(d). A branch element 3-1 connected to the network at node 1 and a loop element 3-2 connected directly to network node 2 and linked through element 3-1 to node 1.

$$\begin{bmatrix} I_{31}\\ I_{32}\end{bmatrix} = \begin{array}{cc} & \begin{array}{cc}1 & 3\text{-}2\end{array}\\ \begin{array}{c}3\text{-}1\\ 3\text{-}2\end{array} & \begin{bmatrix}1 & -1\\ 0 & 1\end{bmatrix}\end{array} \begin{bmatrix} I_3\\ I_{32}\end{bmatrix} \quad \ldots\ldots\ldots (102)$$

Fig.9(e). Two branch elements 3-1 and 4-2 connected to network nodes 1 and 2 respectively and loop element 4-3 linked to nodes 1 and 2 through group elements 3-1 and 4-2.

$$\begin{bmatrix} I_{31}\\ I_{42}\\ I_{43}\end{bmatrix} = \begin{array}{cc} & \begin{array}{ccc}3 & 4 & 4\text{-}3\end{array}\\ \begin{array}{c}3\text{-}1\\ 4\text{-}2\\ 4\text{-}3\end{array} & \begin{bmatrix}1 & 0 & 1\\ 0 & 1 & -1\\ 0 & 0 & 1\end{bmatrix}\end{array} \begin{bmatrix} I_3\\ I_4\\ I_{43}\end{bmatrix} \quad \ldots\ldots\ldots (103)$$



(f) (g)

Fig.(9)

Fig.9(f). Two branch elements 2-1 and 3-1 both connected to network node 1 with loop element 2-1 connected to network node 1 and linked through parallel group element 2-1 to network node 2.

$$\begin{bmatrix} I_{21a}\\ I_{31}\\ I_{21b}\end{bmatrix} = \begin{array}{cc} & \begin{array}{ccc}2 & 3 & 2\text{-}1\end{array}\\ \begin{array}{c}2\text{-}1\\ 3\text{-}1\\ 2\text{-}1\end{array} & \begin{bmatrix}1 & 0 & -1\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}\end{array} \begin{bmatrix} I_2\\ I_3\\ I_{21b}\end{bmatrix} \quad \ldots\ldots\ldots (104)$$

Fig.9(g). Branch elements 3-1 and 2-1 respectively connected directly and indirectly through element 3-1 to network node 3 and loop element 2-1 having node 1 linked via element 3-1 to network node 3 and node 2 linked via parallel element 2-1 and element 3-1 to node 3. This is the same group of elements as in 9(f) but with network node 3 instead of node 1 established so that element 3-1 must be processed first to establish node 1 whereas in 9(f) either element 3-1 or one of the parallel elements 2-1 can be processed first.

$$\boxed{\begin{array}{c} I_{31} \\ I_{21a} \\ I_{21b} \end{array}} = \begin{array}{c|ccc} & 1 & 2 & 2\text{-}1 \\ \hline 3\text{-}1 & -1 & -1 & 0 \\ 2\text{-}1 & 0 & 1 & -1 \\ 2\text{-}1 & 0 & 0 & 1 \end{array} \; \boxed{\begin{array}{c} I_1 \\ I_2 \\ I_{21b} \end{array}} \quad . \; . \; . \; . \; . \; . \; . \quad (105)$$

From eqns.(99) to (105) it can be seen that each row in C_b corresponds to a group element while columns correspond to new nodes in the case of branch elements and to the element itself for loop elements, i.e. as each group element is processed a new row and column is added to $\boxed{C_b}$. The diagonal term in the new row and column is +1 for a loop element and, depending on whether the new node is greater or lesser in magnitude than the established node is +1 or -1 respectively for a branch element (the follows from the assumption that the element current flows from the higher to the lower numbered node). Because the current in the element being processed may affect currents in previously processed group elements but is itself not affected by those currents, the off-diagonal terms in the corresponding column of $\boxed{C_b}$ are +1, -1 or zero and those in the corresponding row are zero, i.e. all terms below the main diagonal are zero.

The column terms of $\boxed{C_b}$ are found by tracing element connections back through the group to the network taking into account relations between node numbers. If the element is connected directly to the network then the corresponding column terms are zero, e.g. in Fig.9(a) as branch element 4-2 is connected to network node 2 the off-diagonal term in column 4 of eqn.(99) is zero; in Fig.9(b) loop element 4-3 is connected is connected to network nodes 4 and 3 there fore the off-diagonal term in column 4-3 of eqn.(100) is zero; similarly for element 4-3 of Fig .9(c) the term in column 4 of eqn.(101) is zero, for element 4-2 of Fig.9(e) the term in column 4 of eqn.(103) is zero and for element 3-1 of Fig.9(f) the term in column 3 of eqn.(104) is zero. For a branch element connected indirectly to the original network, the column term corresponding to an element linking the branch to the network is the same as the diagonal term for the link element, e.g. in Fig.9(c) branch element 5-4 is linked to the network via element 4-3, hence the term in column 5 row 4-3 of $\boxed{C_b}$ in eqn.(101) is +1, which is the same as the diagonal term in row 4-3; similarly for branch element 2-1 and linking element 3-1 in Fig.9(g) the term in column 2 row 3-1 of eqn.(105) is the same as the row 3-1 diagonal term. Thus the sign of the column term is dependant on the node numbering of the corresponding link element, but is independant of the node numbering of the branch element concerned - this is illustrated for example in Fig.9(c) if nodes 4 and 5 are interchanged the current in the link element is unaltered at I_4+I_5.

For a loop element it is necessary to examine the mode of connection from both nodes to the original network. If it is indirectly connected from the higher and, or the lower numbered nodes then the column terms corresponding to linking elements are found in the same way as for a branch element, except that the sign of the diagonal term is reversed for elements linking the higher numbered node of the loop element;-in Fig.9(d) node 3 of loop element 3-2 is linked to network node 1 via element 3-1 hence the term in column 3-2 row 3-1 of eqn.(102) is -1 which is the diagonal term in row 3-1 with sign reversed; similarly for $\boxed{C_b}$ in eqn.(103) the terms of column 4-3 are -1 in row 4-2 and +1 in row 3-1 being the reverse and the same sign respectively as the corresponding diagonal terms. In Fig.9(g) element 3-1 links both nodes of loop element 2-1 to network node 3, hence in eqn.(105) the term in column 2-1 row 3-1 is -1+1=0.

Summarising, the connection sub- matrix $\boxed{C_b}$ has a row and column corresponding to each group element with diagonal term +1 for loop elements and +1 or -1 for branch elements depending respectively on whether the new node being established is the greater or the lesser of the branch nodes. The off-diagonal terms are zero except for elements connected indirectly, i.e. linked through other group elements, to the network in which case the term in the element column and linking element row is :-

(a) for a branch element plus the link element diagonal term; and

(b) for a loop element - minus or plus the link element diagonal term or zero depending respectively on whether the link element connects the higher or the lower numbered node or both nodes of the loop element to the network.

As shown in eqns.(99) - (105) all terms below the main diagonal of $\boxed{C_b}$ are zero when this matrix is set up in the order in which the group elements are processed.

After $\boxed{C_b}$ has been found the triple matrix product, $\boxed{C_{bt} z_m C_b}$, is calculated to give the mutual coupling values which are added into $\boxed{Z'}$ forming $\boxed{Z''}$ as in eqn.(32). Because matrix $\boxed{C_b}$ is a real matrix, the triple matrix product can be separated into real and imaginary parts which may be calculated separately :-

$$\boxed{C_{bt} z_m C_b} = \boxed{C_{bt} r_m C_b} + \boxed{C_{bt} x_m C_b}\, j \quad . \quad . \quad . \quad . \quad . \quad (106)$$

Hence when adding a group of mutually coupled elements to a network, the network bus impedance matrix $\boxed{Z_A}$ is modified in a step by step procedure as follows :-

(a) the group elements are sorted into a list with branch elements, in the order in which they are added to the network, first and loop elements last;

(b) $\boxed{Z'}$ is formed using the element self-impedances in processing the elements one at a time in the listed order, adding a corresponding row and column to $\boxed{Z_A}$ for each branch element and an augmenting row and column for each loop element according to the branch and loop element rules respectively;

(c) the connection sub-matrix $\boxed{C_b}$ and the matrix of mutual impedances $\boxed{z_m}$ are set up using the group element list from (a); the product $\boxed{C_{bt} z_m C_b}$ calculated and the result added to $\boxed{Z'}$ to give $\boxed{Z''}$; and

(d) the augmenting rows and columns are eliminated from $\boxed{Z''}$ one at a time by Kron's rule giving the bus impedance matrix for the interconnected network.

In steps (b) and (c) only addition and subtraction of matrix and impedance terms are involved so the real and imaginary parts of the matrixes $\boxed{Z'}$ and $\boxed{Z''}$ can be calculated separately, but as step (d) involves multiplication and division both parts of the matrix $\boxed{Z''}$ are used in these calculations.

A.4. FAULT CALCULATIONS USING THE BUS IMPEDANCE MATRIX.

In power system analysis, an important application of the bus impedance matrix is in fault calculations for which the positive, negative and zero sequence networks are used with mutual coupling between network elements taken into account in the zero sequence network only. Because a power system has components such as transmission lines operating at different voltages connecting generators and transformers having different power ratings, all element sequence impedances are converted to per unit values on a suitable power base such as 100 MVA before calculating the bus impedance matrix. Let the relation between node voltages and currents and bus impedance matrix of a power system sequence network be given by :-

$$\begin{bmatrix} \cdot \\ V_i \\ \cdot \\ V_j \\ \cdot \\ V_k \\ \cdot \end{bmatrix} = \begin{array}{c} \\ i \\ \\ j \\ \\ k \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccccc} & i & & j & & k & \end{array} \\ \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{ii} & \cdot & Z_{ij} & \cdot & Z_{ik} & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{ji} & \cdot & Z_{jj} & \cdot & Z_{jk} & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & Z_{ki} & \cdot & Z_{kj} & \cdot & Z_{kk} & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \end{array} \begin{bmatrix} \cdot \\ I_i \\ \cdot \\ I_j \\ \cdot \\ I_k \\ \cdot \end{bmatrix} \quad . \quad . \quad (107)$$

where i, j and k are network nodes.

If all currents except that at node "i" are zero, then :-

$$V_i = Z_{ii}I_i \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (108)$$

and $$V_j = Z_{ji}I_i \qquad V_k = Z_{ki}I_i \quad . \quad . \quad . \quad . \quad . \quad (109)$$

From eqn.(108) it follows that Z_{ii} is the sequence network driving point impedance in per unit for node "i" and the per unit power is given by :-

$$V_iI_i^* = \frac{V_iV_i^*}{Z_{ii}^*} = \frac{1}{Z_{ii}^*} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (110)$$

if V_i is 1 per unit voltage - in the case of the positive sequence network this is the fault power for a three phase fault on node "i", i.e. fault power

$$S = \frac{\text{base MVA}}{Z_{ii}^*} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (111)$$

From eqn.(109), Z_{ji} and Z_{ki} are the sequence transfer impedances in per unit between nodes "i" - "j" and "i" - "k" respectively. If I_i=1, i.e. unit current flowing into the network at node "i", then the voltages appearing on nodes "j" and "k" are :-

$$V_j = Z_{ji} \qquad V_k = Z_{ki} \quad . \quad . \quad . \quad . \quad . \quad . \quad (112)$$

The current flowing in an element k-j, which is not mutually coupled to any other network element, is :-

$$I_{kj} = \frac{V_k - V_j}{z_{kj}} = \frac{Z_{ki} - Z_{ji}}{z_{kj}} \quad . \quad . \quad . \quad . \quad . \quad . \quad (113)$$

where z_{kj} is the element self impedance. I_{kj} is the "current distribution factor" for the element k-j.

If e-d and g-f form a group of mutually coupled elements, then the relation between element currents and node voltages is :-

$$\begin{bmatrix} V_e - V_d \\ V_g - V_f \end{bmatrix} = \begin{bmatrix} z_{ed} & z_{ml} \\ z_{ml} & z_{gf} \end{bmatrix} \begin{bmatrix} I_{ed} \\ I_{gf} \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (114)$$

where z_{ed}, z_{gf} are the element self impedances and z_{ml} the mutual coupling impedance. Inverting the impedance matrix in eqn.(114) :-

$$\begin{bmatrix} I_{ed} \\ I_{gf} \end{bmatrix} = \begin{bmatrix} y_{ed} & y_{ml} \\ y_{ml} & y_{gf} \end{bmatrix} \begin{bmatrix} V_e - V_d \\ V_g - V_f \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (115)$$

The current distribution factor for element e-d when unit current is flowing into node "i" is, from eqns.(115) and (112) :-

$$\begin{aligned} I_{ed} &= y_{ed}(V_e - V_d) + y_{ml}(V_g - V_f) \\ &= y_{ed}(Z_{ei} - Z_{di}) + y_{ml}(Z_{gi} - Z_{fi}) \quad . \quad . \quad . \quad . \quad . \quad (116) \end{aligned}$$

$$\text{Similarly} \quad I_{gf} = y_{ml}(Z_{ei} - Z_{di}) + y_{gf}(Z_{gi} - Z_{fi}) \quad . \quad . \quad . \quad . \quad . \quad (117)$$

A.5. MODIFICATION OF BUS IMPEDANCE MATRIX.

Consider a network having bus impedance matrix $\boxed{Z_A}$ and in which there is a group of mutually coupled elements with self and mutual impedances given by the matrix $\boxed{z_{old}}$. If one or more of the impedances in the group are modified so that the matrix of self and mutual impedances becomes $\boxed{z_{new}}$, then the bus impedance matrix incorporating this modification can be found by adding a group of fictitious elements with self and mutual impedances given by $\boxed{z_{fic}}$ where $\boxed{z_{fic}}$ is such that the result of paralleling impedances $\boxed{z_{old}}$ and $\boxed{z_{fic}}$ is $\boxed{z_{new}}$, i.e.

$$\boxed{y_{new}} = \boxed{y_{old}} + \boxed{y_{fic}} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (118)$$

$$\text{or} \quad \boxed{y_{fic}} = \boxed{y_{new}} - \boxed{y_{old}} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (119)$$

where $\boxed{y_{new}}$ and $\boxed{y_{old}}$ are the inverse of $\boxed{z_{new}}$ and $\boxed{z_{old}}$ respectively. From eqn.(119) $\boxed{z_{fic}}$ is calculated as the inverse of $\boxed{y_{fic}}$

i.e. $$\boxed{z_{fic}} = \boxed{y_{fic}}^{-1} \quad \ldots\ldots\ldots \quad (120)$$

The diagonal terms of $\boxed{z_{fic}}$ are the self impedances of, and the off-diagonal terms the mutual couplings between the fictitious elements to be added to $\boxed{Z_A}$ forming the bus impedance matrix of the modified network. All elements in the fictitious group are loop elements as they connect established nodes, hence from eqn.(100) the connection sub-matrix $\boxed{C_b}$ is a unit matrix of order equal to the number of elements in the group.

The equations derived above can be applied to the particular case of modifying an element which is not mutually coupled to any other element; in this case the matrixes in eqns.(118)-(120) degenerate into the self admittance and self impedance of the single element. When changing the impedance of a network element from z_{old} to z_{new}, it follows from eqns.(119) and (120) that the fictitious self impedance for adding in parallel to z_{old} is :-

$$z_{fic} = \frac{z_{old} z_{new}}{z_{old} - z_{new}} \quad \ldots\ldots\ldots \quad (121)$$

When an uncoupled element is switched out of the network, i.e. $y_{new}=0$, it follows from eqns.(119) and (120) that :-

$$z_{fic} = -z_{old} \quad \ldots\ldots\ldots \quad (122)$$

A.6. <u>DERIVATION OF EQUIVALENT NETWORK.</u>

Consider a network having bus impedance matrix given by eqn.(107), then the partial matrix formed by selecting the terms indicated in this equation is the bus impedance matrix of a network with nodes "i", "j" and "k" which is equivalent to the original network, i.e.

$$\boxed{Z_{eqiv}} = \begin{array}{c|ccc} & i & j & k \\ \hline i & Z_{ii} & Z_{ij} & Z_{ik} \\ j & Z_{ji} & Z_{jj} & Z_{jk} \\ k & Z_{ki} & Z_{kj} & Z_{kk} \end{array} \quad \ldots\ldots \quad (123)$$

Inspection of eqn.(123) shows that the terms of the matrix are the transfer and driving point impedances of the original network for nodes "i", "j" and "k", hence all properties of the original network relevant to these nodes can be derived from $\boxed{Z_{eqiv}}$.

As the partial bus impedance matrix $\boxed{Z_{eqiv}}$ represents the effect of the original network at the selected nodes, it can be used in the calculation of the bus impedance matrix for a second network which is connected to one or more of the selected nodes. The calculation may start with $\boxed{Z_{eqiv}}$ then adding elements of the second network - uncoupled elements one at a time and coupled elements by groups - until all elements are processed or, alternatively the impedance $\boxed{Z_{eqiv}}$ can be added to the bus impedance of the second network by considering it to represent elements i-0, j-0 and k-0 having self impedances Z_{ii}, Z_{jj} and Z_{kk} respectively and mutual impedances Z_{ij}, Z_{ik} and Z_{jk}. The final bus impedance matrix is that of the second network including the effect of the original network.

If required the impedances of the equivalent network elements can be found by inverting the corresponding admittance values obtained from $\boxed{Y_{eqiv}}$, the inverse of $\boxed{Z_{eqiv}}$:-

$$\boxed{Y_{eqiv}} = \boxed{Z_{eqiv}}^{-1}$$

$$= \begin{array}{c|ccc|} & i & j & k \\ i & Y_{ii} & Y_{ij} & Y_{ik} \\ j & Y_{ji} & Y_{jj} & Y_{jk} \\ k & Y_{ki} & Y_{kj} & Y_{kk} \end{array} \quad . \; . \; . \; . \; . \; . \; . \quad (124)$$

From $\boxed{Y_{eqiv}}$ the admittances of the elements are easily derived; the off-diagonal terms are the negative of the admittances between nodes and the sum of the terms in each row (or column) is the admittance between node and reference, i.e. $-Y_{ij}$ is the admittance of element i-j and $Y_{ii}+Y_{ij}+Y_{ik}$ is the admittance of element i-0, etc.

An example of an equivalent circuit and network modification are included in the following numerical example.

A.7. EXAMPLE OF BUS IMPEDANCE MATRIX CALCULATION.

The rules which have been derived are now applied to the calculation of the bus impedance matrix for the sample network shown in Fig.10 in which all the self and mutual impedance values are in per unit and, for ease of calculation, element resistances are neglected - a common practice in power system fault studies before use of digital computers became general. By using tables, all network information is set out in a simple systematic form which, for the digital computer programme, are data for calculating the bus impedance matrix/ element current distribution factors, etc.

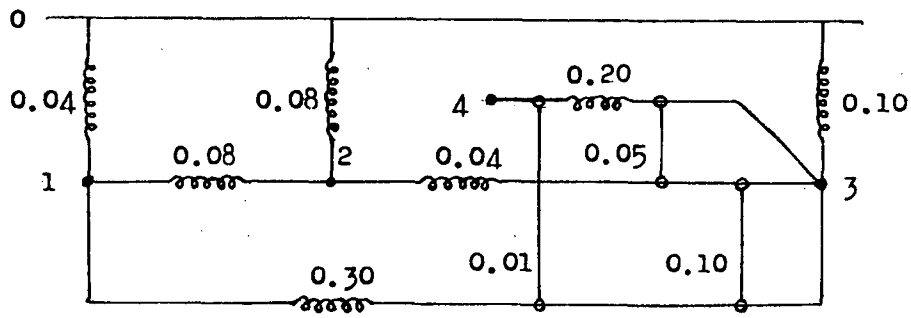


Fig.10

The first step is coding the network, a procedure which simply involves assigning zero to the reference node and a positive integer to all other nodes, thus identifying network nodes by integers and network elements by two node numbers. Three tables are used to list all the network information; the first giving basic data, i.e. numbers of nodes (excluding the reference node), network elements and mutual couplings; the second element node connections and self impedances and the third mutually coupled elements together with their mutual impedances. For the network in Fig.10 the three tables are :-

TABLE 1. BASIC DATA.

Number of nodes	4
Number of elements	7
Number of mutual couplings	3

TABLE 2. ELEMENT DATA.

Element Number	Node Connections		Self Impedance
1	0	1	0.04
2	0	2	0.08
3	1	2	0.08
4	0	3	0.10
5	1	3	0.30
6	3	4	0.20
7	2	3	0.40

TABLE 3. MUTUAL COUPLING DATA.

Coupling Number	Elements Linked		Mutual Impedance
1	5	6	-0.01
2	5	7	0.10
3	6	7	-0.05

In Table 2, each element is assigned a number which is used in Table 3 for identifying mutually coupled elements - a simpler procedure than using the two node numbers which also avoids the difficulty of distinguishing between parallel elements as these have the same node numbers. The mutual impedance value is listed with a positive or a negative sign determined by the directions of current flow in the coupled elements relative to one another which follow from network geometry and node numbering, e.g. from Fig.10 the directions from higher to lower numbered nodes in elements 3-1 and 3-2 is parallel, hence the corresponding mutual impedance value is entered in Table 3 with a positive sign, but for the mutually coupled elements 4-3, 3-2 and 4-3, 3-1 the directions from higher to lower numbered nodes is anti-parallel, hence the corresponding impedances appear in Table 3 with negative signs.

The calculation commences with the formation of group mutual impedance matrixes from Table 3 and in this example there is only one group for which $\boxed{z_m}$ is set up in the order in which the coupled elements are listed in Table 3 :-

$$\boxed{z_m} = \begin{array}{c|c|c|c|} & 3\text{-}1 & 4\text{-}3 & 3\text{-}2 \\ \hline 3\text{-}1 & 0 & -0.01 & 0.10 \\ \hline 4\text{-}3 & -0.01 & 0 & -0.05 \\ \hline 3\text{-}2 & 0.10 & -0.05 & 0 \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \quad (125)$$

The calculation of the network bus impedance matrix begins with a branch element connected to the reference node (a "generator" element) because initially no network nodes are established. In Table 2 the first element listed is in this category so the calculation starts with element 1-0 establishing node 1. Applying the branch element rule for an element connected to the reference node, eqn.(46) :-

$$\boxed{Z_A} = \boxed{Z'} = \begin{array}{c|c|} & 1 \\ \hline 1 & 0.04 \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (126)$$

As node 1 is now established, the next element processed could be a branch element connected to either the reference or node 1, or a loop element parallel to node 1. Inspection of Table 2 shows that the next listed element, 2-0, is a branch connected to the reference node and application of eqn.(46) gives :-

$$\boxed{Z_A} = \boxed{Z'} = \begin{array}{c|c|c|} & 1 & 2 \\ \hline 1 & 0.04 & 0 \\ \hline 2 & 0 & 0.08 \\ \hline \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \quad (127)$$

The third element in Table 2, 2-1, forms a loop connecting established nodes 1 and 2. Applying the rules for adding a loop element, eqns. (67) and (70) :-

$$\boxed{Z'} = \begin{array}{c|c|c|c|} & 1 & 2 & 2\text{-}1 \\ \hline 1 & 0.04 & 0 & 0.04 \\ \hline 2 & 0 & 0.08 & -0.08 \\ \hline 2\text{-}1 & 0.04 & -0.08 & 0.04+0.08+0.08 \\ \hline \end{array}$$

$$= \begin{array}{c|c|c|c|} & 1 & 2 & 2\text{-}1 \\ \hline 1 & 0.04 & 0 & 0.04 \\ \hline 2 & 0 & 0.08 & -0.08 \\ \hline 2\text{-}1 & 0.04 & -0.08 & 0.20 \\ \hline \end{array} \quad . \ . \ . \ . \ . \ . \quad (128)$$

Eliminating the augmenting row and column, 2-1, by Kron's rule, eqn.(71)

$$\boxed{Z_A} = \begin{array}{c|c|c|} & 1 & 2 \\ \hline 1 & 0.032 & 0.016 \\ \hline 2 & 0.016 & 0.048 \\ \hline \end{array} \quad . \ . \ . \ . \ . \ . \quad (129)$$

This is the bus impedance matrix for the network formed by the three elements 1-0, 2-0 and 2-1. Inspection of Table 2 shows that the next listed element 3-0 is a branch element establishing node 3; by application of eqn.(46) :-

$$\boxed{Z_A} = \boxed{Z'} = \begin{array}{c|c|c|c|} & 1 & 2 & 3 \\ \hline 1 & 0.032 & 0.016 & 0 \\ \hline 2 & 0.016 & 0.048 & 0 \\ \hline 3 & 0 & 0 & 0.10 \\ \hline \end{array} \quad . \ . \ . \ . \ . \quad (130)$$

The remaining three elements in Table 2 form a mutually coupled group in which 4-3 is a branch element from node 3, and 3-1 and 3-2 are loop elements. To the impedance matrix in eqn.(130) element 4-3 is added by the branch element rule, eqn.(41), and then elements 3-1, 3-2 by the loop element rule, eqns.(67) and (70) :-

$$\boxed{Z'} = \begin{array}{c|c|c|c|c|c|c|} & 1 & 2 & 3 & 4 & 3\text{-}1 & 3\text{-}2 \\ \hline 1 & 0.032 & 0.016 & 0 & 0 & 0.032 & 0.016 \\ \hline 2 & 0.016 & 0.048 & 0 & 0 & 0.016 & 0.048 \\ \hline 3 & 0 & 0 & 0.10 & 0.10 & -0.10 & -0.10 \\ \hline 4 & 0 & 0 & 0.10 & 0.30 & -0.10 & -0.10 \\ \hline 3\text{-}1 & 0.032 & 0.016 & -0.10 & -0.10 & 0.432 & 0.116 \\ \hline 3\text{-}2 & 0.016 & 0.048 & -0.10 & -0.10 & 0.116 & 0.548 \\ \hline \end{array} \quad (131)$$

The connection sub-matrix $\boxed{C_b}$ for the group is - from eqns.(99) and (100) :-

$$\boxed{C_b} = \begin{array}{c|ccc} & 4 & 3\text{-}1 & 3\text{-}2 \\ \hline 4\text{-}3 & 1 & 0 & 0 \\ 3\text{-}1 & 0 & 1 & 0 \\ 3\text{-}2 & 0 & 0 & 1 \end{array} \quad . . . . (132)$$

The group mutual impedance matrix is obtained by rearranging the terms in eqn.(125) to correspond with the order in which the elements are processed :-

$$\boxed{z_m} = \begin{array}{c|ccc} & 4\text{-}3 & 3\text{-}1 & 3\text{-}2 \\ \hline 4\text{-}3 & 0 & -0.01 & -0.05 \\ 3\text{-}1 & -0.01 & 0 & 0.10 \\ 3\text{-}2 & -0.05 & 0.10 & 0 \end{array} \quad . . . . (133)$$

The triple matrix product $\boxed{C_{bt}z_mC_b}$ is now calculated, a trivial calculation in this case as $\boxed{C_b}$ is a unit matrix :-

$$\boxed{C_{bt}z_mC_b} = \begin{array}{c|ccc} & 4 & 3\text{-}1 & 3\text{-}2 \\ \hline 4 & 0 & -0.01 & -0.05 \\ 3\text{-}1 & -0.01 & 0 & 0.10 \\ 3\text{-}2 & -0.05 & 0.10 & 0 \end{array} \quad . . . . (134)$$

and $\boxed{Z''}$ for the interconnected network is derived from eqns.(32), (131) and (134) :-

$$\boxed{Z''} = \boxed{Z'} + \boxed{C_{bt}z_mC_b}$$

$$= \begin{array}{c|cccccc} & 1 & 2 & 3 & 4 & 3\text{-}1 & 3\text{-}2 \\ \hline 1 & 0.032 & 0.016 & 0 & 0 & 0.032 & 0.016 \\ 2 & 0.016 & 0.048 & 0 & 0 & 0.016 & 0.048 \\ 3 & 0 & 0 & 0.10 & 0.10 & -0.10 & -0.10 \\ 4 & 0 & 0 & 0.10 & 0.30 & -0.11 & -0.15 \\ 3\text{-}1 & 0.032 & 0.016 & -0.10 & -0.11 & 0.432 & 0.216 \\ 3\text{-}2 & 0.016 & 0.048 & -0.10 & -0.15 & 0.216 & 0.548 \end{array} \quad . . (135)$$

The bus impedance matrix for the network in Fig.10 is now found by eliminating the last two rows of $\boxed{Z''}$ one at a time by Kron's rule; eliminating the row and column corresponding to loop element 3-2 :-

	1	2	3	4	3-1
1	0.0315	0.0146	0.0029	0.0044	0.0257
2	0.0146	0.0438	0.0087	0.0131	-0.0029
3	0.0029	0.0087	0.0818	0.0727	-0.0606
4	0.0044	0.0131	0.0727	0.2590	-0.0510
3-1	0.0257	-0.0029	-0.0606	-0.0510	0.3470

and then the row and column corresponding to loop element 3-1 :-

$\boxed{Z_A}$ = ... (136)

	1	2	3	4
1	0.0296	0.0148	0.0074	0.0082
2	0.0148	0.0438	0.0082	0.0127
3	0.0074	0.0082	0.0712	0.0638
4	0.0082	0.0127	0.0638	0.2515

This is the imaginary part of the complex bus impedance matrix, the real part being zero as element resistance has been neglected.

If $I_3 = 1$ and $I_1 = I_2 = I_4 = 0$ . . . . . (137)

then, from eqns.(112) and (136) the voltages at the network nodes are :- $V_1 = 0.0074$ $V_2 = 0.0082$

$V_3 = 0.0712$ $V_4 = 0.0638$ . . . . (138)

From eqn.(113) the current distribution factors for the uncoupled elements connected to node 1 are :-

$$I_{10} = \frac{V_1}{z_{10}} = \frac{0.0074}{0.04} = 0.185 \qquad \ldots\ldots \quad (139)$$

$$I_{21} = \frac{V_2-V_1}{z_{21}} = \frac{0.0082-0.0074}{0.08} = 0.010 \qquad \ldots \quad (140)$$

As element 3-1 is mutually coupled to elements 4-3, 3-2 the inverse of the group self and mutual impedance matrix, $\boxed{z_{qp}}$, is required for calculating its current distribution factor.

$\boxed{z_{qp}}$ = ... (141)

	4-3	3-1	3-2
4-3	0.2	-0.01	-0.05
3-1	-0.01	0.3	0.10
3-2	-0.05	0.10	0.4

Inverting :-

$\boxed{y_{qp}}$ = . . . (142)

	4-3	3-1	3-2
4-3	5.1619	-0.0469	0.6570
3-1	-0.0469	3.6368	-0.9151
3-2	0.6570	-0.9151	2.8109

Therefore, as in eqn.(116) :-

$$I_{31} = -0.0469(V_4-V_3)+3.6368(V_3-V_1)-0.9151(V_3-V_2)$$

Substituting the voltage values from eqn.(138) :-

$$I_{31} = -0.0469(-0.0074)+3.6368(0.0638)-0.9151(0.0630)$$

$$= 0.1748 \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (143)$$

Checking, from eqns.(140) and (143) $I_{21}+I_{31} = 0.1848$ which, within the degree of accuracy of the calculations, is equal to I_{10} from eqn.(139).

A.7.1. EXAMPLES OF BUS IMPEDANCE MATRIX MODIFICATION.

The bus impedance matrix for two examples of modifications to the network in Fig.10 is now calculated.

(a) The self impedance of uncoupled element 2-1 is changed from 0.08 to 0.04 per unit.

From eqn.(121) :-

$$z_{f1c} = 0.08 \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (144)$$

i.e. add an element having impedance 0.08 in parallel with element 2-1. As this element forms a loop, an augmenting row and column is added to the bus impedance matrix of eqn.(136); applying eqns.(67) and (70) :-

$\boxed{Z'}$ = . . (145)

	1	2	3	4	2-1
1	0.0296	0.0148	0.0074	0.0082	0.0148
2	0.0148	0.0438	0.0082	0.0127	-0.0290
3	0.0074	0.0082	0.0712	0.0638	-0.0008
4	0.0082	0.0127	0.0638	0.2515	-0.0045
2-1	0.0148	-0.0290	-0.0008	-0.0045	0.1238

The augmenting row and column 2-1 are eliminated leaving the bus impedance matrix of the modified network :-

$\boxed{Z}$ = ... (146)

	1	2	3	4
1	0.0278	0.0183	0.0075	0.0087
2	0.0183	0.0370	0.0080	0.0116
3	0.0075	0.0080	0.0712	0.0638
4	0.0087	0.0116	0.0638	0.2513

(b) The self and mutual impedances of the mutually coupled group of elements 4-3, 3-1 and 3-2 are modified as follows :-

self impedance of element 4-3 changed from 0.2 to 0.4 per unit,

self impedance of element 3-1 changed from 0.3 to 0.5 per unit,
self impedance of element 3-2 changed from 0.4 to 0.2 per unit,
and mutual impedance between elements 4-3, 3-1 reduced to zero
and similarly between elements 4-3, 3-2.

The new impedance matrix for the group of elements is :-

$$\boxed{z_{new}} = \begin{array}{c|ccc} & 4\text{-}3 & 3\text{-}1 & 3\text{-}2 \\ \hline 4\text{-}3 & 0.4 & 0 & 0 \\ 3\text{-}1 & 0 & 0.5 & 0.1 \\ 3\text{-}2 & 0 & 0.1 & 0.2 \end{array} \quad \ldots\ldots\ldots (147)$$

Inverting :-

$$\boxed{y_{new}} = \begin{array}{c|ccc} & 4\text{-}3 & 3\text{-}1 & 3\text{-}2 \\ \hline 4\text{-}3 & 2.5 & 0 & 0 \\ 3\text{-}1 & 0 & 2.2222 & -1.1111 \\ 3\text{-}2 & 0 & -1.1111 & 5.5556 \end{array} \quad \ldots\ldots (148)$$

y_{old} for the group of mutually coupled elements is given by eqn.(142) and hence from eqn.(119) :-

$$\boxed{y_{fic}} = \boxed{y_{new}} - \boxed{y_{old}}$$

$$= \begin{array}{c|ccc} & 4\text{-}3 & 3\text{-}1 & 3\text{-}2 \\ \hline 4\text{-}3 & -2.6619 & 0.0469 & -0.6570 \\ 3\text{-}1 & 0.0469 & -1.4146 & -0.1960 \\ 3\text{-}2 & -0.6570 & -0.1960 & 2.7447 \end{array} \quad \ldots\ldots (149)$$

Inverting :-

$$\boxed{z_{fic}} = \begin{array}{c|ccc} & 4\text{-}3 & 3\text{-}1 & 3\text{-}2 \\ \hline 4\text{-}3 & -0.3547 & 0 & -0.0849 \\ 3\text{-}1 & 0 & -0.7000 & -0.0500 \\ 3\text{-}2 & -0.0849 & -0.0500 & 0.3404 \end{array} \quad \ldots\ldots (150)$$

The diagonal terms of $\boxed{z_{fic}}$ are self impedances and the off-diagonal terms mutual impedances of loop elements 4-3, 3-1 and 3-2 which are now added to the bus impedance matrix, eqn.(136). As all group elements form loops the corresponding connection sub-matrix $\boxed{C_b}$ is, from eqn.(100), a unit matrix and hence applying the loop element rule and eqn.(32) :-

$\boxed{Z''}$ =

	1	2	3	4	4-3	3-1	3-2
1	0.0296	0.0148	0.0074	0.0082	-0.0008	0.0222	0.0074
2	0.0148	0.0438	0.0082	0.0127	-0.0045	0.0066	0.0356
3	0.0074	0.0082	0.0712	0.0638	0.0074	-0.0638	-0.0630
4	0.0082	0.0127	0.0638	0.2515	-0.1877	-0.0556	-0.0511
4-3	-0.0008	-0.0045	0.0074	-0.1877	-0.1596	-0.0082	-0.0968
3-1	0.0222	0.0066	-0.0638	-0.0556	-0.0082	-0.6140	0.0204
3-2	0.0074	0.0356	-0.0630	-0.0511	-0.0968	0.0204	0.4390

. . . . (151)

Eliminating the three augmenting rows and columns leaves the bus impedance matrix of the modified network :-

	1	2	3	4	4-3	3-1
1	0.0295	0.0142	0.0085	0.0091	0.0008	0.0219
2	0.0142	0.0409	0.0133	0.0168	0.0033	0.0049
3	0.0085	0.0133	0.0621	0.0564	-0.0065	-0.0609
4	0.0091	0.0168	0.0564	0.2455	-0.1990	-0.0532
4-3	0.0008	0.0033	-0.0065	-0.1990	-0.1810	-0.0037
3-1	0.0219	0.0049	-0.0609	-0.0532	-0.0037	-0.6150

	1	2	3	4	4-3
1	0.0303	0.0144	0.0063	0.0072	0.0007
2	0.0144	0.0409	0.0128	0.0164	0.0033
3	0.0063	0.0128	0.0681	0.0617	-0.0061
4	0.0072	0.0164	0.0617	0.2501	-0.1987
4-3	0.0007	0.0033	-0.0061	-0.1987	-0.1810

$\boxed{Z}$ =

	1	2	3	4
1	0.0303	0.0144	0.0063	0.0064
2	0.0144	0.0410	0.0127	0.0128
3	0.0063	0.0127	0.0683	0.0684
4	0.0064	0.0128	0.0684	0.4686

. . . . (152)

A.7.2. <u>EQUIVALENT NETWORK AND PARTIAL NETWORK MATRIX COMBINATION.</u>

Fig.11 shows the network of Fig.10 divided at nodes 1 and 2 into two sub-networks "A" and "B". For sub-network "A" the bus impedance matrix $\boxed{Z_A}$ is given in eqn.(129) :-

$$\boxed{Z_A} = \begin{array}{c|cc} & 1 & 2 \\ \hline 1 & 0.032 & 0.016 \\ 2 & 0.016 & 0.048 \end{array} \quad \ldots\ldots \quad (153)$$

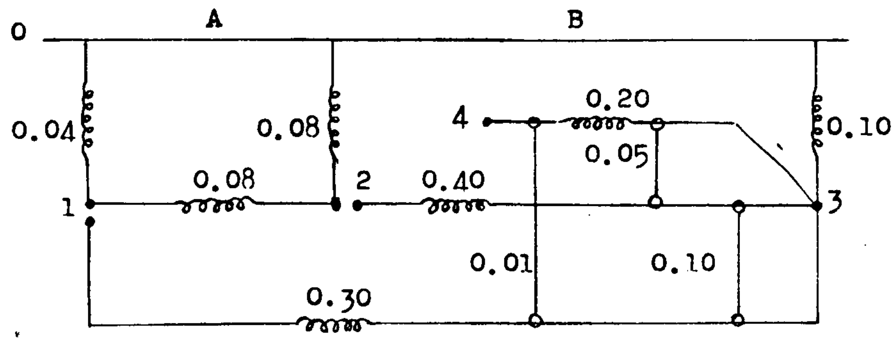


Fig.11

The bus impedance matrix $\boxed{Z_B}$ of sub-network "B" is calculated commencing with generator element 3-0 :-

$$\boxed{Z_B'} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0.10 \end{array} \quad \ldots\ldots \quad (154)$$

The group of three mutually coupled elements 3-1, 4-3 and 3-2 which form branches establishing nodes 1, 4 and 2 respectively are now added to eqn.(154) by the branch element rule, eqn.(41) :-

$$\boxed{Z'} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0.40 & 0.10 & 0.10 & 0.10 \\ 2 & 0.10 & 0.50 & 0.10 & 0.10 \\ 3 & 0.10 & 0.10 & 0.10 & 0.10 \\ 4 & 0.10 & 0.10 & 0.10 & 0.30 \end{array} \quad \ldots\ldots \quad (155)$$

The group mutual impedance matrix $\boxed{z_m}$ is given in eqn.(125) and from eqns.(99) and (101) the connection sub-matrix $\boxed{C_b}$ is :-

$$\boxed{C_b} = \begin{array}{c|ccc} & 1 & 4 & 2 \\ \hline 3\text{-}1 & -1 & 0 & 0 \\ 4\text{-}3 & 0 & 1 & 0 \\ 3\text{-}2 & 0 & 0 & -1 \end{array} \quad \ldots\ldots \quad (156)$$

and the triple matrix product :-

$$\boxed{C_{bt} z_m C_b} = \begin{array}{c|ccc} & 1 & 4 & 2 \\ \hline 1 & 0 & 0.01 & 0.10 \\ 4 & 0.01 & 0 & 0.05 \\ 2 & 0.10 & 0.05 & 0 \end{array} \quad \ldots\ldots \quad (157)$$

As all group elements are branches, adding eqn. (157) to eqn. (155) according to eqn. (32) gives the bus impedance matrix of sub-network "B" :-

$$\boxed{Z_B} = \boxed{Z''} = \boxed{Z'} + \boxed{C_{bt} z_m C_b}$$

$$= \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0.40 & 0.20 & 0.10 & 0.11 \\ 2 & 0.20 & 0.50 & 0.10 & 0.15 \\ 3 & 0.10 & 0.10 & 0.10 & 0.10 \\ 4 & 0.11 & 0.15 & 0.10 & 0.30 \end{array} \quad \ldots\ldots \quad (158)$$

From this equation the bus impedance matrix for B's equivalent network retaining nodes 1 and 2 is derived :-

$$\boxed{Z_{eqiv}} = \begin{array}{c|cc} & 1 & 2 \\ \hline 1 & 0.40 & 0.20 \\ 2 & 0.20 & 0.50 \end{array} \quad \ldots\ldots \quad (159)$$

Inverting :-

$$\boxed{Y_{eqiv}} = \begin{array}{c|cc} & 1 & 2 \\ \hline 1 & 3.125 & -1.25 \\ 2 & -1.25 & 2.50 \end{array} \quad \ldots\ldots \quad (160)$$

Hence the per unit admittances of the elements of the equivalent circuit are :-

$$y_{21} = 1.25 \qquad y_{10} = 3.125-1.25 = 1.875$$
$$y_{20} = 2.5-1.25 = 1.25 \quad \ldots\ldots \quad (161)$$

Inverting these admittances gives the per unit impedances :-

$$z_{21} = 0.8 \qquad z_{10} = 0.5333 \qquad z_{20} = 0.8 \quad \ldots\ldots \quad (162)$$

and the equivalent circuit is shown below in Fig.12 :-

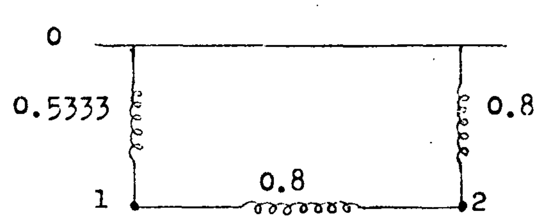


Fig.12

The bus impedance matrix of sub-network "A" including the effect of sub-network "B" is now calculated by two methods :-

(a) elements 2-0, 1-0 and 2-1 with self impedances given by eqn. (162) are added to the bus impedance matrix of sub-network "A", eqn. (153). Starting with element 2-0 and applying the rules for adding a loop element connected to the reference node, eqns. (73)

and (74) :-

$$\boxed{Z'} = \begin{array}{cc} & \begin{array}{ccc} 1 & 2 & 2\text{-}0 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 2\text{-}0 \end{array} & \begin{bmatrix} 0.032 & 0.016 & -0.016 \\ 0.016 & 0.048 & -0.048 \\ -0.016 & -0.048 & 0.848 \end{bmatrix} \end{array} \quad . \quad . \quad . \quad . \quad (163)$$

Eliminate row and column 2-0 and add loop element 1-0 by eqns.(73) and (74) :-

$$\boxed{Z'} = \begin{array}{cc} & \begin{array}{ccc} 1 & 2 & 1\text{-}0 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 1\text{-}0 \end{array} & \begin{bmatrix} 0.0317 & 0.0151 & -0.0317 \\ 0.0151 & 0.0453 & -0.0151 \\ -0.0317 & -0.0151 & 0.5650 \end{bmatrix} \end{array} \quad . \quad . \quad . \quad (164)$$

Eliminate row and column 1-0 and add element 2-1 by the loop element rules, eqns.(67) and (70) :-

$$\boxed{Z'} = \begin{array}{cc} & \begin{array}{ccc} 1 & 2 & 2\text{-}1 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 2\text{-}1 \end{array} & \begin{bmatrix} 0.0299 & 0.0143 & 0.0156 \\ 0.0143 & 0.0449 & -0.0306 \\ 0.0156 & -0.0306 & 0.8462 \end{bmatrix} \end{array} \quad . \quad . \quad . \quad (165)$$

Eliminating the augmenting row and column 2-1 gives the required bus impedance matrix :-

$$\boxed{Z} = \begin{array}{cc} & \begin{array}{cc} 1 & 2 \end{array} \\ \begin{array}{c} 1 \\ 2 \end{array} & \begin{bmatrix} 0.0296 & 0.0149 \\ 0.0149 & 0.0438 \end{bmatrix} \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \quad (166)$$

(b) It is assumed that $\boxed{Z_{eqiv}}$ given in eqn.(159) is the impedance matrix for a group of two generator elements 1-0 and 2-0 having per unit self impedances 0.40 and 0.50 respectively and mutual coupling impedance 0.20. Applying eqns.(73) and (74) to add the two loop elements 1-0 and 2-0 to $\boxed{Z_A}$ from eqn.(153) :-

$$\boxed{Z'} = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 1\text{-}0 & 2\text{-}0 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 1\text{-}0 \\ 2\text{-}0 \end{array} & \begin{bmatrix} 0.032 & 0.016 & -0.032 & -0.016 \\ 0.016 & 0.048 & -0.016 & -0.048 \\ -0.032 & -0.016 & 0.432 & 0.016 \\ -0.016 & -0.048 & 0.016 & 0.548 \end{bmatrix} \end{array} \quad . \quad . \quad . \quad (167)$$

As the two loop elements connect established nodes, the connection sub-matrix $\boxed{C_b}$ is a unit matrix and hence the product :-

$$\boxed{C_{bt} z_m C_b} = \begin{array}{c|cc} & 1\text{-}0 & 2\text{-}0 \\ \hline 1\text{-}0 & 0 & 0.20 \\ 2\text{-}0 & 0.20 & 0 \end{array} \quad \ldots\ldots \quad (168)$$

Combining eqns.(167) and (168) according to eqn.(32) :-

$$\boxed{Z''} = \begin{array}{c|cccc} & 1 & 2 & 1\text{-}0 & 2\text{-}0 \\ \hline 1 & 0.032 & 0.016 & -0.032 & -0.016 \\ 2 & 0.016 & 0.048 & -0.016 & -0.048 \\ 1\text{-}0 & -0.032 & -0.016 & 0.432 & 0.216 \\ 2\text{-}0 & -0.016 & -0.048 & 0.216 & 0.548 \end{array} \quad \ldots\ldots \quad (169)$$

Eliminate row and column 2-0 :-

	1	2	1-0
1	0.0315	0.0146	-0.0257
2	0.0146	0.0438	0.0029
1-0	-0.0257	0.0029	0.3470

Eliminating row and column 1-0 leaves the bus impedance matrix of sub-network "A" including the effect of sub-network "B" :-

$$\boxed{Z} = \begin{array}{c|cc} & 1 & 2 \\ \hline 1 & 0.0296 & 0.0148 \\ 2 & 0.0148 & 0.0438 \end{array} \quad \ldots\ldots \quad (170)$$

Eqns.(166) and (170) are both equal to the bus impedance matrix of the equivalent circuit retaining nodes 1 and 2 for the network shown in Fig.10 and which can be derived from eqn.(136) by inspection.

A.8. COMBINATION OF NETWORK BUS IMPEDANCE MATRIXES.

The addition of the bus impedance matrixes $\boxed{Z_{eqiv}}$ in eqn.(159) and $\boxed{Z_A}$ in eqn.(153) to give $\boxed{Z}$ in eqn.(170) is an example of adding the bus impedance matrixes of networks joined at two nodes; a procedure which can be extended to networks joined at one or more nodes.

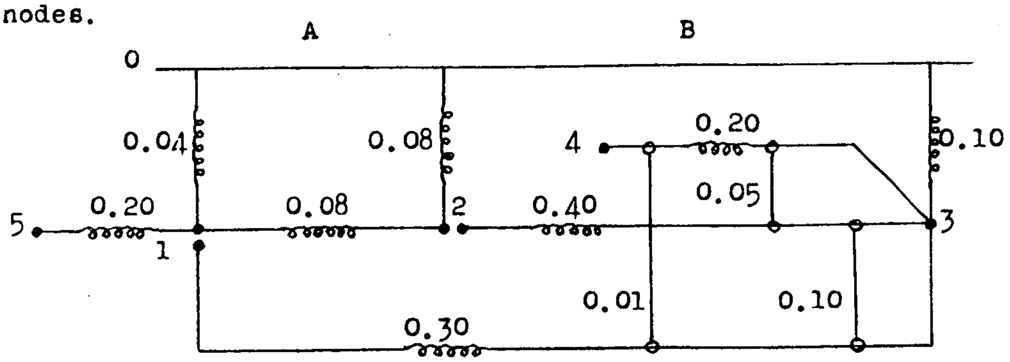


Fig.13

Consider the two networks "A"and "B" shown in Fig.13 which are to be connected together at nodes 1 and 2. The bus impedance matrix of network "B" is given in eqn.(158) and that for network "A" is found by adding branch element 5-1, self impedance 0.2 per unit, to the bus impedance matrix given in eqn.(153), i.e. :-

$\boxed{Z_A}$ = ... (171)

	1	2	5
1	0.032	0.016	0.032
2	0.016	0.048	0.016
5	0.032	0.016	0.232

When adding network "A" to network "B", $\boxed{Z_A}$ can be considered as the impedance matrix of an equivalent group of mutually coupled generator elements 1-0, 2-0 and 5-0, i.e. the diagonal and off-diagonal terms are respectively self and mutual impedances of these equivalent elements. As nodes 1 and 2 are common to both networks, elements 1-0 and 2-0 form loops with element 5-0 forming a branch, hence starting with $\boxed{Z_B}$ and adding the equivalent element self impedances by the generator branch and loop rules followed by the mutual impedances (noting that $\boxed{C_b}$ is a unit matrix) the matrix $\boxed{Z''}$ for the combined network is :-

$\boxed{Z''}$ =

	1	2	3	4	5	1-0	2-0
1	0.40	0.20	0.10	0.11	0	-0.40	-0.20
2	0.20	0.50	0.10	0.15	0	-0.20	-0.50
3	0.10	0.10	0.10	0.10	0	-0.10	-0.10
4	0.11	0.15	0.10	0.30	0	-0.11	-0.15
5	0	0	0	0	0.232	0.032	0.016
1-0	-0.40	-0.20	-0.10	-0.11	0.032	0.432	0.216
2-0	-0.20	-0.50	-0.10	-0.15	0.016	0.216	0.548

. . . (172)

Eliminating row and column 2-0 :-

	1	2	3	4	5	1-0
1	0.3270	0.0175	0.0635	0.0553	0.0058	-0.3212
2	0.0175	0.0438	0.0088	0.0131	0.0146	-0.0029
3	0.0635	0.0088	0.0818	0.0726	0.0029	-0.0606
4	0.0553	0.0131	0.0726	0.2589	0.0044	-0.0509
5	0.0058	0.0146	0.0029	0.0044	0.2315	0.0257
1-0	-0.3212	-0.0029	-0.0606	-0.0509	0.0257	0.3470

then row and column 1-0 leaves the bus impedance matrix of the connected networks :-

$$\boxed{Z} = \begin{array}{c|c|c|c|c|c|} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0.0296 & 0.0148 & 0.0074 & 0.0082 & 0.0296 \\ \hline 2 & 0.0148 & 0.0438 & 0.0083 & 0.0127 & 0.0148 \\ \hline 3 & 0.0074 & 0.0083 & 0.0712 & 0.0637 & 0.0074 \\ \hline 4 & 0.0082 & 0.0127 & 0.0637 & 0.2514 & 0.0082 \\ \hline 5 & 0.0296 & 0.0148 & 0.0074 & 0.0082 & 0.2296 \\ \hline \end{array} \quad . \quad . \quad (173)$$

Comparing this with eqn.(136) shows that the bus impedance matrix of the connected networks is the same as that derived from the bus impedance matrix given in eqn.(136) by adding a branch element 5-1, self impedance 0.2, to the network in Fig.10. Hence, if two networks having bus impedance matrixes $\boxed{Z_A}$ and $\boxed{Z_B}$ are joined at one or more nodes, then the impedance matrix $\boxed{Z''}$ of the combined network is given by (Ref. 2):-

$$\boxed{Z''} = \begin{array}{c|c|c|c|} & & \text{branch} & \text{loop} \\ \hline & Z_B & 0 & -Z_{col} \\ \hline \text{branch} & 0 & \multicolumn{2}{c|}{} \\ \cline{1-2} \text{loop} & -Z_{row} & \multicolumn{2}{c|}{Z_A+Z_{eqiv}} \\ \hline \end{array} \quad . \quad . \quad . \quad (174)$$

where $\boxed{Z_{col}}$ and $\boxed{Z_{row}}$ are respectively the column and row terms of $\boxed{Z_B}$ corresponding to nodes common to both networks and $\boxed{Z_{eqiv}}$ is the sub-matrix of $\boxed{Z_B}$ corresponding to the common nodes. The bus impedance matrix $\boxed{Z}$ of the combined networks is derived from $\boxed{Z''}$ by eliminating the "loop" rows and columns.

The impedance matrix $\boxed{Z''}$ of eqn.(174) resulting from the combination of two network bus impedance matrixes has the simple form shown because the added bus impedance matrix is assumed to be the self and mutual impedance matrix of a group of generator elements which form loops in the case of nodes common to both networks, otherwise branches, and for which the connection sub-matrix $\boxed{C_b}$ is a unit matrix.

A.9. DIGITAL COMPUTER SHORT CIRCUIT PROGRAMME.

In the digital computer programme (HEI 2a) the network bus impedance calculation set out in the above example is generalised for application to any power system network with any combination of node numbering and listing orders for element self and mutual coupling

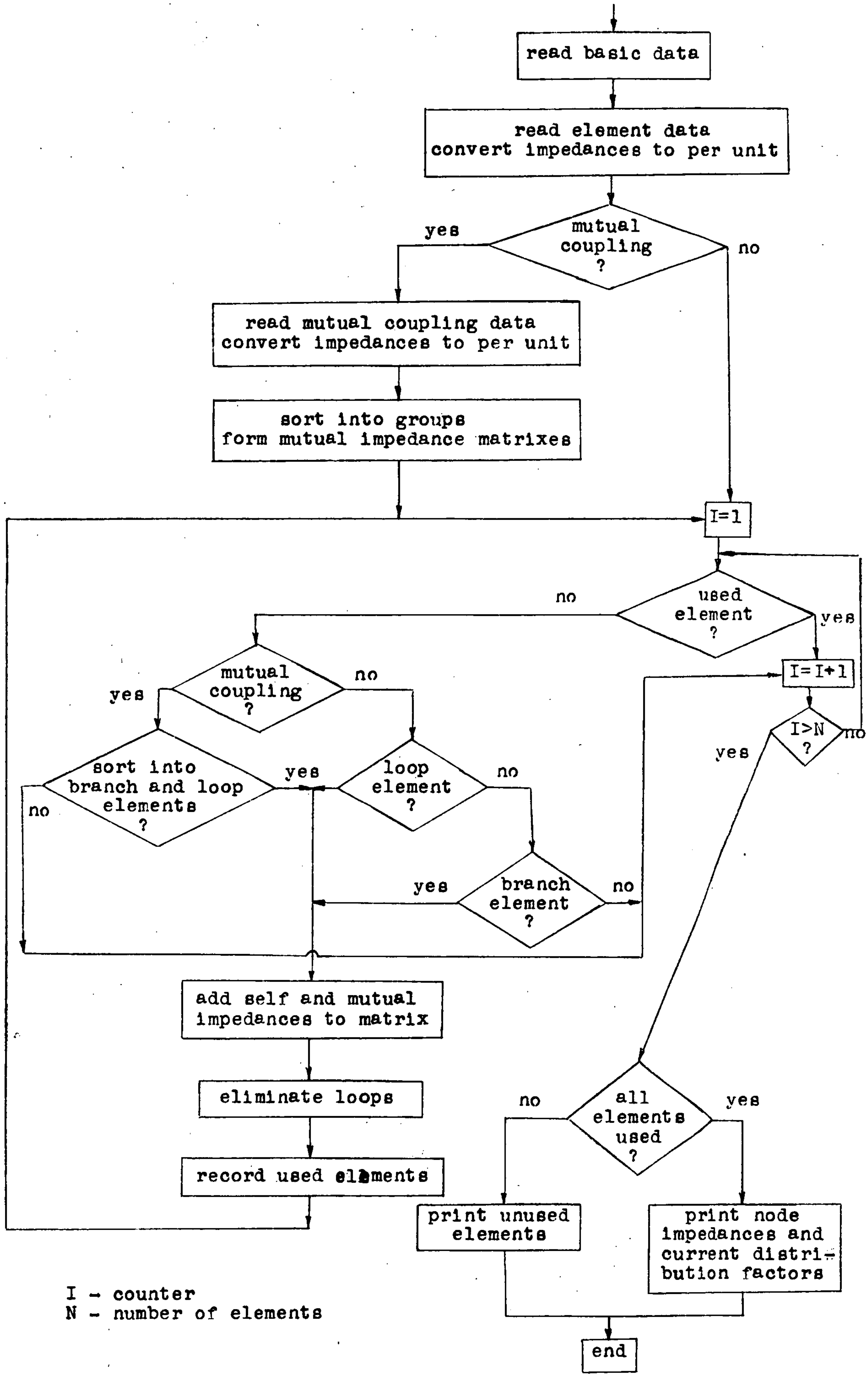

read basic data
read element data
convert impedances to per unit
mutual coupling ?
yes
no
read mutual coupling data
convert impedances to per unit
sort into groups
form mutual impedance matrixes
I=1
used element ?
no
yes
I=I+1
I>N ?
no
yes
mutual coupling ?
yes
no
sort into branch and loop elements ?
yes
no
loop element ?
no
branch element ?
yes
no
add self and mutual impedances to matrix
eliminate loops
record used elements
all elements used ?
no
yes
print unused elements
print node impedances and current distribution factors
end
I - counter
N - number of elements

Fig.14

impedances. The flow diagram for the computer programme is given in Fig.14. Because power system transmission line impedances are usually given in ohms, provision is made for listing element self and mutual impedances either in ohms or per unit by including the base MVA in Table 1, Basic Data, and nominal voltages (in kV) for each impedance in Tables 2 and 3. As each line of data is read, the impedance is converted to per unit and stored together with node numbers for the self impedances and element numbers for the mutual impedances.

The next step in the calculation is sorting the mutual couplings into groups, i.e. each set of elements directly or indirectly coupled together is listed with its associated matrix of mutual impedances. As this matrix is symmetrical only the lower triangular part is stored.

After this preliminary re-organisation of the data, the calculation of the network bus impedance matrix starts with the first generator element from the list (mutual coupling to generator elements is not permitted in the programme, a minor restriction with power system networks). Thus a network node, other than the reference, is now established and the next element processed may be connected to the reference or to this node.

The network bus impedance matrix is formed in a step-by-step procedure involving the repeated searching, until all elements are used, of the list for an element or a group of mutually coupled elements which can be processed. During the calculation only the lower triangular part of the bus impedance matrix is stored and used as it is symmetrical. If the element being processed is a branch then a corresponding row is formed in the matrix using the rule in eqn.(41) or (46), if a loop then an augmenting row is formed using the rule in eqns.(67) and (70) or eqns.(73) and (74) which is then eliminated by application of Kron's rule.

When a group of mutually coupled elements is found, it is examined to determine whether there is an order in which these elements can be processed at this stage of the calculation; if so they are listed in the processing order with branch elements first and loop elements last. The self impedances of the group elements are added to the impedance matrix according to the branch and loop element rules followed by derivation of the group connection sub-matrix $\boxed{C_b}$ which is built up by examining the group elements one at a time in the processing order and forming a corresponding row and column - row terms zero, column terms found by tracing the connections from element nodes to an established network node taking account of relative node magnitudes as shown in eqns. (99) to (105). The matrix of group mutual impedances is set up by rearranging the the terms of the matrix formed in the initial sorting to correspond with the processing order, the triple matrix product $\boxed{C_{bt} z_m C_b}$ calculated and added to the network impedance matrix and then augmenting rows corresponding to loop elements are eliminated.

As mentioned above mutual coupling with generator elemnts is not permitted in the computer programme - in power system networks this is not a significant restriction which can be overcome by using a "dummy" node to divide the element into two parts and lumping the mutual impedance in the part remote from the reference node - no error is introduced by this procedure as all self and mutual impedances are lumped in any case.

When all listed elements are used, the calculation of the bus impedance matrix is complete but before printing results the group self and mutual impedance matrixes are inverted as the admittances $\boxed{y_{qp}}$ are required in the calculation of current distribution factors. For each network node or busbar the current distribution factors for elements connected to that node are calculated and printed, the per unit driving point impedance and the base MVA divided by this impedance are also printed. In addition, current distribution factors for other specified elements may be obtained and the impedance matrix terms for anequivalent network retaining specified nodes punched on paper tape in a form that can be used as data for a computer programme for calculation of the elements of the equivalent network.

B. THE BUS ADMITTANCE MATRIX AND POWER SYSTEM FAULT CALCULATIONS.

B.1. INTRODUCTION.

In contrast with the bus impedance matrix which is full, i.e. every term has a value other than zero, the bus admittance matrix reflects network structure and, as in power systems usually there are few connections between nodes or busbars, it is sparse, i.e. most off-diagonal terms are zero. Hence, if the admittance instead of the impedance matrix could be used for power system fault studies, a considerable saving in computer storage requirements would be possible with a large network by storing only non-zero terms. Also there would be a reduction in the number of arithmetic operations during computation thus reducing computer running time and possibly round-off errors, although the latter is not likely to be significant.

However in power system fault studies, terms of the bus impedance matrix are required for calculating fault powers and current distribution factors - eqns.(111), (113) and (116) - and if complete inversion of the admittance matrix is necessary the advantages of sparsity are lost. W.F.Tinney and J.W.Walker (ref.46) of the Bonneville Power Administration have devised a method for calculating terms of the bus impedance matrix from a factored inverse derived from the bus admittance matrix by Gaussian elimination of terms below the main diagonal and show also that by selecting a preferred elemination order, the resulting factored inverse has almost as many zero terms as the original admittance matrix. In a power system the preferred order is determined by the number of network connections to the nodes, excluding connections to the reference node, i.e. the number of off-diagonal terms in the corresponding row or column. The elimination starts with nodes connected to one other node only, followed by those with connections to two other nodes, then those connected to three, etc. While this is not necessarily the optimum order it has the advantage of simplicity and, in general does produce for power systems a factored inverse which reflects the sparsity of the admittance matrix. From the factored inverse it is easy to derive one row at a time, or only part of a row if all the terms in the row are not required, of the bus impedance matrix from which fault powers and current distribution factors can be calculated.

B.2. CONNECTION MATRIX RELATIONS

For the network shown in Fig.1(a) the relation between node currents and voltages is given by the equation :-

$$\boxed{I_A'} = \boxed{Y_A}\,\boxed{V_A} \quad . \; . \; . \; . \; . \; . \; . \; . \; . \; . \qquad (175)$$

where $\boxed{Y_A}$ is the bus admittance which like the bus impedance matrix has a row and column corresponding to each network node except the reference node, $\boxed{I_A'}$ are the currents flowing into the nodes and $\boxed{V_A}$ the node voltages referred to the reference. Comparing with eqn.(1) :-

$$\boxed{Y_A} = \boxed{Z_A}^{-1} \quad . \; . \; . \; . \; . \; . \; . \; . \qquad (176)$$

The equation connecting element currents and node voltages for the mutually coupled group of elements shown in Fig.1(b) can be written :-

$$\boxed{I_{qp}} = \boxed{y_{qp}}\,\boxed{V_q - V_p} \quad . \; . \; . \; . \; . \; . \; . \qquad (177)$$

where $\boxed{y_{qp}}$, the group self and mutual admittance matrix, is the inverse of the self and mutual impedance matrix $\boxed{z_{qp}}$ of eqn.(2) :-

$$\boxed{y_{qp}} = \boxed{z_{qp}}^{-1} \quad . \; . \; . \; . \; . \; . \; . \; . \qquad (178)$$

Corresponding to eqn.(29), $\boxed{y_{qp}}$ may be considered as the sum of a diagonal matrix $\boxed{y_s}$ and a matrix $\boxed{y_m}$ having zero diagonal terms :-

$$\boxed{y_{qp}} = \boxed{y_s} + \boxed{y_m} \quad . \; . \; . \; . \; . \; . \; . \qquad (179)$$

The terms of $\boxed{y_s}$ are "equivalent" self admittances of the group elements and are not reciprocals of the corresponding element self impedances in $\boxed{z_s}$ while $\boxed{y_m}$ are "equivalent" mutual admittances between group elements. In general $\boxed{y_{qp}}$ is a full matrix with "equivalent" mutual admittances between each pair of group elements even if $\boxed{z_{qp}}$ is sparse and all pairs are not directly coupled.

Combining eqns.(175) and (177) into one equation :-

$$\begin{array}{|c|}\hline I_A' \\ \hline I_{qp} \\ \hline\end{array} = \begin{array}{|c|c|}\hline Y_A & 0 \\ \hline 0 & y_{qp} \\ \hline\end{array}\;\begin{array}{|c|}\hline V_A \\ \hline V_q - V_p \\ \hline\end{array} \quad . \; . \; . \; . \; . \; . \qquad (180)$$

If the network shown in Fig.1(a) and the group of elements Fig.1(b) are connected together so that the power before and after interconnection is unchanged, i.e. the current flowing in each element is unaltered by interconnection, then the relationship between the currents before and after interconnection is given by the equation :-

$$\begin{bmatrix} I_A \\ I_B \end{bmatrix} = \begin{bmatrix} 1 & A_{at} \\ 0 & A_{bt} \end{bmatrix} \begin{bmatrix} I_A' \\ I_{qp} \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (181)$$

where $\boxed{I_A}$ are currents at the nodes of network (a) after interconnection and $\boxed{I_B}$ are, after interconnection, node currents corresponding to branch elements of group (b) and element currents corresponding to loop elements. Each term of the connection sub-matrixes $\boxed{A_{at}}$ and $\boxed{A_{bt}}$ is 0, +1 or -1 and they are the transpose of matrixes $\boxed{A_a}$ and $\boxed{A_b}$ respectively which are related to the bus incidence matrix as shown in eqn.(204). Comparing eqn.(181) with eqn.(4) :-

$$\begin{bmatrix} 1 & A_{at} \\ 0 & A_{bt} \end{bmatrix} = \begin{bmatrix} 1 & C_a \\ 0 & C_b \end{bmatrix}^{-1} = \begin{bmatrix} 1 & -C_a C_b^{-1} \\ 0 & C_b^{-1} \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \qquad (182)$$

Therefore, equating sub-matrixes on both sides of the equation :-

$$\boxed{A_{bt}} = \boxed{C_b^{-1}} \quad \text{and} \quad \boxed{A_{at}} = -\boxed{C_a}\boxed{C_b}^{-1} \quad . \quad . \qquad (183)$$

$\boxed{A_{bt}}$ and $\boxed{C_b}$ are square matrixes of size equal to the number of elements in the group Fig.1(b). As the sub-matrix $\boxed{C_b}$ can be expressed in the form having zero for all terms below the main diagonal and +1 or -1 for its diagonal terms, its determinant is equal to the product of the diagonal terms, i.e. +1 or -1, and hence it has an inverse. The relation between the connection matrixes shown in eqns.(182) and (183) holds in general for connecting any group of mutually coupled elements to a network as no restrictions have been placed on the network and group shown in Figs.1(a) and (b).

Assuming that power is invariant when the group of elements Fig.1(b) is connected to the network Fig.1(a) then the relation between voltages and currents before and after interconnection is :-

$$\begin{bmatrix} V_{At} & V_{Bt} \end{bmatrix} \begin{bmatrix} I_A^* \\ I_B^* \end{bmatrix} = \begin{bmatrix} V_{At} & (V_q-V_p)_t \end{bmatrix} \begin{bmatrix} I_A'^* \\ I_{qp}^* \end{bmatrix} \quad . \quad . \quad . \quad . \qquad (184)$$

where "t" denotes the transposed matrix and "*" the complex conjugate. Substituting for $\begin{bmatrix} I_A^* \\ I_B^* \end{bmatrix}$ from eqn.(181) :-

$$\begin{bmatrix} V_{At} & V_{Bt} \end{bmatrix} \begin{bmatrix} 1 & A_{at} \\ 0 & A_{bt} \end{bmatrix} \begin{bmatrix} I_A^* \\ I_{qp}^* \end{bmatrix} = \begin{bmatrix} V_{At} & (V_q-V_p)t \end{bmatrix} \begin{bmatrix} I_A^* \\ I_{qp}^* \end{bmatrix}$$

Cancelling the current terms and transposing :-

$$\begin{bmatrix} 1 & 0 \\ A_a & A_b \end{bmatrix} \begin{bmatrix} V_A \\ V_B \end{bmatrix} = \begin{bmatrix} V_A \\ V_q-V_p \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \qquad (185)$$

In this equation $\boxed{V_A}$ are voltages at the nodes of network (a) and

$\boxed{V_B}$ are voltages at the nodes established in the interconnected network shown in Fig.1(c) by branch elements of group (b) and zero corresponding to loop elements of the group.

Premultiplying both sides of eqn.(180) by $\begin{bmatrix} 1 & A_{at} \\ 0 & A_{bt} \end{bmatrix}$ and substituting for currents and voltages from eqns.(181) and (185) :-

$$\begin{bmatrix} I_A \\ I_B \end{bmatrix} = \begin{bmatrix} 1 & A_{at} \\ 0 & A_{bt} \end{bmatrix} \begin{bmatrix} Y_A & 0 \\ 0 & y_{qp} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A_a & A_b \end{bmatrix} \begin{bmatrix} V_A \\ V_B \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad (186)$$

From this equation it follows that the admittance matrix of the network formed by connecting the group of elements shown in Fig.1(b) to the network Fig.1(a) is :-

$$\begin{aligned} \boxed{Y''} &= \begin{bmatrix} 1 & A_{at} \\ 0 & A_{bt} \end{bmatrix} \begin{bmatrix} Y_A & 0 \\ 0 & y_{qp} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A_a & A_b \end{bmatrix} \\ &= \begin{bmatrix} Y_A & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} A_{at}y_{qp}A_a & A_{at}y_{qp}A_b \\ A_{bt}y_{qp}A_a & A_{bt}y_{qp}A_b \end{bmatrix} \\ &= \boxed{Y_A} + \begin{bmatrix} A_{at} \\ A_{bt} \end{bmatrix} \boxed{y_{qp}} \begin{bmatrix} A_a & A_b \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (187) \end{aligned}$$

From eqns.(186) and (187) it follows that :-

(a) as the terms of $\boxed{I_B}$ and $\boxed{V_B}$ corresponding to loop elements are respectively element currents and zero, the bus admittance matrix $\boxed{Y}$ relating node currents and voltages of the interconnected network is found by discarding the rows and columns of $\boxed{Y''}$ corresponding to these loop elements, i.e. :-

$$\begin{aligned} \boxed{Y} &= \begin{bmatrix} 1 & A_{at} \\ 0 & A'_{bt} \end{bmatrix} \begin{bmatrix} Y_A & 0 \\ 0 & y_{qp} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A_a & A'_b \end{bmatrix} \\ &= \boxed{Y_A} + \begin{bmatrix} A_{at} \\ A'_{bt} \end{bmatrix} \boxed{y_{qp}} \begin{bmatrix} A_a & A'_b \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (188) \end{aligned}$$

where $\boxed{A'_b}$ is $\boxed{A_b}$ with columns corresponding to loop elements omitted. For comparison this is a simpler procedure thanthat for obtaining the bus impedance matrix $\boxed{Z}$ which requires the elimination of rows and columns from $\boxed{Z''}$ corresponding to loop elements;

(b) in forming $\boxed{Y''}$ the terms of $\boxed{Y_A}$ are modified by the addition of $\boxed{A_{at}y_{qp}A_a}$ calculated from the group admittance matrix as compared with the the formation of $\boxed{Z''}$, eqn.(31), in which $\boxed{Z_A}$ is unchanged;

(c) in $\boxed{Y''}$ the terms in the rows and columns augmenting $\boxed{Y_A}$ are calculated from the group admittance matrix $\boxed{y_{qp}}$ whereas the rows and columns augmenting $\boxed{Z_A}$ in eqn.(31) contain terms of $\boxed{Z_A}$;

(d) from eqns.(179) and (187) it follows that group equivalent mutual admittances are in all four sub-matrixes of $\boxed{Y''}$, whereas in eqn.(31) the group mutual impedances appear only in the lower diagonal sub-matrix of $\boxed{Z''}$;

(e) as, in general, for a group with mutual coupling $\boxed{y_{qp}}$ is a full matrix, the bus admittance matrix for a network with mutual coupling is not as sparse as that for the corresponding network without mutual coupling; and

(f) as $\boxed{Y''}$ is determined by adding to $\boxed{Y_A}$ terms calculated from products of $\boxed{y_{qp}}$ and the connection sub-matrixes $\boxed{A_a}$ and $\boxed{A_b}$ and the bus admittance matrix $\boxed{Y}$ derived from $\boxed{Y''}$ by discarding rows and columns corresponding to group loop elements, the order in which the group elements are processed is immaterial. This is in contrast to forming the bus impedance matrix where group elements are processed in a definite order as new rows and columns are formed from existing rows and columns, eqn.(31), and elimination of loop rows and columns modifies the terms of $\boxed{Z''}$ giving the bus impedance matrix $\boxed{Z}$.

Note :- As $\boxed{y_{qp}}$ is symmetrical, the diagonal sub-matrixes $\boxed{A_{at}y_{qp}A_a}$ and $\boxed{A_{bt}y_{qp}A_b}$ are symmetrical and the off-diagonal sub-matrix $\boxed{A_{at}y_{qp}A_b}$ is the transpose of off-diagonal sub-matrix $\boxed{A_{bt}y_{qp}A_a}$, hence $\boxed{Y''}$ is $\boxed{Y_A}$ plus a symmetrical matrix; and as $\boxed{Y_A}$ has been built up from groups of elements and is therefore symmetrical it follows that $\boxed{Y''}$ and $\boxed{Y}$ are symmetrical.

B.3. STEP-BY-STEP CALCULATION OF BUS ADMITTANCE MATRIX.

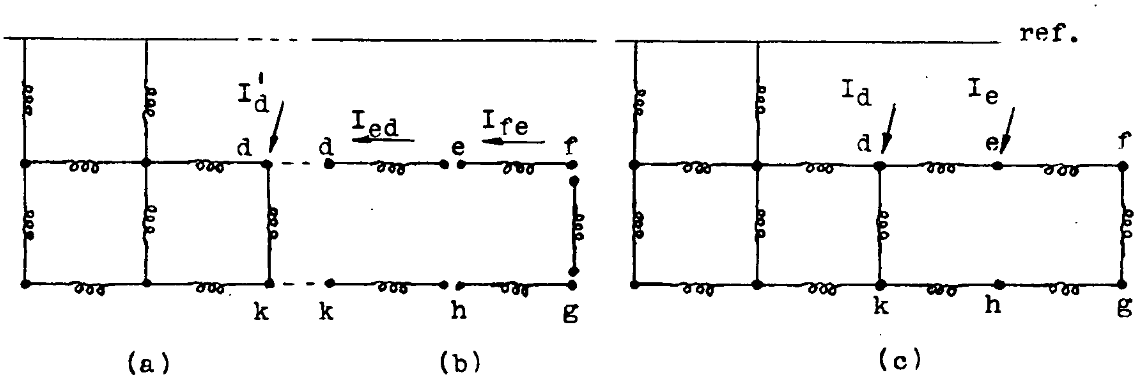


Fig.15

Fig.15 is Fig.1 repeated and from eqn.(181) the relation between element currents $\boxed{I_{qp}}$ and the currents $\boxed{I_B}$ after interconnection is :-

$$\boxed{I_B} = \boxed{A_{bt}}\ \boxed{I_{qp}} \qquad \qquad (189)$$

From Fig.15 assuming that e>d, f>e, k>h, h>g and g>f and that when the elements of (b) are added to network (a) e-d, f-e, k-h and h-g form branches and g-f forms a loop then the terms of the matrix eqn.(189) are :-

$$I_e = I_{ed}-I_{fe} \quad \ldots \ldots \quad (190)$$

$$I_f = I_{fe}-I_{gf} \quad \ldots \ldots \quad (191)$$

$$I_h = I_{hg}-I_{kh} \quad \ldots \ldots \quad (192)$$

$$I_g = I_{gf}-I_{hg} \quad \ldots \ldots \quad (193)$$

$$I_{gf} = I_{gf} \quad \ldots \ldots \quad (194)$$

or, writing as one matrix equation corresponding to eqn.(189) :-

$$\begin{bmatrix} I_e \\ I_f \\ I_h \\ I_g \\ I_{gf} \end{bmatrix} = \begin{array}{c|ccccc|} & e\text{-}d & f\text{-}e & k\text{-}h & h\text{-}g & g\text{-}f \\ e & 1 & -1 & 0 & 0 & 0 \\ f & 0 & 1 & 0 & 0 & -1 \\ h & 0 & 0 & -1 & 1 & 0 \\ g & 0 & 0 & 0 & -1 & 1 \\ g\text{-}f & 0 & 0 & 0 & 0 & 1 \end{array} \begin{bmatrix} I_{ed} \\ I_{fe} \\ I_{kh} \\ I_{hg} \\ I_{gf} \end{bmatrix} \quad \ldots \ldots \quad (195)$$

The product of $\boxed{A_{bt}}$ from this equation and $\boxed{C_b}$ from eqn.(11) is a unit matrix as in eqn.(183).

From eqn.(181) :-

$$\boxed{I_A} = \boxed{I'_A} + \boxed{A_{at}}\,\boxed{I_{qp}} \quad \ldots \ldots \quad (196)$$

For nodes "d" and "k" of the network shown in Fig.15(a), and to which elements of group (b) are connected to form the interconnected network Fig.15(c), the terms of this equation are :-

$$I_d = I'_d-I_{ed} \quad \ldots \ldots \quad (197)$$

$$I_k = I'_k+I_{kh} \quad \ldots \ldots \quad (198)$$

and writing in matrix form :-

$$\begin{bmatrix} I_d \\ I_k \end{bmatrix} = \begin{bmatrix} I'_d \\ I'_k \end{bmatrix} + \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I_{ed} \\ I_{kh} \end{bmatrix} \quad \ldots \ldots \quad (199)$$

Combining eqns.(195) and (199) into one equation relating currents in the interconnected network to node currents in network (a) and element currents in group (b) :-

$$\begin{bmatrix} I_d \\ I_k \\ I_e \\ I_f \\ I_h \\ I_g \\ I_{gf} \end{bmatrix} = \begin{array}{c} \\ d \\ k \\ e \\ f \\ h \\ g \\ g\text{-}f \end{array} \begin{array}{c} \begin{array}{ccccccc} d & k & e\text{-}d & f\text{-}e & k\text{-}h & h\text{-}g & g\text{-}f \end{array} \\ \left[\begin{array}{cc|ccccc} 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \end{array} \begin{bmatrix} I'_d \\ I'_k \\ I_{ed} \\ I_{fe} \\ I_{kh} \\ I_{hg} \\ I_{gf} \end{bmatrix} \quad . \quad . \quad . \quad . \quad (200)$$

The connection matrix in this equation gives the terms in $\boxed{A_{at}}$ and $\boxed{A_{bt}}$ of eqn.(181) omitting the rows of $\boxed{A_{at}}$ in which all terms are zero, i.e. omitting rows and columns corresponding to nodes not common to network (a) and group (b). The equation connecting the node voltages before and after connecting the network and group of elements shown in Figs.15(a) and (b) given in eqn.(185) is found by transposing the connection matrix in eqn.(200) :-

$$\begin{array}{c} \\ d \\ k \\ e\text{-}d \\ f\text{-}e \\ k\text{-}h \\ h\text{-}g \\ g\text{-}f \end{array} \begin{array}{c} \begin{array}{ccccccc} d & k & e & f & h & g & g\text{-}f \end{array} \\ \left[\begin{array}{cc|ccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \hline -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 1 \end{array}\right] \end{array} \begin{bmatrix} V_d \\ V_k \\ V_e \\ V_f \\ V_h \\ V_g \\ 0 \end{bmatrix} = \begin{bmatrix} V_d \\ V_k \\ V_e - V_d \\ V_f - V_e \\ V_k - V_h \\ V_h - V_g \\ V_g - V_f \end{bmatrix} \quad . \quad . \quad . \quad (201)$$

In this equation the columns of $\boxed{A_a}$ in which all terms are zero have been omitted.

Substituting the connection matrix $\boxed{A_a \mid A_b}$ from eqn.(201) into eqn.(187) and using eqn.(179), the matrix of equivalent group self admittancesfor the elements shown in Fig.15(b) which is added to the admittance matrix $\boxed{Y_A}$ of network (a) in finding $\boxed{Y''}$ for the inter-connected network Fig.15(c) is :-

$$\begin{bmatrix} A_{at} \\ A_{bt} \end{bmatrix} \boxed{y_s} \begin{bmatrix} A_a & A_b \end{bmatrix} =$$

$$
\begin{array}{c}
d\\ k\\ e\\ f\\ h\\ g\\ g\text{-}f
\end{array}
\begin{bmatrix}
-1 & 0 & 0 & 0 & 0\\
0 & 0 & 1 & 0 & 0\\
1 & -1 & 0 & 0 & 0\\
0 & 1 & 0 & 0 & -1\\
0 & 0 & -1 & 1 & 0\\
0 & 0 & 0 & -1 & 1\\
0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
y_{ed} & 0 & 0 & 0 & 0\\
0 & y_{fe} & 0 & 0 & 0\\
0 & 0 & y_{kh} & 0 & 0\\
0 & 0 & 0 & y_{hg} & 0\\
0 & 0 & 0 & 0 & y_{gf}
\end{bmatrix}
\begin{array}{c}
\begin{array}{ccccccc} d & k & e & f & h & g & g\text{-}f \end{array}\\
\begin{bmatrix}
-1 & 0 & 1 & 0 & 0 & 0 & 0\\
0 & 0 & -1 & 1 & 0 & 0 & 0\\
0 & 1 & 0 & 0 & -1 & 0 & 0\\
0 & 0 & 0 & 0 & 1 & -1 & 0\\
0 & 0 & 0 & -1 & 0 & 1 & 1
\end{bmatrix}
\end{array}
\quad . \qquad (202)
$$

Multiplying the three matrixes on the right hand side gives the matrix of equivalent self admittances :-

$$
\begin{bmatrix} A_{at} \\ A_{bt} \end{bmatrix}
\begin{bmatrix} y_e \end{bmatrix}
\begin{bmatrix} A_a & A_b \end{bmatrix} =
$$

	d	k	e	f	h	g	g-f
d	y_{ed}	0	$-y_{ed}$	0	0	0	0
k	0	y_{kh}	0	0	$-y_{kh}$	0	0
e	$-y_{ed}$	0	$y_{ed}+y_{fe}$	$-y_{fe}$	0	0	0
f	0	0	$-y_{fe}$	$y_{fe}+y_{gf}$	0	$-y_{gf}$	$-y_{gf}$
h	0	$-y_{kh}$	0	0	$y_{kh}+y_{hg}$	$-y_{hg}$	0
g	0	0	0	$-y_{gf}$	$-y_{hg}$	$y_{hg}+y_{gf}$	y_{gf}
g-f	0	0	0	$-y_{gf}$	0	y_{gf}	y_{gf}

(203)

To find the admittance matrix $\boxed{Y''}$ of the interconnected network shown in Fig.15(c), the matrix in eqn.(203) is added to $\boxed{Y_A}$ the bus admittance matrix of network (a) followed by the addition of group mutual admittances from the product $\begin{bmatrix} A_{at} \\ A_{bt} \end{bmatrix} \begin{bmatrix} y_m \end{bmatrix} \begin{bmatrix} A_a & A_b \end{bmatrix}$. The bus admittance matrix $\boxed{Y}$ of the interconnected network Fig.15(c) is then found by discarding the row and column of $\boxed{Y''}$ corresponding to the loop element g-f.

From eqn.(201) it follows that the connection matrix $\begin{bmatrix} A_a & A_b \end{bmatrix}$ is the bus incidence matrix of the group of elements, Fig.15(b), augmented by a column for loop element g-f, i.e. disregarding column g-f, the row corresponding to each element has two non-zero terms -1 in the column for the lower and +1 in the column for the higher numbered node. Hence, the connection matrixes in eqn.(181) are transposes of matrixes $\boxed{A_a}$ and $\boxed{A_b}$. As shown in eqns.(187) and (188) rows and columns of $\boxed{Y''}$ corresponding to loop elements are discarded to give the bus admittance matrix $\boxed{Y}$, therefore the column of the connection matrix in eqn.(201) and the row of its transpose in eqn.(200) which correspond to loop element g-f can be omitted leaving the bus incidence matrix and its transpose. Writing :-

$$\boxed{A} = \boxed{A_a \mid A_b'} \qquad \ldots \qquad (204)$$

where $\boxed{A}$ is the bus incidence matrix of the group of elements, eqn.(188) may be written :-

$$\begin{aligned}\boxed{Y} &= \boxed{Y_A} + \boxed{A_t}\,\boxed{y_{qp}}\,\boxed{A} \\ &= \boxed{Y_A} + \boxed{A_t}\,\boxed{y_s}\,\boxed{A} + \boxed{A_t}\,\boxed{y_m}\,\boxed{A} \qquad \ldots \qquad (205)\end{aligned}$$

splitting $\boxed{y_{qp}}$ into components as in eqn.(179).

Note It follows from eqns.(187) and (188) that the discarded rows and columns of $\boxed{Y''}$ involve terms of $\boxed{y_{qp}}$ only, and from eqn.(201) that the the column omitted from $\boxed{A_b}$ has +1 for the diagonal and zero for all other terms.

Omitting row and column g-f from the matrix in eqn.(203) it follows that the rule for adding the equivalent self admittance y_{ji} of a mutually coupled element j-i to the bus admittance matrix is :- add y_{ji} into the diagonal terms jj and ii corresponding to the element nodes and $-y_{ji}$ into the off-diagonal terms ji and ij. This is the same as the well known rule for forming a bus admittance matrix from uncoupled network elements; also inspection of eqn.(203) shows that the rule applies to both branch and loop elements of the group and, as only addition is involved, the order in which group elements are processed is immaterial. As the bus admittance matrix does not have a row and column corresponding to the reference node, if the group element is a generator its equivalent self admittance is added to the diagonal term corresponding to its other node.

As there is no row and column in $\boxed{Y}$ for the reference node, the column of $\boxed{A}$, eqn.(205), corresponding to the reference node can be omitted when the group contains a generator element, i.e. the row of $\boxed{A}$ corresponding to a generator element has the one entry +1 in the column for the node other than the reference. After the element self, or equivalent self admittances have been added as above, the equivalent group mutual admittances are added in accordance with eqn.(205) from the product $\boxed{A_t}\ \boxed{y_m}\ \boxed{A}$.

Thus in contrast to the bus impedance matrix calculation, the bus admittance matrix is formed by taking the network elements in any order and adding the appropriate self admittance values into row and column terms, while group mutual admittance values may be added afterwards or at any stage of the calculation. Also, as it is immaterial whether group elements form branches or loops, the triple matrix product $\boxed{A_t y_m A}$ can be set up taking the group elements in any order, in contrast to the mutual impedance product $\boxed{C_{bt} z_m C_b}$ which is formed from the group elements in a definite order.

B.3.1. GROUPS OF MUTUALLY COUPLED ELEMENTS.

Typical examples of calculating the matrix product $\boxed{A_t y_m A}$ for groups of mutually coupled elements are set out below.

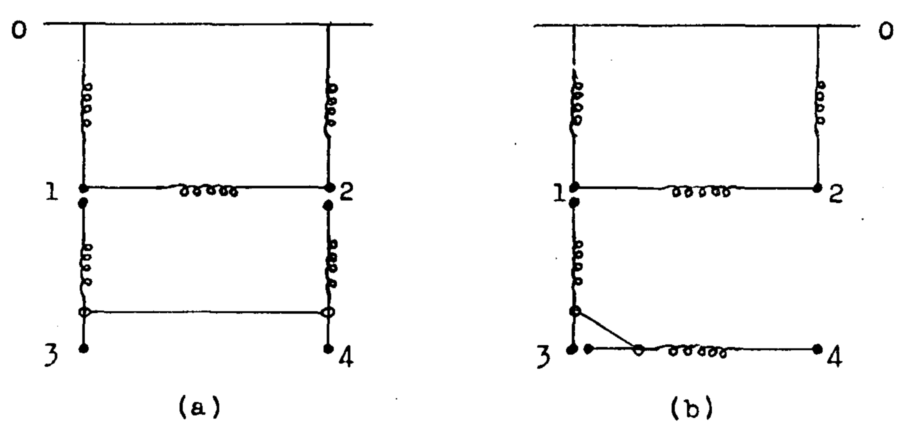


Fig.16

Fig.16(a). Two mutually coupled elements 3-1 and 4-2.

$$\boxed{A} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 3\text{-}1 & -1 & 0 & 1 & 0 \\ 4\text{-}2 & 0 & -1 & 0 & 1 \end{array} \qquad \ldots \ldots \ldots (206)$$

For this group the mutual admittance matrix is :-

$$\boxed{y_m} = \begin{array}{c|cc} & 3\text{-}1 & 4\text{-}2 \\ \hline 3\text{-}1 & 0 & y_{m1} \\ 4\text{-}2 & y_{m1} & 0 \end{array} \qquad \ldots \ldots \ldots (207)$$

and the right hand term of eqn.(205) for the mutual coupling is :-

$$\boxed{A_t y_m A} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array}\begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & y_{m1} \\ y_{m1} & 0 \end{bmatrix} \begin{array}{cccc} 1 & 2 & 3 & 4 \\ \end{array}\begin{bmatrix} -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}$$

$$= \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & y_{m1} & 0 & -y_{m1} \\ 2 & y_{m1} & 0 & -y_{m1} & 0 \\ 3 & 0 & -y_{m1} & 0 & y_{m1} \\ 4 & -y_{m1} & 0 & y_{m1} & 0 \end{array} \quad \ldots\ldots\ldots \quad (208)$$

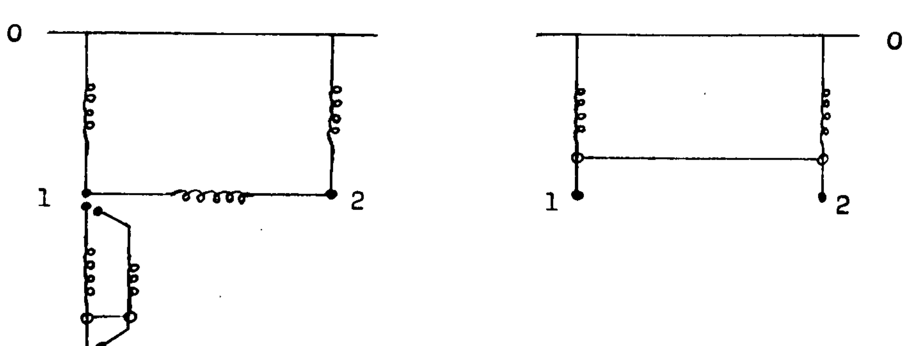
Fig.16(b). Two mutually coupled elements 3-1 and 4-3 with a common node.

$$\boxed{A} = \begin{array}{c|ccc} & 1 & 3 & 4 \\ \hline 3\text{-}1 & -1 & 1 & 0 \\ 4\text{-}3 & 0 & -1 & 1 \end{array} \quad \ldots\ldots\ldots \quad (209)$$

The mutual coupling product for the two element group is :-

$$\boxed{A_t y_m A} = \begin{array}{c} 1 \\ 3 \\ 4 \end{array}\begin{bmatrix} -1 & 0 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & y_{m1} \\ y_{m1} & 0 \end{bmatrix} \begin{array}{ccc} 1 & 3 & 4 \\ \end{array}\begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix}$$

$$= \begin{array}{c|ccc} & 1 & 3 & 4 \\ \hline 1 & 0 & y_{m1} & -y_{m1} \\ 3 & y_{m1} & -2y_{m1} & y_{m1} \\ 4 & -y_{m1} & y_{m1} & 0 \end{array} \quad \ldots\ldots\ldots \quad (210)$$



(c) (d)

Fig.16

Fig.16(c). Two mutually coupled parallel elements 3-1.

$$\boxed{A} = \begin{array}{c|cc} & 1 & 3 \\ \hline 3\text{-}1 & -1 & 1 \\ 3\text{-}1 & -1 & 1 \end{array} \qquad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (211)$$

The group mutual coupling product is :-

$$\boxed{A_t y_m A} = \begin{array}{c} 1 \\ 3 \end{array} \begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & y_{m1} \\ y_{m1} & 0 \end{bmatrix} \begin{array}{c} \begin{array}{cc} 1 & 3 \end{array} \\ \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} \end{array} = \begin{array}{c|cc} & 1 & 3 \\ \hline 1 & 2y_{m1} & -2y_{m1} \\ 3 & -2y_{m1} & 2y_{m1} \end{array} \quad . \quad (212)$$

Fig.16(d). Two mutually coupled generator elements 1-0 and 2-0.

$$\boxed{A} = \begin{array}{c|cc} & 1 & 2 \\ \hline 1\text{-}0 & 1 & 0 \\ 2\text{-}0 & 0 & 1 \end{array} \qquad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (213)$$

The mutual coupling product for the two generator elements is :-

$$\boxed{A_t y_m A} = \begin{array}{c} 1 \\ 2 \end{array} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & y_{m1} \\ y_{m1} & 0 \end{bmatrix} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{array} = \begin{array}{c|cc} & 1 & 2 \\ \hline 1 & 0 & y_{m1} \\ 2 & y_{m1} & 0 \end{array} \quad . \quad . \quad . \quad (214)$$

From the above examples it is seen that besides being symmetrical, the triple matrix product of mutual admittances has rows and columns which sum to zero except when one or more of the group elements is a generator, e.g. eqn.(214), but unless two or more elements have nodes, other than the reference node, in common, eqns.(210) and (212), the diagonal terms are zero. As is well known, the row and column sums of the bus admittance matrix formed from element self, or equivalent, admittances using the product $\boxed{A_t y_s A}$ are zero except when generator elements are included, but the diagonal term is not zero, e.g. eqn.(203) omitting row and column g-f, - this rule has been used for deriving the self admittances, eqn.(161), from eqn.(160). Thus the form of the bus admittance matrix is unchanged when there is mutual coupling between network elements, but some sparsity is lost because mutual admittances appear in terms which otherwise would be zero, e.g. in eqn.(210) a value appears in terms 4-1, 1-4 due to mutual coupling between elements 4-3 and 3-1, similarly mutual coupling between elements 2-0 and 1-0 causes terms 1-2 and 2-1 in eqn.(214) to have a value other than zero.

B.4. EXAMPLE OF BUS ADMITTANCE MATRIX CALCULATION.

Consider the network shown in Fig.17 which is the network given in Fig.10 repeated and for which the bus impedance matrix is given by eqn.(136).

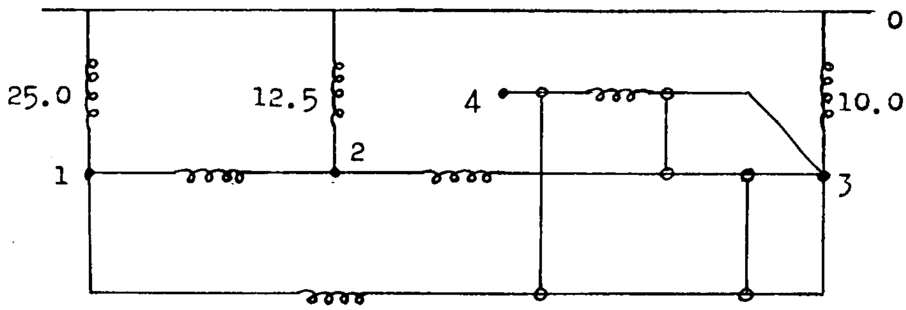


Fig.17

The per unit self and mutual impedances listed in Tables 2 and 3 are are converted to admittances.

TABLE 4. Element self and equivalent self admittances.

Element Number	Node Connections		Admittance
1	0	1	25.0
2	0	2	12.5
3	1	2	12.5
4	0	3	10.0
5	1	3	3.6368
6	3	4	5.1619
7	2	3	2.8109

The admittances for elements 1 to 4 are obtained by inversion of the impedances listed in Table 2 while the equivalent admittances for elements 5, 6 and 7 are obtained from the mutual admittance matrix for the group given in eqn.(142). This equation also gives the group mutual admittance matrix :-

$$\boxed{y_m} = \begin{array}{c|c|c|c|} & 4\text{-}3 & 3\text{-}1 & 3\text{-}2 \\ \hline 4\text{-}3 & 0 & -0.0469 & 0.6570 \\ \hline 3\text{-}1 & -0.0469 & 0 & -0.9151 \\ \hline 3\text{-}2 & 0.6570 & -0.9151 & 0 \\ \hline \end{array} \quad . \quad . \quad . \quad (215)$$

The formation of the network bus admittance matrix starts by taking the element admittances one at a time from Table 4 in the listed order. The first element is a generator connected to node 1, so the admittance value adds into the diagonal term only :-

$$\boxed{Y} = \begin{array}{c|c} & 1 \\ 1 & \boxed{25.0} \end{array} \qquad (216)$$

The next element 2-0 is also a generator :-

$$\boxed{Y} = \begin{array}{c|cc} & 1 & 2 \\ 1 & 25.0 & 0 \\ 2 & 0 & 12.5 \end{array} \qquad (217)$$

As the third listed element connects nodes 2 and 1, its admittance is added to the two diagonal terms and subtracted from the two off-diagonal terms :-

$$\boxed{Y} = \begin{array}{c|cc} & 1 & 2 \\ 1 & 37.5 & -12.5 \\ 2 & -12.5 & 25.0 \end{array} \qquad (218)$$

Element 3-0 is another generator :-

$$\boxed{Y} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ 1 & 37.5 & -12.5 & 0 \\ 2 & -12.5 & 25.0 & 0 \\ 3 & 0 & 0 & 10.0 \end{array} \qquad (219)$$

The bus admittance matrixes for partial networks of Fig.17 given in eqns.(216) to (219) are respectively inverses of the bus impedance matrixes in eqns.(126),(127), (129) and (130).

The equivalent self admittances of the three mutually coupled elements 3-1, 4-3 and 3-2 are added to $\boxed{Y}$ in eqn.(219) :-

$$\boxed{Y'} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ 1 & 41.1368 & -12.5 & -3.6368 & 0 \\ 2 & -12.5 & 27.8109 & -2.8109 & 0 \\ 3 & -3.6368 & -2.8109 & 21.6096 & -5.1619 \\ 4 & 0 & 0 & -5.1619 & 5.1619 \end{array} \qquad (220)$$

The bus incidence matrix for the group of mutually coupled elements is :-

$$\boxed{A} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ 4\text{-}3 & 0 & 0 & -1 & 1 \\ 3\text{-}1 & -1 & 0 & 1 & 0 \\ 3\text{-}2 & 0 & -1 & 1 & 0 \end{array} \qquad (221)$$

From this and eqn.(215) the triple matrix product is derived :-

$\boxed{A_t y_m A}$ = . . . (222)

	1	2	3	4
1	0	-0.9151	0.8682	0.0469
2	-0.9151	0	1.5721	-0.6570
3	0.8682	1.5721	-3.0504	0.6101
4	0.0469	-0.6570	0.6101	0

which is added to $\boxed{Y'}$, eqn.(220), giving the bus admittance matrix of the network shown in Fig.17 :-

$\boxed{Y}$ = . . . (223)

	1	2	3	4
1	41.1368	-13.4151	-2.7686	0.0469
2	-13.4151	27.8109	-1.2388	-0.6570
3	-2.7686	-1.2388	18.5592	-4.5518
4	0.0469	-0.6570	-4.5518	5.1619

$\boxed{Y}$ is the inverse of the bus impedance matrix in eqn.(136). Comparison of eqn.(220), which includes all element self admittances, with eqn.(223) shows that sparsity of the bus admittance matrix for the network, Fig.17, is lost when the mutual couplings are included. In general, for power systems the bus admittance matrix of the zero sequence network is less sparse than that for the corresponding positive or negative sequence network because mutual coupling between network elements is included in the zero sequence but not in the positive or negative sequence network.

B.5. DERIVATION OF FACTORED INVERSE.

The factored inverse of the bus admittance matrix in eqn.(223) is now calculated by taking each row and column in turn, dividing the row terms by the diagonal, then eliminating the column terms below the diagonal by Gaussian elimination and finally replacing the diagonal term by its reciprocal (Ref. 46). Starting with row and column 1 :-

	1	2	3	4
1	0.0243	-0.3261	-0.0673	0.0011
2	0	23.4361	-2.1417	-0.6417
3	0	-2.1417	18.3729	-4.5486
4	0	-0.6417	-4.5486	5.1618

followed by row and column 2 :-

	1	2	3	4
1	0.0243	-0.3261	-0.0673	0.0011
2	0	0.0427	-0.0914	-0.0274
3	0	0	18.1772	-4.6072
4	0	0	-4.6072	5.1442

then row and column 3 and finally replacing the diagonal term in the fourth row by its reciprocal gives the factored inverse :-

$\boxed{Z_{factor}}$ = . . . (224)

	1	2	3	4
1	0.0243	-0.3261	-0.0673	0.0011
2	0	0.0427	-0.0914	-0.0274
3	0	0	0.0550	-0.2535
4	0	0	0	0.2515

As the bus admittance matrix for power system networks is symmetrical, the upper triangular part only is required for calculation of the factored inverse thus reducing storage requirements for the matrix and inverse. If the factored inverse for a power system network is calculated by operating on rows and columns of the bus admittance matrix in a preferred order - commencing with columns having one term below the diagonal, then those with two, three, etc. until all columns are processed - the resulting inverse is almost as sparse as the original matrix, thereby minimising the number of arithmetic operations during computation and space required for storing the inverse.

The bus impedance matrix can be calculated one row at a time from the factors in eqn.(224), e.g. the terms of row 3 are :-

$$Z_{34} = 0.2535\times0.2515 = 0.0638$$
$$Z_{33} = 0.0550+0.2535\times0.0638 = 0.0712$$
$$Z_{32} = 0.0914\times0.0712+0.0274\times0.0638 = 0.0083$$
$$Z_{31} = 0.3261\times0.0083+0.0673\times0.0712-0.0011\times0.0638 = 0.0074$$
. . . (225)

As the terms for each row are calculated progressively commencing with that in the last column, the calculation may therefore be stopped when all the required terms are derived. Comparison with eqn.(136) shows that the terms calculated in eqn.(225) are the same as the row three terms of the bus impedance matrix. Hence the fault powers, eqn.(111), and current distribution factors, eqns.(113), (116) and (117), for the network can now be derived using element self admittances and group admittance matrixes which have been used in calculating the bus admittance matrix.

B.6. DIGITAL COMPUTER SHORT CIRCUIT PROGRAMME.

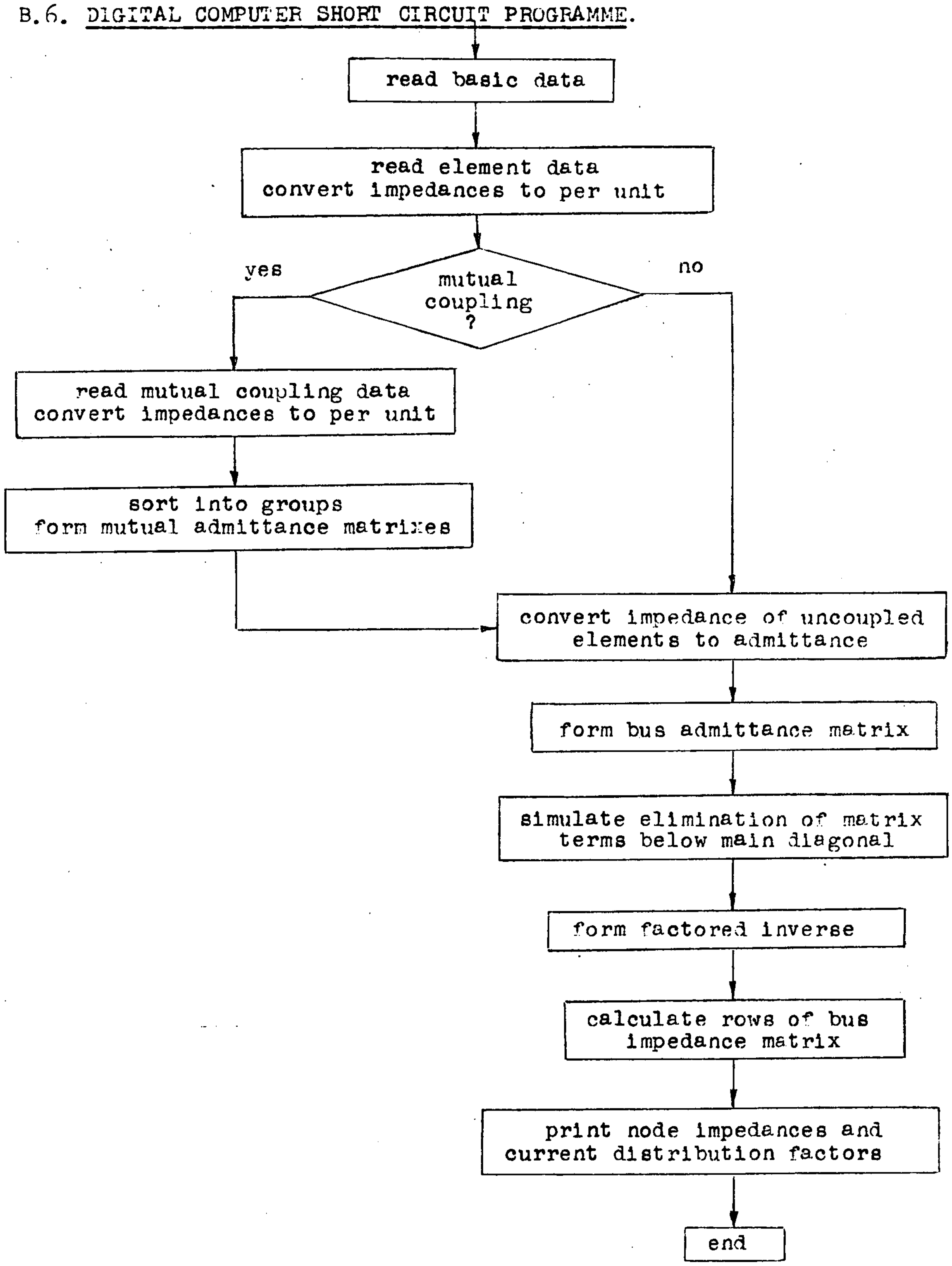


Fig.18

The flow chart of the digital computer programme HEI 13 for calculating power system short circuit problems by the bus admittance and factored inverse method is shown in Fig.18. The data for this programme while basically the same as for the bus impedance matrix programme HEI 2 allows for more flexibility in node numbering and network elements are given specific numbers for use in all studies.

As the element self and mutual impedance data is read from Tables 2 and 3, the impedances are converted to per unit if in ohms before storing with the element and node numbers. Then the mutual impedances are sorted into groups setting up as each group is formed the corresponding self and mutual impedance matrix $\boxed{z_{qp}}$ which is inverted to give the admittance matrix $\boxed{y_{qp}}$ and the upper triangular part of this is retained in storage. The matrix $\boxed{z_{qp}}$ is inverted by first calculating its factored inverse, then one row at a time of $\boxed{y_{qp}}$ using the same programme procedures as are used for deriving the bus impedance matrix from the network admittance matrix. In this programme there is no restriction on mutual couplings as any element, including a generator, may be mutually coupled to any other element.

The next step in the programme is to replace the per unit self impedances of elements without mutual coupling by their corresponding self admittance values and commence calculating the bus admittance matrix. As each element is processed, the appropriate admittance value is added to the diagonal and off-diagonal lists and, at the same time, recording row and column numbers of off-diagonal terms. Then the mutually coupled groups of elements are processed by :-

(a) adding "equivalent" self admittances to the diagonal and off-diagonal lists as for uncoupled elements; and

(b) forming the group bus incidence matrix $\boxed{A}$, calculating the triple matrix product $\boxed{A_t y_{qp} A}$ and adding the result to the diagonal and off-diagonal lists keeping a record of the number and location of the terms in the bus admittance matrix.

On completion of the bus admittance matrix calculation, a simulated elimination is carried out to determine the order in which matrix rows and columns are processed and the storage requirements for the real and imaginary parts of the factored inverse. The rows are listed in order so that at each step of the elimination the row operated on is the one with the fewest number of non-zero terms (Ref. 46)- as the original matrix is symmetrical, this corresponds to the column with the fewest number of non-zero terms below the main diagonal. The process starts by searching for a row with one off-diagonal term, simulating elimination of the corresponding column term and continuing with the next row having one off-diagonal term. When no row with one off-diagonal term remains, the first row with two off-diagonal terms is selected, elimination of the column terms simulated and the process continued until all rows with two off-diagonal terms are used. Next those with three, then those with four, etc. off-diagonal terms are processed until all rows are used. During the simulated elimination new non-zero terms may be formed; the location of these is recorded and their effect, if any, on the elimination order taken into account. The last row processed has one term only, namely the diagonal term and the second last has one off-diagonal term. Resulting from the simulation lists of the node processing order (essentially a node renumbering) and of the non-zero terms in each row of the factored inverse are formed.

The factored inverse of the admittance matrix is then calculated followed by computation of the network bus impedance matrix terms row by row. In the determination of the row terms, only non-zero terms of the factored inverse are used thus reducing the number of arithmetic operations in the calculation.

Finally, from the terms of each row of the impedance matrix, the per unit system impedance to the node is derived and current distribution factors for network elements when unit current is flowing into the node are calculated and printed. In the computer programme, element admittances and mutual admittance matrixes used in deriving the bus admittance matrix are used in calculating current distribution factors applying eqns.(113) and (116).

B.7. EFFECTIVENESS OF SPARSITY TECHNIQUE - EXAMPLES.

The effect on computer storage requirements of using the bus admittance matrix - factored inverse instead of the bus impedance matrix method for power system fault studies is illustrated by the figures in Table 5 which lists factored inverse and matrix sizes for typical power system networks with and without mutual coupling. The numbers of nodes, elements and mutual couplings are from the network input data, while the numbers of coupled elements, groups, terms in the bus admittance matrix and factored inverse are found by the com-

TABLE 5

Network	1	2	3	4	5	6	7
Number of							
nodes	15	40	40	23	67	89	99
elements	21	61	61	34	117	143	177
mutual couplings	0	0	29	20	55	129	139
mutually coupled elements	0	0	27	15	35	75	89
mutually coupled groups	0	0	6	2	7	16	22
terms in upper triangle of bus admittance matrix	30	86	149	105	221	362	427
terms in factored inverse	30	92	152	105	241	370	436
terms in upper triangle of bus impedance matrix	120	820	820	276	2278	4005	4950

puter during the short circuit calculation. The numbers of terms in the bus impedance matrix for each network are for a full matrix. The figures listed for the factored inverse, bus admittance and impedance matrixes apply to single matrixes; for power systems having complex element and mutual impedances the computer storage requirements for these items would be double the figures shown.

From Table 5 the following conclusions can be drawn for power system networks :-

(a) the sparsity of the admittance matrix is modified by mutual coupling between network elements, e.g. the number of terms in the upper triangle of the bus admittance matrix for network 2 increases from 86 to 149 when there are 29 mutual couplings (network 3). However the matrix remains sparse withover 80% of its terms zero;

(b) over 90% of the terms in the bus admittance matrixes for large networks are zero, e.g. network 7 with 99 nodes and mutual coupling between 89 of the 177 elements - the number of non-zero terms is approximately proportional to the number of nodes whereas the total number of terms is proportional to the number of nodes squared; and

(c) generally the method of forming the factored inverse is effective in preserving sparsity, in most cases the number of terms in the factored inverse is less than 3% more than in the admittance matrix.

C. ORDERED GAUSSIAN ELIMINATION FOR POWER SYSTEM LOAD FLOW.

C.1. INTRODUCTION.

For power system fault studies by digital computer the derivation of the bus impedance and admittance matrixes for the positive, negative and zero sequence networks has been discussed, and it has been shown that the required fault powers and current distribution factors can be calculated directly from the bus impedance matrix. In contrast, for power system load flow studies the network nodal equations cannot be solved directly because at most nodes the known information is power, a product of voltage and current, and hence the equations are solved by an iterative process.

Although the bus impedance matrix form of the network equations :-

$$\boxed{V} = \boxed{Z}\,\boxed{I} \quad . . . . (226)$$

has been used for load flow studies (Ref. 43), the network nodal equations :-

$$\boxed{I} = \boxed{Y}\,\boxed{V} \quad . . . . (227)$$

involving the bus admittance matrix are generally used (Ref. 36, 38) and have the advantages that the admittance matrix is sparse and easier to derive than the impedance matrix.

The three types of network busbars or nodes in load flow studies are :-

(a) the slack node where voltage is specified in magnitude and phase angle. There is one node, usually a generator, of this type in power system networks as transmission line losses are unknown initially and therefore the total generation cannot be specified;

(b) generator or voltage regulated nodes where real power and voltage magnitude are specified. Upper and lower limits of reactive power generation are also specified; when either limit is exceeded the node concerned reverts to type (c); and

(c) load or unregulated nodes at which real and reactive powers are specified. Generally most nodes are of this type - for a generator it is often more convient to specify real and reactive power than use a type (b) node.

The iterative procedure chosen for solving the load flow problem was the Newton-Raphson or Ward-Hale method, described in Ref. 36 using rectangular co-ordinates for the components of $\boxed{V}$ and $\boxed{I}$. Initially a voltage of 1+j0 per unit is assumed at all nodes and eqn. (227) solved at node "k" for the nodal current I_k; a correction factor is then calculated for the assumed voltage V_k such that the calculated current and corrected voltage give the specified power S_k' - at a generator node the specified voltage magnitude and real power are used to obtain a voltage correction. When the voltages have been modified at all nodes except the slack node, thus completing one iteration, the process is repeated until the voltage correction at all nodes is less than a specified value or, until the maximum number of iterations is reached.

It has been found in practice that the convergence process is slow but may, in many cases, be improved by acceleration, i.e. over correcting the nodal voltages during the first few iterations. However, some power system studies still required a large number of iterations before a satisfactory result was achieved and in a few cases the calculation stopped at the specified maximum number of iterations without converging. Some factors that adversely affect the rate of convergence are an open type network configuration having loads supplied from radial feeders or a slack node situated remote from the centre of the network.

From a study of the large amount of literature available (1965) on the load flow problem, it appeared that adoption of the elimination method proposed by Van Ness and Griffen (Ref. 44) would lead to solutions in fewer iterations and might also provide solutions to some of the problems which were not converging by the Ward-Hale method. On testing the elimination method by the simple expedient of replacing, in the existing load flow programme, the Ward-Hale iteration procedures by corresponding elimination procedures, it was found that an accurate solution to most problems is obtained in 3 or 4 iterations, including ones that required several hundred iterations by the Ward-Hale method. However a major disadvantage of the elimination method is that for storing the voltage correction equations a matrix approximately twice the size of the bus admittance matrix, i.e. four times as many terms, is required. Hence, for the success of the elimination, or Newton's method as it is now known it is essential to retain sparsity of the matrix during solution of the correction equations thus minimising computer storage requirements and computation time. The procedure devised is based on the analogy between network reduction and elimination of columns in the matrix storing the terms of the voltage correction equations.

C.2. DERIVATION OF VOLTAGE CORRECTION EQUATIONS.

The network bus admittance matrix $\boxed{Y}$, which is formed from the transmission line per unit admittances and susceptances of the nominal pi representation, may be split into real and imaginary parts :-

$$\boxed{Y} = \boxed{G} + j\boxed{B} \quad \ldots\ldots\ldots \quad (228)$$

where the term in row k, column m is :-

$$Y_{km} = G_{km} + jB_{km} \quad \ldots\ldots\ldots \quad (229)$$

From eqn.(227), if admittances and voltages are in per unit, the per unit current is :-

$$I_k = \sum_m Y_{km}V_m \quad \ldots\ldots\ldots \quad (230)$$

where the summation is from m=1 to m=n, n being the number of nodes in the network. The components of the nodal current I_k and nodal voltage V_k at node "k" are :-

$$I_k = a_k + jb_k \quad \ldots\ldots\ldots \quad (231)$$

and

$$V_k = e_k + jf_k \quad \ldots\ldots\ldots \quad (232)$$

The calculated power at node "k" is :-

$$S_k = P_k + jQ_k$$

$$= V_k I_k^* = (e_k + jf_k)(a_k - jb_k) \quad \ldots\ldots \quad (233)$$

If the correct nodal voltages for the problem differ from those assumed by $\boxed{\Delta V}$ and the correct nodal currents differ by $\boxed{\Delta I}$ from those calculated using eqn.(227), then the equation for the solution to the problem is :-

$$\boxed{I+\Delta I} = \boxed{Y}\boxed{V+\Delta V}$$

or

$$\boxed{I} + \boxed{\Delta I} = \boxed{Y}\boxed{V} + \boxed{Y}\boxed{\Delta V} \quad \ldots\ldots\ldots \quad (234)$$

Subtracting eqn.(227) gives :-

$$\boxed{\Delta I} = \boxed{Y}\boxed{\Delta V} \quad \ldots\ldots\ldots\ldots \quad (235)$$

for the equation relating the current and voltage corrections. From this equation the current correction at node "k" is :-

$$\Delta I_k = \sum_m Y_{km}\Delta V_m \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (236)$$

and the components of the voltage and current corrections at node "k" are :-

$$\Delta V_k = \Delta e_k + j\Delta f_k \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (237)$$

$$\Delta I_k = \Delta a_k + j\Delta b_k \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (238)$$

The specified power at node "k" is :-

$$S_k' = P_k' + jQ_k' = (V_k+\Delta V_k)(I_k^*+\Delta I_k^*) \quad . \quad . \quad . \quad (239)$$

S_k' may be expressed in terms of the calculated power S_k and a correction ΔS_k :-

$$S_k' = S_k+\Delta S_k \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (240)$$

Hence, from eqns.(233), (239) and (240) the power correction at node "k" is :-

$$\Delta S_k = S_k'-S_k$$

$$= \Delta V_k I_k^* + (V_k+\Delta V_k)\Delta I_k^* \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (241)$$

ΔS_k can also be written in terms of the real and reactive components of power :-

$$\Delta S_k = P_k'+jQ_k'-(P_k+jQ_k)$$

$$= \Delta P_k+j\Delta Q_k \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (242)$$

where $\Delta P_k = P_k'-P_k \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (243)$

and $\Delta Q_k = Q_k'-Q_k \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (244)$

Substituting for ΔI_k^* from eqn.(236) in eqn.(241) :-

$$\Delta S_k = \Delta V_k I_k^* + (V_k+\Delta V_k)\sum_m Y_{km}^*\Delta V_m^* \quad . \quad . \quad . \quad . \quad . \quad (245)$$

In this equation, Y_{km}^* is known from the bus admittance matrix, the value of V_k is assumed, I_k^* is calculated from eqn.(227) and at load nodes ΔS_k is derived from the difference in specified power S_k' and power S_k calculated from eqn.(233), but at generator nodes only the real component of ΔS_k is known. However at a generator node "k" an equation connecting the magnitude of the assumed voltage with that specified can be written down :-

$$|V_k'|^2-|V_k|^2 = (V_k+\Delta V_k)(V_k^*+\Delta V_k^*)-V_kV_k^*$$

$$= V_k\Delta V_k^*+\Delta V_k(V_k^*+\Delta V_k^*) \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (246)$$

Hence by equating real and imaginary parts on each side of eqn.(245) for load nodes and equating the real parts of eqn.(245) and using eqn.(246) for generator nodes two equations for the unknown voltage corrections ΔV_k can be derived at each of the (n-1) load and generator nodes, i.e. a set of simultaneous second order equations in the components of $\boxed{\Delta V}$ can be written down. As power system voltages normally do not differ from nominal by more than a few per cent, then if all voltages are assumed initially to be 1 per unit the corrections $\boxed{\Delta V}$ will be small and a reasonable approximation to their values obtained by solving the simultaneous linear equations that result when squares and products in eqns.(245) and (246) are neglected. The digital computer can be programmed to solve these linear equations for the voltage corrections which are then applied to the assumed nodal voltages and the process repeated until the voltage corrections at all nodes are negligible.

Neglecting second order terms in ΔV eqns.(245) and (246) are:-

$$\Delta S_k = \Delta V_k I_k^* + V_k \sum_m Y_{km}^* \Delta V_m^* \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (247)$$

and
$$|V_k'|^2 - |V_k|^2 = V_k \Delta V_k^* + \Delta V_k V_k^* \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (248)$$

Substituting for Y_{km}, I_k, V_k, ΔV_k, ΔV_m and ΔS_k from eqns.(229), (231), (232), (237) and (242) in eqn.(247) :-

$$\Delta P_k + j\Delta Q_k = (\Delta e_k + j\Delta f_k)(a_k - jb_k) + (e_k + jf_k)\sum_m (G_{km} - jB_{km})(\Delta e_m - j\Delta f_m) \quad . \quad (249)$$

which on equating real and imaginary parts gives the two equations :-

$$\Delta P_k = a_k \Delta e_k + b_k \Delta f_k + \sum_m (e_k G_{km} + f_k B_{km})\Delta e_m + \sum_m (-e_k B_{km} + f_k G_{km})\Delta f_m \quad . \quad . \quad (250)$$

$$\Delta Q_k = -b_k \Delta e_k + a_k \Delta f_k + \sum_m (-e_k B_{km} + f_k G_{km})\Delta e_m - \sum_m (e_k G_{km} + f_k B_{km})\Delta f_m \quad . \quad (251)$$

Substituting for V_k and ΔV_k from eqns.(232) and (237) in eqn.(248) :-

$$|V_k'|^2 - |V_k|^2 = (e_k + jf_k)(\Delta e_k - j\Delta f_k) + (\Delta e_k + j\Delta f_k)(e_k - jf_k)$$
$$= 2e_k \Delta e_k + 2f_k \Delta f_k \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (252)$$

For a load node eqns.(250) and (251) hold and at a generator node eqns.(250)and (252) hold.

Eqns.(250) - (252) may be written as one matrix equation :-

$$\begin{bmatrix} U & W \\ H & T \end{bmatrix} \begin{bmatrix} \Delta e \\ \Delta f \end{bmatrix} = \begin{bmatrix} \Delta Q \\ \Delta P \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (253)$$

for the (n-1) voltage corrections, where $\boxed{U}$, $\boxed{W}$, $\boxed{H}$ and $\boxed{T}$ are square matrixes of size (n-1) and $\boxed{\Delta e}$, $\boxed{\Delta f}$, $\boxed{\Delta Q}$ and $\boxed{\Delta P}$ are column matrixes with (n-1) rows. For simplicity eqn.(253) will be written :-

$$\boxed{M}\,\boxed{\Delta V} = \boxed{\Delta S} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (254)$$

where $\boxed{M}$ is a square matrix of size 2(n-1) and $\boxed{\Delta V}$, $\boxed{\Delta S}$ represent the voltage and power components as in eqn.(253).

For load nodes, the terms $\boxed{\Delta Q}$ and $\boxed{\Delta P}$ in eqn.(253) are given by eqns.(243) and (244), while for generator nodes $\boxed{\Delta P}$ is given by eqn.(243) and from eqn.(248) the term $\boxed{\Delta Q}$ is :-

$$\Delta Q_k = |V_k'|^2 - |V_k|^2 \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (255)$$

From eqns.(252) and (251) the diagonal terms of the sub-matrixes $\boxed{U}$ and $\boxed{W}$ are for a generator node :-

$$U_{kk} = 2e_k \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (256)$$

$$W_{kk} = 2f_k \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (257)$$

and for a load node :-

$$U_{kk} = -b_k - e_k B_{kk} + f_k G_{kk} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (258)$$

$$W_{kk} = a_k - e_k G_{kk} - f_k B_{kk} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (259)$$

The diagonal terms of sub-matrixes $\boxed{H}$ and $\boxed{T}$ for load and generator nodes are from eqn.(250) :-

$$H_{kk} = a_k + e_k G_{kk} + f_k B_{kk} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (260)$$

$$T_{kk} = b_k - e_k B_{kk} + f_k G_{kk} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (261)$$

The off-diagonal terms of the four sub-matrixes are from eqn.(252) for a generator node :-

$$U_{km} = W_{km} = 0 \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (262)$$

from eqn.(251) for a load node :-

$$U_{km} = -e_k B_{km} + f_k G_{km} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (263)$$

$$W_{km} = -e_k G_{km} - f_k B_{km} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (264)$$

and from eqn.(250) for a load and generator node :-

$$H_{km} = e_k G_{km} + f_k B_{km} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (265)$$

$$T_{km} = -e_k B_{km} + f_k G_{km} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (266)$$

From eqns.(256) - (266) it follows, that in general sub-matrixes $\boxed{U}$, $\boxed{W}$, $\boxed{H}$ and $\boxed{T}$ have the same degree of sparsity as the network bus admittance matrix when the row and column corresponding to the slack node are omitted, i.e. for a non-zero term in the admittance matrix there is a corresponding non-zero term in each of the sub-matrixes except for rows of $\boxed{U}$ and $\boxed{W}$ when eqn.(262) applies and all off-diagonal terms are zero. However, it is convient to retain all terms in the sub-matrixes which correspond to non-zero terms in the network bus admittance matrix thus keeping the symmetrical form of the four sub-matrixes and, because there are only a few generator nodes the number of zero terms retained in $\boxed{U}$ and $\boxed{W}$ is small. By symmetry from eqns.(263) - (266) the m-k off-diagonal terms are :-

$$U_{mk} = -e_m B_{mk} + f_m G_{mk} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (267)$$

$$W_{mk} = -e_m G_{mk} - f_m B_{mk} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (268)$$

$$H_{mk} = e_m G_{mk} + f_m B_{mk} \quad . . . . (269)$$

$$T_{mk} = -e_m B_{mk} + f_m G_{mk} \quad . . . . (270)$$

Comparing with eqns. (263) - (266) it is seen that because the voltage components e_k and e_m, f_k and f_m are equal except for the first iteration when all voltages are 1+j0, corresponding off-diagonal terms U_{km} and U_{mk}, etc., are not equal although their values will not be very different (the admittance matrix terms B_{km} and B_{mk}, G_{km} and G_{mk} are equal). Hence the sub-matrixes $\boxed{U}$, $\boxed{W}$, $\boxed{H}$ and $\boxed{T}$ are not symmetrical in value even when all nodes, other than the slack node, are load nodes; therefore all terms of the matrixes must be stored in the computer calculation.

C.3. GAUSSIAN ELIMINATION PROCEDURE.

As outlined below, eqn. (253) is solved by Gaussian elimination using the main diagonal terms as pivots. Writing down the terms of eqn. (254) :-

$$\begin{bmatrix} m_{11} & m_{12} & m_{13} & \cdot \\ m_{21} & m_{22} & m_{23} & \cdot \\ m_{31} & m_{32} & m_{33} & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix} \begin{bmatrix} \Delta V_1 \\ \Delta V_2 \\ \Delta V_3 \\ \cdot \end{bmatrix} = \begin{bmatrix} \Delta S_1 \\ \Delta S_2 \\ \Delta S_3 \\ \cdot \end{bmatrix} \quad . . . . (271)$$

Starting with the first row and column divide the terms on both sides of eqn. (271) in row 1 by the diagonal term m_{11} :-

$$\begin{bmatrix} 1 & m'_{12} & m'_{13} & \cdot \\ m_{21} & m_{22} & m_{23} & \cdot \\ m_{31} & m_{32} & m_{33} & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix} \begin{bmatrix} \Delta V_1 \\ \Delta V_2 \\ \Delta V_3 \\ \cdot \end{bmatrix} = \begin{bmatrix} \Delta S'_1 \\ \Delta S_2 \\ \Delta S_3 \\ \cdot \end{bmatrix} \quad . . . . (272)$$

where $m'_{12} = m_{12}/m_{11} \qquad m'_{13} = m_{13}/m_{11} \qquad \Delta S'_1 = \Delta S_1/m_{11} \quad . (273)$

The off-diagonal terms in column 1 are now eliminated giving :-

$$\begin{bmatrix} 1 & m'_{12} & m'_{13} & \cdot \\ 0 & m'_{22} & m'_{23} & \cdot \\ 0 & m'_{32} & m'_{33} & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix} \begin{bmatrix} \Delta V_1 \\ \Delta V_2 \\ \Delta V_3 \\ \cdot \end{bmatrix} = \begin{bmatrix} \Delta S'_1 \\ \Delta S'_2 \\ \Delta S'_3 \\ \cdot \end{bmatrix} \quad . . . . (274)$$

where $m'_{22} = m_{22} - m_{21}m'_{12} \qquad m'_{23} = m_{23} - m_{21}m'_{13}$

$m'_{32} = m_{32} - m_{31}m'_{12} \qquad m'_{33} = m_{33} - m_{31}m'_{13}$

$\Delta S'_2 = \Delta S_2 - m_{21}\Delta S'_1 \qquad \Delta S'_3 = \Delta S_3 - m_{31}\Delta S'_1 \quad . . . . . (275)$

This completes pivoting on the first diagonal term m_{11} so the

the second diagonal term from eqn.(274) is now used as a pivot and the off-diagonal terms in column 2 eliminated, after which the remaining diagonal terms are used in turn as pivots and all the off-diagonal terms eliminated leaving a unit matrix :-

$$\begin{bmatrix} 1 & 0 & 0 & . \\ 0 & 1 & 0 & . \\ 0 & 0 & 1 & . \\ . & . & . & . \end{bmatrix} \begin{bmatrix} \Delta V_1 \\ \Delta V_2 \\ \Delta V_3 \\ . \end{bmatrix} = \begin{bmatrix} \Delta S_1'' \\ \Delta S_2'' \\ \Delta S_3'' \\ . \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (276)$$

Therefore $\Delta V_1 = \Delta S_1''$ $\quad \Delta V_2 = \Delta S_2''$ $\quad \Delta V_3 = \Delta S_3''$ . . . (277)

and the solution of the eqn.(271) is complete.

As it is assumed initially that $e_k = 1$ and $f_k = 0$ at all nodes it follows from eqns.(256) and (257) that :-

$$U_{kk} = 2 \qquad W_{kk} = 0 \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (278)$$

for a generator node "k". Hence the diagonal terms of the sub-matrix $\boxed{W}$ cannot be used as pivots and the equations for the voltage corrections are written in the form given in eqn.(253) using the diagonal terms U_{kk} and T_{kk} as pivots.

From eqns.(258), (260), (267) and (269) with $e_k = 1$ and $f_k = 0$:-

$$U_{kk} = -b_k - B_{kk} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (279)$$

$$H_{kk} = a_k + G_{kk} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (280)$$

$$U_{mk} = -B_{mk} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (281)$$

$$H_{mk} = G_{mk} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (282)$$

and from eqn.(230) :-

$$I_k = a_k + jb_k = \sum_m Y_{km} = \sum_m (G_{km} + jB_{km}) \quad . \quad . \quad . \quad . \quad . \quad (283)$$

In power systems the sum of the terms in a row or column of the bus admittance matrix is equal to the admittance between the corresponding node and reference which, in most cases is the nominal pi susceptance of the transmission lines connected to the node. Hence in eqn.(283) $a_k = \sum_m G_{km} = 0$ and $b_k = \sum_m B_{km}$ is small compared with B_{kk}. Because in high voltage networks, the transmission line reactance is generally greater than the resistance, the terms B_{kk}, B_{mk} are greater than G_{kk}, G_{mk} respectively and also B_{kk}, being the sum of B_{mk} for node "k", is greater than or equal to B_{mk}. Hence it follows from eqns.(279) - (282) that the diagonal term U_{kk} is the greatest in absolute magnitude in its row or column of $\boxed{M}$ for a load node. If "k" is a generator node U_{kk} may not be the largest term in column "k", but in this case all other terms, as shown in eqns.(262) and (278), are zero. After the first iteration, e_k and f_k are still 1 and 0 approximately as the voltage corrections are small, hence eqns.(278) - (283) remain true to a first approximation and the above conclusions regarding the relative absolute magnitudes of row and column terms hold for subsequent iterations. Similarly it can be shown that in general, T_{kk} is the term of greatest absolute magnitude in its row and column of $\boxed{M}$ in eqn.(254). Thus the $\boxed{U}$ and $\boxed{T}$ diagonal terms should be suitable pivots for solving eqn.(253) and, in practice this has been found to hold for power system networks.

C.4. OPTIMAL ORDERING SCHEME FOR ELIMINATION.

From eqns.(256) - (266) it is seen that, as the terms of the sub-matrixes in eqn.(253) depend on voltage components e_k, f_k which vary from iteration to iteration, the equations must be set up anew for each iteration. It has also been shown that the sub-matrixes $\boxed{U}$, $\boxed{W}$, $\boxed{H}$ and $\boxed{T}$ have the same degree of sparsity as the bus admittance matrix of the network omitting the slack node. The fundamental requirement for success in using the elimination method for solving load flow problems is a preferred order of eliminating the off-diagonal column terms of matrix $\boxed{M}$ that preserves sparsity, thereby minimising computer storage and number of arithmetic operations.

As the sub-matrixes reflect the network structure or geometry, i.e. if there is no network connection between nodes "i" and "j" then corresponding terms i-j and j-i are zero in each sub-matrix, elimination of column terms in $\boxed{M}$ is analogous to network reduction: When, in the network, nodes at the end of branches are eliminated no new connections are formed; similarly eliminating a column with one off-diagonal term in the sub-matrixes of $\boxed{M}$ does not form any new terms in the matrix. Elimination of a node connected to two other network nodes introduces one new connection which may be parallel to an existing connection; similarly eliminating a column with two off-diagonal terms in the sub-matrixes creates a new pair of off-diagonal terms which may modify existing terms. Elimination of a network node connected to three other nodes forms three new connections (a star-delta transformation) some or all of which may be parallel to existing connections; similarly eliminating a column with three off-diagonal terms in the sub-matrixes of $\boxed{M}$ creates or modifies three pairs of matrix terms. In practice the number of terms formed or modified is less than that indicated because columns in which some of these terms appear have already been eliminated.

The application of this to eqn.(253) takes the form of eliminating columns of the matrix in pairs, pivoting first on the $\boxed{U}$ then on the corresponding $\boxed{T}$ diagonal terms. At each step the pair of columns with the fewest number of off-diagonal terms is selected for elimination and if more than one pair satisfies this criterion, then the first listed pair is selected.

Starting with matrix columns corresponding to nodes at the end of radial lines in the network, the one off-diagonal term in the $\boxed{U}$, $\boxed{W}$, $\boxed{H}$ and $\boxed{T}$ sub-matrixes and the diagonal terms in the $\boxed{W}$ and $\boxed{H}$ matrixes are eliminated. As no new terms are formed in this process the sparsity of $\boxed{M}$ is unaffected. When all such columns have been processed, the first column with two off-diagonal terms in the sub-matrixes is selected and elimination of its terms may form a new pair of terms in each sub-matrix or , may add to existing terms. Thus after all columns with one and two off-diagonal terms in the sub-matrixes are processed the sparsity of $\boxed{M}$ is not affected to any great extent.

When all columns with two off-diagonal terms in the sub-matrixes have been processed those with three are processed in turn, then those with four, five, etc. until all the off-diagonal terms of $\boxed{M}$ are eliminated. As processing columns with two, three, or more off-diagonal terms can form new terms in $\boxed{M}$, the record of the number of terms per column must be modified during the elimination,e.g. a column with two may become a column with three off-diagonal terms in the sub-matrixes.

The elimination procedure is detailed for the 9 node network shown diagrammatically in Fig.19 for which connections to the slack node are made in broken lines.

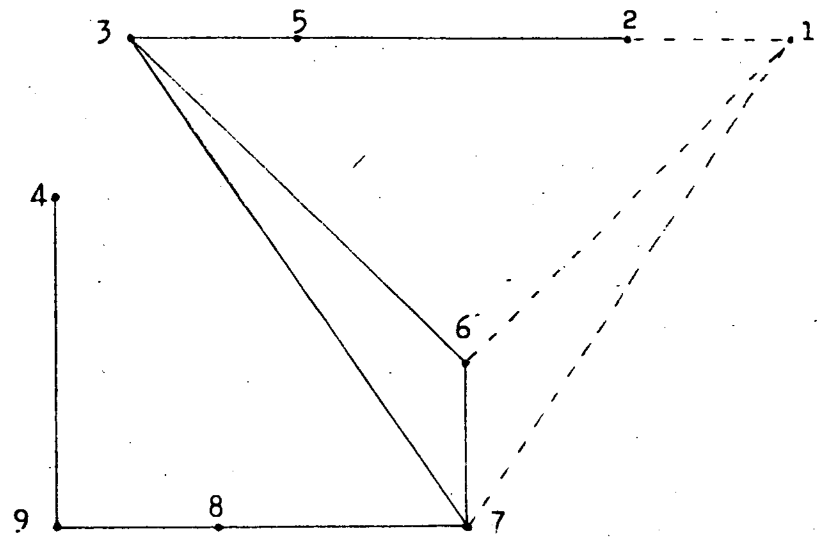


Fig.19

The sub-matrix $\boxed{U}$ for this network is indicated below :-

$$\boxed{U} = \begin{array}{c|c|c|c|c|c|c|c|c|}
 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline
2 & X & & & X & & & & \\ \hline
3 & & X & & X & X & X & & \\ \hline
4 & & & X & & & & & X \\ \hline
5 & X & X & & X & & & & \\ \hline
6 & & X & & & X & X & & \\ \hline
7 & & X & & & X & X & X & \\ \hline
8 & & & & & & X & X & X \\ \hline
9 & & & X & & & & X & X \\ \hline
\end{array} \qquad \ldots\ldots \qquad (284)$$

where "X" denotes a term having a value and a blank indicates zero. The columns with one off-diagonal term are 2 and 4. Dividing row 2 by its diagonal term and eliminating term 5-2 involves modification to terms 2-5 and 5-5, while elimination of the term 9-4 requires division of term 4-9 by diagonal term 4-4 and modification of term 9-9, after which $\boxed{U}$ has the form :-

$$\boxed{U} = \begin{array}{c|c|c|c|c|c|c|c|c|}
 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline
2 & 1 & & & X & & & & \\ \hline
3 & & X & & X & X & X & & \\ \hline
4 & & & 1 & & & & & X \\ \hline
5 & & X & & X & & & & \\ \hline
6 & & X & & & X & X & & \\ \hline
7 & & X & & & X & X & X & \\ \hline
8 & & & & & & X & X & X \\ \hline
9 & & & & & & & X & X \\ \hline
\end{array} \qquad \ldots\ldots \qquad (285)$$

The first column, 5, with two off-diagonal terms is now selected. Dividing row 5 by the diagonal term 5-5 modifies term 5-3 and eliminating terms 2-5 and 3-5 modifies term 3-3 and adds one new term 2-3 as the off-diagonal term in column 2 has already been eliminated. The matrix now has the form :-

$\boxed{U} =$ . . . . (286)

	2	3	4	5	6	7	8	9
2	1	X						
3		X			X	X		
4			1					X
5		X		1				
6		X			X	X		
7		X			X	X	X	
8						X	X	X
9							X	X

Column 3 now has four off-diagonal terms and 6 is the next column with two terms plus the diagonal. Dividing row 6 by the diagonal term 6-6 modifies terms 6-3 and 6-7 while eliminating terms 3-6 and 7-6 results in modification to terms 3-3, 3-7, 7-3 and 7-7 without forming new terms. The next column with two off-diagonal terms is 8 and after dividing row 8 terms 8-7 and 8-9 by the diagonal 8-8 eliminating term 7-8 modifies 7-7 and forms a new term 7-9, while elimination of term 9-8 results in modification of 9-9 and formation of a new term 9-7. At this stage the matrix has the form :-

$\boxed{U} =$ . . . . (287)

	2	3	4	5	6	7	8	9
2	1	X						
3		X				X		
4			1					X
5		X		1				
6		X			1	X		
7		X				X		X
8						X	1	X
9						X		X

So far 8 terms have been eliminated, 3 new terms formed and modifications made to 16 terms leaving no columns with two off-diagonal terms and only one column, 9, with 3. Dividing row 9 by its diagonal 9-9 modifies term 9-7, eliminating term 4-9 forms a new term 4-7, eliminating terms 7-9 and 8-9 modifies terms 7-7 and 8-7 leaving the matrix with the form :-

$$\boxed{U} = \begin{array}{c|cccccccc|} & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline 2 & 1 & X & & & & & & \\ 3 & & X & & & & X & & \\ 4 & & & 1 & & & X & & \\ 5 & & X & & 1 & & & & \\ 6 & & X & & & 1 & X & & \\ 7 & & X & & & & X & & \\ 8 & & & & & & X & 1 & \\ 9 & & & & & & X & & 1 \\ \hline \end{array} \quad . \; . \; . \; . \; . \; . \; . \quad (288)$$

Only two columns remain, 3 with 4 off-diagonal terms and 7 with 5 and processing column 3 involves dividing term 3-7 by the diagonal term 3-3, forming new terms 2-7 and 5-7, modifying terms 6-7 and 7-7. This leaves the matrix with one full column, 7 :-

$$\boxed{U} = \begin{array}{c|cccccccc|} & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline 2 & 1 & & & & & X & & \\ 3 & & 1 & & & & X & & \\ 4 & & & 1 & & & X & & \\ 5 & & & & 1 & & X & & \\ 6 & & & & & 1 & X & & \\ 7 & & & & & & X & & \\ 8 & & & & & & X & 1 & \\ 9 & & & & & & X & & 1 \\ \hline \end{array} \quad . \; . \; . \; . \; . \; . \; . \quad (289)$$

On eliminating the off-diagonal terms of column 7 a unit matrix is left. Hence during the elimination process 6 new terms are formed in the $\boxed{U}$ matrix and modifications to 22 terms are required, i.e. the number of terms used during the elimination is 30 compared with 24 in the original $\boxed{U}$ matrix and 64 terms in the full matrix.

Elimination in the preferred order - 2, 4, 5, 6, 8, 9, 3 and 7 - is now compared with processing the columns in numerical order - 2, 3, 4, 5, 6, 7, 8 and 9. Commencing with column 2, as above 2 terms are modified and no new terms are formed, then elimination of the 3 off-diagonal terms in column 3 creates 4 new terms 5-6, 5-7, 6-5 and 6-7 and modifies 8 terms 3-5, 3-6, 3-7, 5-5, 6-6, 6-7, 7-6 and 7-7 leaving the sub-matrix in the form :-

$$\boxed{U} = \begin{array}{c|c|c|c|c|c|c|c|c|} & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline 2 & 1 & & & X & & & & \\ \hline 3 & & 1 & & X & X & X & & \\ \hline 4 & & & X & & & & & X \\ \hline 5 & & & & X & X & X & & \\ \hline 6 & & & & X & X & X & & \\ \hline 7 & & & & X & X & X & X & \\ \hline 8 & & & & & & X & X & X \\ \hline 9 & & & X & & & & X & X \\ \hline \end{array} \qquad \ldots\ldots \qquad (290)$$

Column 4 is processed in the same way as in the preferred system, i.e. no new terms are formed and two terms 4-9 and 9-9 are modified, while elimination of the 4 off-diagonal terms in column 5 creates 2 new terms 2-6 and 2-7 and modifies 8 terms 5-6, 5-7, 3-6, 3-7, 6-6, 6-7, 7-6 and 7-7 giving the sub-matrix the form :-

$$\boxed{U} = \begin{array}{c|c|c|c|c|c|c|c|c|} & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline 2 & 1 & & & & X & X & & \\ \hline 3 & & 1 & & & X & X & & \\ \hline 4 & & & 1 & & & & & X \\ \hline 5 & & & & 1 & X & X & & \\ \hline 6 & & & & & X & X & & \\ \hline 7 & & & & & X & X & X & \\ \hline 8 & & & & & & X & X & X \\ \hline 9 & & & & & & & X & X \\ \hline \end{array} \qquad \ldots\ldots \qquad (291)$$

Eliminating the 4 off-diagonal terms in column 6 does not form any new terms, but the 5 terms 6-7, 2-7, 3-7, 5-7 and 7-7 are modified and elimination of the 5 off-diagonal terms in column 7 creates 4 new terms 2-8, 3-8, 5-8 and 6-8 and modifies the two terms 7-8 and 8-8 so that the sub-matrix now has the form :-

$$\boxed{U} = \begin{array}{c|c|c|c|c|c|c|c|c|} & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline 2 & 1 & & & & & & X & \\ \hline 3 & & 1 & & & & & X & \\ \hline 4 & & & 1 & & & & & X \\ \hline 5 & & & & 1 & & & X & \\ \hline 6 & & & & & 1 & & X & \\ \hline 7 & & & & & & 1 & X & \\ \hline 8 & & & & & & & X & X \\ \hline 9 & & & & & & & X & X \\ \hline \end{array} \qquad \ldots\ldots \qquad (292)$$

Processing column 8 to eliminate the 6 off-diagonal terms forms 5 new terms 2-9, 3-9, 5-9, 6-9 and 7-9 and modifies 2 terms 8-9 and 9-9 and leaves column 9 full. Hence by processing columns in numerical order 15 new terms are formed and 29 terms are modified during the elimination, i.e. the number of terms used is 39. Thus comparing the preferred order of processing columns with processing in numerical order, there is a significant saving in storage and in arithmetic operations - 120 terms in the $\boxed{M}$ matrix as against 156 terms and 22+6=28 as against 29+15=46 arithmetic operations on $\boxed{U}$ matrix terms by the former compared with the latter process. This saving is achieved in each iteration in the computer programme and could be considerable for a large power system network.

The preferred elimination order is not necessarily the optimum for all power system networks, but is easy to apply as it is based on a simple rule - at each stage the column pair of $\boxed{M}$ chosen for processing is the one with the least number of terms and if more than one pair satisfies this criterion then select that first listed. The procedure discussed for the preferred elimination order of the $\boxed{U}$ matrix is applied, with slight alterations, to the $\boxed{M}$ matrix, eqn.(254), by processing columns in pairs and pivoting first on the $\boxed{U}$ diagonal term, then on the corresponding $\boxed{T}$ diagonal term.

Consider the $\boxed{M}$ matrix corresponding to Fig.19 - when pivoting on U_{22} to eliminate U_{52}, H_{22} and H_{52} the terms U_{25}, W_{22} and W_{25} are divided by U_{22} and the terms U_{55}, W_{52}, W_{55}, H_{25}, T_{22}, T_{25}, H_{55}, T_{52} and T_{55} are modified; then pivoting on T_{22} to eliminate T_{52}, W_{22} and W_{52} the terms H_{25} and T_{25} are divided by T_{22} and the terms U_{25}, W_{25}, U_{55}, W_{55}, H_{55} and T_{55} are modified, i.e. no new terms are formed but 20 terms are modified. Comparing this with pivoting on U_{22}, discussed above, in the $\boxed{U}$ matrix alone when 2 terms are modified highlights the increased number of operations due to $\boxed{M}$ being twice the size of $\boxed{U}$ and emphasises the importance of the preferred elimination order. After pivoting on U_{22}, T_{22} the column pairs are processed in the order 4, 5, 6, 8, 9, 3 and 7 - note that where one new term is formed in the $\boxed{U}$ matrix 4 new terms are formed in $\boxed{M}$, e.g. pivoting on U_{55}, T_{55} forms the new terms U_{23}, W_{23}, H_{23} and T_{23}.

As in each iteration the error in the nodal voltage corrections

Δe and Δf is proportional to their squares, the procedure is quadratically convergent and for most power system problems the values of Δe and Δf are 10^{-4} or less after 3 or 4 iterations. This corresponds to differences of less than 0.005MW and 0.005MVAr between the calculated and specified real and reactive powers at all nodes, i.e. for practical purposes the problem is solved.

Summarising, a procedure for solving eqn.(253) which retains sparsity in the matrix is :-

(a) processing in numerical order the column pairs having one off-diagonal term in $\boxed{U}$ and $\boxed{T}$, these correspond to network nodes with one connection in the network; no new terms are formed in $\boxed{M}$ by their elimination;

(b) processing in numerical order the column pairs having two off-diagonal terms in $\boxed{U}$ and $\boxed{T}$. These correspond to network nodes connected to two other nodes and the elimination of each pair forms none, one or two new terms in the sub-matrixes of $\boxed{M}$ depending on network connections and columns (or nodes) already eliminated;

(c) processing in order the column pairs with three off-diagonal terms in $\boxed{U}$ and $\boxed{T}$. These correspond to star-delta transformations in the network with the possibility of forming up to six new terms in each of the sub-matrixes of $\boxed{M}$. However, in practice not more than two or three new terms are likely and existing terms will be modified, also columns in which new terms could appear have already been eliminated, e.g. eliminating the three off-diagonal terms in column 9, eqn(287), forms the one new term, 4-7, in the sub-matrixes; and

(d) processing in order the column pairs with four , then those with five, etc., off-diagonal terms in $\boxed{U}$ and $\boxed{T}$. Large numbers of new matrix terms could be formed, but in practice this is unlikely because at this stage many columns where new terms might appear have been eliminated, e.g. eliminating the four off-diagonal terms in column 3 of eqn.(288) results in only two new terms 2-7 and 5-7 in the sub-matrixes of $\boxed{M}$.

C.5. EFFECT OF OPTIMAL ORDERING ON POWER SYSTEM MATRIXES.

TABLE 6

Network	1	2	3	4	AEP14	AEP30	AEP57
Number of nodes	9	10	36	83	14	30	57
lines and regulators	14	15	43	94	20	41	78
network connections	11	12	42	94	20	41	78
terms in bus admittance matrix excluding the slack node	24	31	111	268	49	107	206
terms in $\boxed{U}$ matrix used in elimination	30	42	175	518	71	193	531
terms in a matrix of size (n-1)	64	81	1225	6724	169	841	3136

Table 6 lists computer storage required for the $\boxed{U}$ sub-matrix for typical power system networks including the three IEEE Standard Test Systems with, for comparison the numbers of terms in the bus admittance matrix and the total including zero terms. The figures for lines and regulators differs from that for network connections in some cases because there are parallel lines. The initial number of terms in the $\boxed{U}$, $\boxed{W}$, $\boxed{H}$ and $\boxed{T}$ sub-matrixes is given in the row for the bus admittance matrix excluding the slack node, while the corresponding figure allowing for terms formed during the elimination is given in the row labelled "number of terms in $\boxed{U}$ matrix used in elimination" - this latter figure being derived by the computer during solution of the problem.

Examination of Table 6 shows that the preferred elimination order is effective in maintaining sparsity, e.g. for the 83 node system more than 91% of terms in the $\boxed{M}$ matrix remain zero throughout the calculation leading to a considerable saving in storage requirements and by operating only on non-zero terms, keeping computation time down.

C.6.1. DATA PREPARATION FOR COMPUTER PROGRAMME.

In preparing for a digital computer load flow study, the

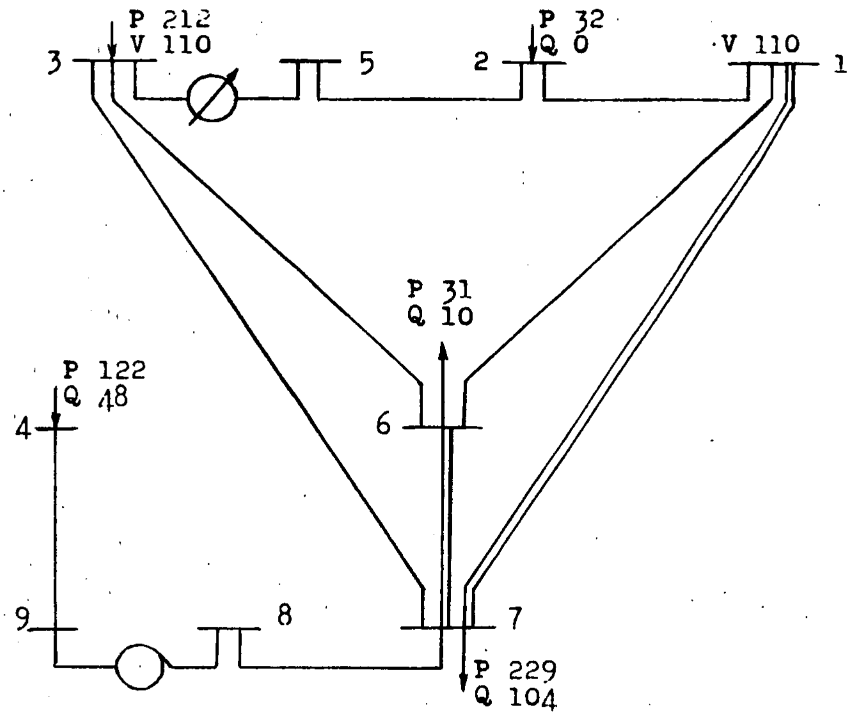


Fig. 20

power system network is coded by numbering the active nodes sequentially commencing with 1, the same as for short circuit studies except that the reference node is not referred to directly. The data for a load flow study is in three lists :-

(a) the basic data giving the numbers of nodes, transmission lines, generators, etc.;

(b) the network data detailing node connections and impedances of transmission lines, voltage regulators and transformers; and

(c) the problem data listing the generation and loading at the network nodes.

For the sample system shown in Fig.20 Tables 7, 8 and 9 list the basic, network and problem data respectively.

TABLE 7. BASIC DATA.

MVA base	100.0
Number of nodes	9
Number of transmission lines	12
Number of regulators	1
Number of generators	2
Number of loads	7
Number of shunt capacitors	0
Number of adjustable regulators	0

The slack node is included in the generator nodes and all those for which real and reactive powers are specified are included in the load nodes. Adjustable regulators are those for which upper and lower voltage limits are specified, the tap being automatically adjusted during computation to bring the voltage at the node concerned within these limits.

TABLE 8. TRANSMISSION LINE AND VOLTAGE REGULATOR DATA.

Line	Connection		Resistance	Reactance	Total Shunt Susceptance	Nominal Voltage
1	5	2	1.25	3.4	42.0	110.0
2	2	1	1.34	3.85	48.0	110.0
3	1	6	10.36	18.25	230.0	110.0
4	1	7	29.34	40.68	134.0	110.0
5	7	6	7.2	44.3	15.0	110.0
6	7	8	0.18	0.69	5.0	110.0
7	7	3	11.25	19.72	252.0	110.0
8	6	3	18.21	45.09	102.0	110.0
9	8	9	0.0	0.04	0.0	1.0
10	9	4	4.83	19.37	233.0	220.0
11	1	7	29.34	40.68	134.0	110.0
12	7	6	7.2	44.3	15.0	110.0
Regulator					Tap	
501	3	5	0.0	2.45	1.08	110.0

The nominal voltage is used to convert the resistance, reactance (which are in ohms) and susceptance (which is in micromhos) to per unit; but these may be entered on the data list in per unit and 1.0 entered under nominal voltage. For voltage regulators the tap, entered in per unit, is assumed to act at the higher numbered node, the regulator being represented in the computer programme by the equivalent circuit given in Ward and Hale (Ref.36).

TABLE 9. GENERATOR AND LOAD DATA.

Generator node	Voltage magnitude	Real power	Reactive power limits minimum	maximum	Nominal voltage
1	110.0				110.0
3	110.0	212.0	80.0	110.0	110.0

Load node	Real power	Reactive power	Nominal voltage
2	32.0	0.0	110.0
4	122.0	48.0	220.0
5	0.0	0.0	110.0
6	-31.0	-10.0	110.0
7	-229.0	-104.0	110.0
8	0.0	0.0	110.0
9	0.0	0.0	220.0

The slack node, for which only voltage magnitude is specified, is listed first in Table 9 and for generator nodes besides voltage magnitude and real power limits for reactive power generation are specified; if it is impossible to achieve reactive power generation within the limits then the node concerned is converted from a generator to a load allowing the voltage to vary. The sign convention adopted is that power flowing into a node is positive.

C.6.2. DIGITAL COMPUTER PROGRAMME.

The outline of the flow diagram for the digital computer programme HEI 8, calculation of power system load flows by the elimination method using a preferred column processing order, is given in Fig.21. After reading the sets of data and converting impedances, etc. to per unit, voltages at all nodes are set to 1+j0 per unit except at generator nodes where the real component is set to the per unit value of the specified voltage magnitude. The transmission line and regulator lists are examined, the diagonal terms of the bus admittance matrix formed and stored, line admittances stored and those of parallel lines combined thus making all terms of the bus admittance matrix available. Lists are set up giving row and column numbers of off-diagonal terms and the location of the corresponding values in the line list.

After the "formation" of the bus admittance matrix, a simulated elimination of the voltage correction equations is performed taking

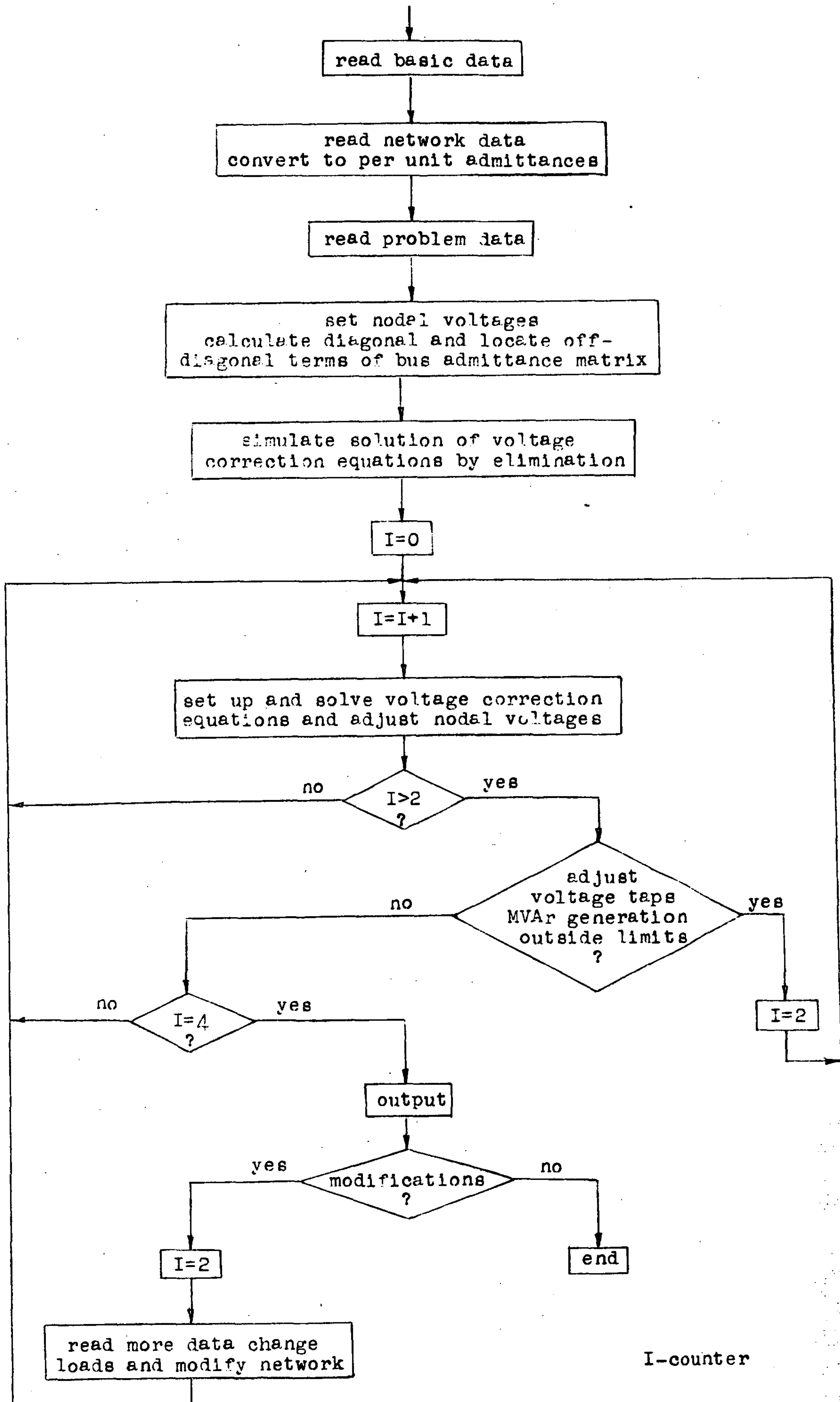

read basic data
read network data
convert to per unit admittances
read problem data
set nodal voltages
calculate diagonal and locate off-diagonal terms of bus admittance matrix
simulate solution of voltage correction equations by elimination
I=0
I=I+1
set up and solve voltage correction equations and adjust nodal voltages
no
I>2 ?
yes
adjust voltage taps MVAr generation outside limits ?
no
yes
I=2
no
I=4 ?
yes
output
yes
modifications ?
no
I=2
end
read more data change loads and modify network
I-counter

Fig.21

into consideration omission of the slack node and its connections. Firstly the nodes at the ends of radial lines, i.e. those corresponding to one off-diagonal term in the sub-matrixes of $\boxed{M}$, are listed in numerical order and corresponding storage requirements for $\boxed{M}$ recorded.

Next the effect of eliminating the first listed node with two network connections is examined and if new terms are formed in $\boxed{M}$ their location is recorded before proceeding to the remaining nodes having two network connections. When the elimination of all nodes corresponding to columns in $\boxed{U}$ with two off-diagonal terms has been examined, those with three, then those with four, etc., are processed until finally all nodes are eliminated. As a result of this simulation, a processing order for the columns of $\boxed{M}$ is established and lists formed of the storage required, and terms used, in each column of its sub-matrixes.

The voltage correction equations are set up and solved, the nodal voltages adjusted and the process repeated using the corrected voltage values. Before calculating the next set of voltage corrections, the voltage at tap changing transformers is examined and any required tap changes made and the MVAr at generator nodes is calculated and if outside the specified limits the node concerned is converted to a load. After a minimum of four iterations, or two following an adjustment to taps or conversion of a generator to a load, the output procedure is entered. The results consist of listing for every node MW and MVAr generation or load, voltage in kV and per unit and its angle relative to that of the slack node and the MVAr line charging while for transmission lines, transformers and regulators the power flows at each node, MW and MVAr losses and tap settings for regulators are listed. In addition the total generation, line charging and losses are printed. The calculation is then repeated for modifications such as changing loads or altering transmission impedances (including line switching).

The accuracy of the solution to the problem is determined by the difference between specified and calculated parameters at the network nodes and in most cases calculated real and reactive powers are within 0.005 MW and 0.005 MVAr respectively of the specified values at all nodes, which for practical purposes is an exact solution.

C.7. EXAMPLE OF RATE OF CONVERGENCE.

TABLE 10. VOLTAGE CORRECTIONS.

Node	Iteration 1	Iteration 2	Iteration 3
2	0.0395+j0.0396	-0.0041-j0.0004	-0.00003-j0.00004
3	0.0000+j0.0872	-0.0033-j0.0061	-0.00001-j0.00014
4	0.0069+j0.0723	-0.0297+j0.0060	-0.00097+j0.00030
5	0.0715+j0.0652	-0.0072-j0.0006	-0.00005-j0.00008
6	-0.0392-j0.0091	-0.0104-j0.0007	-0.00029+j0.00002
7	-0.0629-j0.0253	-0.0226-j0.0009	-0.00070+j0.00003
8	-0.0571-j0.0194	-0.0230-j0.0004	-0.00072+j0.00005
9	-0.0290+j0.0294	-0.0267+j0.0024	-0.00085+j0.00016

Table10 lists the components of the per unit voltage corrections calculated in three iterations of the power system network shown in Fig.20 using the data in Tables 7, 8 and 9. As on the fourth iteration all voltage corrections are 0.000001 or less, it follows that every nodal voltage has been corrected in three iterations. The pattern of convergence shown in Table 10, i.e. an error of the same magnitude in both components of all nodal voltages after three or four iterations, holds generally for power system networks.

CONCLUSION

By considering the combination of a network with a group of mutually coupled elements using a connection matrix and its inverse rules, suitable for digital computer programming, have been devised for forming network bus impedance and admittance matrixes from randomly ordered lists of element self and mutual impedamces.

In calculating the bus impedance matrix, an uncoupled element self impedance is added to the diagonal term of a new row and column which, for a branch element correspond to the branch node and for a loop element augment the matrix. The new row and column are derived from existing matrix terms by repetition or subtraction and augmenting terms are eliminated by matrix reduction. For a mutually coupled group element self impedances are added to the bus impedance matrix as uncoupled branch or loop elements, then the group mutual impedances are added to the appropriate matrix terms and finally any augmenting rows and columns are eliminated by matrix reduction.

A given network bus impedance matrix can be modified to allow for variations in element self and mutual impedances by adding loop element impedances which when parallelled with the existing impedances give the required new values. Power system fault parameters are calculated by arithmetic operations on sequence network bus impedance matrix terms and a desired equivalent network can be derived from the appropriate matrix terms. Because a network bus impedance matrix can be considered as the self and mutual impedance matrix of a group of coupled generator elements, the rules for adding the impedances of a group of mutually coupled elements to a bus impedance matrix can be applied to combine two bus impedance matrixes.

The method of forming a network bus admittance matrix from randomly ordered lists of element self admittances has been extended to elements with mutual coupling. For a mutually coupled group of elements, the diagonal terms of its self and mutual admittance matrix are added in the same way as uncoupled element self admittances and the result of the triple matrix product of the off-diagonal terms with the group element bus incidence matrix and its transpose is added. As well as being simpler to construct than the bus impedance matrix, the admittance matrix is sparse and from it, by an optimal ordering procedure, a sparse factored inverse is derived which is suitable for digital computer fault studies. This method has the advantage of requiring less storage than the bus impedance matrix method.

To solve power system load flow problems on the digital computer using Newton's method, an optimal order for solving the voltage correction equations has been devised which is based on the anology between network reduction and the Gaussian elimination procedure. Initially the voltage correction matrix has the same form as the bus admittance matrix and, for typical power system networks, by this ordering scheme sparsity is retained during the elimination. Most power system load flow problems are solved in four iterations by this procedure.

REFERENCES.

Books.

1. Kron,G. - Tensor Analysis of Networks. New York, Wiley, 1937.
2. Kron,G. - Diakoptics - The Piecewise Solution of Large Scale Systems. London, Macdonald, 1963.
3. George,E.E. - Matrix Handbook for Electrical Engineers. Ebasco Services Inc., U.S.A., 1958.
4. Bickley,W.G. and Thompson,R.S.H.G. - Matrixes, Their Meaning and Manipulation. London, English Universities Press, 1964.
5. Stagg,G.W. and El-Abiad,A.H. - Computer Methods in Power System Analysis. New York, McGraw-Hill, 1968.
6. Brameller,A., John,M.N. and Scott,M.R. - Practical Diakoptics for Electrical Networks. London, Chapman & Hall, 1969.

Short Circuit - General.

7. Glimm,A.F., Haberman,R., Henderson,J.M. and Kirkmayer,L.K. - Digital Calculation of Network Impedances. Trans.A.I.E.E., Pt.III, Vol.74, Dec.1955, p.1285.
8. Hale,H.W. and Ward,J.B. - Digital Computation of Driving Point and Transfer Impedances. Trans.A.I.E.E., Pt.III, Vol.76, 1957, p.476.
9. Byerly,R.T., Long,R.W. and King,C.W. - Logic for Applying Topological Methods to Electric Networks. Trans.A.I.E.E., Pt.I, Vol.77, Nov.1958, p.657.
10. Converti,V. - Application of Row-by-Row Matrix Inversion to Power System Problems. Trans.A.I.E.E., Pt.IIIA, Vol.78, June 1959, p.413.
11. Shipley,R.B. and Coleman,D. - A New Direct Matrix Inversion Method. Trans.A.I.E.E., Pt.I, Vol.78, Nov.1959, p.568.
12. Sato,N. - Digital Calculation of Network Inverse and Mesh Transformation Matrixes. Trans.A.I.E.E., Pt.III, Vol.79, Oct.1960, p.719.
13. Hale,H.W. - A Logic for Identifying the Trees of a Graph. A.I.E.E. Power Apparatus and Systems, June 1961, p.195.

Short Circuit - Nodal Iterative.

14. Coombe,L.W. and Lewis,D.G. - Digital Calculation of Short Circuit Currents in Large Complex Impedance Networks. Trans.A.I.E.E., Pt.III, Vol.75, 1956 (Feb.1957), p.1394.

15. Siegel,J.C. and Bills,G.W. - Nodal Representation of Large Complex Element Networks Including Mutual Reactances. Trans.A.I.E.E., Pt.III, Vol.77, 1958(Feb.1959), p.1226.

16. Taylor,G.E., Carter,G.K. and MacDonald,E.H. - New Digital Computer Short Circuit Programme for Relay Studies. Trans.A.I.E.E., Pt.III, Vol.79, 1960(Feb.1961), p.1257.

17. Shipley,R.B., Coleman,D. and Nason,W.L. - Short Circuit Comparison Study. A.I.E.E. Power Apparatus and Systems, Feb.1962, p.1162.

Short Circuit - Loop Matrix.

18. Lantz,M.J. - Digital Short Circuit Solution of Power System Networks Including Mutual Impedance. Trans.A.I.E.E., Pt.III, Vol.76, 1957(Feb.1958), p.1230.

19. Byerly,R.T., Long,R.W., Baldwin,C.J. and King,C.W. - Digital Calculation of Power System Networks Under Faulted Conditions. Trans. A.I.E.E., Pt.III, Vol.77, 1958(Feb.1959), p.1296.

20. Tinney,W.F. and McIntyre,C.M. - A Digital Method for Obtaining a Loop Connection Matrix. Trans.A.I.E.E., Pt.III, Vol.79, Oct.1960, p.740.

21. Rindt,L.J. and Long,R.W. - New Short Circuit Provides Many Automatic Features. I.E.E.E. Trans. on Power Apparatus and Systems, Special Supplement, 1963, p.740.

Short Circuit - Bus Impedance Matrix.

22. Toalston,A.L. - Digital Solution of Short Circuit Currents for Networks Including Mutual Impedances. Trans.A.I.E.E., Pt.IIIB, Vol.78, 1959(Feb.1960), p.1720.

23. El-Abiad,A.H. - Digital Calculation of Line-to-Ground Short Circuits by Matrix Method. Trans.A.I.E.E., Pt.III, Vol.79, June 1960, p.323.

24. Brown,H.E. and Person,C.E. - Digital Calculation of Single Phase-to-Ground Faults. Trans.A.I.E.E., Pt.III, Vol.79, Oct.1960, p.657.

25. Brown,H.E., Person,C.E., Kirkmayer,L.K. and Stagg,G.W. - Digital Calculation of 3-Phase Short Circuits by Matrix Method. Trans. A.I.E.E., Pt.III, Vol.79, 1960(Feb.1961), p.1277.

26. El-Abiad,A.H., Guidone,R. and Stagg,G.W. - Calculation of Short Circuits Using a High Speed Digital Computer. A.I.E.E. Power Apparatus and Systems, Dec.1961, p.702.

27. Storry,J.O. and Brown,H.E. - An Improved Method of Incorporating Mutual Couplings in Single Phase Short Circuit Calculations. I.E.E.E. Trans. on Power apparatus and Systems, Vol.PAS 89, Jan. 1970, p.71.

28. Tarsi,D.C.- Simultaneous Solution of Line-Out and Open-End Line-to-Ground Short circuits. I.E.E.E. Trans. on Power Apparatus and Systems, Vol.PAS 89, July/Aug.1970, p.1220.

29. Dy Liacco,T.E. and Ramarao,K.A. - Short Circuit Calculations for Multiline Switching and End Faults. I.E.E.E. Trans. on Power Apparatus and Systems, Vol.PAS 89, July/Aug.1970, p.1226.

30. Daniels,H.A. and Chen,M. - Compensation of Mutual Coupling in the Formation of the Bus Impedance Matrix. I.E.E.E. Trans. on Power Apparatus and Systems, Vol.PAS 90, May/June 1971, p.1207.

31. Gonzalez,F.A. and Gastineau,E.L. - A Complete 500 Bus Short Circuit Program for Use with a Small Computer. I.E.E.E. Trans. on Power Apparatus and Systems, Vol.PAS 91, May/June 1972, p.1259.

Short Circuit - Bus Admittance Matrix.

32. Nagappan,K. - Step-by-Step Formation of the Bus Admittance Matrix. I.E.E.E. Trans. on Power Apparatus and Systems, Vol.PAS 89, May/June 1970, p.812.

33. Anderson,P.M., Bowen,D.W. and Shah,A.P. - An Indefinite Admittance Network Description for Fault Computation. I.E.E.E. Trans on Power Apparatus and Systems, Vol.PAS 89, July/Aug.1970, p.1215.

34. Tinney,W.F. - Compensation Methods for Network Solutions by Optimally Ordered Triangular Factorisation. I.E.E.E. Trans. on Power Apparatus and Systems, Vol.PAS 91, Jan./Feb.1972, p.123.

Power System Load Flow.

35. Laughton,M.A. and Davies,M.W.H. - Numerical Techniques in Solution of Power System Load Flow Problems. Proc.I.E.E., Vol.111, Sept. 1964, p.1575.

36. Ward,J.B. and Hale,H.W. - Digital Computer Solution of Power Flow Problems. Trans.A.I.E.E., Vol.75, June 1956, p.398.

37. Furst,G.B. and Heal,N.F. - Digital Load Flow Studies. The SNOCOM Load Flow Programme. Jour.I.E.Aust., Vol.34, Dec.1962, p.333.

38. Glimm,A.F. and Stagg,G.W. - Automatic Calculation of Load Flows. Trans.A.I.E.E., Vol.76, Oct.1957, p.817.

39. Gupta,P.P. and Davies,M.W.H. - Digital Computers in Power System Analysis. Proc.I.E.E., Pt.A, Vol.108, Oct.1961, p.383.

40. John,M.N. - A General Method of Digital Network Analysis Particularly Suitable for Use with Low Speed Computers. Proc.I.E.E., Pt.A, Vol.108, Oct.1961, p.369.

41. Brameller,A. and Denmead,J.K. - Some Improved Methods for Digital Network Analysis. Proc.I.E.E., Pt.A, Vol.109, Feb.1962, p.109.

42. Hale,H.W. and Goodrich,R.W. - Digital Computation of Power Flow - Some New Aspects. Trans.A.I.E.E., Pt.IIIA, Vol.78, Oct.1959, p.919.

43. Brown,H.E., Carter,G.K., Happ,H.H. and Person,C.E. - Power Flow Solution by Impedance Matrix Iterative Method. I.E.E.E. Trans. on Power apparatus and Systems, April 1963, p.1.

44. Van Ness,J.E. and Griffin,J.H. - Elimination Methods for Load Flow Studies. A.I.E.E. Power Apparatus and Systems, June 1961, p.299.

45. Sato,N. and Tinney,W.F. - Techniques for Exploiting the Sparsity of the Network Admittance Matrix. I.E.E.E. Trans. on Power Apparatus and Systems, Dec.1963, p.944.

46. Tinney,W.F. and Walker,J.W. - Direct Solutions of Network Equations by Optimally Ordered Triangular Factorisation. Proc.I.E.E.E., Vol.55, Nov.1967, p.1801.

47. Tinney,W.F. and Hart,C.E. - Power Flow Solution by Newton's Method. I.E.E.E. Trans. on Power Apparatus and Systems, Vol.PAS 86, Nov. 1967, p.1449.

48. Wallach,Y. and Even,R.K. - Application of Newton's Method to Load Flow Calculations. Proc.I.E.E., Vol.114, Mar.1967, p.372.

49. Freris,L.L. and Sasson,A.M. - Investigation of the Load Flow Problem. Proc.I.E.E., Vol.115, Oct.1968, p.1459.

50. Ogbuobiri,E.C., Tinney,W.F. and Walker,J.W. - Sparsity Directed Decomposition for Gaussian Elimination on Matrixes. I.E.E.E. Trans. on Power apparatus and Systems, Vol.PAS 89, Jan.1970, p.141.

51. Peterson,N.M. and Meyer,W.S. - Automatic Adjustment of Transformer and Phase Shifter Taps in the Newton Power Flow. I.E.E.E. Trans. on Power Apparatus and Systems, Vol.PAS 90, Jan./Feb.1971, p.1459.

52. Stott,B. and Hobson,E. - Solution of Large Power System Networks by Ordered Elimination. A comparison of Ordering Schemes. Proc. I.E.E., Vol.118, Jan.1971, p.125.

53. Sasson, A.M., Trevino, C. and Aboytes, F. - Improved Newton's Load Flow Through a Minimisation Technique. I.E.E.E. Trans. on Power Apparatus and Systems, Vol.PAS 90, Sept./Oct.1971, p.1974.

54. Stott, B. - Decoupled Newton Load Flow. I.E.E.E. Trans. on Power Apparatus and Systems, Vol.PAS 91, Sept./Oct.1972, p.1955.

55. Peterson, N.M., Tinney, W.F. and Bree, D.W. - Iterative Linear AC Power Flow Solution for Fast Approximate Outage Studies. I.E.E.E. Trans. on Power Apparatus and Systems, Vol.PAS 91, Sept./Oct.1972, p.2048.

S.I.

S

PROCEEDINGS

THE INSTITUTION OF ELECTRICAL ENGINEERS

Volume 112 **Power**

Digital calculation of sequence networks including mutual impedances

W. A. Prebble, B.Sc., B.E., A.M.I.E.E.

Synopsis

A method of including mutual impedances in the bus-impedance matrix by addition and matrix reduction is derived. It is shown how this procedure can be extended, to take into account the modification of an established matrix when the impedance of network elements is changed, and to the calculation of the matrix of a subnetwork which is part of a larger network system. A sample calculation of a four-bus network with three mutually coupled line elements, showing the derivation and modification of its bus-impedance matrix and the derivation of the matrixes when it is divided into two networks, is included in the Appendix.

List of symbols

V_p = voltage of bus p above reference bus

I_p = current injected into network at bus p

Z_{pq} = term of bus-impedance matrix in row p and column q

I, V, Z = nodal-current, voltage and bus-impedance matrixes (primed quantity refers to value before interconnection of partial networks)

I_{pq} = current in element p–q

z_{pq} = self impedance of element p–q

z_m = mutual impedance between elements

z = self-impedance and mutual-impedance matrix of coupled elements

y = inverse of matrix z

y_{pq}, y_m = terms of y

C = connection matrix (all terms zero, plus or minus one)

C_t = transpose of matrix C

C_m = part of matrix C involving mutually coupled elements

In all cases, reference bus is number 0

1 Introduction

Until recent years, problems involving short circuits on transmission systems have been solved mainly by the use of symmetric components and the short-circuit board; so far, this approach has proved satisfactory. As the 3-phase method requires a more complicated system arrangement if mutual impedances, earthing conditions and phase relations are to be represented, symmetric components have continued to be used for the digital-computer solution of these problems. This has the added advantage that network parameters, such as impedances, are readily available.

The two most important conditions for the computer programme are:

(*a*) the representation of mutual couplings between the transmission lines; this can exist in any of the phase networks, or between them, but is taken into account only in the zero-phase-sequence system. This has proved one of the most difficult aspects of solving short-circuit problems on a digital computer

(*b*) flexibility in representing the opening and closing of circuit breakers, to simulate various fault conditions.

Other important conditions are ease of coding and programming, the inclusion of resistances and negative impedances and the accuracy of the calculation with low impedances.

Bearing these conditions in mind, a digital-computer programme should be automatic, accepting any network configuration and doing the equivalent of plugging up the network on a calculating board; i.e. it should construct a mathematical model of the network which contains all the information for the complete solution of the network equations, the model to be formed from a list of impedances and bus connections.

The procedures developed for solving this problem on a digital computer can be classified as:

(*a*) the mesh method[1]

(*b*) the nodal method.[2]

The mesh or loop-equation approach has the disadvantages of being more difficult to code than the nodal method and of involving a matrix inversion, which increases the storage space required and introduces rounding errors. The earlier nodal methods required an iteration procedure, but these

Paper 4700 P, first received 27th August and in revised form 8th December 1964

Mr. Prebble is with the Hydro-Electric Commission, Hobart, Tasmania

have now been superseded by the driving-point- and transfer-impedance matrix or, more simply, the bus-impedance-matrix[3] approach. In this method, the matrix is formed automatically, without matrix inversion or iterative techniques, and all fault calculations are done by simple arithmetic operations on related portions of the matrix. Furthermore, the matrix can be easily and simply modified to represent changes in the transmission network, without having to construct a new matrix, and it is therefore possible to programme for the automatic solution of all fault conditions.

Present methods of handling mutual couplings involve inverting a small matrix and applying corrections to each term of the new row and column that are being added to the matrix of the established partial network.[4] These corrections allow for the direct and indirect couplings of the new element with elements of the established partial network. It is the purpose of this paper to show how this somewhat involved procedure can be replaced by simple addition; the key to this being adding each group of mutually coupled lines to the partial-network matrix, and closing loops after all the elements of the group have been added.

2 Derivation of equations

2.1 Equations for construction of bus-impedance matrix

These are derived from the consideration of two simple basic cases.

2.1.1 Adding to partial network mutually coupled elements which establish new buses

Fig. 1*a* represents a partial network in which buses 1 and 2 are established; Fig. 1*b* represents a pair of elements

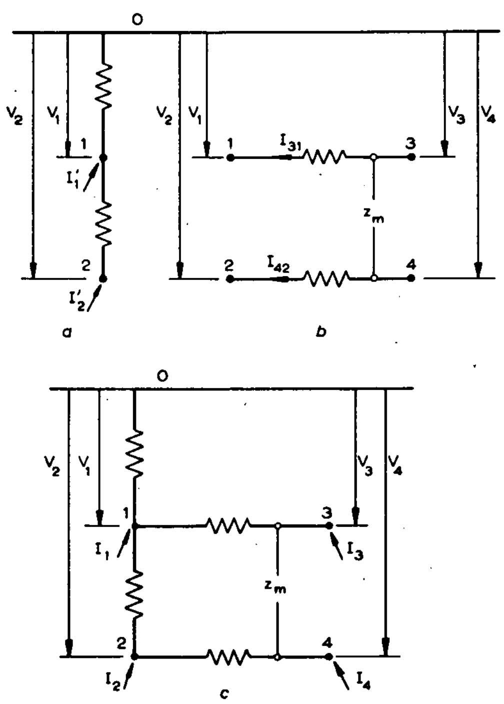


Fig. 1

Adding mutually coupled elements which establish new buses

a Initial network
b Two mutually coupled line elements
c Final network

of self impedances z_{31}, z_{42} and mutual impedance z_m, which are to be added to the partial network, establishing the new buses 3 and 4. Fig. 1*c* represents the final interconnected network.

The nodal equations for the partial network of Fig. 1*a* are

$$\begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix} \begin{bmatrix} I'_1 \\ I'_2 \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad (1)$$

where V_1, V_2 are the voltages applied to the nodes and I'_1, I'_2 are the currents flowing into the nodes 1 and 2; the Zs are the terms of the bus-impedance matrix.

The equations for the network of Fig. 1*b* are

$$\begin{bmatrix} V_3 - V_1 \\ V_4 - V_2 \end{bmatrix} = \begin{bmatrix} z_{31} & z_m \\ z_m & z_{42} \end{bmatrix} \begin{bmatrix} I_{31} \\ I_{42} \end{bmatrix} \quad . \quad . \quad . \quad . \quad (2)$$

where $V_3 - V_1$, $V_4 - V_2$ are the voltages applied and I_{31}, I_{42} are the resulting currents in elements 3–1, 4–2, respectively.

Combining eqns. 1 and 2 into one matrix equation:

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 - V_1 \\ V_4 - V_2 \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} & 0 & 0 \\ Z_{21} & Z_{22} & 0 & 0 \\ 0 & 0 & z_{31} & z_m \\ 0 & 0 & z_m & z_{42} \end{bmatrix} \begin{bmatrix} I'_1 \\ I'_2 \\ I_{31} \\ I_{42} \end{bmatrix} \quad (3)$$

The networks of Figs. 1*a* and *b* are now connected at the common buses so that the power in each element is the same before and after interconnection. For this to be so, the currents before and after interconnection are related by the equations $I'_1 = I_1 + I_3$, $I'_2 = I_2 + I_4$, $I_{31} = I_3$ and $I_{42} = I_4$, which may be written as a single matrix equation:

$$\begin{bmatrix} I'_1 \\ I'_2 \\ I_{31} \\ I_{42} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} \quad . \quad . \quad . \quad . \quad (4)$$

From this it follows that the voltages of the interconnected network are given by[5]

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 - V_1 \\ V_4 - V_2 \end{bmatrix} = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} \quad . \quad . \quad (5)$$

and the bus-impedance matrix by

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} Z_{11} & Z_{12} & 0 & 0 \\ Z_{21} & Z_{22} & 0 & 0 \\ 0 & 0 & z_{31} & z_m \\ 0 & 0 & z_m & z_{42} \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

On multiplying out expression 6 by the rules of matrix algebra, the equations for the interconnected network become[5]

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} & Z_{11} & Z_{12} \\ Z_{21} & Z_{22} & Z_{21} & Z_{22} \\ Z_{11} & Z_{12} & Z_{11} + z_{31} & Z_{12} + z_m \\ Z_{21} & Z_{22} & Z_{21} + z_m & Z_{22} + z_{42} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad (7)$$

2.1.2 Adding mutually coupled elements which complete loops in the established partial network

This includes adding elements in parallel with elements of the partial network. Fig. 2*a* represents a partial network with established buses 1, 2, 3 and 4; Fig. 2*b* the mutually coupled elements which are to be added to the partial network, resulting in the final network of Fig. 2*c*. The equations for the networks of Figs. 2*a* and *b* can be written as a single equation

$$V' = Z'I' \qquad (8)$$

where V', Z' and I' are the following matrixes:

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_3 - V_1 \\ V_4 - V_2 \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} & Z_{13} & Z_{14} & 0 & 0 \\ Z_{21} & Z_{22} & Z_{23} & Z_{24} & 0 & 0 \\ Z_{31} & Z_{32} & Z_{33} & Z_{34} & 0 & 0 \\ Z_{41} & Z_{42} & Z_{43} & Z_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & z_{31} & z_m \\ 0 & 0 & 0 & 0 & z_m & z_{42} \end{bmatrix} \begin{bmatrix} I_1' \\ I_2' \\ I_3' \\ I_4' \\ I_{31} \\ I_{42} \end{bmatrix} \qquad (9)$$

When these two networks are connected together at the common buses and the power is maintained constant, the relation between currents and voltages before and after interconnection is given by

$$I' = CI \qquad (10)$$

and $\quad C_t V' = V \qquad (11)$

Writing out in full the corresponding matrixes for these two equations:

$$\begin{bmatrix} I_1' \\ I_2' \\ I_3' \\ I_4' \\ I_{31} \\ I_{42} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_{31} \\ I_{42} \end{bmatrix} \qquad (12)$$

and

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & -1 & 0 & 1 & 0 \\ 0 & 1 & 0 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_3 - V_1 \\ V_4 - V_2 \end{bmatrix} = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ 0 \\ 0 \end{bmatrix} \qquad (13)$$

Multiplying eqn. 8 by C_t and substituting in eqns. 10 and 11 gives

$$V = ZI \qquad (14)$$

where $Z = C_t Z' C$

After multiplying these three matrixes, Z is given by

$$\begin{bmatrix} Z_{11} & Z_{12} & Z_{13} & Z_{14} & Z_{11} - Z_{13} & Z_{12} - Z_{14} \\ Z_{21} & Z_{22} & Z_{23} & Z_{24} & Z_{21} - Z_{23} & Z_{22} - Z_{24} \\ Z_{31} & Z_{32} & Z_{33} & Z_{34} & Z_{31} - Z_{33} & Z_{32} - Z_{34} \\ Z_{41} & Z_{42} & Z_{43} & Z_{44} & Z_{41} - Z_{43} & Z_{42} - Z_{44} \\ Z_{11} - Z_{31} & Z_{12} - Z_{32} & Z_{13} - Z_{33} & Z_{14} - Z_{34} & Z_{11} - Z_{13} - Z_{31} + Z_{33} + z_{31} & Z_{12} - Z_{14} - Z_{32} + Z_{34} + z_m \\ Z_{21} - Z_{41} & Z_{22} - Z_{42} & Z_{23} - Z_{43} & Z_{24} - Z_{44} & Z_{21} - Z_{23} - Z_{41} + Z_{43} + z_m & Z_{22} - Z_{24} - Z_{42} + Z_{44} + z_{42} \end{bmatrix} \qquad (15)$$

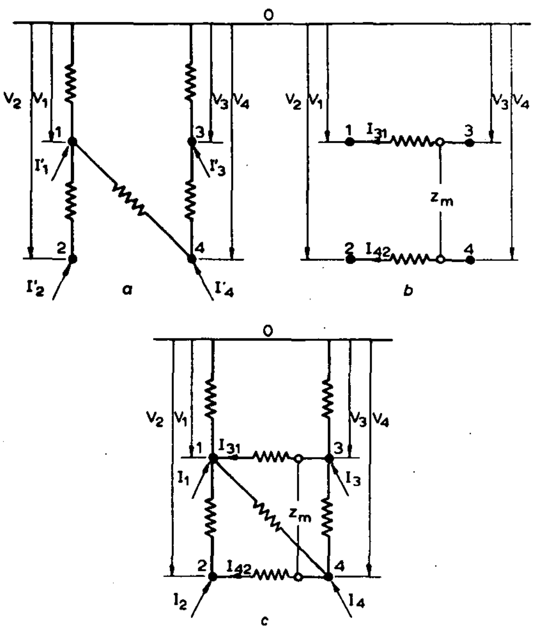


Fig. 2
Adding mutually coupled elements which form loops
a Initial network
b Two mutually coupled line elements
c Final network

As the last two terms of the voltage matrix set out in eqn. 13 are zero, the last two rows and columns of the matrix in expression 15 can be eliminated one at a time by Kron's method; i.e. if r is the last row and column, the term Z_{pq} is replaced by $Z_{pq} - (Z_{pr}Z_{rq}/Z_{rr})$. This procedure is the mathematical equivalent of closing the loops in the network, and, in this case, the final bus-impedance matrix has four rows and columns.

Eqns. 7 and 15 show that the bus-impedance matrix for any network including mutual couplings can be built up by simple addition, followed by closing of loops after all the elements of the group have been added. The elements of the group which establish new buses are added first into their correct rows and columns; e.g. for an element connected between buses p and q, where q is a new bus, make the offdiagonal terms of row and column q equal to the corresponding terms of row and column p, and the diagonal term qq equal to the diagonal term pp plus the self impedance of the element. For a generator element, i.e. $p = 0$, the terms of this row and column are zero. If the element being added is coupled to another element of the group which has already been added to the bus-impedance matrix, the mutual coupling between these elements is added to the appropriate terms; e.g. if this element establishes a new bus r, this mutual impedance is added into terms rq and qr.

When all the elements of the group which establish new buses have been added, those elements which complete loops are added, forming rows and columns outside the established matrix. For an element connected between the established buses p and q, the offdiagonal terms of row r and column r are the differences between the corresponding terms of the rows and columns p and q; the diagonal term is the difference between the terms pr and qr plus the self impedance of the new element. For a generator element, the terms of one of the rows and columns p and q are zero. If the element being added forms a new row and column s and has mutual coupling to the element which formed row and column r, this mutual impedance is added into terms rs and sr.

When all the elements of the group have been added to the matrix, the loops are closed by eliminating one row and column for each loop. If the element which has been added has no mutual coupling to any other element and forms a loop in the network, the corresponding row and column of the matrix are eliminated immediately following the addition of this element to the partial network.

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} & 0 & 0 & Z_{11} & Z_{12} \\ Z_{21} & Z_{22} & 0 & 0 & Z_{21} & Z_{22} \\ 0 & 0 & Z_{33} & Z_{34} & -Z_{33} & -Z_{34} \\ 0 & 0 & Z_{43} & Z_{44} & -Z_{43} & -Z_{44} \\ Z_{11} & Z_{12} & -Z_{33} & -Z_{34} & Z_{11}+Z_{33}+z_{31} & Z_{12}+Z_{34} \\ Z_{21} & Z_{22} & -Z_{43} & -Z_{44} & Z_{21}+Z_{43} & Z_{22}+Z_{44}+z_{42} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_{31} \\ I_{42} \end{bmatrix} \qquad (19)$$

2.2 Modification of established network[6,7]

The bus-impedance matrix of an established network can easily be modified when the impedance of its elements is altered. If z_{old} represents the impedance matrix of a group of coupled elements, and z_{new} their impedance matrix after changes in one or more of the group,

$$y = y_{new} - y_{old} \qquad (16)$$

where $y_{new} = 1/z_{new}$, and $y_{old} = 1/z_{old}$ is first determined. A group of coupled lines with self and mutual impedances given by the terms of the matrix

$$z = 1/y \qquad (17)$$

is then added to the existing bus-impedance matrix. Because all the buses are established, each element of this group therefore makes a loop in the network. When removing a line with mutual coupling from the network, a very high self impedance and zero mutual impedance are put in the corresponding terms of the matrix z_{new}. For studying the effects of switching operations in a network, a partial matrix for selected buses is formed from the complete matrix, and modifications are carried out on this smaller matrix.

To change the impedance of an element which has no mutual couplings, add a parallel element of admittance equal to the difference between the new and the old admittances.

In calculating a bus-impedance matrix, this is the only occasion that a matrix inversion is required, and the inversion of the small matrixes involved can be done by division by the major diagonal terms.[8]

2.3 Interconnection of networks[9]

Fig. 3*a* represents two established subnetworks, and Fig. 3*b* two line elements which connect these two networks, as shown in Fig. 3*c*. As in Sections 2.1.1 and 2.1.2, the matrix equation for the networks of Figs. 3*a* and *b* is

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_3 - V_1 \\ V_4 - V_2 \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} & 0 & 0 & 0 & 0 \\ Z_{21} & Z_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_{33} & Z_{34} & 0 & 0 \\ 0 & 0 & Z_{43} & Z_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & z_{31} & 0 \\ 0 & 0 & 0 & 0 & 0 & z_{42} \end{bmatrix} \begin{bmatrix} I'_1 \\ I'_2 \\ I'_3 \\ I'_4 \\ I_{31} \\ I_{42} \end{bmatrix} \qquad (18)$$

If the networks of Figs. 3*a* and *b* are so connected that the power in each element remains constant, the matrix equation for the resulting network (Fig. 3*c*) is

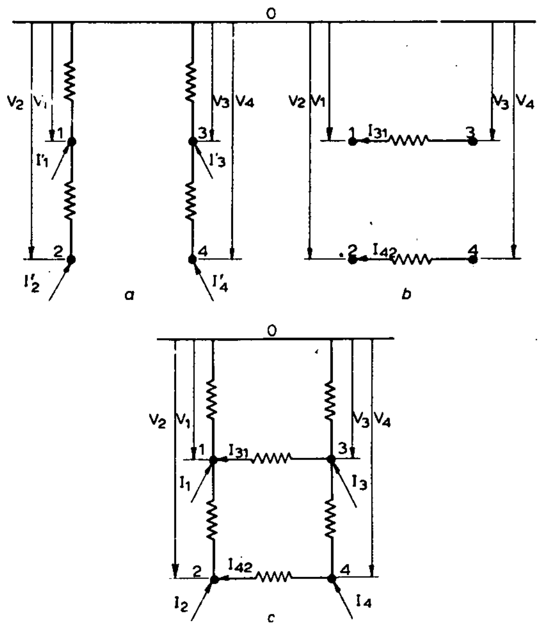


Fig. 3
Interconnection of two networks
a Initial networks
b Two line elements
c Final network

The matrix equation for the left subnetwork of Fig. 3a, which includes the effect of the right subnetwork, is therefore given by

$$\begin{bmatrix} V_1 \\ V_2 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} & Z_{11} & Z_{12} \\ Z_{21} & Z_{22} & Z_{21} & Z_{22} \\ Z_{11} & Z_{12} & Z_{11}+Z_{33} & Z_{12}+Z_{34} \\ Z_{21} & Z_{22} & Z_{21}+Z_{43} & Z_{22}+Z_{44} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_{31} \\ I_{42} \end{bmatrix} \quad . \; . \; . \; . \; (20)$$

The final bus-impedance matrix for this subnetwork is found by eliminating the last two rows and columns of the matrix Z in eqn. 20. From this can be derived the rule for so cutting a large network into smaller subnetworks that the bus-impedance matrix of each subnetwork includes the effect of the complete network. The network is cut at specified buses into smaller networks A, B, C etc., and the bus-impedance matrix for each of these is calculated. The bus-impedance matrix of the subnetwork A, including the effect of the subnetwork B, is found by selecting the partial matrix of the common buses from B's bus-impedance matrix and adding this as a group of mutually coupled elements, between the reference and common buses, to A's matrix. The bus-impedance matrix for A, including the effect of B, is formed by eliminating the rows and columns so added.

In this way short-circuit studies of, for example, proposed generation and transmission arrangements can be performed without calculating the complete bus-impedance matrix for each arrangement. The matrix for the basic network is established, followed by those for each additional subnetwork. The required short circuits are calculated from appropriate combinations of the subnetworks with the basic network.

3 Computer calculation of network faults

Fig. 4 outlines the main steps for calculating the bus-impedance matrix of a network, using a digital computer, from a list of line-and-generator self-and-mutual impedances and bus identifications. Before an element can be added to the bus-impedance matrix, it must be connected at one or

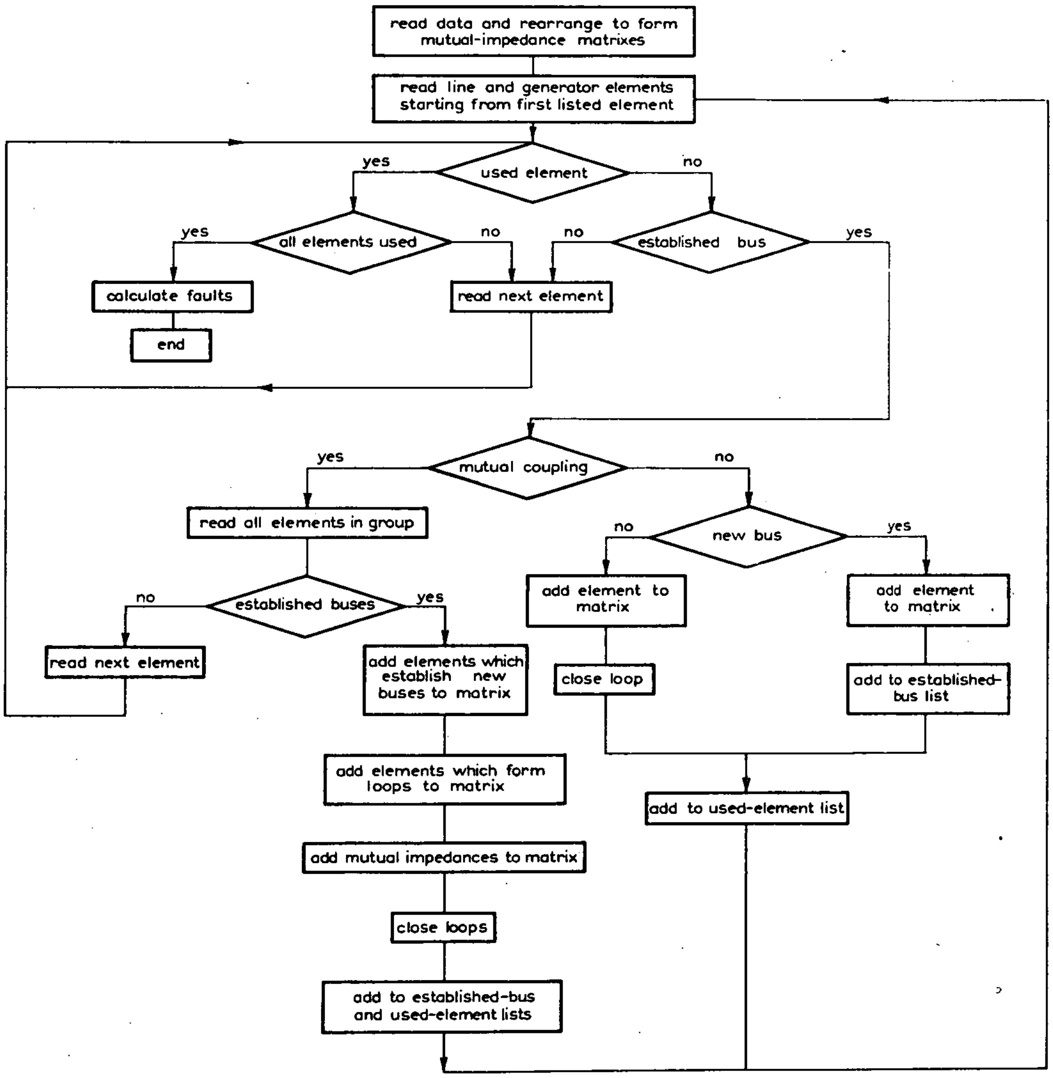


Fig. 4
Basic flow diagram for calculation of bus-impedance matrix

both terminals to an already established bus (the reference bus being considered established). The formation of the matrix is commenced with the first element of the list connected to the reference bus. In general, the matrix is symmetric about the main diagonal, and so one half only is stored in the computer.

At the beginning of the programme, the mutual impedances are sorted into groups; a list of the elements in each group is stored and a matrix z_g of their mutual impedances is set up. When an element with mutual coupling is read from the line-and-generator list, all the elements of its group are tested to determine whether they can be added to the bus-impedance matrix being formed, and the order in which they will be added is established (those which form new buses first). During this test, the mutual-impedance connection matrix C_m is formed as shown in Section 7.2. If the test is successful, the elements of the group are added one at a time to the bus-impedance matrix, then mutual impedances found from the product $C_{mt} z_g C_m$ are added and the loops closed by reduction. If all the elements of the group cannot be added at this stage of the calculation, the next listed element is read.

4 Network-fault power and line currents

The calculation of the fault power at a bus and the resulting sequence currents in the elements of a network is illustrated with reference to Fig. 2c. Assuming that all impedances are expressed per unit, and that the calculated bus-impedance matrix for the network is

$$\begin{bmatrix} Z_{11} & Z_{12} & Z_{13} & Z_{14} \\ Z_{21} & Z_{22} & Z_{23} & Z_{24} \\ Z_{31} & Z_{32} & Z_{33} & Z_{34} \\ Z_{41} & Z_{42} & Z_{43} & Z_{44} \end{bmatrix} \quad \ldots \ldots \ldots \quad (21)$$

where Z_{11}, Z_{12} etc. do not have the same values as Z_{11}, Z_{12} etc. in eqn. 15 but are the values after the last two rows and columns of that equation have been eliminated; then, for a fault on bus 2,

$$\begin{aligned} &\text{bus fault power} && = 1/Z_{22} \text{ per unit} \\ &\text{voltage on bus 1} && = 1 - Z_{21}/Z_{22} \text{ per unit} \\ &\text{fault current in line 4–3} && = (Z_{23} - Z_{24})/Z_{22}z_{43} \text{ per unit} \\ &\text{fault current in line 3–1} && = y_{31}(Z_{21} - Z_{23})/Z_{22} + \\ &&& \quad y_m(Z_{22} - Z_{24})/Z_{22} \text{ per unit} \end{aligned}$$

where y_{31} and y_m are terms of the admittance matrix y, the inverse of the mutual-impedance matrix z involving the coupled lines 3–1 and 4–2. The phase powers and currents for different types of fault are then found by combinations of the sequence networks.

5 Conclusion

The problem of calculating system short circuits has been analysed by Kron's tensor methods and translated into matrix operations which can easily be programmed for a digital computer. From the final bus-impedance matrix, the faults at any point in the network (with any modifications such as opening or closing of lines) can be determined by simple arithmetic. The method also has the advantage of simplicity of coding, and hence the data preparation is no more complex than for a short-circuit-board study; further, it can be programmed to provide a complete record of the short-circuit power at all locations, and the resulting current in every element of the network.

6 References

1 LANTZ, M. J.: 'Digital short-circuit solution of power system networks including mutual impedance', *Trans. Amer. Inst. Elect. Engrs.*, 1958, **77**, Pt. III, p. 1230
2 COOMBE, L. W., and LEWIS, D. G.: 'Digital calculation of short-circuit currents in large complex impedance networks', *ibid.*, 1957, **76**, Pt. III, p. 1394
3 BROWN, H. E., PERSON, C. E., KIRCHMAYER, L. K., and STAGG, G. W.: 'Digital calculation of three-phase short-circuits by matrix method', *ibid.*, 1960, **79**, Pt. III, p. 1277
4 EL-ABIAD, A. H.: 'Digital calculation of line-to-ground short-circuits by matrix method', *ibid.*, 1960, **79**, Pt. III, p. 323
5 KRON, G.: 'Tensor analysis of networks' (Wiley, 1939)
6 BRAMELLER, A., and DENMEAD, J. K.: 'Some improved methods for digital network analysis', *Proc. IEE*, 1962, **109** A, p. 109
7 LAUGHTON, M. A., and DAVIES, M. W. H.: 'Numerical techniques in solution of power-system load-flow problems', *ibid.*, 1964, **111**, (9), p. 1575
8 SHIPLEY, R. B., and COLEMAN, D.: 'A new direct matrix inversion method', *Trans. Amer. Inst. Elect. Engrs.*, 1959, **78**, Pt. I, p. 568
9 KRON, G.: 'Diakoptics—the piecewise solution of large scale systems' (Macdonald, 1963)

7 Appendix

7.1.1 Calculation of bus-impedance matrix for small network

No account is taken of prefault conditions in the network (Fig. 5), and the numerical work is simplified by neglecting the resistance of the line and generator elements.

Start with generator elements 1–0:

$$[0{\cdot}04]$$

Add generator element 2–0, establishing a new bus 2:

$$\begin{bmatrix} 0{\cdot}04 & 0 \\ 0 & 0{\cdot}08 \end{bmatrix}$$

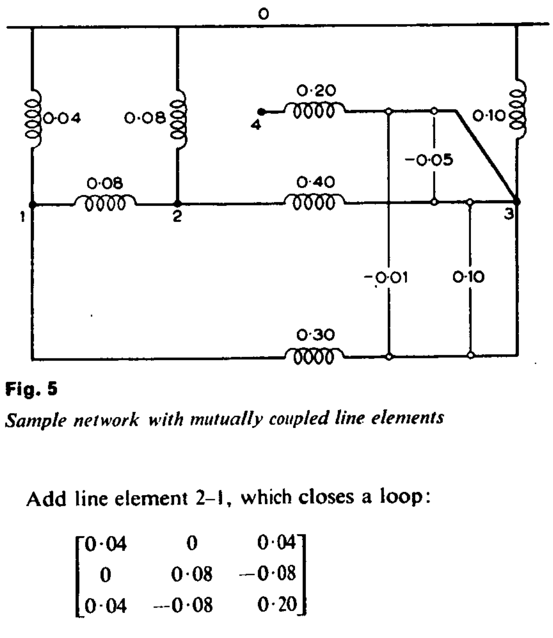


Fig. 5
Sample network with mutually coupled line elements

Add line element 2–1, which closes a loop:

$$\begin{bmatrix} 0{\cdot}04 & 0 & 0{\cdot}04 \\ 0 & 0{\cdot}08 & -0{\cdot}08 \\ 0{\cdot}04 & -0{\cdot}08 & 0{\cdot}20 \end{bmatrix}$$

Eliminate the last row and column and add generator element 3–0, establishing a new bus 3:

$$\begin{bmatrix} 0{\cdot}032 & 0{\cdot}016 & 0 \\ 0{\cdot}016 & 0{\cdot}048 & 0 \\ 0 & 0 & 0{\cdot}100 \end{bmatrix}$$

Add the mutually coupled group of lines, 4–3 being added first, establishing a new bus 4, then 3–1 and 3–2, which close loops:

$$\begin{array}{cc} & \begin{array}{cccccc} & & & 4\text{–}3 & 3\text{–}1 & 3\text{–}2 \end{array} \\ \begin{array}{c} \\ \\ \\ 4\text{–}3 \\ 3\text{–}1 \\ 3\text{–}2 \end{array} & \begin{bmatrix} 0\cdot032 & 0\cdot016 & 0 & 0 & 0\cdot032 & 0\cdot016 \\ 0\cdot016 & 0\cdot048 & 0 & 0 & 0\cdot016 & 0\cdot048 \\ 0 & 0 & 0\cdot100 & 0\cdot100 & -0\cdot100 & -0\cdot100 \\ 0 & 0 & 0\cdot100 & 0\cdot300 & -0\cdot110 & -0\cdot150 \\ 0\cdot032 & 0\cdot016 & -0\cdot100 & -0\cdot110 & 0\cdot432 & 0\cdot216 \\ 0\cdot016 & 0\cdot048 & -0\cdot100 & -0\cdot150 & 0\cdot216 & 0\cdot548 \end{bmatrix} \end{array}$$

The term in row 4 and column 4 is $0\cdot100 + 0\cdot200 = 0\cdot300$
the term in row 4 and column 5 is $0 - 0\cdot100 - 0\cdot010 = - 0\cdot110$
the term in row 6 and column 6 is $0\cdot048 - (- 0\cdot100) + + 0\cdot400 = 0\cdot548$ etc.

Eliminate row 6 and column 6:

$$\begin{bmatrix} 0\cdot032 & 0\cdot015 & 0\cdot003 & 0\cdot004 & 0\cdot026 \\ 0\cdot015 & 0\cdot044 & 0\cdot009 & 0\cdot013 & -0\cdot003 \\ 0\cdot003 & 0\cdot009 & 0\cdot082 & 0\cdot073 & -0\cdot061 \\ 0\cdot004 & 0\cdot013 & 0\cdot073 & 0\cdot259 & -0\cdot051 \\ 0\cdot026 & -0\cdot003 & -0\cdot061 & -0\cdot051 & 0\cdot347 \end{bmatrix}$$

Eliminating row 5 and column 5 gives the bus-impedance matrix of the sample network:

$$\begin{bmatrix} 0\cdot030 & 0\cdot015 & 0\cdot007 & 0\cdot008 \\ 0\cdot015 & 0\cdot044 & 0\cdot008 & 0\cdot013 \\ 0\cdot007 & 0\cdot008 & 0\cdot071 & 0\cdot064 \\ 0\cdot008 & 0\cdot013 & 0\cdot064 & 0\cdot251 \end{bmatrix} \quad . \quad . \quad . \quad . \quad (22)$$

7.1.2 Modification of bus-impedance matrix

The self and mutual impedances of the group of coupled lines in the network of Fig. 5 are changed as follows:

self impedance of line 4–3 from 0·20 to 0·40

self impedance of line 3–1 from 0·30 to 0·50

self impedance of line 3–2 from 0·40 to 0·20

mutual impedance between lines 4–3 and 3–1 from −0·01 to 0

mutual impedance between lines 4–3 and 3–2 from −0·05 to 0

$$z_{old} = \begin{array}{cc} & \begin{array}{ccc} 4\text{–}3 & 3\text{–}1 & 3\text{–}2 \end{array} \\ \begin{array}{c} 4\text{–}3 \\ 3\text{–}1 \\ 3\text{–}2 \end{array} & \begin{bmatrix} 0\cdot20 & -0\cdot01 & -0\cdot05 \\ -0\cdot01 & 0\cdot30 & 0\cdot10 \\ -0\cdot05 & 0\cdot10 & 0\cdot40 \end{bmatrix} \end{array}$$

Inverting z_{old} gives

$$y_{old} = \begin{array}{cc} & \begin{array}{ccc} 4\text{–}3 & 3\text{–}1 & 3\text{–}2 \end{array} \\ \begin{array}{c} 4\text{–}3 \\ 3\text{–}1 \\ 3\text{–}2 \end{array} & \begin{bmatrix} 5\cdot162 & -0\cdot047 & 0\cdot657 \\ -0\cdot047 & 3\cdot637 & -0\cdot915 \\ 0\cdot657 & -0\cdot915 & 2\cdot811 \end{bmatrix} \end{array}$$

$$z_{new} = \begin{array}{cc} & \begin{array}{ccc} 4\text{–}3 & 3\text{–}1 & 3\text{–}2 \end{array} \\ \begin{array}{c} 4\text{–}3 \\ 3\text{–}1 \\ 3\text{–}2 \end{array} & \begin{bmatrix} 0\cdot40 & 0 & 0 \\ 0 & 0\cdot50 & 0\cdot10 \\ 0 & 0\cdot10 & 0\cdot20 \end{bmatrix} \end{array}$$

Inverting z_{new} gives

$$y_{new} = \begin{array}{cc} & \begin{array}{ccc} 4\text{–}3 & 3\text{–}1 & 3\text{–}2 \end{array} \\ \begin{array}{c} 4\text{–}3 \\ 3\text{–}1 \\ 3\text{–}2 \end{array} & \begin{bmatrix} 2\cdot500 & 0 & 0 \\ 0 & 2\cdot222 & -1\cdot111 \\ 0 & -1\cdot111 & 5\cdot556 \end{bmatrix} \end{array}$$

$$y = y_{new} - y_{old} = \begin{array}{cc} & \begin{array}{ccc} 4\text{–}3 & 3\text{–}1 & 3\text{–}2 \end{array} \\ \begin{array}{c} 4\text{–}3 \\ 3\text{–}1 \\ 3\text{–}2 \end{array} & \begin{bmatrix} -2\cdot662 & 0\cdot047 & -0\cdot657 \\ 0\cdot047 & -1\cdot415 & -0\cdot196 \\ -0\cdot657 & -0\cdot196 & 2\cdot745 \end{bmatrix} \end{array}$$

Invert y to give

$$z = \begin{array}{cc} & \begin{array}{ccc} 4\text{–}3 & 3\text{–}1 & 3\text{–}2 \end{array} \\ \begin{array}{c} 4\text{–}3 \\ 3\text{–}1 \\ 3\text{–}2 \end{array} & \begin{bmatrix} -0\cdot355 & 0 & -0\cdot085 \\ 0 & -0\cdot700 & -0\cdot050 \\ -0\cdot085 & -0\cdot050 & 0\cdot340 \end{bmatrix} \end{array} \quad . \quad . \quad . \quad . \quad (23)$$

Adding a group of coupled lines 4–3, 3–1 and 3–2, with self and mutual impedances given by eqn. 23, to the bus-impedance matrix 22 calculated in Section 7.1.1:

$$\begin{array}{cc} & \begin{array}{ccccccc} & & & & 4\text{–}3 & 3\text{–}1 & 3\text{–}2 \end{array} \\ \begin{array}{c} \\ \\ \\ \\ 4\text{–}3 \\ 3\text{–}1 \\ 3\text{–}2 \end{array} & \begin{bmatrix} 0\cdot030 & 0\cdot015 & 0\cdot007 & 0\cdot008 & -0\cdot001 & 0\cdot023 & 0\cdot008 \\ 0\cdot015 & 0\cdot044 & 0\cdot008 & 0\cdot013 & -0\cdot005 & 0\cdot007 & 0\cdot036 \\ 0\cdot007 & 0\cdot008 & 0\cdot071 & 0\cdot064 & 0\cdot007 & -0\cdot064 & -0\cdot063 \\ 0\cdot008 & 0\cdot013 & 0\cdot064 & 0\cdot251 & -0\cdot187 & -0\cdot056 & -0\cdot051 \\ -0\cdot001 & -0\cdot005 & 0\cdot007 & -0\cdot187 & -0\cdot161 & -0\cdot008 & -0\cdot097 \\ 0\cdot023 & 0\cdot007 & -0\cdot064 & -0\cdot056 & -0\cdot008 & -0\cdot613 & 0\cdot021 \\ 0\cdot008 & 0\cdot036 & -0\cdot063 & -0\cdot051 & -0\cdot097 & 0\cdot021 & 0\cdot439 \end{bmatrix} \end{array}$$

The term in row 2 column 5 is $0\cdot008 - 0\cdot013 = -0\cdot005$,
the term in row 6 column 7 is $0\cdot007 - (-0\cdot064) - 0\cdot050 = 0\cdot021$
the term in row 6 column 6 is $0\cdot023 - (-0\cdot064) - 0\cdot700 = - 0\cdot613$ etc.

Eliminating row 7 and column 7:

$$\begin{bmatrix} 0\cdot030 & 0\cdot014 & 0\cdot008 & 0\cdot009 & 0\cdot001 & 0\cdot022 \\ 0\cdot014 & 0\cdot041 & 0\cdot013 & 0\cdot017 & 0\cdot003 & 0\cdot005 \\ 0\cdot008 & 0\cdot013 & 0\cdot062 & 0\cdot056 & -0\cdot006 & -0\cdot061 \\ 0\cdot009 & 0\cdot017 & 0\cdot056 & 0\cdot246 & -0\cdot199 & -0\cdot053 \\ 0\cdot001 & 0\cdot003 & -0\cdot006 & -0\cdot199 & -0\cdot181 & -0\cdot004 \\ 0\cdot022 & 0\cdot005 & -0\cdot061 & -0\cdot053 & -0\cdot004 & -0\cdot615 \end{bmatrix}$$

Eliminating row 6 and column 6:

$$\begin{bmatrix} 0\cdot030 & 0\cdot014 & 0\cdot006 & 0\cdot007 & 0\cdot001 \\ 0\cdot014 & 0\cdot041 & 0\cdot013 & 0\cdot016 & 0\cdot003 \\ 0\cdot006 & 0\cdot013 & 0\cdot068 & 0\cdot062 & -0\cdot006 \\ 0\cdot007 & 0\cdot016 & 0\cdot062 & 0\cdot250 & -0\cdot199 \\ 0\cdot001 & 0\cdot003 & -0\cdot006 & -0\cdot199 & -0\cdot181 \end{bmatrix}$$

Eliminating row 5 and column 5 gives the bus-impedance matrix for the modified network:

$$\begin{bmatrix} 0\cdot030 & 0\cdot014 & 0\cdot006 & 0\cdot006 \\ 0\cdot014 & 0\cdot041 & 0\cdot013 & 0\cdot013 \\ 0\cdot006 & 0\cdot013 & 0\cdot068 & 0\cdot068 \\ 0\cdot006 & 0\cdot013 & 0\cdot068 & 0\cdot468 \end{bmatrix} \quad . \quad . \quad . \quad . \quad (24)$$

7.1.3 Network subdivision

The network of Fig. 5 is cut at buses 1 and 2 into two subnetworks *a* and *b* (Fig. 6). The bus-impedance matrix for the subnetwork of Fig. 6*a* is:

$$\begin{bmatrix} 0\cdot032 & 0\cdot016 \\ 0\cdot016 & 0\cdot048 \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (25)$$

and that for subnetwork *b* is

$$\begin{bmatrix} 0\cdot400 & 0\cdot200 & 0\cdot100 & 0\cdot110 \\ 0\cdot200 & 0\cdot500 & 0\cdot100 & 0\cdot150 \\ 0\cdot100 & 0\cdot100 & 0\cdot100 & 0\cdot100 \\ 0\cdot110 & 0\cdot150 & 0\cdot100 & 0\cdot300 \end{bmatrix} \quad . \quad . \quad . \quad . \quad (26)$$

The bus-impedance matrix for the subnetwork of Fig. 6*a*, which includes the effect of subnetwork *b*, is formed by

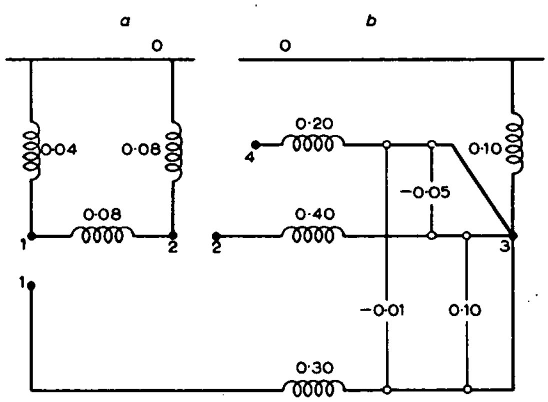


Fig. 6

Sample network divided into subnetworks a and b

adding elements 1–0 and 2–0 of self and mutual impedances 0·400, 0·500 and 0·200, respectively, to the matrix 25 for *a*:

$$\begin{bmatrix} 0\cdot032 & 0\cdot016 & 0\cdot032 & 0\cdot016 \\ 0\cdot016 & 0\cdot048 & 0\cdot016 & 0\cdot048 \\ 0\cdot032 & 0\cdot016 & 0\cdot432 & 0\cdot216 \\ 0\cdot016 & 0\cdot048 & 0\cdot216 & 0\cdot548 \end{bmatrix}$$

Eliminating row 4 and column 4:

$$\begin{bmatrix} 0\cdot032 & 0\cdot015 & 0\cdot026 \\ 0\cdot015 & 0\cdot044 & -0\cdot003 \\ 0\cdot026 & -0\cdot003 & 0\cdot347 \end{bmatrix}$$

Eliminating row 3 and column 3 gives the bus-impedance matrix for subnetwork *a*, including the effect of *b*:

$$\begin{bmatrix} 0\cdot030 & 0\cdot015 \\ 0\cdot015 & 0\cdot044 \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (27)$$

The bus-impedance matrix for subnetwork *b*, which includes the effect of subnetwork *a*, is formed by adding elements 1–0 and 2–0, of self and mutual impedances 0·032, 0·048 and 0·016, respectively, to the matrix 26 for *b*:

$$\begin{bmatrix} 0\cdot400 & 0\cdot200 & 0\cdot100 & 0\cdot110 & 0\cdot400 & 0\cdot200 \\ 0\cdot200 & 0\cdot500 & 0\cdot100 & 0\cdot150 & 0\cdot200 & 0\cdot500 \\ 0\cdot100 & 0\cdot100 & 0\cdot100 & 0\cdot100 & 0\cdot100 & 0\cdot100 \\ 0\cdot110 & 0\cdot150 & 0\cdot100 & 0\cdot300 & 0\cdot110 & 0\cdot150 \\ 0\cdot400 & 0\cdot200 & 0\cdot100 & 0\cdot110 & 0\cdot432 & 0\cdot216 \\ 0\cdot200 & 0\cdot500 & 0\cdot100 & 0\cdot150 & 0\cdot216 & 0\cdot548 \end{bmatrix}$$

Eliminating row 6 and column 6:

$$\begin{bmatrix} 0\cdot327 & 0\cdot018 & 0\cdot064 & 0\cdot055 & 0\cdot321 \\ 0\cdot018 & 0\cdot044 & 0\cdot009 & 0\cdot013 & 0\cdot003 \\ 0\cdot064 & 0\cdot009 & 0\cdot082 & 0\cdot073 & 0\cdot061 \\ 0\cdot055 & 0\cdot013 & 0\cdot073 & 0\cdot259 & 0\cdot051 \\ 0\cdot321 & 0\cdot003 & 0\cdot061 & 0\cdot051 & 0\cdot346 \end{bmatrix}$$

Eliminating row 5 and column 5 gives the bus-impedance matrix for subnetwork *b* including the effect of *a* (this is the same matrix as that calculated in Section 7.1.1):

$$\begin{bmatrix} 0\cdot030 & 0\cdot015 & 0\cdot007 & 0\cdot008 \\ 0\cdot015 & 0\cdot044 & 0\cdot008 & 0\cdot013 \\ 0\cdot007 & 0\cdot008 & 0\cdot071 & 0\cdot064 \\ 0\cdot008 & 0\cdot013 & 0\cdot064 & 0\cdot251 \end{bmatrix} \quad . \quad . \quad . \quad . \quad . \quad (28)$$

7.2 Derivation of the mutual-impedance connection matrix C_m

In Fig. 7*a*, buses 2 and 3 have been established, and the group of mutually coupled lines are added to the bus-impedance matrix in the order:

2–1 establishing a new bus 1
4–3 establishing a new bus 4
5–4 establishing a new bus 5.

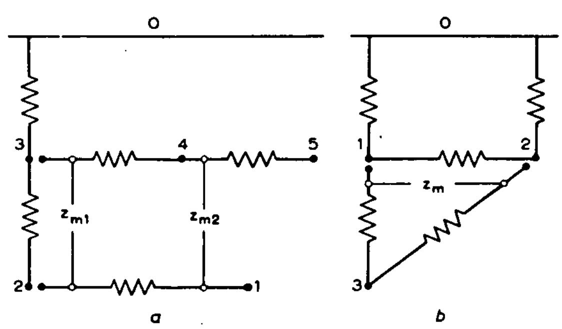


Fig. 7

Adding mutually coupled elements to networks

a Elements which establish new buses
b Elements which establish new bus and form loop

The matrix C_m for this group of coupled lines is:

$$C_m = \begin{array}{c} \\ 2\text{–}1 \\ 4\text{–}3 \\ 5\text{–}4 \end{array} \begin{array}{c} \begin{array}{ccc} 2\text{–}1 & 4\text{–}3 & 5\text{–}4 \end{array} \\ \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \end{array} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (29)$$

In Fig. 7*b*, buses 1 and 2 have been established, and the mutually coupled lines are added to the bus-impedance matrix in the order:

3–1 establishing a new bus 3

3–2 closing a loop.

The matrix C_m for these coupled lines is:

$$C_m = \begin{array}{c} \\ 3\text{–}1 \\ 3\text{–}2 \end{array} \begin{array}{c} \begin{array}{cc} 3\text{–}1 & 3\text{–}2 \end{array} \\ \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix} \end{array} \quad \ldots \ldots \ldots \quad (30)$$

The matrix C_m is derived by examining the bus connection numbers of each element of the group in the order in which it is added to the bus-impedance matrix. C_m has the properties:

Its order is equal to the number of elements in the group.
In general, it is asymmetric.
The major diagonal terms are +1 for loop elements and +1 or −1 for new bus elements, depending on the relation between the element bus connection numbers.

All terms below the major diagonal are zero, but terms above the major diagonal are +1 or —1 if an element of the group is connected to a new bus already established by the group, the sign depending on the relation between the bus connection numbers.

S.2.

Thus, the Cauer realization may be developed from purely algebraic considerations.

Frank M. Brown, Captain U.S.A.F.,
Department of Electrical Engineering,
Air Force Institute of Technology,
Wirght-Patterson A.F.B., Ohio, U.S.A.

REFERENCES

1. *L. M. Milne-Thomson* The Calculus of Finite Differences, London, Macmillan and Co., Ltd., 1933, pp. 108 – 110.

The Calculation of the Transfer and Driving Point Impedance Matrix by Digital Computer

The transfer and driving point impedance matrix is important for digital computer studies of power system networks. By examining the connection matrix simple rules are derived for the construction of this matrix; in particular, each line in a mutually coupled group of lines is initially considered as uncoupled the necessary mutual impedance values being derived separately for the whole group. These rules involve impedance values and node numbers only and are therefore suitable for digital computer programming.

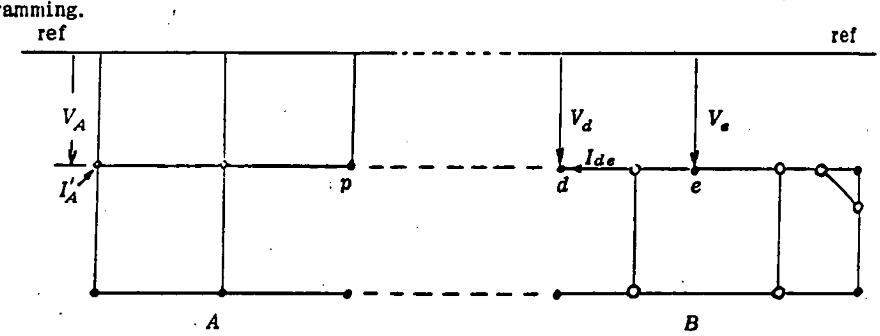


Fig. 1: Network A and group of coupled lines B.

In the following discussion all impedances are lumped, as is usual in power system studies.

In Fig. 1, A represents a network of $n+1$ nodes, one of which is chosen as the reference. The relation between the node voltages, referred to the reference, and the node currents is given by the matrix equation

$$[V_A] = [Z_A][I'_A] \tag{1}$$

where $[Z_A]$ is the $n \times n$ transfer and driving point impedance matrix of the network A.

In Fig. 1, B represents a group of m mutually coupled lines and a reference node for which the m equations

$$[V_d - V_e] = [Z_B][I_{de}] \tag{2}$$

hold, V_d, V_e are the node voltages, referred to the reference, of line $d-e$, I_{de} the line current and $[Z_B]$ the $m \times m$ matrix of line self and mutual impedances (the convention assumed for the direction of current flow is from the higher to the lower numbered node).

Combining equations (1) and (2):

$$\begin{bmatrix} V_A \\ V_d - V_e \end{bmatrix} = \begin{bmatrix} Z_A & 0 \\ 0 & Z_B \end{bmatrix} \begin{bmatrix} I'_A \\ I_{de} \end{bmatrix} \tag{3}$$

It is required to find the transfer and driving point impedance matrix of the interconnected network formed when the lines B are added to the network A. Connect the group of lines B to the network A so that the voltage of each node is the same before and after interconnection, i.e., the current is unchanged in each branch and the power in the connected network is the same as in the two parts A and B. The equation relating the currents before and after interconnection:

$$\begin{bmatrix} I'_A \\ I_{de} \end{bmatrix} = \begin{bmatrix} 1 & C \\ 0 & C_g \end{bmatrix} \begin{bmatrix} I_A \\ I_B \end{bmatrix} \tag{4}$$

where $[I_A]$ are the currents at the nodes in the interconnected network represented by A and $[I_B]$ are node currents when a line of B establishes a node in the interconnected network and line currents otherwise.

The terms of the $n \times m$ matrix $[C]$ and the $m \times m$ matrix $[C_g]$ are +1, -1 and 0. Consider the line k of B with nodes d and e. In the interconnected network two cases arise:

1. A node is established by this line; let this node be e. If p is the node of A to which d is connected either directly (in which case $p = d$) or indirectly through other lines of B, then the term pk of $[C]$ is +1 and the other terms in column k zero. The diagonal term kk of $[C_g]$ is +1 if $e > d$, -1 if $e < d$. The remaining column k terms of $[C_g]$ are zero unless there are lines k_1, k_2, ... between nodes d and p in which case the terms k_1k, k_2k, ... are +1, if in the path from d to p the higher numbered node occurs first, -1 if the lower numbered node occurs first.

2. The line connects two nodes which have already been established. Let p, q be the nodes of A to which d, e are either directly or indirectly connected, then if $e > d$ the terms pk and qk of $[C]$ are +1 and -1 respectively, the other terms in column k are zero. The diagonal term kk of $[C_g]$ is +1. For direct connection of the line k to A the remaining column k terms of $[C_g]$ are zero; otherwise there are terms +1 or -1 determined as in 1 for lines between nodes d and p, but with the opposite sign for lines between nodes e and q.

The voltages before and after interconnection are related by the equation:

$$\begin{bmatrix} 1 & 0 \\ C_t & C_{gt} \end{bmatrix} \begin{bmatrix} V_A \\ V_d - V_e \end{bmatrix} = \begin{bmatrix} V_A \\ V_B \end{bmatrix} \tag{5}$$

where $[V_B]$ are node voltages when the line of B establishes a node in the interconnected network, otherwise zero.

The impedance matrix after interconnection is given by:

$$\begin{bmatrix} 1 & 0 \\ C_t & C_{gt} \end{bmatrix} \begin{bmatrix} Z_A & 0 \\ 0 & Z_B \end{bmatrix} \begin{bmatrix} 1 & C \\ 0 & C_g \end{bmatrix} = \begin{bmatrix} Z_A & Z_A C \\ C_t Z_A & C_t Z_A C + C_{gt} Z_B C_g \end{bmatrix} \tag{6}$$

The matrix $[Z_B]$ can be expressed as the sum of two matrices:

$$[Z_B] = [Z_S] + [Z_M] \tag{7}$$

where $[Z_S]$ is the matrix of the self-impedances of the lines B, i.e., all off-diagonal terms zero, and $[Z_M]$ is the matrix of the mutual impedances between the lines.

It follows from equations (6) and (7) that the impedance matrix for the interconnected network can be derived in three steps.

1. Add the lines $d-e$ one at a time. If the node d is already established then repeat row and column d in row and column e, the principal diagonal term ee is the self-impedance of the line plus the term de. If both nodes d and e are established then form a new row and column f with terms $(e > d)$ row d minus row e, column d minus column e and principal diagonal term ff the self-impedance of the line plus term df minus term ef. In the case in which d is the reference node its row and column terms are taken as zero.
2. The mutual impedance values are then added into the matrix. These are determined by the matrix product $[C_{gt} Z_M C_g]$, where $[C_{gt}]$, $[C_g]$ are derived from the order in which the lines B are added and the relation between their node numbers.
3. The rows and columns f are eliminated by matrix reduction, as the voltage terms in equation (5) which correspond to lines of B added between established nodes, are zero.

W. A. Prebble,
Hydro-Electric Commission,
Hobart, Tasmania.

REFERENCES

1. *G. Kron* Tensor Analysis of Networks, Wiley, 1939.
2. *G. Kron* Diakoptics—the Piecewise Solution of Large Scale Systems, MacDonald.
3. *A. H. El-Abiad* Digital Calculation of Line-to-Ground Short-Circuits by Matrix Method. Transactions American Institute of Electrical Engineers, 1960, Vol. 79, Part III, p. 323.
4. *M. A. Laughton and M.W.H. Davies* Numerical Techniques in Solution of Power-System Load-Flow Problems, Proceedings I.E.E., 1964, Vol. III, p. 1575.
5. *W. A. Prebble* Digital Calculation of Sequence Networks Including Mutual Impedances, ibid., (to be published).

Matrices or Tensors?*

Since there are *thousands* of Journals that publish papers on the application of matrices, but only *one* Quarterly that publishes on the application of tensors, it strikes me that the Tensor Quarterly should give preference to tensorial papers. As a matter of fact serious thought should be given to a change in the title to 'Tensor Quarterly' instead of the present 'Matrix and Tensor Quarterly'.

* Editor's Note: Matrices can exist without tensors whilst tensors cannot exist without matrices. This summarizes the title and policy of the Quarterly.

S. 3.

Reprint

of Paper published in

THE ELECTRICAL ENGINEERING TRANSACTIONS

OF

THE INSTITUTION OF ENGINEERS, AUSTRALIA

Digital Calculation of Short-Circuit Networks

By W. A. Prebble, B.Sc., B.E.
(*Associate Member*)*

Summary.—It is shown how a power system network is coded by means of node numbers and lists of self and mutual impedances assembled for input to the computer. After reading, sorting and reorganising these data, the computer calculates the bus impedance matrix for any sequence network by starting with the reference bus and establishing, one by one, the network busses. The calculation processes, according to simple rules, one element or one group of mutually coupled elements at a time, until all the listed elements are used.

The output consists of the sequence impedance for every bus in the sequence network solved plus current distribution factors in selected line and generator elements. When a network is too large for solution with the available computer storage, it is divided into sub-networks and information is printed out to derive their equivalent circuits. These are then combined in the appropriate way to obtain the required information about the original network.

The mathematical rules for calculating line flows and equivalent circuits are summarised in the Appendix together with information on computer running times.

A bibliography relating to digital computer programmes for fault studies is included.

List of Principal Symbols.

C_g = Connection matrix for group of coupled lines.
C_{gt} = Transpose of C_g.
Z_m = Mutual impedance matrix for group of coupled lines.
Z_{sm} = Self and mutual impedance matrix for group of coupled lines.
Y_{sm} = Inverse of Z_{sm}.
Z_l = Self impedance of network element l.
Z_{mr} = Mutual impedance between two lines.
Z_{ij} = Bus impedance matrix term in row i, column j.
V_i = Voltage of node i above reference.
I_{ij} = Current in line connecting nodes i and j.

In the flow diagrams:

N = Number of line and generator elements.
M = Number of mutual impedances.
R, G = Number of lines in a group, group number.
J, K, L, P Line elements.
D, E = Busses of line J.
I = A counting integer.

1.—Introduction.

During the past few years a number of papers have been published on the application of digital computers to power system short-circuit problems. The methods derived for solving these problems can be divided into two principal groups:

(i) those which use an iterative procedure (Refs. 9 to 13), and
(ii) those in which an impedance matrix is calculated directly.

Method (i) has the disadvantage that an iterative solution of the complete network for each fault condition is required. Method (ii) can be sub-divided into:

(*a*) the mesh or loop equation approach (Refs. 15 to 18), and
(*b*) the nodal approach (Refs. 19 to 26).

The mesh method requires a matrix inversion and is more difficult to code than the nodal method. The computer programme described in this paper is based on the nodal method which has been developed in recent years and described in other publications (Refs. 8, 20 and 26). In this programme the computer calculates the transfer and driving point impedance matrix, referred to more simply as the bus impedance matrix directly from lists of sequence impedances identified by node numbers by making simple logical decisions followed by arithmetic operations.

*Paper No. 2039, presented before the Engineering Conference, 1966, in Newcastle from 21st to 25th March, 1965.

The author is a Section Leader in the Electrical Investigations Division, The Hydro-Electric Commission, Tasmania.

The programme is written in the Algol language and all input and output to the computer are in the form of paper tapes. All data must be in the form of numbers except for strings which are input and output without being operated on by the computer and are used for headings. The nodal method is particularly suited to this type of data input because the network can be simply and easily coded with numbers. It has the added advantage that the data are kept to a minimum and are easily arranged in tables.

All calculations during the running of the programme are with per unit values of impedance, etc., provision being made for the selection of any convenient mVA base; the voltage base is the nominal kV of the various sections of the system. As the bus impedance matrix is symmetrical, only the lower triangular part is stored during the calculation.

The computer programme is considered under the three parts into which it naturally divides:

(1) the reading and sorting of data,
(2) the calculation of the bus impedance matrix, and
(3) the output.

2.—Reading and Sorting Data.

An essential preliminary to the calculation of the bus impedance matrix is the sorting and re-organising of the input data. By making the computer do this, the rules for coding the network and listing the data are reduced to a minimum. The generator or source bus is numbered zero and the remaining busses or nodes distinguished by positive integers which need not be consecutive although the omission of any number leads to the non-utilisation of allocated storage. In addition to the actual system busses, dummy busses may be placed at any desired location, e.g., along a line element.

Fig. 1 is a diagram showing the coding of a small zero sequence network with mutual impedances; Tables I, II and III list the data as presented to the computer for solving this network. Note that only the figures are read by the computer during the data input, the remaining letters, etc., punched on the tape are ignored.

2.1 *Basic Data:*

This consists of an ordered set of positive integers as shown in Table I for the network of Fig. 1. The first number is used in converting any resistance and reactance values listed in ohms in Tables II and III to the per unit values required for the computer calculations. The second and third numbers tell the computer how many items of data are in Tables II and III and, hence, the amount of storage to allocate; if the third number is zero, then there

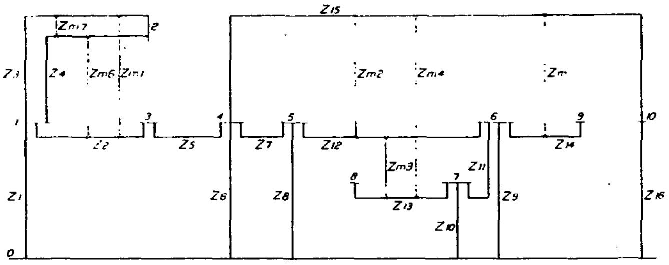


Fig. 1.—Partial System Zero Sequence Network.

are no mutual couplings and Table III is omitted. The highest bus number is required to allocate storage for the bus impedance matrix. The last number in Table I controls the arithmetic operations in the programme; if zero, the resistance columns in Tables II and III are omitted and the working is with real numbers; if greater than zero the working is with complex numbers. This arises because in the past sequence calculations have been done using reactances only; thus the required resistance values are not always available.

TABLE I.

Basic Data.

mVA base	100
Number of line and generator elements	16
Number of mutual impedances	7
Highest bus number	10
Number of resistances	8

2.2 Line and Generator Data:

The line and generator elements are listed in any order, each being distinguished by the numbers of the busses which it connects. Table II gives this data list for the network of Fig. 1. The numbering of each element of the list is used to check the total number of elements and to distinguish those which are mutually coupled to other elements. The column for voltage allows the resistance and reactance to be listed in ohms or in per unit by entering the nominal voltage in kV or the integer one respectively. As the data for each element are read, the following operations are performed.—

(1) the bus connection numbers are stored in order, the lower number first, and

(2) if necessary, the resistance and reactance are converted to per unit using the kV voltage and mVA base, then the per unit values are stored.

This does not slow down the reading process as the computer makes the decisions and performs the calculations much faster than the numbers are read.

In general, line elements are entered in the table with resistance and reactance in ohms while generator and transformer elements are entered with per unit reactances which may be negative for a transformer represented by an equivalent circuit.

2.3 Mutual Coupling Data:

At present the programme allows for mutual coupling between any network elements except generator elements, i.e., those connected to the reference bus. These may be listed in any order as shown in Table III, each mutual impedance being distinguished by the two line numbers from the line list. Each impedance is numbered as a check on the total number of entries. As in the line and generator list, there are columns for line voltage thus permitting the impedance to be entered in ohms when the line voltage is entered in kV or in per unit when the integer one is entered under voltage.

TABLE II.

Line and Generator List.

Element	Bus connections		Resistance	Reactance	Voltage
1	1	0	0.0	0.198	1
2	1	3	15.42	94.63	220
3	2	1	24.53	82.59	110
4	2	1	24.53	82.59	110
5	3	4	15.42	94.63	220
6	4	0	0.0	0.052	1
7	4	5	0.0	0.085	1
8	5	0	0.0	0.085	1
9	6	0	0.0	0.108	1
10	7	0	0.0	5.353	1
11	6	7	0.0	−0.603	1
12	5	6	10.47	57.28	110
13	7	8	9.67	89.98	88
14	6	9	12.27	55.10	110
15	4	10	16.67	93.96	220
16	10	0	0.0	0.15	1

TABLE III.

Mutual Impedance List.

Mutual	Element	Voltage	Element	Voltage	Resistance	Reactance
1	2	220	3	110	4.40	16.39
2	12	110	15	220	6.69	28.80
3	12	110	13	88	−3.24	−16.21
4	13	88	15	220	−3.24	−12.13
5	15	220	14	110	6.81	27.59
6	2	220	4	110	4.40	16.63
7	3	110	4	110	8.21	51.89

The sign of the mutual impedance depends on the network geometry and is found by examining the direction along the lines from the higher to the lower numbered busses in the network diagram. If these directions in two mutually coupled lines are the same, then the mutual impedance is listed as positive; if these directions are opposite then the mutual impedance is negative; e.g., if in Fig. 1 busses 7 and 8 are interchanged then the mutual couplings 3 and 4 in Table III would be positive.

During the reading of the data, two operations are performed for each mutual impedance.—

(1) the two line numbers are sorted into order and stored temporarily with the lower number first, and

(2) after conversion to per unit, if necessary, the impedance value is placed in temporary storage.

After this preliminary working the data are not in a form which is suitable for use in the programme and hence the data must be re-arranged. The basic flow diagram by which the computer carries out this re-arrangement is given in Fig. 2. Starting with the first element from the line and generator list, the mutual line list is searched for coupling with this element; if no coupling is found then the process is repeated with the second element, and so on until a pair of mutually coupled lines are located. These two lines commence a group and the mutual line list is searched until all the other lines directly, or indirectly, coupled to these two lines are found and listed in the group. If all the mutually coupled lines are grouped the calculation proceeds to the next part of the programme, otherwise the search continues for another group of coupled lines.

During the sorting, the lower triangular matrix of mutual impedance values for each group is derived and these are stored one after the other in linear arrays. For reference during subsequent calculations it is necessary to store the number of lines in each group and the total number of groups, in addition to the locations of each group of lines and impedance values.

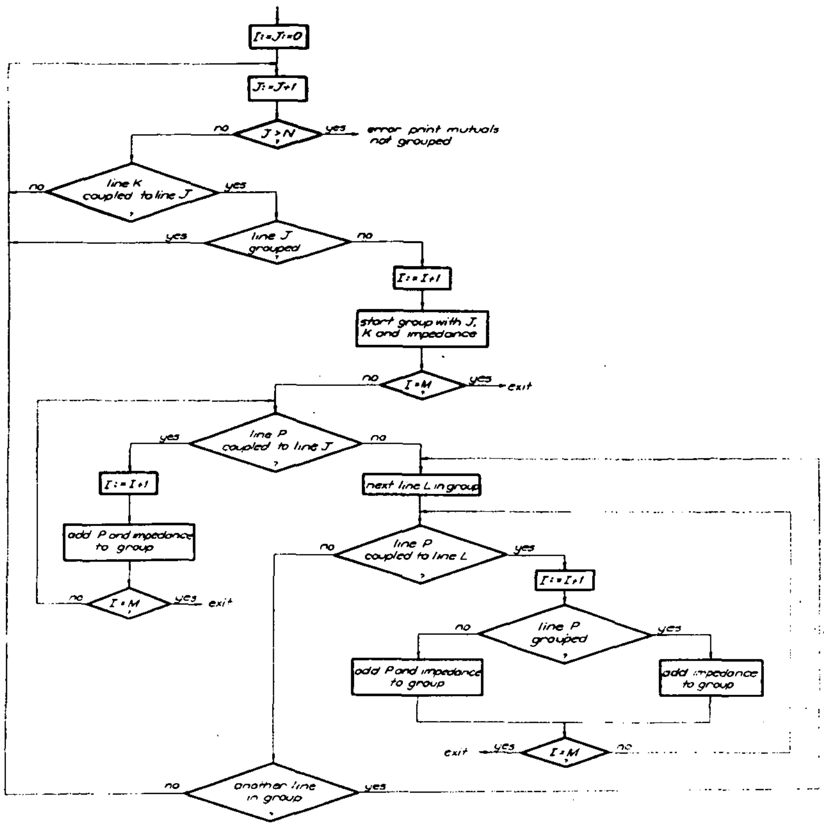


Fig. 2.—Basic Flow Diagram for Sorting the Mutually Coupled Lines into Groups.

3.—Calculation of the Bus Impedance Matrix.

The first step in the calculation of the bus impedance matrix is the setting of the controls which direct the working of the programme. These consist of:

(1) a used element indicator in which each element is marked when the calculation adding it to the matrix is finished;

(2) an established bus indicator in which each bus is marked as the calculation establishes it in the matrix;

(3) the mutual element indicator has been set up during the procedure of grouping the mutual impedances. This indicates which elements have mutual coupling and into which group the element has been sorted; and

(4) an integer variable which has, as the calculation proceeds, the value of the highest established bus thus controlling the use, when working with elements referred to below as type (*b*), of the space allotted for the storage of the bus impedance matrix.

The bus impedance matrix is calculated by establishing the network busses, one at a time, and closing loops as they are formed. The mathematical rules for these operations have been covered in the literature, e.g., Refs. 8, 20, 22, 23, 24, 25 and 26; the explanations below refer to the following matrix:

$$\begin{array}{cc} & \begin{array}{ccccccccccc} & & i & & & & & & k & & \end{array} \\ \begin{array}{c} i \\ \\ j \\ \\ k \end{array} & \left[\begin{array}{ccccccccccc} . & Z_{ii} & . & . & . & Z_{ij} & . & . & . & Z_{ik} & . \\ . & . & . & . & . & . & . & . & . & . & . \\ . & Z_{ji} & . & . & . & Z_{jj} & . & . & . & Z_{jk} & . \\ . & . & . & . & . & . & . & . & . & . & . \\ . & Z_{ki} & . & . & . & Z_{kj} & . & . & . & Z_{kk} & . \end{array}\right] \end{array} \quad \text{.........(1)}$$

For clarity the complete matrix is shown, but, in practice, for sequence networks it is symmetrical and only the lower triangular part is stored by the computer.

In the calculation of the bus impedance matrix, outlined in the basic flow diagram, Fig. 3, two types of element are recognised:

(*a*) a branch element, i.e., one from an established bus and therefore establishing a new bus, and

(*b*) a loop element, i.e., one connecting two established busses.

As the network elements are listed in any order and, therefore, are not necessarily processed in this order, the basis for the programme is repeated searches of the list for elements of type (*a*) or (*b*) until all the elements are used and the calculation of the impedance matrix is complete. The computer selects the first unused element l from the list and places it in one of the following three categories.—

(i) *A generator element*, i.e., one connected to the reference bus. If this is a type (*a*) element establishing a new bus k then a new row and column k are formed in the impedance matrix having off-diagonal terms zero and diagonal term equal to the self impedance of the element.

$$Z_{ki} = Z_{kj} = 0 \quad \text{..................................(2)}$$

$$Z_{kk} = Z_l \quad \text{..............................(3)}$$

Initially the only bus established is the reference bus, hence the first network element processed will be in this category.

For a type (*b*) element connected from the reference to the established bus i then a dummy row and column k are formed with the off-diagonal terms equal to the corresponding terms of the row and column i and the diagonal term equal to the element self-impedance plus the diagonal term of row i.

$$Z_{ki} = Z_{ii}; \quad Z_{kj} = Z_{ij} \quad \text{........................(4)}$$

$$Z_{kk} = Z_{ii} + Z_l \quad \text{............................(5)}$$

The dummy row and column k are eliminated by replacing each term Z_{ij} of the matrix by:

$$Z_{ij} - \frac{Z_{ik}Z_{kj}}{Z_{kk}} \quad \text{..................................(6)}$$

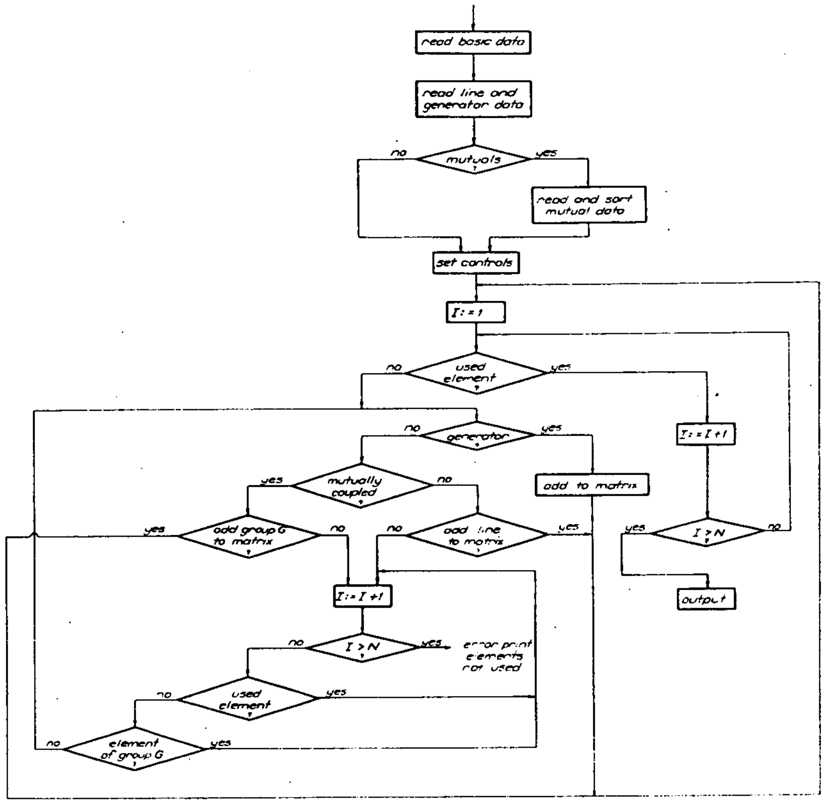


Fig. 3.—Basic Flow Diagram for Calculating the Bus Impedance Matrix.

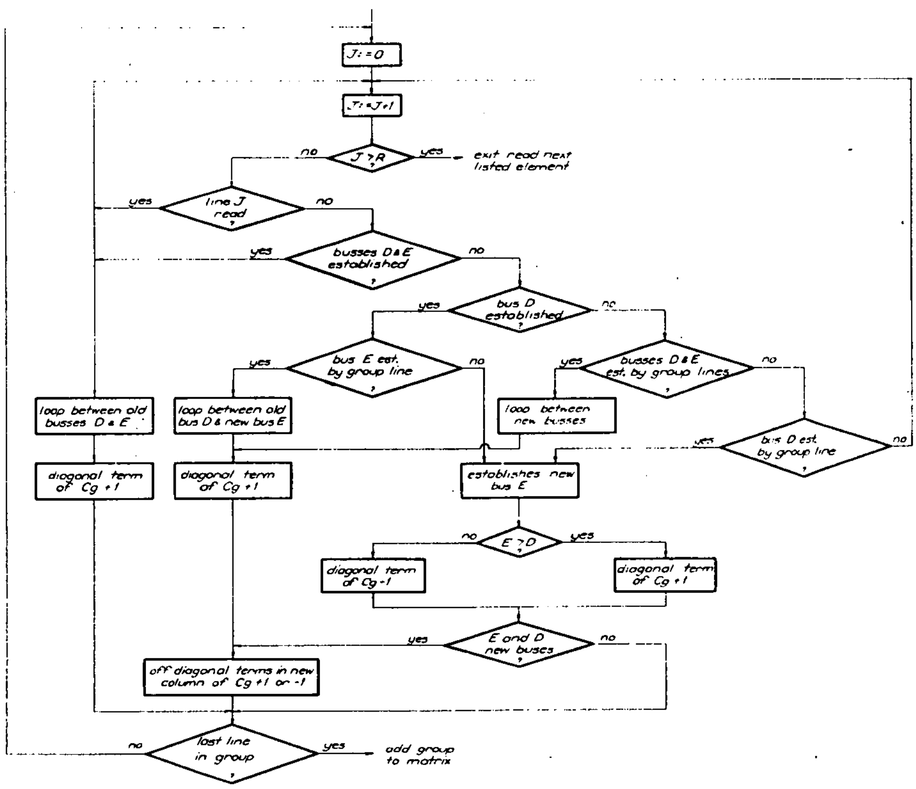


Fig. 4.—Basic Flow Diagram Determining if a Mutually Coupled Group of Lines can be Processed.

(ii) *A line element without mutual coupling.* If neither bus of this element is established the programme proceeds to the next unused element in the line list. For a line from an established bus i to a new bus k then a new row and column k are formed with the off-diagonal terms equal to the corresponding terms of row and column i and the diagonal term equal to the self impedance of the element plus the diagonal term of row i.

$$Z_{ki} = Z_{ii}; \quad Z_{kj} = Z_{ij} \quad \text{.......................(7)}$$
$$Z_{kk} = Z_{ii} + Z_l \quad \text{..............................(8)}$$

When the line is of type (*b*), connecting the established busses i and j, then a dummy row and column k are formed in the impedance matrix with the off-diagonal terms equal to the difference between the corresponding terms in the rows and columns i and j while its diagonal term is the self impedance of the element plus the difference between the i and j terms of row k.

$$Z_{ki} = Z_{ii} - Z_{ji}; \quad Z_{kj} = Z_{ij} - Z_{jj} \quad \text{...............(9)}$$
$$Z_{kk} = Z_{ki} - Z_{kj} + Z_l \quad \text{........................(10)}$$

The dummy row and column k are eliminated by the rule of Eq. (6).

(iii) *A line with mutual coupling,* in which case a more complicated procedure is entered because all elements of the group, i.e., all elements either directly or indirectly coupled to this element, must be processed together. The basic flow diagram for this procedure is set out in Fig. 4 and this does the following:

(1) determines whether all elements of the group can be processed at this stage, the order in which they are to be processed (those which establish new busses first) and whether they establish new busses or loops; and

(2) derives the group impedance matrix C_g which has $+1$ or -1 in the main diagonal, zero for all terms below the main diagonal and depending on the bus numbers of the elements and the way they are connected together and to the partial established network, zero, $+1$ or -1 for terms above the main diagonal.

The elements of the group are separated into two lists, those which establish new busses and those which form loops. The programme starts by testing the group elements to find the first one which can be added to the bus impedance matrix and putting it into one of the two lists. The elements are then searched again to find the next element of the group which can be added to the bus impedance matrix assuming that the first element has been added. The procedure is then repeated until it has been determined in what order the elements of the group are to be added to the bus impedance matrix or that they cannot be added in any possible way. In the latter case, the programme proceeds to the next element in the list which is not in the group just tested.

On completion of the above two steps the elements are processed in the order determined and the group mutual impedance matrix Z_m formed. This latter operation is simply a re-arrangement of the group mutual impedances which are not necessarily stored in the order in which the line elements have been processed. The matrix product $C_{gt}Z_mC_g$ is then derived and the values from this product added into their respective terms of the bus impedance matrix. Finally, any loops formed by the group are closed one at a time using Eq. (6). In the Appendix, this process is outlined with reference to the network of Fig. 1.

4.—Output.

Because of the present storage limitations on the computer, the storing of the complete programme leaves insufficient working space for solving system problems. It has therefore been divided into two separate programmes, the first of which ends with the determination of the bus impedance matrix and the output on paper tape of the terms of the matrix plus all the sorted self and mutual impedance data. The input to the second programme consists of the output from the first programme plus the numbers of specified elements for which line flows are wanted and specified busses if an equivalent network is required.

The second programme firstly inverts the self and mutual impedance matrix, Z_{sm}, of each group of coupled line elements. The inversion is by the method of pivotal condensation, the successive pivots being the main diagonal terms (Refs. 5 and 14). This

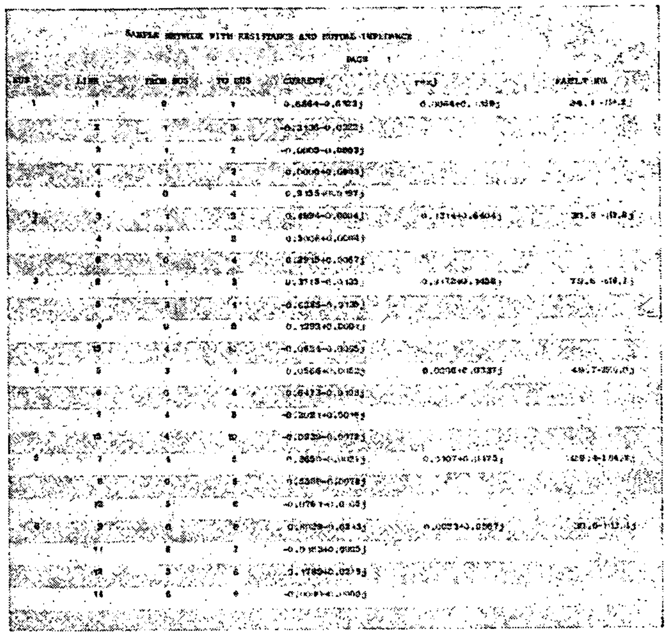


Fig. 5.—Part Print-up of Results for the Network, Fig. 1.

gives an inverse to the required degree of accuracy because, in general, the self impedance diagonal terms have a greater modulus than the off-diagonal mutual impedance terms. The symmetry of the matrices is utilised to save space and operations during the inversion process. The lower triangular parts of the resulting admittance matrices, Y_{sm}, are then stored one after the other as calculated and called when required for the calculation of the current distribution factors.

Taking the busses one at a time, in the order of their numbers, a list of all the elements connected to the bus under consideration plus any other specified elements is set up. The per unit sequence impedance of the system to the bus and the mVA base divided by this impedance are printed. The latter represents the three-phase fault power at the bus when the positive phase sequence network is solved. The magnitude and direction of the current distribution factors (considering unit current flowing out of the bus and in at the reference bus, no current into or out of any other bus) are then calculated and printed out for the listed elements. The rules for these calculations are given in Eqs. (14), (16) and (17). The output is punched on paper tape by the computer and is then printed on a flexowriter; part of the flexowriter print-up for the network of Fig. 1 is shown in Fig. 5.

If required the terms of the partial bus impedance matrix for selected network busses are also printed and can be used to derive an equivalent circuit for the network. On inverting this matrix, the resulting off-diagonal terms are minus the admittances of lines between busses and the sum of the terms in a row or a column is the admittance of an element connected to the reference bus.

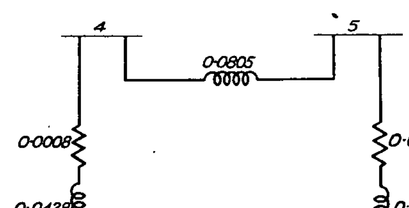


Fig. 6.—Equivalent Circuit for the Network, Fig. 1.
Impedance values in per unit.

Fig. 6 is the equivalent circuit for the network of Fig. 1 retaining busses 4 and 5. As the partial bus impedance matrix represents the effect of the complete network at the selected busses, it can be used in the calculation of the matrix for a second network connected to the first network at one or more of the selected busses. The calculation starts with this matrix, the final bus impedance matrix being that for the second network including the effect of the first network.

5.—Programme Running.

The computer running time for solving a network depends on a number of factors such as the presence of resistance and mutual coupling, the coding of the network, and the order in which the line and generator elements are listed. While the programme solves a problem coded in any way with the self and mutual impedances listed in any order, it is more efficient if the node numbers follow regularly round the network starting at a generator element and the elements with the lowest node numbers listed first. It is also better from the aspect of programme running time, if the line elements with mutual couplings are listed in such a way that all those in a group can be processed as soon as one element of the group is read. The actual order in which the mutual impedances are listed is not significant as these are sorted by the computer in any case; the speed of sorting is improved slightly by listing line elements with mutual coupling early in the line and generator list.

By varying the network coding and data listing, the network elements are processed in a different order which may cause a difference in the fourth significant figure of the resulting impedances and distribution factors. As the data are not known to any higher degree of accuracy, there is no advantage in varying the coding or listing for the purpose of obtaining more accurate results.

6.—Conclusion.

The direct calculation of the bus impedance matrix by the computer leads to simple rules for coding the network with node numbers. By making the computer organise and sort the data the input is kept to easily understood lists of self and mutual impedances which can be assembled with little experience and a minimum chance of error.

The computer automatically calculates the bus impedance matrix by simple arithmetical operations and logical decisions based on the node numbers and finally calculates impedances and line flows for all busses of the sequence network. At present the sequence networks are solved independently, but the programme could be extended to combine these networks to determine line-to-line, line-to-ground faults, etc., and resulting power flows anywhere on the system.

Acknowledgments.

The author acknowledges the permission of the Hydro-Electric Commissioner, Mr. A. W. Knight, C.M.G., B.Sc., M.E., B.Com., M.I.E.Aust., to publish this paper. He is also indebted to Mr. R. W. Wadsley, B.E., M.I.E.Aust., Deputy Chief Electrical Engineer, for helpful criticism of the subject-matter.

References.

The papers referring to short-circuit problems have been divided into sections and include some on load flow analysis where these contain relevant material. Also included are some books and papers on matrix methods which are used in electrical network analysis.

General:

1. GLIMN, A. F., HABERMANN, R., HENDERSON, J. M. and KIRCHMAYER, L. K.—Digital Calculation of Network Impedances. *Trans. A.I.E.E. Pt. III*, Vol. 74, Dec., 1955, p. 1285, (Paper No. 55-680).
2. HALE, H. W. and WARD, J. B.—Digital Computation of Driving Point and Transfer Impedances. *Trans. A.I.E.E. Pt. III*, Vol. 76, Aug., 1957, p. 476, (Paper No. 57-146).
3. BYERLY, R. T., LONG, R. W. and KING, C. W.—Logic for Applying Topological Methods to Electric Networks. *Trans. A.I.E.E. Pt. I*, Vol. 77, Nov., 1958, p. 657, (Paper No. 58-826).
4. CONVERTI, V.—Application of Row-by-Row Matrix Inversion to Power System Problems. *Trans. A.I.E.E. Pt. III-A*, Vol. 78, June, 1959, p. 413, (Paper No. 59-99).
5. SHIPLEY, R. B. and COLEMAN, D.—A New Direct Matrix Inversion Method. *Trans. A.I.E.E. Pt. I*, Vol. 78, Nov., 1959, p. 568, (Paper No. 59-502).
6. SATO, N.—Digital Calculation of Network Inverse and Mesh Transformation Matrices. *Trans. A.I.E.E. Pt. III*, Vol. 79, Oct., 1960, p. 719, (Paper No. 60-780).
7. HALE, H. W.—A Logic for Identifying the Trees of A Graph. *A.I.E.E. Power Apparatus and Systems*, No. 54, June, 1961, p. 195, (Paper No. 61-54).
8. PREBBLE, W. A.—The Calculation of the Transfer and Driving Point Impedance Matrix by Digital Computer. *Matrix and Tensor Quarterly*, Vol. 15, June, 1965, p. 126.

Nodal Iterative:

9. COOMBE, L. W. and LEWIS, D. G.—Digital Calculation of Short-Circuit Currents in Large Complex-Impedance Networks. *Trans. A.I.E.E. Pt. III*, Vol. 75, 1956, (Feb., 1957), p. 1394, (Paper No. 56-920).
10. SIEGEL, J. C. and BILLS, G. W.—Nodal Representation of Large Complex-Element Networks Including Mutual Reactances. *Trans. A.I.E.E. Pt. III*, Vol. 77, 1958, (Feb., 1959), p. 1226, (Paper No. 58-1023).
11. TAYLOR, G. E., CARTER, G. K. and MACDONALD, E. H.—New Digital Computer Short Circuit Programme for Relay Studies. *Trans. A.I.E.E. Pt. III*, Vol. 79, 1960, (Feb., 1961), p. 1257, (Paper No. 60-1388).
12. SHIPLEY, R. B., COLEMAN, D. and NASON, W. L.—Short-Circuit Comparison Study. *A.I.E.E. Power Apparatus and Systems*, No. 58, Feb., 1962, p. 1162, (Paper No. 61-932).
13. JOHN, M. N.—A General Method of Digital Network Analysis Particularly Suitable for Use with Low-Speed Computers. *Proc. I.E.E. Pt. A*, Vol. 108, Oct., 1961, p. 369, (Paper No. 3259S).
14. BRAMELLER, A. and DENMEAD, J. K.—Some Improved Methods for Digital Network Analysis. *Proc. I.E.E. Pt. A*, Vol. 109, Feb., 1962, p. 109, (Paper No. 3778S).

Loop:

15. LANTZ, M. J.—Digital Short-Circuit Solution of Power System Networks including Mutual Impedance. *Trans. A.I.E.E. Pt. III*, Vol. 76, 1957, (Feb., 1958), p. 1230, (Paper No. 57-796).
16. BYERLY, R. T., LONG, R. W., BALDWIN, C. J. and KING, C. W.—Digital Calculation of Power System Networks under Faulted Conditions. *Trans. A.I.E.E. Pt. III*, Vol. 77, 1958, (Feb., 1959), p. 1296, (Paper No. 58-1099).
17. TINNEY, W. F. and MCINTYRE, C. M.—A Digital Method for Obtaining a Loop Connection Matrix. *Trans. A.I.E.E. Pt. III*, Vol. 79, Oct., 1960, p. 740, (Paper No. 60-781).
18. RINDT, L. J. and LONG, R. W.—New Short Circuit Provides Many Automatic Features. *I.E.E.E. Trans. on Power Apparatus and Systems, Special Supplement*, 1963, p. 740, (Paper No. T-151).

Nodal:

19. TOALSTON, A. L.—Digital Solution of Short-Circuit Currents for Networks Including Mutual Impedances. *Trans. A.I.E.E. Pt. III-B*, Vol. 78, 1959, (Feb., 1960), p. 1720, (Paper No. 59-1164).
20. EL-ABIAD, A. H.—Digital Calculation of Line-to-Ground Short Circuits by Matrix Method. *Trans. A.I.E.E. Pt. III*, Vol. 79, June, 1960, p. 323, (Paper No. 60-180).
21. BROWN, H. E. and PERSON, C. E.—Digital Calculation of Single-Phase-to-Ground Faults. *Trans. A.I.E.E. Pt. III*, Vol. 79, Oct., 1960, p. 657, (Paper No. 60-255).
22. BROWN, H. E., PERSON, C. E., KIRCHMAYER, L. K. and STAGG, G. W.—Digital Calculation of 3-Phase Short Circuits by Matrix Method. *Trans. A.I.E.E. Pt. III*, Vol. 79, 1960, (Feb., 1961), p. 1277, (Paper No. 60-181).
23. EL-ABIAD, A. H., GUIDONE, R., and STAGG, G. W.—Calculation of Short Circuits Using a High-Speed Digital Computer. *A.I.E.E. Power Apparatus and Systems*, No. 57, Dec., 1961, p. 702, (Paper No. 61-797).
24. BROWN, H. E., CARTER, G. K., HAPP, H. H. and PERSON, C. E.—Power Flow Solution by Impedance Matrix Iterative Method. *I.E.E.E. Trans. on Power Apparatus and Systems*, No. 65, April, 1963, p. 1, (Paper No. 62-214).
25. LAUGHTON, M. A. and DAVIES, M. W. H.—Numerical Techniques in Solution of Power-System Load-Flow Problems. *Proc. I.E.E.*, Vol. 111, Sept., 1964, p. 1575, (Paper No. 4511P).
26. PREBBLE, W. A.—The Digital Calculation of Sequence Networks Including Mutual Impedances. *Proc. I.E.E.*, Vol. 112, April, 1965, p. 711, (Paper No. 4700P).

Textbooks and Reports:

27. KRON, G.—*Tensor Analysis of Networks.* New York, Wiley, 1939, 635 p.
28. KRON, G.—*Diakoptics—The Piecewise Solution of Large-Scale Systems.* London, Macdonald, 1963, 166 p.
29. GEORGE, E. E.—*Matrix Algebra Handbook for Electrical Engineers.* Ebasco Services Inc., U.S.A., 1958, 234 p.
30. STIGANT, S. A.—*Matrix and Tensor Analysis in Electrical Network Theory.* London, Macdonald, 1964, 505 p.

APPENDIX.

1. Addition of Mutually Coupled Elements to the Bus Impedance Matrix:

This will be discussed with reference to Fig. 1 and Tables II and III. After rearranging, the mutual impedances are held in storage in two groups, in the first of which are lines 2, 3, 4, in that order, and in the second four lines 12, 15, 13, 14. The first element read establishes bus 1, the second element is mutually coupled and the computer determines that the group can be added in the order: element 2 establishing bus 3, element 3 establishing bus 2, and element 4 forming a loop. The connection matrix C_g set up for this group by the computer is:

$$\begin{array}{c} \\ 3-1 \\ 2-1 \\ 2-1 \end{array} \begin{array}{c} \begin{array}{ccc} 3-1 & 2-1 & 2-1 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \end{array} \qquad \ldots\ldots\ldots\ldots\ldots\ldots(11)$$

where the numbers outside the matrix refer to the bus connections of the lines 2, 3 and 4. These line elements are then processed in accordance with Eqs. (7) to (10) forming the matrix:

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} Z_{11} & Z_{12} & Z_{13} & Z_{14} \\ Z_{21} & Z_{22} & Z_{23} & Z_{24} \\ Z_{31} & Z_{32} & Z_{33} & Z_{34} \\ Z_{41} & Z_{42} & Z_{43} & Z_{44} \end{bmatrix} \end{array} \qquad \ldots\ldots\ldots\ldots\ldots\ldots(12)$$

The matrix product $C_{gt}Z_mC_g$ is calculated:

$$\begin{array}{c} \\ 3-1 \\ 2-1 \\ 2-1 \end{array} \begin{array}{c} \begin{array}{ccc} 3-1 & 2-1 & 2-1 \end{array} \\ \begin{bmatrix} 0 & Z_{m1} & -Z_{m1}+Z_{m6} \\ Z_{m1} & 0 & Z_{m7} \\ -Z_{m1}+Z_{m6} & Z_{m7} & -2Z_{m7} \end{bmatrix} \end{array} \qquad \ldots\ldots\ldots(13)$$

and then the result Z_{m1} added to Z_{32}, Z_{23}; $-Z_{m1}+Z_{m6}$ added to Z_{43}, Z_{34}; Z_{m7} added to Z_{42}, Z_{24} and $-2Z_{m7}$ added to Z_{44}. Finally, row and column 4 are eliminated by the rule of Eq. (6) and the calculation proceeds to the next element in the line list.

2. Fault Power and Current Distribution Factors:

From Eq. (1) for a fault on bus i

$$\text{fault power} = \frac{\text{mVA base}}{Z_{ii}} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(14)$$

Consider unit current flowing in at the reference bus and out of the network at bus i, then

$$V_i = Z_{ii}$$
$$V_j = Z_{ji} \text{ etc.} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(15)$$

Hence the current distribution factor for a line $j-k$, of impedance Z_l, with a fault on bus i is:

$$I_{jk} = \frac{Z_{ji}-Z_{ki}}{Z_l} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(16)$$

If the lines $j-k$ and $g-h$ are a mutually coupled group:

$$I_{jk} = Y_{ll}(Z_{ji}-Z_{ki}) + Y_{lp}(Z_{gi}-Z_{hi}) \qquad \ldots\ldots\ldots\ldots(17)$$

where Y_{ll}, Y_{lp} are terms of the admittance matrix obtained by inverting the matrix:

$$Z_{sm} = \begin{array}{c} \\ l \\ p \end{array} \begin{array}{c} \begin{array}{cc} l & p \end{array} \\ \begin{bmatrix} Z_l & Z_{mr} \\ Z_{mr} & Z_p \end{bmatrix} \end{array} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(18)$$

where Z_{mr} is the mutual impedance coupling lines l and p.

3. Equivalent Circuit:

The equivalent circuit retaining busses 4 and 5 in the network of Fig. 1 has the bus impedance matrix:

$$\begin{array}{c} \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{cc} 4 & 5 \end{array} \\ \begin{bmatrix} 0.00056+j0.03366 & 0.00043+j0.01648 \\ 0.00043+j0.01648 & 0.00066+j0.04750 \end{bmatrix} \end{array} \qquad \ldots\ldots(19)$$

which inverted gives the equivalent circuit admittance matrix:

$$\begin{array}{c} \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{cc} 4 & 5 \end{array} \\ \begin{bmatrix} 0.441-j35.778 & -0.002+j12.415 \\ -0.002+j12.415 & 0.242-j25.354 \end{bmatrix} \end{array} \qquad \ldots\ldots\ldots(20)$$

From this is derived the per unit admittance of each element:

$$\begin{array}{ll} 4-0 & 0.439-j23.364 \\ 5-0 & 0.240-j12.939 \\ 4-5 & 0.002+j12.415 \end{array} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(21)$$

and inversion of these gives the per unit impedances of Fig. 6.

4. Computer Running Times:

The times quoted refer to the Elliot 503 computer installed in the Hydro-University computing centre.

Because of the size of the two programmes there is insufficient storage space in the computer to hold the Algol compilers together with the translated programme. Hence, as the programme is compiled a version in machine code is output on paper tape. This has the advantage that the programme does not have to be compiled again and the machine code version is more rapidly translated into the machine than the Algol version. The times for the input of the two machine code versions of the programmes are 25 and 20 sec., respectively, compared with several minutes for translating the original Algol versions.

The following times apply to a 33-bus system with 54 line and generator elements, 47 mutual couplings, but not including any resistance. The time for reading in the data and calculating the bus impedance matrix by the first programme is 30 sec. followed by 2 min. to punch out all the terms of this matrix and the sorted self and mutual impedance data for the second programme. The reading in of the data and the inversion of the mutual impedance matrices by the second programme takes 15 sec. while the calculation of the line flows, etc., and punching out the results takes another 35 sec. If more line flows are required, these can be calculated and punched out at the rate of 4 a second. The overall computer time, allowing for winding tapes, etc., is 5 to 6 min. and the time to print up the results on the flexowriter is 10 min.

If the network has resistance the time to punch out the results of the first programme is doubled while that for punching the results of the second programme and printing on the flexowriter is increased by about 25 per cent. It is seen from these times that not a great deal is to be gained by coding the network and listing the data for optimum calculation time.

Discussion

Mr. J. W. Phillips (Associate Member, Sydney Division).—This paper constitutes a worthy addition to the growing literature of computer application to the problems encountered in the planning, design and operation of electricity supply systems. The author set out to develop a programme for the determination of short-circuit currents using a computer just acquired by his organisation and in doing so adopted the nodal approach whereby, by solving the three sequence networks, he obtains the positive, negative and zero sequence impedances for each busbar. This in turn enables him to determine the short-circuit currents at the busbars and in the various lines connected to these busbars. This method is more sophisticated than, for instance, the nodal iterative approach which is in common use. The author should be complimented on the vast amount of study and the quality and simplicity of the exposition of the subject matter. I would think that his method would serve well for a fault study of a large existing system with a view to deciding on protective settings, but would it be as suitable for the purpose of forward system planning as an adequate A.C. network analyser? As is known the system designer has no control over the study once it is put into the computer, whereas with a network analyser he retains this control throughout the exercise and could change the course of study at any point. How would the time taken in preparing and carrying out a computer study by the proposed method compare with that required for a similar network analyser study ? I also note that the programme is written in the " Algol " language. Could it be readily translated into the popular " Fortran " language ?

Mr. R. K. Edgley (Associate Member, Sydney Division).—I would support Mr. Phillips's comments concerning the relative convenience and utility of computers and network analysers for power-system fault studies. Some years ago I had to assist my Company in deciding which of the two to adopt for this purpose, and although it was realised that a computer could carry out studies much more quickly, and with greater accuracy, than a network analyser, our decision was to build a network analyser. The main reasons for the decision were :—

(*a*) The network analyser could have sufficient accuracy, consistent with that of the available data, for the purpose ;

(*b*) As an analogue device, it enabled the engineers concerned to follow a study closely, and to see and understand what was being done.

The decision was made at a time when commercially available computers were in their infancy, but even though a wide range of versatile computers is now available we still prefer the network analyser for fault and power flow studies. This preference could be conditioned by the fact that many of our studies are carried out for other organisations whose engineers in general are not familiar with computing techniques ; in this our usage differs from the author's.

The author mentioned that the application of his method was complicated by lack of sufficient computer capacity. In this connection, my Company has been using computer methods (for other than power system studies) for some years, although it is only just installing a computer. This was made possible by the existence in U.K. of a countrywide high-speed data-transmission network, which among other things allows users to " talk " to computers, and computers to " talk " to one another by Telex methods.

In a recent paper* by S. Dossing, I note mention of the nucleus of such a network, allowing data transmission at a rate approximately 50 times higher than by the P.M.G. Telex network, already in existence in Australia, although it is operated I believe exclusively for the American National Aeronautics & Space Administration. Has the author considered the possibility of extending his computer capacity by " talking " to another computer, and does he know whether there are any plans to establish in Australia a high-speed data-transmission network, either in conjunction with, or separate from, the N.A.S.A. network ?

Mr. P. J. Hoare (Associate Member, Brisbane Division).—The author is to be commended for a clear explanation of the inclusion of mutuals in the impedance matrix solution of short-circuit studies.

Can the programme allow for loads, differing machine voltages, off-nominal transformer taps and line end faults ? The Northern Electric Authority has developed a Fortran short-circuit programme for an IBM 1620 Computer at the University College of Townsville which can allow for these refinements and for mutuals, although in fact they are seldom used. This procedure has been extended with a second programme to calculate total and selected feeder sequence and phase currents and voltages for earth faults, utilising as input data punched card output of the short-circuit programme. This method is simpler, and considered preferable to one in which fault currents are calculated directly from all sequence impedances since a check can be made on the intermediate results, and very little additional computer time is involved.

Would it be possible to develop the bus matrix to include only busbars on which faults are applied or metered lines connected, and what is the procedure for changing fault location or a feeder outage ?

The use of the bus impedance matrix as described would definitely result in faster calculating time than the nodal iterative method used by the Authority. However, with the latter method, ease of programming and saving in programme storage (for example, the one developed by the Authority has only about 200 Fortran instructions including the refinements mentioned above) may offer advantages with a small, reasonably fast computer and small systems of less than 30-50 busbars.

* Dossing, S.—Data Transmission. *Elec. Engg. Trans. I.E.Aust.*, March, 1966, p. 21.

The Author in Reply :

The author appreciates the comments submitted by the discussers. The question of using an A.C. network analyser or a digital computer for solving power system problems is raised. In the past, the network analyser has been used extensively and, depending on factors such as its availability, the type of problem and the training and preferences of the engineers concerned, it will no doubt continue in use. However, the digital computer is now applied, in an increasing extent, to power system analysis ; if the best is to be obtained from this new tool it should not be compared directly with the analyser. In methods of analysis and computation, the computer has brought about a revolution requiring a new approach to power system problems and to the assembling of accurate data. It is doubtful if, at present, any organisation would buy an analyser because the computer has many applications in addition to electrical network problems.

To Mr. Phillips : The programme is particularly suitable for the extensive, detailed short-circuit calculations necessary in determining circuit breaker ratings and protective relay settings required by the existing power system and for its planned extensions. Because many mutual transformers are required, the representation on the analyser of network sections with several parallel transmission lines can present difficulties which do not arise in the digital computer programme.

The initial problem preparation, i.e., drawing a circuit diagram and listing the self and mutual impedances, is similar for the analyser and computer studies, but with a computer no further work is done by the engineer. The preparation of data tapes, running of the programme on the computer and printing the results takes about a day (irrespective of problem size) depending on the workload. The programme could be written in " Fortran " from the flow diagrams ; a direct translation from the " Algol " probably would not be satisfactory.

To Mr. Edgley : The recent addition of a unit of core backing store (16,384 words) has overcome programme limitations caused by insufficient computer storage space. Now a study of the complete power system is possible without division into sub-networks and deriving equivalent circuits ; in addition, programme running time is reduced because the punching of intermediate results is not necessary. The author believes that high speed links are available from the P.M.G. for transmitting data direct to a computer but, to date, little use has been made of this facility. When the distances involved and the services provided by the airlines are considered, the sending of large amounts of data on magnetic tape could be satisfactory as well as more economical than a data-transmission link.

To Mr. Hoare : The author agrees that the nodal method does lead to some simplifications, e.g., the admittance matrix is easy to calculate, has many zero terms and the inclusion of mutual coupling between lines (a difficult problem with the impedance matrix) is relatively simple. However, an iterative solution of the network is necessary for each fault, whereas in the bus impedance method, the fault mVA and sequence current distributions are automatically calculated, by simple arithmetic operations, for all busbars.

The effect of constant impedance loads can be included by listing an equivalent impedance, and off-nominal taps by the impedances for an equivalent circuit (Ref. D1). Different machine voltages, transformer tap changes, feeder outages and line end faults can be included by the following procedure. The bus impedance matrix for the basic network is calculated and held in storage ; from this a sub-matrix for the buses close to the circuit change is selected and modified (e.g., in the case of a line outage, a new line with impedance equal to minus that of the original line, is added as a loop element to the sub-matrix). From the modified sub-matrix, the effect of this line outage on the selected busbars is easily determined.

A line end fault is then derived by adding the self-impedance of this line to the fault impedance of the busbar, to which it is connected, in the modified sub-matrix. For a line end fault on a line with mutual coupling, the corresponding procedure is to remove the group from the sub-matrix, then add them in again with the line concerned establishing a new bus. However, a better method in this case is to place a dummy bus, in the basic network, at a suitable location on the line and lump the mutual impedances in one section, so that the second section is a line without mutual coupling.

By this process, selecting the sub-matrix for any number of busbars from the network bus impedance matrix, the effect of line outages, tap changes, loads, etc. can be calculated at specified busbars.

Reference.

D1. Clark, E.—*Circuit Analysis of A.C. Power Systems.* New York, Wiley, 1943 and 1950, Vol. 1, 540 p., Vol. 2, 396 p.

S.4

Reprint

of Paper published in

THE ELECTRICAL ENGINEERING TRANSACTIONS

OF

THE INSTITUTION OF ENGINEERS, AUSTRALIA

The Digital Solution of the Load Flow Problem by Elimination

BY W. A. PREBBLE, B.Sc., B.E.
(*Associate Member*)*

Summary.—This paper first sets out the requirements of a digital programme for solving the load flow problem and briefly describes, with its limitations, an iterative method.

Then a programme, based on the solution by elimination of a set of linear equations for the voltage corrections at each busbar, is described. This method has advantages in that all practical load flow problems are solved in three or four elimination cycles, there is no difficulty solving problems which fail to converge using the iterative technique, and the resulting solution approaches the specified conditions at all busbars with the same degree of accuracy.

As the voltage correction equations differ from cycle to cycle, a reduction in the number of arithmetic operations and storage required by the computer is achieved by establishing, for the elimination process, the order of column pivoting. From an examination of network connections, it is shown that this is achieved by the selection for pivoting, at each stage in the calculation, of the column with the least number of non-zero terms; when a number of columns satisfy this criterion simultaneously, then the first such column is selected.

A description of the computer programme is illustrated by the solution of a small power system network.

The Appendices include the derivation of the voltage correction equations, their solution by column pivoting and arrangement to limit round-off errors, the calculated busbar voltage corrections for the small network and problem solving times on the computer.

List of Principal Symbols.

V $= e + jf =$ node voltage.
I $= a + jb =$ node current.
Y $= G + jB =$ admittance.
Z $=$ impedance.
S $= P + jQ =$ complex power.
M $=$ voltage correction matrix.
N $=$ number of network busbars.
$U, W, H, T =$ sub-matrixes of matrix M.
Matrixes are identified in the text by [ ].
Subscripts identify particular martix elements and a primed element denotes a specified quantity.
* $=$ complex conjugate.
Δ $=$ incremental value.
$\sum_m$ = summation from $m = 1$ to $m = N$.
$|\ |$ = modulus of a complex quantity.

1.—Introduction.

In planning extensions to an existing network, many load flow studies are necessary for determining the most economical transmission line arrangement to transmit power from the generators to the loads under all anticipated operating conditions. The transmission lines are required to carry the power under all conditions of generation and load, maintaining busbar voltages within specified limits even when some lines are out of service. Also required from these studies are the tapping ranges of transformers or regulators, and the size and location of any capacitors or reactors necessary to ensure an acceptable flow of reactive power in the network.

In the past, a network analyser was used exclusively by the H.E.C. for load flow studies. With the growth of the system, the capacity of the analyser has been exceeded and the digital computer is now used for solving the larger network problems. The computer also has advantages (e.g., the production of a complete printed record of the study, data tapes can be kept and easily altered for repeating, at a later date, studies with changed system conditions); in addition, features such as regulator tap changing, load variations and line switching can be included in the programme.

With the installation of the Hydro-University digital computer in 1964, system load flow studies commenced using an iterative procedure for solving problems (Ref. 1). This method was chosen because of its successful use by other authorities (Ref. 2), it is easily understood and programme writing is straightforward. However, after about a year unsatisfactory features of this programme had become apparent (e.g., some problems took too long to solve while others did not converge to the required tolerance) therefore, a programme using the elimination method was developed.

2.—Formulation of the Problem.

Although some of the early digital load flow studies used mesh equations to describe the relation between the voltages and currents in a transmission network, it is now accepted that the nodal equations are superior. For a particular node k, the nodal equation is

$$I_k = \sum_m Y_{km} V_m \quad \text{.............................(1)}$$

while the system nodal equations are

$$[I] = [Y][V] \quad \text{.............................(2)}$$

or, their inverse

$$[V] = [Z][I].$$

With these equations the network can be simply coded; each busbar is identified by a number and each transmission line, transformer or regulator by the two numbers of the busbars which it connects. Any shunt capacitors or reactors are identified by a busbar number.

For the solution of the problem sets of conditions are specified at each busbar which, therefore, can be classified into one of three types.—

(1) the slack or floating busbar, k, at which the voltage is specified, i.e., $e_k =$ voltage magnitude and $f_k = 0$. There is one busbar, usually a generator, of this type required in the network because the transmission losses have to be found and hence the total generation is unknown;
(2) voltage regulated or generator busbars at which the voltage magnitude and the real power are specified; and
(3) unregulated or load busbars at which the real and reactive power are specified.

In addition, the solution may be required to satisfy restraints such as a limit on the reactive power generated at a type (2) busbar, or limits on the voltage of a busbar controlled by a tap changing transformer.

*Paper No. 2293, presented at The Institution's Power Systems Conference, held in Melbourne on 15th and 16th August, 1967.

The author is a Section Leader in the Electrical Investigations Division, The Hydro-Electric Commission, Tasmania.

3.—Iterative Solution.

For solving the problem, the H.E.C.'s initial programme was based on the Newton-Raphson iterative technique developed by Ward and Hale (Ref. 1) using Eq. (1) and rectangular voltage co-ordinates, i.e.,

$$V_k = e_k + jf_k \qquad \text{.....(3)}$$

This method has advantages in that the network admittance matrix is

(1) easy to derive (Refs. 1 and 2) which leads to a simple computer programme for its construction from a list of line impedances and connections ; and
(2) for power systems, mainly empty ; hence computer space and time are saved by storing and operating only on the non-zero terms.

In each iteration voltage corrections at each busbar, except the slack busbar, are computed sequentially with, at this point in the calculation, the voltages at all the other busbars considered correct. The iterations are repeated until the voltage corrections at all busbars are less than a specified tolerance.

Alternatively, the calculation is stopped if this tolerance is not reached in a prescribed number of iterations. Thus the solution is, at best, an approximation, the tolerance determining the degree of accuracy attained. The rate of convergence depends on several factors, e.g., the number of busbars, the choice of slack busbar and the presence or absence of radial lines remote from the slack busbar are some of the more important. The number of iterations required for convergence to a solution is considerably reduced, if, for a given number of iterations depending on the size of the network, both components of the calculated voltage correction are multiplied by a linear acceleration factor (Ref. 3).

Using a linear acceleration factor of 1.6 and reliable estimates, obtained from system operating conditions and previous studies, for the specified busbar powers and voltages, load flow problems with as many as 44 busbars have converged to a tolerance of 0.00005 per unit voltage in 90-100 iterations. However, some problems (including ones with fewer busbars) require 200-300 iterations even with an increased tolerance ; while others apparently do not converge at all. This applies aprticularly with networks representing system emergency operating conditions such as line outages, especially when the resulting network has a radial type configuration.

4.—More Direct Solution.

From an examination of the various methods (Ref. 3) of solving the load flow problem it appeared that a direct method would result in the solution of problems that have proved difficult and, apparently, impossible to solve by the iterative technique.

These methods may be classified into three groups.—

(1) using the admittance matrix (Ref. 4) ;
(2) using the impedance matrix (Refs. 5 and 6) ; and
(3) using a hybrid matrix, i.e., a matrix with mixed impedance and admittance elements, which is really a combination of direct and iterative methods (Ref. 7).

The main disadvantage of methods (2) and (3) is that the impedance matrix has, in general, no zero terms, therefore the whole matrix must be stored for use in the calculations. Also more calculation is involved in setting up the impedance matrix than for the admittance matrix ; in Refs. 5, 6 and 7 a matrix inversion is used but, this can be avoided (Ref. 8) by assembling directly from the network line impedance and connection list. These disadvantages are not so important on a modern computer with a large fast access store.

In method (1) a voltage, usually $1 + j0$, is assumed for all busbars, then $2(N - 1)$ linear equations for the voltage corrections, i.e., the difference between the assumed and the true voltages at all except the slack busbar, are set up and solved (Appendix I). The corrections so obtained are applied to the busbar voltages but, because the true equations are non-linear, the result is an approximation to the true answer, therefore the cycle is repeated until all corrections are less than a specified limit.

As the existing digital load flow programme used the admittance matrix, it was decided to test method (1) by copying this programme but replacing the Newton-Raphson iterative procedures with ones for setting up and solving, by elimination (Appendix II), the voltage correction equations. So that the process could be examined in detail, facilities for printing the equations and the resulting voltage corrections after each cycle were also included.

This programme was tried with several small networks (up to 20 busbars) ; in particular ones that had proved difficult to solve by the iterative technique. The main features of the results obtained are as follows.—

(1) the number of cycles required for a satisfactory result is independent of the number of busbars, the presence of radial loads, or the location of the slack busbar (in one case this was at the end of a radial line) ;
(2) the voltage correction on the fourth cycle is less than 0.00005 per unit for both voltage co-ordinates at all busbars ;
(3) after the first couple of cycles, the voltage correction is of a similar order for both co-ordinates at all busbars (Appendix IV). This is in contrast to the iterative method, where convergence to the correct voltage is slower at busbars loosely coupled to the slack busbar ; and
(4) usually after three and certainly after four cycles, the computed power differs by less than 0.005 MW and 0.005 MVAr from the specified power at all busbars. With an iterative solution, this difference is as high as 0.8 MW, occasionally higher, at some busbars even with a convergence limit of 0.00005 per unit for the voltage co-ordinates.

Following these successful tests a programme based on the admittance matrix and using the elimination method to solve the correction equations, was developed. The principal difficulty with the programme is in devising techniques for the efficient storing of the admittance and correction matrixes. Although the size of these matrixes depends on the square of the number of network busbars, most of their terms for large systems are zero (Ref. 9). Thus by storing and operating on the non-zero terms only, a considerable saving in computer storage space and operating times can be expected.

The admittance matrix presents no difficulties in this respect and, furthermore, is symmetrical and constant for a particular network. As the terms of the correction matrix depend on the busbar voltages, which vary between cycles, this matrix must be recalculated for each cycle. For this reason, and also because it is unsymmetrical and terms initially zero are assigned values during the elimination, it is desirable to determine for the correction matrix of each problem

(1) the order of eliminating columns (Appendix II) which ensures that a minimum number of locations are assigned values during the calculation ; and
(2) the storage requirements for all the terms used during the elimination ; in particular, that for each column so that the terms can be stored in their right relationship to avoid shifting and facilitate indexing during the calculation.

While the scheme used to achieve these objectives may not be the optimum, it does, however lead to considerable savings in space and arithmetic operations by the computer.

5.—Voltage Correction Equations.

The equations for the busbar voltage corrections can be written

$$[\Delta S] = [M][\Delta V] \qquad \text{.....(4)}$$

or,

$$\begin{bmatrix} \Delta Q \\ \Delta P \end{bmatrix} = \begin{bmatrix} U & W \\ H & T \end{bmatrix} \begin{bmatrix} \Delta e \\ \Delta f \end{bmatrix} \qquad \text{.....(5)}$$

The expressions for the terms of Eq. (5), set out in Eqs. (6) to (17) below, are derived in Appendix I. In the following equations, k and m are two busbars, other than the slack busbar, connected in the network by a transmission line, transformer or regulator. At a load busbar k :

$$\Delta Q_k = Q'_k - Q_k \qquad \text{.....(6)}$$

$$U_{kk} = -b_k - e_k B_{kk} + f_k G_{kk} \qquad \text{.....(7)}$$

$$W_{kk} = a_k - e_k G_{kk} - f_k B_{kk} \qquad \text{.....(8)}$$

At a generator busbar k :

$$\Delta Q_k = |V'_k|^2 - |V|^2 \qquad \text{.....(9)}$$

$$U_{kk} = 2e_k \qquad \text{.....(10)}$$

$$W_{kk} = 2f_k \qquad \text{.....(11)}$$

At a load or generator busbar k:

$$\Delta P_k = P'_k - P_k \qquad \text{.....(12)}$$

$$H_{kk} = a_k + e_k G_{kk} + f_k B_{kk} \qquad \text{.....(13)}$$

$$T_{kk} = b_k - e_k B_{kk} + f_k G_{kk} \qquad \text{.....(14)}$$

$$U_{km} = T_{km} = - e_k B_{km} + f_k G_{km} \qquad \text{.....(15)}$$

$$H_{km} = - W_{km} = e_k G_{km} + f_k B_{km} \qquad \text{.....(16)}$$

except that for a generator busbar k:

$$U_{km} = W_{km} = 0 \qquad \text{.....(17)}$$

Hence, it follows that ΔQ, ΔP, Δe and Δf are column matrixes each with $(N - 1)$ terms, while U, W, H and T are $(N - 1)$ by $(N - 1)$ square matrixes.

It is seen from Eqs. (6) to (17) that, excluding the slack busbar, for each network connection there corresponds a term in the U, W, H and T matrixes (including the zero terms of Eq. (17)). As in general U_{km} is not equal to U_{mk}, etc., the matrixes are not symmetrical, but corresponding terms on either side of their main diagonals have values and these terms correspond with those of the network admittance matrix, Y, when the row and column for the slack bushbar are excluded. This symmetry, and the correspondence with the admittance matrix, provide clues for the efficient solution of Eq. (5) by elimination (Refs. 9 and 10) and for setting up a system to store and locate all the terms of the connection matrix used in the calculations.

6.—Elimination Scheme.

Fig. 1 represents, in outline, the connections of a small power system network; those to the swing busbar being indicated with broken lines. The U-matrix corresponding to this network is

$$\begin{array}{c} \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{array} \begin{array}{c} \begin{array}{cccccccc} 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{array} \\ \left[\begin{array}{cccccccc} U_{22} & & & U_{25} & & & & \\ & U_{33} & & U_{35} & U_{36} & U_{37} & & \\ & & U_{44} & & & & & U_{49} \\ U_{52} & U_{53} & & U_{55} & & & & \\ & U_{63} & & & U_{66} & U_{67} & & \\ & U_{73} & & & U_{76} & U_{77} & U_{78} & \\ & & & & & U_{87} & U_{88} & U_{89} \\ & & U_{94} & & & & U_{98} & U_{99} \end{array}\right] \end{array} \qquad \text{......(18)}$$

where the terms indicated are assigned values by Eqs. (6) to (17). In the following discussion, the elimination method of solving Eq. (5), set out in Appendix II, is discussed with reference to Fig. 1

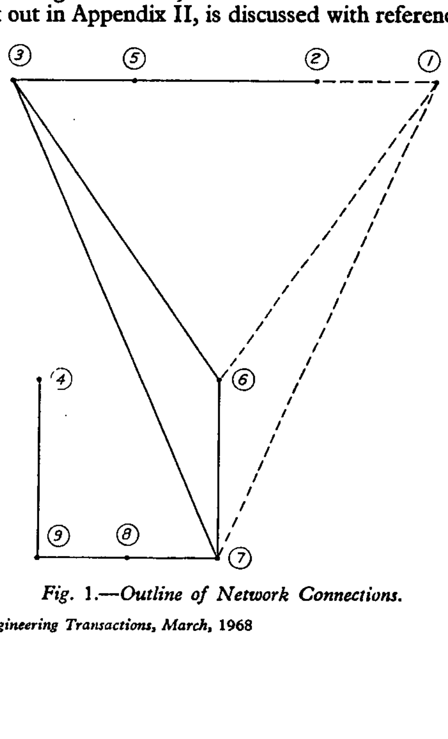


Fig. 1.—*Outline of Network Connections.*

and its corresponding U-matrix Eq. (18). In Eq. (18), columns numbered 2 and 4 each have one off-diagonal term corresponding to a connection in Fig. 1. The elimination of the terms U_{52} and U_{94} introduces no new terms in the U-matrix, just as a network reduction which eliminates busbars at the end of radial lines introduces no new network branches. The first column with two off-diagonal terms is 5, and the elimination of U_{25}, U_{35} introduces a new term U_{23}, but not a term U_{32} as U_{52} in column 2 has already been eliminated. The equivalent operation in the network reduction is the elimination of busbar 5 with the formation of the new branch 2-3.

Column 6 is the next one with two off-diagonal terms and, as the terms U_{37} and U_{73} are already present, the elimination of U_{36} and U_{76} introduces no new matrix terms. This is equivalent, in the network reduction, to eliminating busbar 6 which introduces a branch in parallel with the existing branch 3-7. The elimination of the off-diagonal terms of column 8 introduces the new terms U_{79} and U_{97}, and as this results in three off-diagonal terms in column 9, all the remaining columns have three or more terms. At this stage of the elimination, Eq. (18) has the form:

$$\begin{array}{c} \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{array} \begin{array}{c} \begin{array}{cccccccc} 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{array} \\ \left[\begin{array}{cccccccc} 1 & U_{23} & & & & & & \\ & U_{33} & & & & U_{37} & & \\ & & 1 & & & & & U_{49} \\ & U_{53} & & 1 & & & & \\ & U_{63} & & & 1 & U_{67} & & \\ & U_{73} & & & & U_{77} & & U_{79} \\ & & & & & U_{87} & 1 & U_{89} \\ & & & & & U_{97} & 0 & U_{99} \end{array}\right] \end{array} \qquad \text{......(19)}$$

The values of the terms U_{33}, U_{53}, etc. of Eq. (19) will, in general, be different from those of corresponding terms in Eq. (18). So far the calculation has resulted in the elimination of eight, and the introduction of three off-diagonal terms, i.e., a net loss of five terms from the U-matrix. The remaining columns are processed in the order 9 (introducing a term U_{47}), 3 (introducing terms U_{27} and U_{57}) and finally 7 which completes the calculation.

From the above discussion it follows that an efficient elimination procedure for solving Eq. (5) is as follows.—

(1) processing, in order, the columns with one off-diagonal term. These correspond to busbars with one network branch connection and introduce no new terms in the matrix;

(2) processing, in order, the columns with two off-diagonal terms. These correspond to busbars with two network branch connections and introduce none, one or two new terms in the matrix depending on the network connections and the columns eliminated under (1). Note that the processing of one column may add a term to another column in this category which then becomes a column with three off-diagonal terms;

(3) processing, in order, the columns with three off-diagonal terms, i.e., those corresponding to busbars with three network branch connections. This is equivalent, in a network reduction, to a star-delta transformation and could result in as many as six new matrix terms corresponding to three branches. However, in practice not more than two or three new terms are likely, because (*a*) network connections are usually such that some of the terms already exist in the matrix; and (*b*) columns where new terms could appear have been processed under (1) and (2). This is illustrated in Eq. (19) when the three off-diagonal terms of column 9 are eliminated resulting in one new term U_{47}; and

(4) processing, in order, the columns with four, then those with five, etc., off-diagonal terms until the elimination is complete. Theoretically, large numbers of new matrix terms can occur; but, in practice this is unlikely because of existing network connections and, at this stage, many columns where new terms might appear have been processed, e.g., from Eq. (19) the elimination of the four off-diagonal terms in column 3 results in two new terms U_{27} and U_{57}.

The elimination scheme, discussed above for the U-matrix, can be applied, with slight modifications, to the M-matrix. In this case, the columns are processed in pairs, i.e., referring to Fig. 1

and its M-matrix, the off-diagonal terms U_{52}, H_{52}, H_{22}, W_{22}, W_{52} and T_{52}, in the column corresponding to busbar No. 2, are eliminated first followed, in order, by the off-diagonal terms in the columns corresponding to busbar Nos. 4, 5, 6, 8, 9, 3 and 7. Therefore, after processing the columns corresponding to busbars Nos. 2, 4, 5, 6 and 8, the H-, W- and T-matrixes have terms in corresponding locations to that of the U-matrix, as indicated in Eq. (19), except that the diagonal terms in these columns of the H- and W-matrixes are zero.

The preferred elimination scheme above is now compared with that of eliminating the off-diagonal matrix terms, column by column, in the order of the busbar numbers, i.e., 2, 3, 4, 5, etc. Using the latter order of elimination, 15 new terms appear and 29 terms are modified in the U-matrix of Eq. (18), whereas only 6 new terms and 22 modifications occur with the preferred scheme. It therefore follows that, if these two elimination schemes are applied to the correction equations for Fig. 1 (i.e., Eq. (5)) then the preferred scheme leads to a reduction in the number of M-matrix terms used during the calculation and, also in the number of arithmetic operations on terms of the M- and ΔS-matrixes.

7.—Computer Programme.

In Fig. 2, the flow diagram for the load flow programme is given. The calculation commences by reading in the system data which is in two parts. The first part is basic data listing the number of modifications, MVA base, numbers of busbars, transmission lines (including transformers), regulators, generator busbars (including the slack busbar), load busbars, shunt reactors or capacitors and busbars with voltage limits. The second part is the line data which, for the network of Fig. 3, are set out in Table I.

Any regulators are listed last and given numbers commencing with 501 to distinguish them from the transmission lines and fixed ratio transformers; the tap refers to the higher numbered busbar, i.e., in Table I busbar 5 has an 8 per cent boost. As the data for each line are read, the busbar connections are stored, the resistance and reactance converted to per unit admittance, the shunt susceptance to per unit and these three values stored. For regulators the

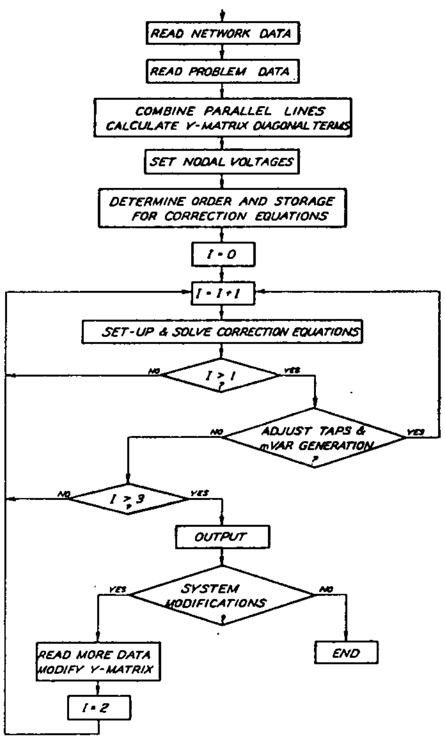


Fig. 2.—Power System Network.

TABLE I.

Line and Regulator Data.

Line No.	Busbar Connections		Resistance (ohms)	Reactance (ohms)	Shunt Susceptance (micromhos, or Tap %)	Nominal Voltage (kV)
1	5	2	1.25	3.4	42.0	110
2	2	1	1.34	3.85	48.0	110
3	1	6	10.36	18.25	230.0	110
4	1	7	14.67	20.34	268.0	110
5	7	6	3.6	22.15	30.0	110
6	7	8	0.18	0.69	5.0	110
7	7	3	11.25	19.72	252.0	110
8	6	3	18.21	45.09	102.0	110
9	8	9	0.0	0.04	0.0	1
10	9	4	4.83	19.37	233.0	220
501	3	5	0.0	2.45	1.08	110

per unit admittance multiplied by the tap is stored, together with the tap setting. Hence the list of admittances stored represents the off-diagonal terms of the admittance matrix, and can be used directly in calculating terms of the M-matrix.

Following the reading of the system data, the problem data are read. Tables II and III set out these data for the network of Fig. 3.

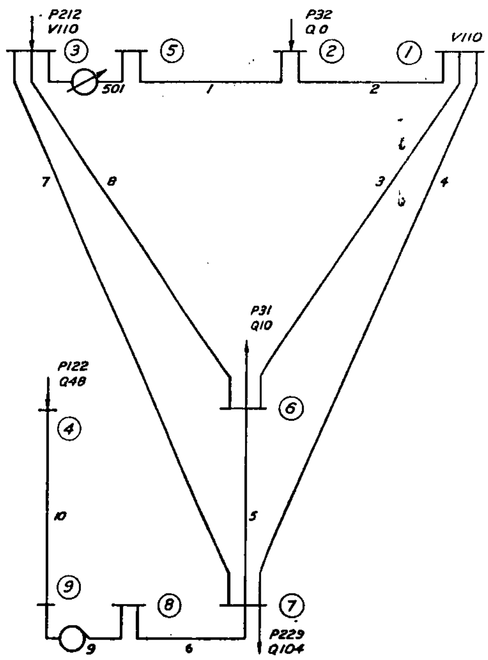


Fig. 3.—Basic Diagram for Load Flow Study.

TABLE II.

Generator Busbar Data.

Busbar No.	Voltage (kV)	Real Power (MW)	Reactive Power Limits (MVAr)		Nominal Voltage (kV)
1	110.0				110
3	110.0	212.0	80.0	110.0	110

The first busbar listed is the slack busbar for which the voltage magnitude and nominal voltage only are specified, but it may be assigned any number. For the remaining generator busbars the real power and upper and lower reactive power limits are also specified.

TABLE III.

Load Busbar Data.

Busbar No.	Real Power (MW)	Reactive Power (MVAr)	Nominal Voltage (kV)
2	32.0	0.0	110
4	122.0	48.0	220
5	0.0	0.0	110
6	−31.0	−10.0	110
7	−229.0	−104.0	110
8	0.0	0.0	110
9	0.0	0.0	220

This list includes all busbars at which the voltage magnitude is not specified; the sign indicates whether the nett power flow is into the system (positive), or out of the system (negative) at the busbar.

Further problem data such as shunt reactors and capacitors, voltage limits at regulator busbars can also be included at this stage. This is followed by system modifications which can be either line switching or load changing.

After reading all the data, the line list is searched and any parallel lines combined, thus determining the number of network connections which is useful in setting up the correction matrix. Next, the diagonal terms of the admittance matrix are calculated and stored, and two lists, one showing the number of connections to each busbar, the other indicating where the corresponding admittance is stored, are set up to give easy access to non-zero Y-matrix terms. The initial per unit voltage of each load busbar is set at $1 + j0$ and of each generator busbar at $e + j0$ where e is the specified per unit voltage magnitude.

Following this, lists showing the number of terms in each column and row of, and their locations in, the U-matrix are set up. The numbers of the busbars corresponding to columns with one off-diagonal term are listed and the row list modified by removing terms relating to these columns. Next listed is the first busbar corresponding to a column with two off-diagonal terms. The elimination of these two terms is then examined to determine whether it results in new terms for any column not listed. If so, the necessary lists are amended before removing the related terms from the row list and proceeding to the next column with two off-diagonal terms. This procedure is continued through columns with two, three, etc., off-diagonal terms until all $(N - 1)$ busbars are listed. The result is three lists, the first of busbar numbers, the remaining two giving the maximum number of terms which appear in each column and row when the elimination proceeds according to the first list.

The calculation now enters the elimination cycle. The correction matrix is set up using the column and row lists to allocate storage space for each set of terms, thus avoiding term shifting

BUS	MW	MVAr	VOLTAGE		kV	per unit	degrees
1	−90.39	−19.04	110.00	+0.00j	110.00	1.0000	0.0000
2	32.00	−0.00	113.89	+4.30j	113.97	1.0361	2.1622
3	212.00	90.87	109.64	+8.91j	110.00	1.0000	4.6469
4	122.00	48.00	214.78	+17.30j	215.47	0.9794	4.6059
5	0.00	0.00	117.06	+7.10j	117.28	1.0662	3.4731
6	−31.00	−10.00	104.51	−1.08j	104.51	0.9501	−0.5909
7	- 229.00	−104.00	100.52	−2.88j	100.57	0.9142	−1.6427
8	0.00	−0.00	101.12	−2.17j	101.14	0.9194	−1.2298
9	0.00	−0.00	207.56	+7.04j	207.67	0.9440	1.9429

LINE NO	BUS	MW	MVAr	BUS	MW	MVAr	MW LOSS	MVAr LOSS	TAP
1	2	114.79	67.61	5	−116.50	−72.26	1.708	4.646	
2	1	144.08	60.99	2	−146.79	−68.78	2.711	7.789	
3	1	−19.13	−22.25	6	18.39	20.95	0.737	1.298	
4	1	−34.57	−26.31	7	32.28	23.14	2.288	3.173	
5	6	−11.45	−16.84	7	11.31	16.00	0.137	0.841	
6	7	119.79	51.95	8	−120.09	−53.11	0.303	1.163	
7	3	−70.47	−15.80	7	65.62	7.30	4.849	8.500	
8	3	−25.03	−4.34	6	24.06	1.94	0.971	2.405	
9	8	120.09	53.06	9	−120.09	−61.21	0.000	8.156	
10	4	−122.00	−58.82	9	120.09	55.16	1.908	7.653	
501	3	−116.50	−75.01	5	116.50	71.68	0.000	3.333	1.080

TOTAL LINE LOSS = 15.61 MW 48.96 MVAr TOTAL LINE CHARGING = 43.12 MVAr

BUS	MISMATCH MW	MVAr	LINE CHARGING MVAr
1	−0.0000	0.0000	6.61
2	0.0000	0.0000	1.17
3	0.0000	0.0000	4.28

Fig. 4.—Partial Print of Results for Load Flow Study.

during the calculation. After two cycles the results are examined, and if necessary, regulator taps adjusted and generator busbars which have exceeded their reactive power limits converted to load busbars. The elimination cycle is then continued until there are no further adjustments or until four cycles have been completed.

When this stage is reached the results are punched out on paper tape for printing. These are

(1) at each busbar—the MW and MVAr power flowing into, or out of, the network; the voltage components and magnitude in kV and per unit; the voltage angle referred to the swing busbar; the line charging lumped at the busbar; the reactive flow in any shunt reactors or capacitors;

(2) for each line, transformer or regulator—the MW and MVAr flowing into or out of each busbar; the MW and MVAr loss; and

(3) the total MW line loss and MVAr line charging for the network.

Any required network modifications are then carried out and the elimination cycle re-entered, otherwise the calculation is complete.

Fig. 4 is a portion of the printed results for a load flow study of the network shown in Fig. 3.

8.—Conclusion.

For solving the load flow problem, the more direct method of using a set of linear equations to calculate voltage corrections at all network busbars simultaneously is superior to the iterative method of sequentially determining busbar voltage corrections. Only three or four correction calculations are required to obtain an accurate solution of any practical load flow problem.

It has disadvantages in that the correction equations require considerable storage space and are recomputed for each cycle. By taking advantage of the sparsity of the admittance matrix, these can be overcome and a scheme established for the elimination process based on the network connections when the slack busbar is excluded. This leads to a considerable saving of computer storage space and to a reduction in the number of arithmetic operations during computation.

The computer programme developed using these techniques efficiently solves system load flow problems.

Acknowledgments.

The author is indebted to the Hydro-Electric Commissioner, Mr. A. W. Knight, C.M.G., B.Sc., M.E., B.Comm., M.I.E.Aust., for permission to publish this paper. He is grateful to colleagues in the Electrical Investigations Division for useful comments regarding the working of the load flow programmes.

References.

1. Ward, J. B. and Hale, H. W.—Digital Computer Solution of Power-Flow Problems. *Trans. A.I.E.E. Pt. III*, Vol. 75, June, 1956, p. 398, (Paper No. 56-164).
2. Furst, G. B. and Heal, N. F.—Digital Load Flow Studies. The SNOCOM Load Flow Programme. *Jour. I.E.Aust.*, Vol. 34, No. 12, Dec., 1962, pp. 333-41.
3. Laughton, M. A. and Davies, M. W. H.—Numerical Techniques in Solution of Power-System Load-Flow Problems. *Proc. I.E.E.*, Vol. 111, No. 9, Sept., 1964, p. 1575 (Paper No. 4511P).
4. Van Ness, J. E. and Griffin, J. H.—Elimination Methods for Load-Flow Studies. *A.I.E.E. Power Apparatus and Systems*, No. 54, June, 1961, p. 299, (Paper No. 61-51). (*Trans. A.I.E.E. Pt. III*, No. 80.)
5. Gupta, P. P. and Davies, M. W. H.—Digital Computers in Power System Analysis. *Proc. I.E.E. Pt. A*, Vol. 108, No. 41, Oct., 1961, p. 383 (Paper No. 3484S).
6. Brameller, A. and Denmead, J. K.—Some Improved Methods for Digital Network Analysis. *Proc. I.E.E. Pt. A*, Vol. 109, No. 43, Feb., 1962, p. 109, (Paper No. 3778S).
7. Hale, H. W. and Goodrich, R. W.—Digital Computation of Power Flow—Some New Aspects. *Trans. A.I.E.E. Pt. III-A*, Vol. 78, Oct., 1959, p. 919, (Paper No. 59-224).
8. Brown, H. E., Carter, G. K., Happ, H. H. and Person, C. E.—Power Flow Solution by Impedance Matrix Iterative Method. *I.E.E.E. Trans. on Power Apparatus and Systems*, No. 65, April, 1963, p. 1, (Paper No. 62-214).
9. Sato, N. and Tinney, W. F.—Techniques for Exploiting the Sparsity of the Network Admittance Matrix. *I.E.E.E. Trans. on Power Apparatus and Systems*, No. 69, Dec., 1963, p. 944 (Paper No. 63-201).
10. Carre, B. A.—The Partitioning of Network Equations for Block Iteration. *The Computer Journal*, Vol. 9, May, 1966, p. 84.
11. Bickley, W. G. and Thompson, R. S. H. G.—*Matrices, Their Meaning and Manipulation.* London, English Universities Press, 1964, 168 p.

APPENDIX I.

Derivation of Equations.

From Eq. (1), the current I_k, at busbar k, is calculated from the network constants and the assumed voltages. The following equation is then used to calculate a power at the busbar:

$$S_k = V_k I^*_k \qquad \text{......(20)}$$

However, as the correct problem voltage differs from the assumed voltage $[V]$ by a quantity $[\Delta V]$ the correct current will differ from $[I]$ by $[\Delta I]$, therefore, the equation for the problem is:

$$[I + \Delta I] = [Y][V + \Delta V] \qquad \text{......(21)}$$

and the specified power at busbar k is:

$$S'_k = S_k + \Delta S_k = (V_k + \Delta V_k)(I^*_k + \Delta I^*_k) \qquad \text{......(22)}$$

From Eqs. (1) and (21)

$$[\Delta I] = [Y][\Delta V] \qquad \text{......(23)}$$

Combining Eqs. (20) and (22) and substituting for ΔI_k from Eq. (23),

$$\Delta S_k = \Delta V_k I^*_k + (V_k + \Delta V_k) \sum_m Y^*_{mk} \Delta V^*_m \qquad \text{......(24)}$$

Equating the real and imaginary terms on both sides of this equation, results in $2(N - 1)$ second-order equations in the unknown voltage corrections—there will not be an equation at the slack busbar as the voltage corrections are zero. At the load busbars all the other terms in Eq. (24) can be calculated from the network and problem data, but at the generator busbars the reactive power is unknown and hence one of the two equations contains another unknown. However, as at a busbar k of this type the voltage magnitude $V'_k = V_k + \Delta V_k$ is specified, the following equation can be written

$$\begin{aligned}|V'_k|^2 - |V_k|^2 &= (V_k + \Delta V_k)(V^*_k + \Delta V^*_k) - V_k V^*_k \\ &= V_k \Delta V^*_k + \Delta V_k(V^*_k + \Delta V^*_k) \qquad \text{......(25)}\end{aligned}$$

Theoretically it is possible to solve, for the unknown voltage corrections, the $2(N - 1)$ equations derived from Eqs. (24) and (25). But, as it is easier to solve linear equations, an approximate solution is obtained by neglecting the terms involving powers and products of the voltage corrections. In a power system network, the busbar voltages do not differ greatly from nominal, in most cases less than 10 per cent, therefore the corrections are small. A result for the load flow problem which is, for practical purposes, exact can be found by repeating, three or four times, the set up and solution of Eqs. (24) and (25) neglecting second-order terms.

Neglecting second order terms, Eq. (24) becomes

$$\Delta S_k = \Delta V_k I^*_k + V_k \sum_m Y^*_{mk} \Delta V_m^*$$

or,

$$\begin{aligned}\Delta P_k + j\Delta Q_k = (\Delta e_k + j\Delta f_k)(a_k - jb_k) + (e_k + jf_k) \times \\ \times \sum_m (G_{mk} - jB_{mk})(\Delta e_m - j\Delta f_m) \qquad \text{...(26)}\end{aligned}$$

Equating the real and imaginary parts of Eq. (26):

$$\begin{aligned}\Delta P_k = a_k\Delta e_k + b_k\Delta f_k + \sum_m (e_kG_{mk} + f_kB_{mk})\Delta e_m + \\ + \sum_m (- e_kB_{mk} + f_kG_{mk})\Delta f_m \qquad \text{......(27)}\end{aligned}$$

$$\begin{aligned}\Delta Q_k = - b_k\Delta e_k + a_k\Delta f_k + \sum_m (- e_kB_{mk} + f_kG_{mk})\Delta e_m - \\ - \sum_m (e_kG_{mk} + f_kB_{mk})\Delta f_m \qquad \text{......(28)}\end{aligned}$$

Similarly from Eq. (25):

$$\begin{aligned}|V'_k|^2 - |V_k|^2 &= V_k\Delta V^*_k + \Delta V_k V^*_k \\ &= 2e_k\Delta e_k + 2f_k\Delta f_k \qquad \text{......(29)}\end{aligned}$$

Eqs. (6) to (17) follow from Eqs. (27), (28) and (29).

APPENDIX II.

Digital Solution of Equations.

The procedure used for solving a set of linear equations is outlined briefly. Consider the set of equations

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} \qquad \text{.................(30)}$$

in which it is required to determine x_1, x_2, and x_3, assuming that the a-matrix is not singular. The procedure on the computer is to use the main diagonal terms as pivots and eliminate the off-diagonal terms in successive columns (Ref. 11). For example, starting with column 2, all terms in row two of the a- and c-matrixes are divided by a_{22}. The term a_{12} is then eliminated by subtracting $a_{12}a_{21}/a_{22}$ from a_{11}; $a_{12}a_{23}/a_{22}$ from a_{13} and $a_{12}c_2/a_{22}$ from c_1. In a similar way, a_{32} is eliminated modifying the row 3 terms; then the off-diagonal terms in columns 1 and 3 are eliminated using first a_{11}, then a_{33} as the pivot. With this procedure, no additional storage is required by the computer and the final c-matrix terms are the solution to the equations.

APPENDIX III.

Accuracy of Method of Solution.

In the above method, it can be shown (Ref. 11) that the result is satisfactory and not subject to round-off errors if the element of greatest absolute value in each column is used as the pivot. Assuming that $e = 1$ and $f = 0$ at all busbars, then from Eqs. (7) and (13):

$$U_{kk} = -b_k - B_{kk} \qquad \text{..........................(31)}$$

$$H_{kk} = a_k + G_{kk} \qquad \text{..........................(32)}$$

From Eq. (1)

$$a_k + jb_k = \sum_m (G_{mk} + jB_{mk}) \qquad \text{.....................(33)}$$

For a network of lines having resistance, reactance and shunt susceptance $\sum_m G_{mk}$ is zero and $\sum_m B_{mk}$ is small, being the summation of the shunt susceptances at the busbar, k. Hence from Eq. (33), a_k is zero and b_k is small. As in power system networks the reactance of a line is usually greater than its resistance, it follows that B_{kk} is greater than G_{kk} and hence, from Eqs. (31) and (32), U_{kk} is higher in absolute value than H_{kk}.

From Eqs. (15) and (16):

$$U_{mk} = B_{mk} \qquad \text{................................(34)}$$

$$H_{mk} = G_{mk} \qquad \text{................................(35)}$$

For a power system, the diagonal term B_{kk} is, in general, greater than the off-diagonal terms B_{mk}, G_{mk} of the Y-matrix. Hence, from Eqs. (31), (34) and (35), U_{kk} is greater than U_{mk} and H_{mk}. If k is a generator busbar, then from Eq. (10)

$$U_{kk} = 2 \qquad \text{................................(36)}$$

This, in general, is not greater than other terms in column k of the U- and H-matrixes, but in this case all other terms in row k are zero.

Similarly it can be shown that, in general, T_{kk} is the largest element in column k of the W- and T-matrixes. After the first elimination, e_k and f_k will not differ greatly from one and zero respectively, therefore the conclusions of the above analysis still hold. Therefore, in the majority of power system networks, $1 + j0$ as the starting busbar voltage ensures that each time the correction matrix M is set up, its main diagonal terms are the elements of greatest absolute value in their respective columns (except as mentioned above for generator busbars).

In lower voltage networks of the distribution type, the resistance of a line may be greater than its reactance and the presence of any cables increases the shunt susceptance at the busbars concerned. As can be seen from Eqs. (7) to (16) for busbar k, the elements of $[M]$ depend not only on network admittances and susceptances but also on e_k and f_k; and it can be shown that for this type of network, a starting busbar voltage of $0.6 + j0.8$ usually ensures that $[M]$ has a dominant major diagonal.

APPENDIX IV.

Voltage Corrections.

Table IV lists the per unit voltage corrections calculated for the network given in Fig. 3 and the data listed in Tables I, II and III.

For the fourth cycle, the maximum value of the corrections to any busbar voltage is -10^{-6} per unit for the in-phase component of the voltage at No. 4 busbar.

APPENDIX V.

Computer Running Times.

The programme is written in Algol and takes 58 sec. to compile on the Hydro-University Elliott 503 computer. For a load flow study of a 35-busbar system, the time taken from starting to read the data until punching of the output commences is 37 sec. Although the M- and ΔS-matrixes are held in the core backing store, thus requiring transfers to and from this store during computation, only 25 sec. is required for setting up and solving the elimination equations four times. The calculation and punching on paper tape of the busbar voltages and loads, the line losses, charging and power flows takes a further 1 min. 8 sec. giving a total computing time of 1 min. 45 sec. for the study. The overall computing time, including winding of tapes, is about 4 min. and the time taken for printing the results on the flexowriter is 17 min.

TABLE IV.

Busbar Voltage Corrections.

Busbar	First Cycle	Second Cycle	Third Cycle
2	0.0395 +j0.0396	−0.0041 −j0.0004	−0.00003 −j0.00004
3	0.0000 +j0.0872	−0.0033 −j0.0061	−0.00001 −j0.00014
4	0.0069 +j0.0723	−0.0297 +j0.0060	−0.00097 +j0.00030
5	0.0715 +j0.0652	−0.0072 −j0.0006	−0.00005 −j0.00008
6	−0.0392 −j0.0091	−0.0104 −j0.0007	−0.00029 +j0.00002
7	−0.0629 −j0.0253	−0.0226 −j0.0009	−0.00070 +j0.00003
8	−0.0571 −j0.0194	−0.0230 −j0.0004	−0.00072 +j0.00005
9	−0.0290 +j0.0294	−0.0267 +j0.0024	−0.00085 +j0.00016

Discussion

Mr. J. A. Callow (Member, Sydney Division).—The author is kind enough to refer to work by the Snowy Mountains Authority in Ref. 2, the load flow programme for the SNOCOM computer. That was our first attempt at producing a load flow programme, and we have since produced programmes for the National Elliott 405 and for the GE225, the latter machine being in use at present. In writing the later programmes, we made various modifications to reduce preparation and computing times and improve convergence. We have established clearly that for any particular system the best result will be obtained by a programme which is to some extent tailored to suit the system. For instance, we found that for the particular configuration of the interconnected 330-kV system the package GE225 programme was not satisfactory. Our present programmes include various options that permit a degree of " tailoring ".

We considered the elimination method for the GE225 programme but found that for an extensive system with a large number of busbars the large matrices required an amount of fast access storage which was not available to us. Could the author indicate how many busbars his computer can handle?

In view of the author's recent experience with both network analysers and computers, could he comment on whether he finds it possible to get the feel of a new problem on the computer as quickly as on the network analyser? We have recently been trying to simulate the interconnected system as a three-machine system in which the machines would have the same voltages and angles behind transient reactance as selected machines in the full representation. This requires the shifting of impedance from one line section to another line section (i.e., to either side of critical loads) and a step-by-step process is required with the computer because of the need to check the results, particularly the voltages at the loads, before proceeding with the punched cards for the next adjustment. We had the feeling that we could have achieved the required result in less time on a network analyser. However, for the following stability studies, the punched cards for several different fault locations could be produced in one batch, and left with the operator.

The Author in Reply:

Resulting from the growth and interconnection of power systems, the need has arisen to solve load flow problems for large complicated networks which has led to a great deal of effort being expended on the improvement of existing, and the development of more sophisticated computer programmes. Such programmes are made possible by improvements in computers such as the construction of large random access core storages, a considerable increase in the speed of arithmetic and logic operations and the introduction of languages like " Algol " by means of which the power system engineer (without becoming a skilled programmer) can communicate directly with the digital computer in a language in which he can think.

For the successful application of the elimination (or Newton's) method to the load flow problem, it is essential to determine a preferred order of solving the voltage correction equations and this is achieved by using the logical versatility of the computer to, in effect renumber the network busbars. By this preliminary network analysis the necessity of storing large matrixes is avoided, thereby increasing the size of the problem that can be handled by the computer and also decreasing the computation time by limiting the number of arithmetic operations required. As it is becoming recognised that Newton's is one of the best and most reliable methods for solving load flow problems, many organisations can be expected to adopt it as techniques for controlling the size of the matrixes used are developed.

Using the H.E.C.'s programme which occupies some 6,500 locations of the 8K core store in the Elliott 503 computer, problems with 70 busbars have been solved. For a problem of this size, the remainder of the core store plus about 2,000 locations in the 16K core backing store are used, leaving plenty of space available for solving larger networks.

There is a physical correspondence between the system parameters and controls and their representation on a network analyser, which is therefore suitable for experimentation and an excellent educational instrument enabling an engineer to readily acquire a knowledge of the operating characteristics of a small system. However, engineers using a digital computer soon become familiar with the effect on system behaviour of altering various parameters and tend to plan studies in a systematic and disciplined way; furthermore the computer's powerful logical facilities can be used to incorporate engineering decisions in programmes.

The author would agree that the network analyser is as fast as the digital computer for studies on small systems, but it is a specialised item of equipment which would not be bought today because the digital computer can do all the work of the analyser and have time available for solving other problems. As Mr. Callow has mentioned, punched cards or tape can be produced by a computer thus making it easy to interrupt a study or, at some time in the future, do subordinate studies not originally contemplated.

TABLE X

Type of Problem	Gauss–Seidel Method	Newton's Method
Heavily loaded systems	Usually cannot solve systems with phase shift beyond 70 degrees	Solves systems with phase shifts up to 90 degrees
Systems containing negative reactance, such as 3-winding transformers or series capacitors	Unable to solve	Solves with ease
Systems with slack bus at a desired location	Often requires trial-and-error to find a slack bus location which will yield a solution	More tolerant of slack bus location
Long and short lines terminating on same bus	Usually cannot solve a system with a long-to-short line ratio at any bus beyond 1000 to 1	Can solve a system with a long-to-short line ratio at any bus of 1 000 000 to 1
Long radial type system	Difficulty in solving	Solves a wider range of such problems
Acceleration factors	Number of iterations depends on choice of acceleration factors	None required

automatically, with print-out of overloaded lines only, takes three minutes and six seconds using a flat voltage start for each case, and takes 26 seconds longer using voltages from the previous case. It is believed that the 11 percent savings in computer time is not worth risking a possible convergence to a wrong solution.

The main disadvantages of Newton's method are: 1) the programming logic is considerably more complex, and 2) the memory requirements dictate a computer with at least 32K memory.

The advantages of Newton's method include greater speed and accuracy, and the ability to solve a wider variety of ill-conditioned systems than ever before. The authors are to be commended for their work in presenting this method to the industry.

W. A. Prebble (The Hydro-Electric Commission, Hobart, Tasmania, Australia): The following comments refer to the program, based on the elimination method suggested by Van Ness and Griffin,[1] developed and tested early in 1966 and now used for solving the Commission's load flow problems.

Using rectangular Cartesian coordinates, i.e.,

$$V = e + jf \tag{11}$$

the network voltage correction equations, written in matrix form, are

$$\begin{bmatrix} \Delta Q \\ \Delta P \end{bmatrix} = \begin{bmatrix} U & W \\ S & T \end{bmatrix} \begin{bmatrix} \Delta e \\ \Delta f \end{bmatrix} \tag{12}$$

in which there are no rows or columns corresponding to the slack busbar. The values used in the U, W, S and T submatrixes are given by equations (14) and (15) of Van Ness and Griffin;[1] for generator busbars the equation for voltage magnitude is included in ΔQ, U and W. Equation (12) is solved by column pivoting, with the main diagonal U and T terms as the pivots.

As, for a particular problem, (12) is set up and solved more than once, it is advantageous to predetermine the order of column pivoting which keeps the number of arithmetic operations (also the storage requirements) to a minimum for this problem. The network connections are examined and a list, establishing the order of column pivots, is set up so that at each stage of the calculation, the pivot is the one in the column with the least number of off-diagonal terms; when several columns satisfy this requirement simultaneously, the first of these is used.

From the network, omitting the slack busbar and the connections to it, indexes are set up for locating the off-diagonal terms of the T matrix. The columns with one off-diagonal term (corresponding to busbars at the end of radial lines) are listed first, because when used as pivotal columns no new terms are introduced into the matrix. Next, the first column with two off-diagonal terms is listed and the result of using it as the pivotal column examined; if this introduces new terms in the matrix, then the indexes are modified. The procedure is repeated with the remaining columns having two off-diagonal terms, then those with three, etc., until all the columns are listed.

Equation (12) is then set up and solved by pivoting on the U and T diagonal terms pairs according to the order list, the indexes giving the storage location of the terms used during the calculation.

It is interesting to note that with rectangular voltage coordinates the U, W, S, and T submatrix terms depend directly on the values of e and f chosen. In a high-voltage system, the choice of $1 + j0$ for the initial busbar voltages makes the U and T diagonal terms the ones of greatest magnitude in their respective columns each time (12) is set up (except for generator busbars where the U diagonal term is initially two and all other terms in this U and W row zero). For low-voltage distribution line and cable networks, in which the resistance may be greater than the reactance, a starting voltage such as $0.6 + j0.8$ causes dominance of the major diagonal terms. However, in practice no advantage has been found for starting with a voltage other than $1 + j0$; the solution is obtained just as quickly and accurately even when the main diagonal is not dominant.

From the experience gained with the program, which is written in Algol for the Elliott 503 computer, it is concluded that:

1) Any load flow problem can be solved accurately, i.e., within 0.005 MW, 0.005 Mvar of the specified real, reactive power at load busbars and 0.005 MW, 0.005 kV of the specified real power, voltage magnitude at generator busbars; for practical problems this requires two or three elimination cycles.

2) The choice of slack busbar is not important: radial type networks with slack busbar at the beginning have been solved successfully.

3) It is faster than the nodal iterative method using linear acceleration.

Manuscript received February 28, 1967.

[1] J. E. Van Ness and J. H. Griffin, "Elimination methods for load-flow studies," *Trans. AIEE (Power Apparatus and Systems)*, vol. 80, pp. 299–304, June 1961.

P. L. Dandeno (Hydro-Electric Power Commission, Toronto, Ont., Canada): The authors have presented examples which give convincing evidence of extremely high accuracies for load flow solutions. There may not be the need for such high accuracy for *normal* load flow cases; however, this discusser believes that the authors' method could be used to great advantage in stability calculations. This was hinted at in their comments about related developments in the paper.

Manuscript received February 24, 1967.

UNITED STATES
DEPARTMENT OF THE INTERIOR
BONNEVILLE POWER ADMINISTRATION
PORTLAND, OREGON 97208

October 3, 1967

In reply refer to:
ESBM

Mr. Allen Prebble
The Hydro-Electric Commission
Box 355D G.P.O.
Hobart, Tasmania, Australia

Dear Mr. Prebble:

Thank you for the papers and information which you sent to me in your letter of May 4, 1967. Although you may be isolated by geography, you certainly are keeping up with developments in your field.

During this last year it has become widely recognized that Newton's Method is the best way to solve the power flow problem and most of the major U. S. companies either have adopted this method or have plans to do so. You were years ahead of most everyone in this effort.

I am enclosing a copy of our latest paper. We recognize that it probably is too brief for easy reading but there was a 10-page limitation. We intend to write more on this subject later. Although the problem formulation is in terms of the polar form of solution, it could be adqpted to the rectangular form which you use. We expect that the optimal power flow will eventually replace the present cut and try solutions.

Our organization is acquiring a CDC 6400 computer that will enable us to solve power flow problems of 2000 busses. With this computing power many new applications should be possible.

I was also interested in your paper on short circuit calculations. Our present program is similar, but we are considering the possibilities of using the impedance matrix in factored form with sparsity techniques when we rewrite the program.

I would appreciate maintaining our correspondence and exchange of papers.

Sincerely yours,

William F. Tinney, Head
Methods Analysis Group

Enclosure

S.7

as far as possible, the best features of previous methods into one process, with the accent on the rapid solution of the larger modern systems.

Today, there is a widely held view that the N.R. algorithm with ordered elimination constitutes the best general-purpose load-flow solver currently available, for well and ill conditioned systems of all sizes.

My own experience strongly confirms these conclusions, and the following example may be of interest. Having developed a number of different production programs in FORTRAN (coded efficiently within the limitations of the language), I ran the 118 busbar test system used by the author of paper 5830 P on the Manchester University Atlas machine, in order to compare their total execution times. The result for the A.G.S. method was similar to that given in the paper. The N.R. solution converged with a maximum mismatch of less than 0·01 MVA in 23% of the A.G.S. time, at a commercial cost of £2. Much better comparative performances can be obtained with machine-orientated N.R. programs. In addition, the method is generally highly successful on ill-conditioned systems, and has nonrestrictive computer-memory requirements.

The 118 busbar system could not be accommodated in the Atlas 16k core store using the nodal Zmatrix program, but a rough calculation established that one iteration cycle would have taken virtually as long as the complete N.R. solution, ignoring the time required to obtain the Zmatrix. This is in keeping with the trend that, for most purposes, power-system-network solutions using nonsparse inverse network matrices are rapidly becoming obsolete.

For adjusted solutions, the advantages of the N.R. method over the A.G.S. or the author's new boosted G.S. (B.G.S.) method may become even more marked than for unadjusted solutions. Hence, for the larger systems and with machine-orientated programs, the margin of superiority of the N.R. method will still be considerable, and much more marked than indicated in the example.

A. Brameller: It is very interesting to see that some attempt has been made to compare the nodal and mesh approach. For commercial purposes, it is important that the total time of computation is a minimum. Could the authors give us a cost or total-time comparison for the different methods they propose?

The standard AEP networks (on which the authors of Paper 5639 P have based their conclusions) are highly interconnected systems and relatively well conditioned. There is a class of networks which display mathematical ill-conditioning effects. Among such systems are those in which l.v. and h.v. networks are interconnected through relatively high-impedance transformers and also h.v. transmission systems which carry bulk power over several hundreds of miles. Such test systems are available, and I wonder whether the authors investigated their proposed methods to such networks and, if so, what are the results?

In the G.S. or direct solution, whether based on mesh or nodal analysis, the selection of a reference/slack busbar can in general affect the convergence rate. Work carried out in America[B] and confirmed by results obtained at UMIST, suggest that the applications of N.R. methods gives convergence in about four to eight iterations for the most difficult networks. The convergence is virtually independent of the choice of the reference/slack busbar. Could the authors compare their methods with the N.R. methods, bearing in mind that highly efficient topologically controlled elimination processes are available which take advantage of the sparsity of the admittance matrix both in computation and storage requirement.

D. W. Wells: I have tried to compare the convergence rates of the examples quoted in the paper with some results which I obtained using the N.R. method. For the convergence rate ρ, I have used the log of the largest eigenvalue of the iteration matrix which I estimated from the convergence obtained. After n iterations, the residuals are reduced by a factor of about exp ρ. For the various methods, ρ is as follows:

A.G.S. $\rho = 0{\cdot}005–0{\cdot}05$
B.G.S. $\rho = 0{\cdot}01–0{\cdot}1$
N.R. $\rho = 2{\cdot}2–5{\cdot}0$

This means that the B.G.S. method will take at least 20 times as many iterations as the N.R. method to produce a result of the same accuracy. The relative times taken for an iteration are less easy to estimate, but if the Jacobian in the N.R. method is factorised, each iteration will take considerably less than 20 times as long as a B.G.S. iteration. Even allowing for the time taken to form the Jacobian, the N.R. method is faster but the program is more complex.

W. A. Prebble (*communicated*): The hybrid method, shown by the authors of Paper 5639 P to have advantages with regard to the number of iterations, the choice of slack busbar etc., has the disadvantages that the system impedance and admittance matrixes are required, and two iterative procedures must be programmed. For solving load-flow problems in the Hydro-Electric Commission of Tasmania, the variational matrix, or Newton's method, is used, and this experience, together with the following remarks, indicates that it could be superior to the authors' hybrid method.

Table A summarises the results of solving by Newton's method (starting with a voltage of $1 + j0$p.u. at all except the slack busbar) the IEEE standard test systems described in Appendixes 9.6.1, 9.6.3, and 9.6.4 of the paper.

Table A

CONVERGENCE USING NEWTON'S METHOD

System	Iterations	Number of nodes for which the error is greater than:			
		0·0005 MW	0·0005 MVAr	0·001 MW	0·001 MVAr
AEP 14 busbar	3	0	0	0	0
AEP 30 busbar	3	4	4	1	1
AEP 57 busbar	3	9	8	2	2

For each of the test systems, the error in the voltage magnitude at the generator busbars is less than 0·00005p.u. after three iterations, and a fourth iteration reduces the errors at all busbars in the two larger systems to less than 0·0005 MW and 0·0005 MVAr. From Table A, it is seen that the system size has very little effect on the number of iterations required, and for practical purposes these systems are solved in three iterations.

Table B, which lists the voltage corrections calculated in each iteration for the AEP 14-busbar system, shows that all voltages converge quadratically at about the same rate to their final values. For a fourth iteration, all corrections are of a magnitude 10^{-6} or less.

Table B

PER-UNIT VOLTAGE CORRECTIONS

Node	Iteration 1 $\times 10^{-2}$	Iteration 2 $\times 10^{-3}$	Iteration 3 $\times 10^{-4}$
2	$0{\cdot}00 - 8{\cdot}56j$	$-3{\cdot}91 - 4{\cdot}99j$	$-0{\cdot}34 - 1{\cdot}72j$
3	$0{\cdot}00 - 22{\cdot}16j$	$-25{\cdot}14 - 5{\cdot}00j$	$-3{\cdot}60 - 1{\cdot}34j$
4	$2{\cdot}79 - 17{\cdot}64j$	$-22{\cdot}74 - 6{\cdot}76j$	$-4{\cdot}99 - 1{\cdot}04j$
5	$2{\cdot}90 - 14{\cdot}91j$	$-18{\cdot}67 - 6{\cdot}85j$	$-4{\cdot}07 - 1{\cdot}39j$
6	$0{\cdot}00 - 25{\cdot}96j$	$-32{\cdot}17 - 2{\cdot}87j$	$-5{\cdot}60 - 2{\cdot}27j$
7	$7{\cdot}11 - 23{\cdot}19j$	$-35{\cdot}83 - 13{\cdot}79j$	$-9{\cdot}95 - 1{\cdot}99j$
8	$0{\cdot}00 - 25{\cdot}28j$	$-29{\cdot}17 + 0{\cdot}61j$	$-3{\cdot}88 + 0{\cdot}55j$
9	$6{\cdot}51 - 26{\cdot}18j$	$-42{\cdot}21 - 10{\cdot}99j$	$-11{\cdot}84 - 0{\cdot}38j$
10	$5{\cdot}89 - 26{\cdot}39j$	$-41{\cdot}83 - 10{\cdot}29j$	$-11{\cdot}51 - 0{\cdot}58j$
11	$6{\cdot}29 - 25{\cdot}55j$	$-39{\cdot}28 - 14{\cdot}11j$	$-11{\cdot}01 - 3{\cdot}32j$
12	$5{\cdot}91 - 25{\cdot}86j$	$-38{\cdot}95 - 15{\cdot}22j$	$-11{\cdot}02 - 4{\cdot}51j$
13	$5{\cdot}44 - 26{\cdot}06j$	$-39{\cdot}17 - 13{\cdot}51j$	$-10{\cdot}81 - 3{\cdot}75j$
14	$4{\cdot}19 - 28{\cdot}04j$	$-44{\cdot}53 - 6{\cdot}06j$	$-12{\cdot}06 + 1{\cdot}08j$

From the practical point of view, Newton's method has the advantages that

(*a*) no acceleration is required and no testing for convergence is necessary
(*b*) the choice of slack busbar is not critical
(*c*) it is extremely reliable in solving problems that fail to converge using the nodal-admittance technique.

However, it does have the disadvantage that, for a system of $n$nodes, $2(n - 1)$ linear equations are set up and solved in each iteration, but, by taking advantage of network geometry to solve these in a preferred order,[C,D] large systems can be handled on a medium-sized computer in times proportional to the number of nodes.

All the methods (including Newton's) mentioned by the authors are basically that of the network analyser adapted for digital computation; it would seem that real progress can now be made only by devising new techniques.

J. A. Treece (*in reply*): All contributors to the discussion have asserted that the N.R. technique has made G.S. obsolete for load-flow calculations; so I will address my remarks to them all.

In my opinion the G.S. technique will continue to survive for some years. Special-purpose programs, built around a basic G.S. load flow, cannot be easily reprogrammed. Although the N.R. technique can be programmed efficiently, with some difficulty, it is not known if it can stand up to a normal production environment. Long-term studies alone amount to over 300 a year, of which 40% exceed 400 nodes in size. These studies sometimes contain 300 program-controlled tap changers, and some have d.c. links built onto the a.c. system; the d.c. link is extremely sensitive to voltage changes. A technique as reliable as the G.S. technique is absolutely essential. The execution times quoted in Table G show that, for large cases, the bootstrap method is superior to G.S. by a factor of up to three times. However, only iteration times are compared. The input–output time has become more significant, as iteration times have fallen, so a new technique has to be extremely fast to produce a reasonable 'pay off'. The only way to improve this situation is to concentrate on a technique which will handle outage cases very quickly. Examples are frequently run with over 50 outages to establish security of the network. It is in this area that both G.S. and N.R. techniques need to be developed; yet nothing seems to have been done. Perhaps this aspect of the problem is not sufficiently glamorous to attract interest.

In comparing techniques, only mismatches provide a reliable guide. The results quoted in many papers using voltage

Table C

MISMATCH PLOT (ACCELERATION)

Nodes	Iterations	Mismatch	Acceleration
IEEE 118	148	1·63	1·60
	134	1·58	1·64
	123	1·57	1·67
	112	1·56	1·70
	104	1·60	1·72
	97	1·57	1·74*
	400†	7·0	1·75
CEGB 276	325	6·9	1·60
	285	6·8	1·65*
	—	—	1·70‡
CEGB 515	900	18·5	1·60
	838	17·0	1·65
	747	16·2	1·70*
	920†	—	1·725

* Near-optimum † Not converged ‡ Diverging

Table D

MISMATCH PLOT (BOOST)

Nodes	Iterations	Mismatch	Boost
IEEE 118	161	0·37	0·75
	130	0·27	0·80
	89	0·13	0·85
	78	0·07	0·86
	67	0·15	0·87*
	78	0·11	0·88
CEGB 276	360	3·5	0·70
	251	2·4	0·80
	121	1·8	0·90*
	247‡	0·4	0·95
CEGB 515	684	3·6	0·85
	456	1·1	0·92*
	—	—	0·95‡

* Near-optimum † Diverging ‡ Unstable

Table E

RATE OF CONVERGENCE (TWO POINTS)

Nodes	$N1$	$M1$	$N2$	$M2$	A = acceleration B = boost
IEEE 118	40	23·0	97	1·57	1·74A
	49	1·05	67	0·15	0·87B
CEGB 276	200	20·0	285	6·8	1·65A
	80	11·0	121	1·8	0·90B
CEGB 515	300	98·0	747	16·2	1·70A
	300	5·8	456	1·1	0·92B

$N1$, $N2$ = number of iterations
$M1$, $M2$ = mismatches

Table F

ITERATIONS FOR SAME MISMATCH SOLUTION

Nodes	Acceleration	Extrapolation*	Boost
IEEE 118	97	—	45
CEGB 276	285	200	95
CEGB 515	747	410	200

* Combined with acceleration

Table G

COMPUTING TIME IN SECONDS (CEGB CASES)

Time to reach	Nodes	Time (A)	Time (B)	Ratio of times
Voltage tolerances	276	40	21	4·0A/B*
	515	175	131	13·5A/B*
100 iterations	276	14·6	17·3	1·19B/A
	515	23·4	28·8	1·23B/A
Same mismatches	276	40	15·6	2·6A/B
	515	175	57·5	3·0A/B

* Mismatch ratio
A = acceleration
B = boost

tolerances only are of little value. I have taken the trouble to provide mismatch values for the test examples used in my paper. They are shown in Table C (A.G.S.) and Table D (B.G.S.). These mismatches are the maximum errors obtained when convergence to the conventional voltage tolerance of 0·0001 p.u. is used. Near-optimum boost values can be obtained using the simple formula $B = 0·94 - (10/N)$ for a network of N nodes. I see no point in further complicating the choice of B. The bootstrap method is applied by simply adding $-\Delta V_s^* Y_{rs}^* V_r$ to a running mismatch sum at each node r (connected to s) during iteration on node s. The successor nodes $r > s$ are ignored during the s iteration, and the partial products in this calculation are already available in the normal G.S. procedure. Consequently B.G.S. iterations only cost about 20% more than A.G.S. iterations as shown in Table G. The inherent simplicity of the G.S. technique is thus seen to be preserved. Table E shows two points on each of the linear convergence curves (obtained by plotting log M against N) for several test examples. However, comparisons based on convergence rates alone are suspect, since they take no account of the logic required to program the processes. We have some experience of techniques which, in terms of number of operations, seemed superior, until they were programmed. Table F shows the number of iterations necessary to reach the same mismatch solutions, regardless of voltage tolerance, and results for the massive extrapolation technique (Reference 4 in Paper 5830 P) are included. For large values, the B.G.S. technique is clearly superior to the other methods when the increased cost of the iterations are taken into account. Only 200 iterations are required, compared with 747 (near-optimum A.G.S.) for the 515 node case, and the iteration time has been reduced by 66%. These networks are not particularly easy to solve, but they are not ill conditioned for the G.S. technique—such cases do not often arise.

L. L. Freris and **A. M. Sasson** (*in reply*): The authors agree with Mr. Cory's statement that methods implicitly

```
*************************************
*************************************

            10/07/75
ALGOL LPRES


SEGMENT         AREA   1  SIZE  94
SEGMENT         AREA   1  SIZE  337
SEGMENT         AREA   1  SIZE  357
SEGMENT         AREA   1  SIZE  464
SEGMENT         AREA   1  SIZE  124
HEI2A SHORT CIRCUIT

FREE STORE         MS 2230     BS 7226
```

```
HEI2a Short circuit;
begin comment calculates the bus impedance matrix for a network with or without mutual couplings;
          integer A,D,I,J,K,L,Q,m,n,p,C5,C6,C9; real a; integer array str[1:40];
          I:=1; instring(str,I); read a,n,m,p;
          comment mva base, number of lines, number of mutuals,highest bus number;
          K:= if m<5 then 5 else m;
               begin integer array connec, linind[1:n,1:2],busind[1:p],mutind[1:K,1:3],muline[1:3*K,1:2];
                    array react,resis[1:n],remat,immat[1:(p+4)*(p+5) div 2],reimped,imimped[1:3*K];
                    comment maximum of four loops if the highest numbered bus is established,more than
                    four if it is not established;
                    switch s:=l1,l2,l3,l4,l5;
                    comment insert procedures start,grouping,radial line,dummy bus,mutuals,
                         reorder,generator,reduce matrix,invert,lineflow,finish;
procedure start;
        comment uses global variables a,m,n,p,connec,resis,react;
               begin comment segment[1]; integer i,j,k,q,t; real r,x,y,v ;
                    t:=0;
                    for i:=1 step 1 until n do
                         begin read q,j,k,r,x,v;
                              if j=k or j>p or k>p or j<0 or k<0 then
                                   begin print ££l?line data error?,q,j,k;
                                        t:=t+1
                                   end;
                              connec[i,1]:=if j>k then k else j;
                              connec[i,2]:=if j>k then j else k;
                              y:=if v>1.1 then a/(v*v) else v; resis[i]:=r*y; react[i]:=x*y
                              end reading of line data,storing line connections in connec with lower
                              numbered bus first,per unit values of reactance and resistance in react and resis;
                    if t>0 then stop
               end t will have a positive value if there is a mistake in the bus numbers, in this case the numbers of
                    the line and its buses are printed and the program stops;
procedure grouping;
comment uses global variables a,m,C5,C6,C9,mutind,muline,react,resis,reimped,imimped,linind;
     begin comment segment[1]; integer d,f,g,h,i,j,k,q,r,t;           real u,v,x,y,z;
          integer array fircon,secon,ind[1:m]; real array ristce,reatce[1:m];
          switch ss:=l11,l12,l13,l14,l15,l16;
          h:=0;
          for i:=1 step 1 until m do
               begin read g,j,u,k,v,x,y;
                    if j=k or j>n or k>n or j<1 or k<1 then
                         begin print ££l?mutual data error?,g,j,k; h:=h+1
                         end;
                    fircon[i]:=j; secon[i]:=k; ind[i]:=0;
                    z:=if abs(u-1.0)>0.1 then a/(u*v) else 1.0;
                    ristce[i]:=x*z; reatce[i]:=y*z
               end reading mutual coupling data storing line numbers and per unit values of resistance and reactance;
          if h>0 then stop;
          C6:=C9:=i:=0;
l11:      i:=i+1;    if i>m then goto l16;
          if ind[i]≠0 then goto l11;
          C5:=C5+1;  ind[i]:=C5;
          j:=fircon[i];  k:=secon[i];
          linind[j,2]:=C5; linind[k,2]:=C5;
          muline[C6+1,1]:=j; muline[C6+2,1]:=k;
          d:=2; g:=1; h:=C9+g+(d-1)*(d-2) div 2;
          reimped[h]:=ristce[i]; imimped[h]:=reatce[i];
l12:      k:=i;
l13:      k:=k+1;    if k>m then goto l14;
```

```
          if ind[k]≠0 then goto 113;
          q:=fircon[k]; r:=secon[k];
          if q=j or r=j then
               begin switch sss:=1111,1112;
                    ind[k]:=C5; t:=if q=j then r else q; f:=g;
          1111:     f:=f+1;
                    if f>d then
                         begin d:=d+1; muline[C6+d,1]:=t; linind[t,2]:=C5; goto 1112
                         end storing line number;
                    if t≠muline[C6+f,1] then goto 1111;
          1112:     h:=C9+g+(f-1)*(f-2) div 2; reimped[h]:=ristce[k]; imimped[h]:=reatce[k]
               end storing resistance and reactance in group impedance matrix;
          goto 113;
114:      g:=g+1;      if g>d then goto 115;
          j:=muline[C6+g,1];      goto 112;
115:      mutind[C5,1]:=d; mutind[C5,2]:=C6+1; mutind[C5,3]:=C9+1;
          C6:=C6+d; C9:=C9+d*(d-1) div 2; goto 111;
116:      for i:=1 step 1 until C6 do muline[i,2]:=0
     end grouping which sorts mutual impedances into groups, C5 number of groups, mutind gives the number of lines in
          each group, the location of these lines in muline and the location of the mutual impedance values in imimped
          and reimped;

          procedure radial line;
          comment uses global variables I,J,K,A,resis,react,remat,immat,busind,linind;
               begin integer i,j,k,q,r;
                    q:=J*(J-1) div 2;  r:=K*(K-1) div 2;
                    for i:=1 step 1 until A do
                         begin j:=if i>J then J+i*(i-1) div 2 else i+q;
                              k:= if i>K then K+i*(i-1) div 2 else i+r;
                              if busind[J]=0 then
                                   begin immat[j]:=immat[k]; remat[j]:=remat[k]
                                   end else
                                   begin immat[k]:=immat[j]; remat[k]:=remat[j]
                                   end
                         end;
                    if busind[J]=0 then
                         begin immat[q+J]:=immat[r+K]+react[I]; remat[q+J]:=remat[r+K]+resis[I];
                              busind[J]:=1
                         end else
                         begin immat[r+K]:=immat[q+J]+react[I]; remat[r+K]:=remat[q+J]+resis[I];
                              busind[K]:=1
                         end;
                    if K>A then A:=K;  linind[I,1]:=1
               end radial line which adds a line from an established bus to a new bus, initiates an established bus
                    and a used line, updates A if necessary;
procedure dummy bus;
comment uses global variables I,J,K,A,D,resis,react,immat,remat,linind;
     begin integer i,j,k,q,r,t;
          q:=(A+D)*(A+D+1) div 2;  r:=J*(J-1) div 2;  t:=K*(K-1) div 2;
          for i:=1 step 1 until A+D do
               begin j:= if i>J then J+i*(i-1) div 2 else i+r;
                    k:= if i>K then K+i*(i-1) div 2 else i+t;
                    immat[q+i]:= immat[j]-immat[k]; remat[q+i]:=remat[j]-remat[k]
               end;
          immat[q+A+D+1]:=immat[q+J]-immat[q+K]+react[I];
          remat [q+A+D+1]:=remat[q+J]-remat[q+K]+resis[I];
          D:=D+1;  linind[I,1]:=1
     end dummy bus which adds an element forming a loop in the network, initiates a used element and adds one to D;
```

```
procedure mutuals(E,R,U,V,W,X,Y);
comment uses global variables I,J,K,immat,remat,connec and procedures dummy bus,radial line;
value E,R,W,X,Y; integer E,R; integer array U,V,W; real array X,Y;
      begin comment segment[1]; integer i,j,k,t,u,v,w;
            integer array ind[1:R,1:2]; real array improd,reprod[1:R,1:R];
            switch ss:=111,112,113,114,115,116,117;
procedure product(T,G,H,Z);
value T,G,H; integer T; integer array G; real array H,Z;
      begin real z;
            for i:=1 step 1 until T do
            for j:=1 step 1 until T do
                  begin Z[i,j]:=0.0;
                        for k:=1 step 1 until T do
                              begin z:=0.0;
                                    for t:=1 step 1 until T do z:=z+H[k,t]*G[t,j];
                                    Z[i,j]:=Z[i,j]+G[k,i]*z
                              end
                  end Z is a second order matrix of size T and is the product transpose of matrix G
                        times matrix H times matrix G and holds the mutual resistances or reactances to be added
      end product;
                        for i:=1 step 1 until R do
                        if i>E then
                              begin I:=V[i-E,2]; J:=connec[I,1]; K:=connec[I,2]; dummy bus
                              end else
                              begin I:=U[i,2]; J:=connec[I,1]; K:=connec[I,2];  radial line
                              end adds first elements which establish new buses to imaat and remat,
                                    then adds elements which form loops, no mutual impedances added at this stage;
                        product(R,W,X,improd); product(R,W,Y,reprod);
                        for i:=1 step 1 until R do
                              begin ind[i,1]:=i;  ind[i,2]:= if i>E then A-E+i else U[i,3]
                              end at this stage the second location of ind lists the new
                                    buses in the order in which they have been established and
                                    also the dummy bus numbers;
                        i:=0;
            111:        i:=i+1;  if i>E then goto 114;  j:=i;
            112:        j:=j+1;  if j>E then goto 111;
                        if ind[i,2]<ind[j,2] then goto 112;  t:=0;
            113:        t:=t+1;  k:=ind[i,t]-ind[j,t];  ind[i,t]:=ind[i,t]-k;
                        ind[j,t]:=ind[j,t]+k;  if t=1 then goto 113;  goto 112;
                        comment the second location of ind now the bus numbers in ascending order and the first
                                    location gives the order in which the bus was established;
            114:                    i:=R;
            115:        w:=ind[i,2]; u:=ind[i,1]; k:=w*(w+1) div 2;
                        immat[k]:=immat[k]+ improd[u,u]; remat[k]:=remat[k]+ reprod[u,u];
                        if i=1 then goto 117;   j:=i;
            116:        j:=j-1;  if j=0 then
                              begin i:=i-1;  goto 115
                              end;
                        t:=ind[j,2]; v:=ind[j,1]; k:=w*(w-1) div 2+t;
                        immat[k]:=immat[k]+improd[u,v]; remat[k]:=remat[k]+reprod[u,v];
                        goto 116;
                        comment adds group mutual impedances to immat and remat;
            117:  end mutuals;
procedure reorder(lab);
comment uses global variables I,Q,linind,busind,muline,mutind,reimped,imimped
            and requires procedure mutuals declared first;
label lab;
      begin integer d,e,f,g,h,i,j,k,q,r,t,u,v,w;
```

```
e:=f:=0;  Q:=linind[I,2];  r:=mutind[Q,1];  k:=mutind[Q,2]-1;
          begin integer array new[1:r,1:4],loop[1:r,1:2],mat[1:r,1:2*r];
                real array immumat,remumat[1:r,1:r];
                switch ss:=l11,l12,l13,l14,l15,l16,l17,l18,l19,l110,l111,l112,l113,l114,l115;
                for i:=1 step 1 until r do
                    begin for j:=1 step 1 until r do immumat[i,j]:=remumat[i,j]:=0;
                          for j:=1 step 1 until 2*r do mat[i,j]:=0
                    end;
                l11:i:=h:=0;
                l12:i:=i+1;  if i>r then
                    begin for i:=1 step 1 until r do muline[k+i,2]:=0;  goto lab
                    end exits as group cannot be added at this stage;
                if muline[k+i,2]≠0 then goto l12;
                q:=muline[k+i,1];  w:=connec[q,1];  t:=connec[q,2];
                if busind[w]≠0 and busind[t]≠0 then goto l14;
                comment this is a loop between established buses w and t;
                if busind[w]≠ 0 and busind[t]=0 then
                    begin switch sss:=again;  j:=0;
                again:      j:=j+1;  if j>e then goto l16;
                            if t≠new[j,3] then goto again;  g:=-1;  goto l13
                    end this is a new bus t from established bus w or a loop
                            between established bus w and new bus t;
                if busind[w]=0 and busind[t]≠0 then
                    begin switch sss:=again;  j:=0;
                again:      j:=j+1;  if j>e then goto l15;
                            if w≠new[j,3] then goto again;  g:=1;  goto l13
                    end this is a new bus w from established bus t or a loop
                            between established bus t and new bus w;
                    begin switch sss:=again;  j:=0;
                again:      j:=j+1;  if j>e then goto l12;
                            if w=new[j,3] then
                                begin switch ssss:=repeat;  v:=j;  h:=1;
                            repeat:     j:=j+1;  if j>e then goto l16;
                                        if t≠new[j,3] then goto repeat;  g:=-1;  goto l13
                                end this is a new bus t from a new bus w or a loop
                                        between new bus t and new bus w;
                                begin switch ssss:=repeat;  if t≠new[j,3] then goto again; v:=j; h:=1;
                            repeat:     j:=j+1; if  j>e then goto l15;
                                        if w ≠ new[j,3] then goto repeat; g:=1; goto l13
                                end this is a new bus w from a new bus t or a loop
                                        between new bus w and new bus t
                    end;
                l13: h:=h+1;
                l14: f:=f+1; loop[f,1]:=i; loop[f,2]:=q; d:=r+f;  goto l18;
                l15: e:=e+1; new[e,3]:=w;  new[e,4]:=t; mat[e,e]:=-1;  goto l17;
                l16: e:=e+1; new[e,3]:=t;  new[e,4]:=w;  mat[e,e]:=1;
                l17: new[e,1]:=i; new[e,2]:=q; d:=e; g:=1; if h=1 then begin u:=j; goto l110 end;
                l18: if h=0 then goto l112;
                l19: mat[j,d]:=mat[j,d]+( if new [j,4]<new[j,3] then g else -g);  u:=j;
                l110:  j:=j-1; if j=0 then goto l111 ;
                if new [u,4]=new[j,3] then goto l19 else goto l110;
                l111:  if h=1 then goto l112; g:=-g;  h:=h-1; j:=v;  goto l19;
                comment updates loop or new bus count, records the order of the line in the group and
                        its number in the line list in the first and second locations of loop or new,
                        if a new bus records the bus numbers in third and fourth locations of new,
                        the new bus number being in the third location. Puts plus or minus one if
                        necessary in the r+f or e columns of mat;
      l112:     muline[k+i,2]:=1;  if e+f <r then goto l11;
```

```
                    for i:=1 step 1 until f do
                          begin for j:=1 step 1 until e do mat[j,e+i]:=mat[j,r+i];
                                mat[e+i,e+i]:=1
                          end completes connection matrix mat and shifts column r+f to column e+f
                                making a square matrix of order r;
                      v:=mutind[Q,3]-1;  i:=0;
              1113:   i:=i+1;  if i=r then goto 1115;
                      if i>e then w:=loop[i-e,1] else w:=new[i,1];  j:=i;
              1114:   j:=j+1;  if j>r then goto 1113;
                      if j>e then t:=loop[j-e,1] else t:=new[j,1];
                      if w>t then u:=(w-1)*(w-2) div 2+t else u:=(t-1)*(t-2) div 2+w;
                      immumat[i,j]:=immumat[j,i]:=imimped[u+v]; remumat[i,j]:=remumat[j,i]:=reimped[u+v];
                      goto 1114;
                      comment forms square matrices immumat and remumat of order r, main diagonal terms zero,
                                off diagonal terms equal to the mutual impedances between the elements of
                                group Q in the order in which these are to be added to immat and remat;
              1115:   mutuals(e,r,new,loop,mat,immumat,remumat)
                    end
                end reorder;
            procedure generator;
            comment uses global variables,I,A,D,connec,resis,react,busind,linind,immat,remat;
                begin integer i,j,k,q,r;
                    k:=connec[I,2];  q:=k*(k-1) div 2;
                    if busind[k]=0 then
                        begin immat[q+k]:=react[I]; remat[q+k]:=resis[I];
                              if k>A then A:=k;  busind[k]:=1
                        end else
                        begin r:=A*(A+1) div 2;
                              for i:=1 step 1 until A do
                                  begin j:= if i>k then k+i*(i-1) div 2 else i+q;
                                        immat[r+i]:=immat[j]; remat[r+i]:=remat[j]
                                  end;
                              immat[r+A+1]:=immat[q+k]+react[I]; remat[r+A+1]:=remat[q+k]+resis[I];
                              D:=D+1
                        end;
                    linind[I,1]:=1
                end generator which adds an element from the reference bus either establishing a new bus in which
                    case an established bus is initiated and A updated if necessary, or forms a loop in which
                    case D is updated, in either case a used line is initiated;
        procedure reduce matrix;
                comment uses global variables A,D,immat,remat;
                begin integer i,j,k,h;  real e,f,g,r,t,x,y,x1,y1;  switch ss:=111,112,113,114;
          111:      k:=(A+D)*(A+D+1)div 2; x:=remat[k];  y:=immat[k];
                    e:=x*x+y*y; g:=y/e;  f:=x/e;
                    k:=(A+D)*(A+D-1) div 2;  i:=1;
          112:      j:=1;  h:=i*(i-1) div 2;
          113:      y:=immat[k+j]; y1:=immat[k+i]; x:=remat[k+j]; x1:=remat[k+i];
                    r:=x*x1-y*y1; t:=x*y1+x1*y;
                          remat[j+h]:=remat[j+h]-r*f-t*g;
                          immat[j+h]:=immat[j+h]+r*g-t*f;
                    j:=j+1;  if j>i then
                        begin i:=i+1;  if i=A+D then goto 114 else goto 112
                        end;
                    goto 113;
          114:      for i:=1 step 1 until A+D do
                        begin immat[k+i]:=0.0; remat[k+i]:=0.0
                        end;
                    D:=D-1;  if D>0 then goto 111
```

```
end this procedure eliminates the last D rows of the complex matrix stored as remat and immat;
      procedure invert(G,H,X,Y);
      value G,H;  integer G,H;  real array X,Y;
      comment uses global variables mutind,muline,resis,react,imimped,reimped;
            begin integer d,g,h,i,j,k;  real z;
                  d:=mutind[G,1];  g:=mutind[G,2]-1;  h:=mutind[G,3]-1;
                        begin real array S,R[1:d,1:d];
                              switch ss:=111,112,113,114,115,116;
                              for i:=1 step 1 until d do
                                    begin k:=muline[g+i,1];  S[i,i]:=react[k];  R[i,i]:=resis[k];
                                          k:=(i-1)*(i-2) div 2+h;
                                          for j:=1 step 1 until i-1 do
                                                begin S[i,j]:=S[j,i]:=imimped[k+j];
                                                      R[i,j]:=R[j,i]:=reimped[k+j]
                                                end
                                    end;
                              for i:=1 step 1 until d do
                                    begin z:=1.0/(R[i,i]*R[i,i]+S[i,i]*S[i,i]);
                                                      R[i,i]:=-z*R[i,i];  S[i,i]:=z*S[i,i];
                                          j:=0;
                              111:        j:=j+1;
                                          if j=i then goto 111;
                                          if j>d then goto 112;
                                          g:=if j<i then i else j;  h:=if j<i then j else i;
                                          R[g,h]:=R[i,i]*R[h,g]-S[i,i]*S[h,g];
                                                      S[g,h]:=R[i,i]*S[h,g]+S[i,i]*R[h,g];
                                          goto 111;
                              112:        j:=0;
                              113:        j:=j+1;  if j=i then goto 113;
                                          if j>d then
                                                begin if i=d then goto 116;  goto 115 end;
                                          k:=0;
                              114:        k:=k+1;
                                          if k=i then goto 114;  if k>j then goto 113;
                                          g:=if j<i then i else j;
                                          h:=if j<i then j else i;
                                          R[j,k]:=if k<i then R[j,k]+R[i,k]*R[h,g]-S[i,k]*S[h,g]
                                                      else R[j,k]+R[j,i]*R[i,k]-S[j,i]*S[i,k];
                                          S[j,k]:=if k<i then S[j,k]+R[i,k]*S[h,g]+S[i,k]*R[h,g]
                                                      else S[j,k]+R[j,i]*S[i,k]+S[j,i]*R[i,k];
                                          goto 114;
                              115:        for j:=1 step 1 until d-1 do
                                          for k:=j+1 step 1 until d do
                                                begin S[j,k]:=S[k,j];  R[j,k]:=R[k,j]
                                                end
                                    end;
                  116:        for i:=1 step 1 until d do
                                    begin k:=i*(i+1) div 2+H-1 ;  X[k]:=-S[i,i];  Y[k]:=-R[i,i];
                                          k:=i*(i-1) div 2+H-1 ;
                                          for j:=1 step 1 until i-1 do
                                                begin X[j+k]:=-S[i,j];  Y[j+k]:=-R[i,j]
                                                end
                                    end
                        end
        end invert which inverts in situ using diagonal pivots the complex matrix R+jS using
                  the properties that the matrix is symmetrical and has no zero main diagonal terms;
procedure lineflow(N,I,E,G,H,F);
value N,I;  integer N,I;  integer array E;  real array G,H,F;
```

```
comment uses global variables muline, mutind,connec,linind,immat,remat;
      begin integer i,j,k,r,s,t,q,w;
            r:=linind[N,2]; t:=mutind[r,1];
                  begin real array T[1:t,1:2];  switch sss:=1111;
                        q:=I*(I-1) div 2;
                        for i:=1 step 1 until t do
                              begin j:=muline[mutind[r,2]+i-1,1];
                                    w:=connec[j,1];  s:=connec[j,2];
                                    j:=if w>I then w*(w-1) div 2+I else q+w;
                                    k:=if s>I then s*(s-1) div 2+I else q+s;
                                    T[i,1]:=remat[k]-remat[j]; T[i,2]:=immat[k]-immat[j];
                              end;
                        F[1]:=F[2]:=0.0; s:=0;
            1111:       s:=s+1;
                        if muline[mutind[r,2]+s-1,1]≠N then goto 1111;
                        comment s gives the position of line N in muline;
                        q:=s*(s-1) div 2;
                        for i:=1 step 1 until t do
                              begin j:=if i>s then i*(i-1) div 2+s else q+i;
                                    k:=j+E[r]-1;
                                    F[1]:=F[1]+T[i,1]*G[k]-T[i,2]*H[k];
                                                F[2]:=F[2]+T[i,1]*H[k]+T[i,2]*G[k]
                              end calculation of per unit current in line N for a fault on bus I
                  end
      end lineflow;
procedure finish;
comment uses global variables a,n,m,p,C5,L,connec,linind,mutind,resis,react,imimped,reimped,
            remat,immat and requires procedures invert and lineflow declared first;
      begin integer i,j,k,d,e,f,g,h,t,w,I,J,K,M,P,q;  real u,v,x,y,z,x1,y1,x2,y2;
            switch sss:=1111;
            k:=0;
            for i:=1 step 1 until n do if linind[i,1]=0 then
                  begin print ££l?error line not used?,i;  k:=k+1
                  end;
            if k>0 then stop;
            g:=M:=0;
            if m>0 then for i:=1 step 1 until C5 do g:=g+mutind[i,1]*(mutind[i,1]+1) div 2;
            read P;
                  begin comment segment[1]; integer array indic[0:C5],buslin[0:P,1:2],line[1:10];
                        real array imadmit,readmit[0:g],cur[1:2];
                        switch ss:=111,112,113,114,115,116,117,118,119,1110;
                        for i:=1 step 1 until P do read buslin[i,1],buslin[i,2];
                        if m>0 then
                              begin j:=1;  for i:=1 step 1 until C5 do
                                          begin indic[i]:=j;
                                                invert(i,j,imadmit,readmit);
                                                j:=j+mutind[i,1]*(mutind[i,1]+1) div 2
                                          end
                              end inverts the mutual impedance matrices and stores the values in imadmit
                        and readmit the location of each set being given by indic;
                        f:=0;
                        for i:=1 step 1 until p do
                              begin k:=i*(i+1) div 2;  if immat[k]=0.0 then goto 1110;
                                    I:=J:=K:=1;
                        111:        e:=0;  if K>1 then goto 112;
                                    comment number of lines connected to bus i less than ten;
                                    for j:=1 step 1 until n do
                                    if connec[j,1]=i or connec[j,2]=i then
```

```
                         begin e:=e+1;  line[e]:=j
                         end;
      112:          if P>f then for j:=J step 1 until P do
                    if buslin[j,1]=i then
                         begin e:=e+1;  if e>10 then
                                   begin J:=j;  K:=K+1; e:=e-1;  goto 113
                                   end;
                              f:=f+1;  line[e]:=buslin[j,2]
                         end stores in sets of ten the lines connected to bus i
                    and the additional lines for which the line flows are required;
      113:          if M=0 then goto 114;
                              if L+e<56 then goto if I>1 then 116 else 115;
                    top of form;
      114:          M:=M+1;  print ££12s60?PAGE?,M;
print ££12?  BUS        LINE     FROM BUS    TO BUS        CURRENT              IMPEDANCE               FAULT MVA?;
                    L:=if M=1 then 5 else 3;  if I>1 then goto 116;
      115:          x:=remat[k]; y:=immat[k]; z:=1.0/(x*x+y*y); x2:=x*z; y2:=y*z;
      116:          print ££12??,i;
                    q:=i*(i-1) div 2;
                    for j:=1 step 1 until e do
                         begin d:=line[j];
                              w:=connec[d,1];  t:=connec[d,2];
                              h:=if t>i then t*(t-1) div 2+i else q+t;
                              if w=0 then
                                   begin u:=remat[h]; v:=immat[h];
                                        goto 118
                                   end;
                              if m=0 then goto 117;
                              if linind[d,2]=0 then goto 117;
                              lineflow(d,i,indic,readmit,imadmit,cur);
                              goto 119;
      117:                    g:=if w>i then w*(w-1) div 2+i else q+w;
                              u:=remat[h]-remat[g];
                                        v:=immat[h]-immat[g];
      118:                    x1:=resis[d]; y1:=react[d]; z:=x1*x1+y1*y1;
                              cur[1]:=(u*x1+v*y1)/z; cur[2]:=(v*x1-u*y1)/z;
                              comment flow is positive from lower to higher numbered bus;
      119:                    if j>1 then print ££1?    ?;
                              print prefix(£      ?),d,w,t;
                              print £    ?,cur[1],special(2),cur[2],£j?;
                              if j=1 then begin print £    ?,x, special (2),y,£j?;
                              print £    ?,aligned(4,1), x2*a, special(2), y2*a,£j?
                                        end
                         end;  L:=L+e+1;
                    I:=I+1;  if K=I then goto 111;
      1110:end
  end;
read P;  if P=0 then goto 1111; scaled (9); punch (2); prefix(££1??);
  begin comment segment[1]; integer array row[1:P];
       for i:=1 step 1 until P do read row[i];
       print £1  £q?equivalent circuit ?; i:=1; outstring(str,i);
       print ££u? ?,P,P*(P+1) div 2,£  1  1  3  3?;
       for i:=1 step 1 until P do
            begin k:=row[i];  g:=k*(k-1) div 2;
                 for j:=1 step 1 until P do
                 if row [j]≤ k then
                      begin d:= row[j]; h:= g+d;
                           print d,k,remat[h],immat[h]
```

```
                                    end
                                    end print out of terms for equivalent circuit;
                              print £END?
                         end;
1111: end  print out of fault at each bus in mva, per unit imepdance to each bus,mva flow in each
            line connected to the faulted bus and in specified lines, also specified terms of
            the bus impedance matrix;
                    for I:= 1 step 1 until n do linind[I,1]:=linind[I,2]:=0;
                    for I:= 1 step 1 until p do busind[I]:=0;
                    Q:=3*K; for I:=1 step 1 until Q do reimped [I]:=imimped[I]:=0.0;
                    Q:=p*(p+1) div 2; for I:=1 step 1 until Q do
                         begin immat[I]:=0.0; remat[I]:=0.0
                         end;
                    sameline; digits(3); aligned(1,4); I:=1; outstring(str,I); C5:=0;
                    start;
                    if m>0 then grouping;
                    A:=D:=0;
            11:     I:=1;   Q:=1000;
            12:     if linind[I,1]=1 then
                         begin I:=I+1;
                              if I>n then
                                   begin finish;   goto 15
                                   end;
                              goto 12
                         end if element has been used then next element is read,if all elements used then
                              goes to finish;
            13:     if connec[I,1]=0 then
                         begin generator;
                              if D>0 then reduce matrix; goto 11
                         end;
                    if linind[I,2]>0 then
                         begin reorder(14);
                              if D>0 then reduce matrix; goto 11
                         end line element with mutual coupling,all lines of the group are tested to determine
                              whether and in what order they can be added to the matrix,if all cannot be added
                              exits to next element;
                    J:=connec[I,1]; K:=connec[I,2];
                    if busind[J]=0 and busind[K]=0 then goto 14;
                    comment element cannot be added to the matrix as neither bus established;
                    if busind[J]=1 and busind[K]=1 then
                         begin dummy bus;  reduce matrix;  goto 11
                         end line element for which both buses are established and therefore makes a loop
                              in the network;
                    radial line;
                    goto 11;
                    comment line element for which one bus is established and which therefore establishes a new bus;
            14:     I:=I+1;  if I>n then
                         begin finish;  goto 15
                         end;
                    if linind[I,1]=1 then goto 14;
                    if linind[I,2]=Q then goto 14;
                    goto 13;
                    comment next element read and tested to determine whether it has been used or whether it
                              belongs to a group of which all the elements cannot be added at this stage;
            15:  end
end;
```

```
£SAMPLE NETWORK WITH MUTUAL COUPLING?
Basic Data
base mva                            100.0
number of elements                  7
number of mutual couplings          3
number of nodes                     4
Element Data
number       node  connections      self impedance          voltage
1            0          1           0.0         0.04        1.0
2            0          2           0.0         0.08        1.0
3            1          2           0.0         0.08        1.0
4            0          3           0.0         0.10        1.0
5            1          3           0.0         0.30        1.0
6            3          4           0.0         0.20        1.0
7            2          3           0.0         0.40        1.0
Mutual Coupling Data
number       element    voltage     element     voltage     mutual   impedance
1            5          1.0         6           1.0         -0.0       -0.01
2            5          1.0         7           1.0         0.0        0.10
3            6          1.0         7           1.0         -0.0       -0.05
Additional Element Current Distribution Factors             17
1            2          1           4           1           6          1          7
2            1          2           4           2           5          2          6
3            1          3           2           3           3          4          1
4            2          4           3           4           4          4          5
4            7
Equivalent Circuit      0 END
```

SAMPLE NETWORK WITH MUTUAL COUPLING

PAGE 1

BUS	LINE	FROM BUS	TO BUS	CURRENT	IMPEDANCE	FAULT MVA
1	1	0	1	0.7407+0.0000J	0.0000+0.0296J	0.0+3375.0J
	3	1	2	-0.1852+0.0000J		
	5	1	3	-0.0741+0.0000J		
	2	0	2	0.1852+0.0000J		
	4	0	3	0.0741+0.0000J		
	6	3	4	0.0000+0.0000J		
	7	2	3	-0.0000+0.0000J		
2	2	0	2	0.5471+0.0000J	0.0000+0.0438J	0.0+2284.6J
	3	1	2	0.3620+0.0000J		
	7	2	3	-0.0909+0.0000J		
	1	0	1	0.3704+0.0000J		
	4	0	3	0.0825+0.0000J		
	5	1	3	0.0084+0.0000J		
	6	3	4	0.0000+0.0000J		
3	4	0	3	0.7117+0.0000J	0.0000+0.0712J	0.0+1405.1J
	5	1	3	0.1747+0.0000J		
	6	3	4	-0.0000+0.0000J		
	7	2	3	0.1136+0.0000J		
	1	0	1	0.1852+0.0000J		
	2	0	2	0.1031+0.0000J		
	3	1	2	0.0105+0.0000J		
4	6	3	4	1.0000+0.0000J	0.0000+0.2515J	0.0 +397.6J
	1	0	1	0.2037+0.0000J		
	2	0	2	0.1589+0.0000J		
	3	1	2	0.0570+0.0000J		
	4	0	3	0.6374+0.0000J		
	5	1	3	0.1467+0.0000J		
	7	2	3	0.2159+0.0000J		

END

```
********************************
********************************

            15/07/75
ALGOL EDIT  LPRES


SEGMENT      AREA  1 SIZE 110
SEGMENT      AREA  1 SIZE 467
SEGMENT      AREA  1 SIZE 439
SEGMENT      AREA  1 SIZE 697
SEGMENT      AREA  1 SIZE 594
HEI13 SHORT CIRCUIT

FREE STORE     MS 3038   BS 6040
```

```
HEI13 Short Circuit;

begin comment solves sequence network including resistance and mutual coupling by means of the factorised impedance matrix.
        Sequence impedance and fault mva calculated for each bus together with current factors for specified lines;
        integer I,m,n,p,C2,C3,C4,C5,C6,C7,C8,C9;
        real a;  integer array str[1:30];    I:=1; instring(str,I);
        read a,m,p,n; comment base mva,number of lines,mutuals and highest bus number;
        C3:= if p>0 then p+m+m else m; C9:= if p>6 then p else 6;
    begin comment CBS:; integer array linnum,linind[1:m],connec[1:m,1:2],lnbus,busind,order,rowpon,term[1:n],
        mutind[0:p,1:3],muline[0:p+p],buslines,conlines[1:C3+C3]; comment CBS:;
        real array admitt[1:m,1:2],busmat[1:n,1:2],line[1:C3,1:2],reped,imped[1:n],reimped,imimped[1:4*C9];
        comment insert procedures start,organise,factor,impedance,renumber,mutual,admat;

        procedure start;
        comment uses global variables a,m,n,linnum,admitt,connec,linind;
            begin comment segment[1]; integer g,h,i,j,k;  real v,x,y,z;
                h:=0;
                for i:=1 step 1 until m do
                    begin read g,j,k,x,y,v;
                        if j>n or k>n or j<0 or k<0 then
                            begin print ££l?line data error?,g,j,k;  h:=h+1
                            end;
                        linnum[i]:=g; linind[i]:=0;
                        connec[i,1]:=if j>k then k else j;
                        connec[i,2]:=if j>k then j else k;
                        z:= if abs(v-1.0)>0.1 then a/(v*v) else 1.0;
                        admitt[i,1]:=x*z; admitt[i,2]:=y*z
                    end;
                if h≠0 then stop
            end start;
        procedure organise(D);
        value D;  integer D;
        comment uses global variables C8,C2,C4,lnbus,conlines,order, busind,rowpon,.Renumbers network busbars so that
              upper triangular matrix formed by elimination operating in order on rows with fewest remaining terms;
            begin comment segment[1]; integer d,e,f,g,h,i,j,k,p,q,r,t,ltcol,nlone;  boolean b;
                integer array linebus[1:D],row[1:C8+20];  switch ss:=111,112,113;
                C4:=d:=1;  ltcol:=C8;  C2:=D;
                for i:=1 step 1 until D do linebus[i]:=lnbus[i];
                for i:=1 step 1 until ltcol do row[i]:=conlines[i]; if D=2 then goto 113;
                for i:=1 step 1 until D do if linebus[i]=0 then
                    begin busind[i]:=1000;  C2:=C2-1
                    end;
                nlone:=C2-1;
        111:    i:=0;  j:=1;
        112:    i:=i+1;
                if i>D then
                    begin i:=0;  j:=j+1;  goto 112
                    end;
                if j≠linebus[i] then goto 112;
                order[d]:=i; busind[i]:=d; rowpon[d]:=C4;
                C4:=C4+j+1;  d:=d+1;
                    begin integer array number[1:j];  switch sss:=1111,1112;
                        k:=e:=0;
                        for q:=1 step 1 until i do e:=e+linebus[q];
                        f:=e-j+1;  ltcol:=ltcol-j;  linebus[i]:=0;
                        for q:=f step 1 until e do
                            begin k:=k+1;  number[k]:=row[q]
                            end the j busbars connected to i are stored in number and will now be removed from row;
```

```
                    for q:=f step 1 until ltcol do row[q]:=row[q+j];
                         for q:=1 step 1 until j do
                              begin g:=number[q];
                                   for p:=q+1 step 1 until j do
                                        begin h:=number[p];  e:=0;
                                             comment search busbar h for a connection to busbar g;
                                             for k:=1 step 1 until h do
                                                  begin if k=h then f:=e+1;  e:=e+linebus[k]
                                                  end;
                                             for k:=f step 1 until e do if g=row[k] then goto 1111;
                                             comment if connection then exit else make connection g to h;
                                             b:= false;  e:=ltcol:=ltcol+2;
                                             for k:=D step -1 until 1 do
                                                  begin f:=linebus[k];
                                                       if k=g or k=h then
                                                            begin linebus[k]:=f+1;
                                                                 row[e]:= if k=g then h else g;
                                                                 if b then goto 1111;
                                                                 b:= true;  e:=e-1 ;
                                                            end;
                                                       r:=e-f+1;
                                                       for t:=e step -1 until r do row[t]:= if b then row[t-1]
                                                       else row[t-2];
                                                       e:=e-f
                                                  end  k;
                         1111:                  end p now remove connection i from busbar g;
                                   e:=0;  ltcol:=ltcol-1 ;
                                   for k:=1 step 1 until g do
                                        begin r:=linebus[k];
                                             if k=g then
                                                  begin linebus[k]:=r-1;  f:=e
                                                  end;
                                             e:=e+r
                                        end;
                         1112:     f:=f+1;
                                   if row[f]≠i then goto  1112;
                                   for k:=f step 1 until  ltcol do row[k]:=row[k+1]
                              end q
                         end block;
                    if d≠nlone then goto 111;
          113:      e:=2;
                    for i:=1 step 1 until D do
                         begin j:=linebus[i];
                              if j≠0 then
                                   begin order[d]:=i; busind[i]:=d; rowpon[d]:=C4;
                                        d:=d+1;  if d=C2 then C4:=C4+2;
                                        if j≠1 then
                                             begin print ££l?error in organise?,d,i,j,ltcol,row[e];
                                                  e:=e-1
                                             end
                                   end
                         end i;
                    if e≠2 then stop
               end organise;

     procedure factor(D,cmn,rmt,imt);
     value D;  integer D; integer array cmn; real array rmt,imt;
     comment sets up table of factors with renumbered network busbars.  Uses global variables C2,busmat,line,lnbus,
```

```
      buslines,conlines,order,busind,rowpon,term;
 begin comment segment[1]; integer d,e,f,g,h,i,j,k,p,q,t;
      real u,v,w,x,y,z;  switch ss:=111,112,113;  e:=0;
      for i:=1 step 1 until D do
           begin d:=lnbus[i];  if d=0 then goto 113;  f:=e+1;  e:=e+d;
                k:=busind[i]; j:=rowpon[k];
                rmt[j]:=busmat[i,1]; imt[j]:=busmat[i,2];
                d:=0;  t:=k+1;
                for q:=f step 1 until e do
                     begin g:=conlines[q];
                          for p:=t step 1 until C2 do if g=order[p]  then
                               begin j:=j+1; cmn[j]:=p;  d:=d+1;
                                    h:=buslines[q];
                                    rmt[j]:=line[h,1]; imt[j]:=line[h,2];
                                    goto 111
                               end p;
                111:  end q; term[k]:=d;
      113:  end i upper half of admittance matrix stored with renumbered busbars,terms below the main
                diagonal are now eliminated and new terms stored in the allotted spaces;
      i:=0;
112:  i:=i+1;
      j:=rowpon[i]; x:=rmt[j]; y:=imt[j];
      z:=x*x+y*y;  x:=x/z;  y:=-y/z;
      rmt[j]:=x; imt[j]:=y; d:=term[i];
      comment x+jy inverse of diagonal term;
           begin integer array bus,loc[0:d];  switch sss:=1111,1112,1113;
                for k:=1 step 1 until d do
                     begin g:=e:=k+j;  f:=cmn[e];  h:=0;
                  1111:     h:=h+1;  if h=k then goto 1112;
                          p:=bus[h];  if p<f then goto 1111;
                          bus[h]:=f;  f:=p;  e:= loc[h];  loc[h]:=g;  g:=e;
                          goto 1111;
                  1112:     bus[k]:=f;  loc[k]:=g
                     end bus lists the column numbers of row i terms in ascending order, loc lists their locations;
                for k:=1 step 1 until d do
                     begin e:=loc[k];  f:=bus[k];
                          u:=rmt[e]; v:=imt[e];
                          z:=x*u-y*v;  w:=x*v+y*u;
                          rmt[e]:=z; imt[e]:=w;
                          comment u+jv is the off-diagonal term in row f which will be eliminated,z+jw is the
                          corresponding term in row i divided by the diagonal term,the diagonal term in row f is now modified;
                          j:=rowpon[f]; rmt[j]:=rmt[j]-u*z+v*w;  imt[j]:=imt[j]-u*w-v*z;
                          comment row f now searched for terms in same column as remaining terms in row i;
                          h:=term[f];  t:=h+j;
                          for q:=k+1 step 1 until d do
                               begin g:=bus[q];  e:=loc[q];
                                    u:=rmt[e]; v:=imt[e];
                                    for p:=j+1 step 1 until t do if g=cmn[p] then
                                         begin rmt[p]:=rmt[p]-u*z+v*w; imt[p]:=imt[p]-u*w-v*z;
                                              goto 1113
                                         end modifying row f term,if no term then add new term;
                                    t:=t+1;  h:=h+1; term[f]:=h; cmn[t]:=g;
                                    rmt[t]:=-u*z+v*w; imt[t]:=-u*w-v*z;
                          1113: end q
                     end k
           end block;
      if i<C2 then goto 112
 end factor;
```

```
procedure impedance(D,E,F,G,cmn,rmt,imt);
value D,E,F,G;  integer D,E,F,G;  integer array cmn;  real array rmt,imt;
comment calculates reped+j imped i.e. row D of the impedance matrix from the table of factors.  Uses global  variables C2,
          rowpon,term,reped,imped;
begin     integer d,e,f,g,i,j,k;  real u,v,w,x,y,z;  switch ss:=111,112; if G=2 then goto 112;
          k:=D+1;
          for i:=k step 1 until C2 do
               begin reped[i]:=0.0; imped[i]:=0.0
               end;
          e:=rowpon[D]; d:=term[D];
          reped[D]:=rmt[e]; imped[D]:=imt[e];
          for i:=1 step 1 until d do
               begin f:=e+i;  g:=cmn[f];
                    reped[g]:=-rmt[f]; imped[g]:=-imt[f]
               end;
          for i:=k step 1 until C2 do
               begin x:=reped[i]; y:=imped[i];
                    if x=0.0 and y=0.0 then goto 111;
                    d:=term[i]; e:=rowpon[i];
                    for j:=1 step 1 until d do
                         begin f:=e+j;  g:=cmn[f];
                              u:=-rmt[f]; v:=-imt[f];
                              reped[g]:=reped[g]+x*u-y*v;  imped[g]:=imped[g]+x*v+y*u
                         end;
                    u:=rmt[e]; v:=imt[e];
                    reped[i]:=x*u-y*v; imped[i]:=x*v+y*u;
111:           end;
112:      for i:=F-1 step -1 until E do
               begin x:= if i<D then 0.0 else reped[i];
                    y:= if i<D then 0.0 else imped[i];
                    e:=rowpon[i]; d:=term[i];
                    for j:=1 step 1 until d do
                         begin f:=e+j;  g:=cmn[f];
                              u:=-rmt[f]; v:=-imt[f];
                              z:=reped[g]; w:=imped[g];
                              x:=x+u*z-v*w;  y:=y+u*w+v*z
                         end j;
                    reped[i]:=x; imped[i]:=y
               end i
end impedance;

procedure renumber(D,N);
comment uses global variables m,linnum;
value D; integer D; integer array N;
     begin integer i,j,k; switch ss:= 111;
          for i:=1 step 1 until D do
               begin k:= N[i];
                    for j:=1 step 1 until m do if k=linnum[j] then goto 111;
          111:      N[i]:= j
               end
     end renumber;

procedure mutual;
comment uses global variables a,p,C3,C4,C5,C6,C7,C8,C9,linnum,linind,mutind,muline,line,reimped,imimped,admitt,busmat,lnbus,
          buslines,conlines,order,busind,rowpon,term,reped,imped and requires procedures renumber, organise, factor and impedance;
     begin integer d,e,f,g,h,i,j,k,q,r,t;  real u,v,x,y,z;
          integer array fircon,secon,ind,mut[1:p];
```

```
        real array resis,react[1:p];  switch ss:=111,112,113,114,115,116,117;
        for i:=1 step 1 until p do
             begin read g,j,u,k,v,x,y;
                  fircon[i]:=j;  secon[i]:=k;  ind[i]:=0;
                  z:= if abs(u-1.0)>0.1 then a/(u*v) else 1.0;
                  resis[i]:=x*z;  react[i]:=y*z
             end reading mutual coupling data,converting to per unit and storing for sorting;
        renumber(p,fircon);  renumber(p,secon);
        i:=C9:=0;
111:    i:=i+1;  if i>p then goto 117;
        if ind[i]≠0 then goto 111;
        C5:=C5+1;  ind[i]:=C5;
        j:= fircon[i];  k:= secon[i];
        linind[j]:=C5; linind[k]:=C5;
        mut[1]:=i;  muline[C6+1]:=j;  muline[C6+2]:=k;
        d:=2;  C3:=g:=1;
112:    k:=i;
113:    k:=k+1;  if k>p then goto 114;
        if ind[k]≠0 then goto 113;
        q:= fircon[k];  r:= secon[k];
        if q=j or r=j then
             begin switch sss:=1111;
                  ind[k]:=C5;  C3:=C3+1;  mut[C3]:=k;
                  t:= if q=j then r else q; f:=g;
             1111: f:=f+1;
                  if f>d then
                       begin d:=d+1;  muline[C6+d]:=t; linind[t]:=C5;  goto 113
                       end;
                  if t≠muline[C6+f] then goto 1111
             end; goto 113;
114:    g:=g+1;  if g>d then goto 115;
        j:=muline[C6+g];  goto 112;
115:    mutind[C5,1]:=d;  mutind[C5,2]:=C6;  mutind[C5,3]:=C9; C8:=2*C3;  e:=0;
        for j:=1 step 1 until d do
             begin h:=muline[C6+j];  t:=0;
                  for k:=1 step 1 until C3 do
                       begin q:=mut[k];  f:= fircon[q];  g:= secon[q];
                            if h=f or h=g then
                                 begin e:=e+1;  t:=t+1; buslines[e]:=k;
                                      r:= if h=f then g else f;
                                      for q:=1 step 1 until d do if r=muline[C6+q] then
                                           begin conlines[e]:=q;  goto 116
                                           end
                                 end;
                  116:  end k; lnbus[j]:=t;
                  busmat[j,1]:=admitt[h,1]; busmat[j,2]:=admitt[h,2]
             end j;
        for j:=1 step 1 until C3 do
             begin  h:=mut[j]; line[j,1]:=resis[h]; line[j,2]:=react[h]
             end setting up impedance matrix for inverting;  organise(d);
             begin integer array column[1:C4]; real array remat,imat[1:C4]; factor(d,column,remat,imat);
                  for j:=1 step 1 until d do
                       begin h:=busind[j];  k:=j*(j-1) div 2;
                            impedance(h,h,d,1,column,remat,imat);
                            for q:=h step 1 until d do
                                 begin g:= if h=q then j else order[q];
                                      f:= if g>j then g*(g-1) div 2+j else k+g;  g:=C9+f;
                                      reimped[g]:=reped[q]; imimped[g]:=imped[q]
```

```
                                   end
                              end
                         end storing admittance matrix of mutual couplings;
                    C9:=C9+d*(d+1) div 2;  C6:=C6+d;  if d>C7 then C7:=d;  goto 111;
117:  end mutual;

procedure admat;
comment uses global variables C3,C5,C7,C8,m,n,linind,mutind,muline,reimped,imimped,admitt,
                    connec,busmat,line,lnbus,buslines,conlines,reped,imped;
     begin comment segment[1]; integer d,e,f,g,h,i,j,k,q,r,t,gr,tr,fp,jp,kp,tp,sw;  real x,y,z;
                    integer array loc[1:n],at[0:C8,0:C7],bus[0:C8];
                    switch ss:=111,112,113,114,115,116,117,118,119,1110,1111,1112,1113,1114;
                    for i:=1 step 1 until n do
                         begin loc[i]:=0; lnbus[i]:=0; for j:=1,2 do busmat[i,j]:=0.0
                         end;
                    i:=C3:=C8:= 0;  sw:=1;
111:                i:=i+1;  if i>m then goto 114;
                    if linind[i]≠0 then goto 111;
                    x:=admitt[i,1]; y:=admitt[i,2];
                    z:=x*x+y*y;  x:=x/z;  y:=-y/z;
                    admitt[i,1]:=x; admitt[i,2]:=y;
                    j:=connec[i,1]; k:=connec[i,2];
112:                for q:=k,j do
                         begin if q=0 then goto ss[sw];
                              busmat[q,1]:=busmat[q,1]+x;  busmat[q,2]:=busmat[q,2]+y
                         end;
113:                f:=lnbus[j];  g:=loc[j];
                    for q:=1 step 1 until f do if k=conlines[q+g] then
                         begin d:=buslines[q+g];
                              line[d,1]:=line[d,1]-x; line[d,2]:=line[d,2]-y;  goto ss[sw]
                         end;  C3:=C3+1;  C8:= C8+2;
                    if C3>p+m+m then print ££l?admat error?,C3,i,sw;
                    line[C3,1]:=-x; line[C3,2]:=-y;
                    for q:=n step -1 until j do
                         begin f:=lnbus[q];  g:=loc[q];
                              t:= if q>k then g+2 else g+1;
                              if q≠j then
                                   begin for r:=f step -1 until 1 do
                                                  begin gr:=g+r;  tr:=t+r;
                                                       buslines[tr]:=buslines[gr];
                                                       conlines[tr]:=conlines[gr]
                                                  end;
                                        loc[q]:=t
                                   end;
                              if q=k or q=j then
                                   begin lnbus[q]:=f+1 ;  tr:=t+( if q=j then f else f+1);
                                        buslines[tr]:=C3;  conlines[tr]:= if q=k then j else k
                                   end
                         end q;
                    goto ss[sw];
114:                i:=0;
115:                i:=i+1;  if i>C5 then goto 1111;
                    sw:=7;  h:=mutind[i,1];  gr:=2*h;  tp:=mutind[i,3];
                    for jp:=1 step 1 until gr do for kp:=1 step 1 until h do at[jp,kp]:=0;
                    fp:=mutind[i,2];  e:= jp:=0;
117:                jp:= jp + 1;  if jp>h then goto 1112;  f:=muline[fp+jp];  j:=connec[f,1]; k:=connec[f,2];
                              for tr:=j,k do if tr≠0 then
                                   begin for kp:=1 step 1 until e do if tr=bus[kp] then goto 116;
```

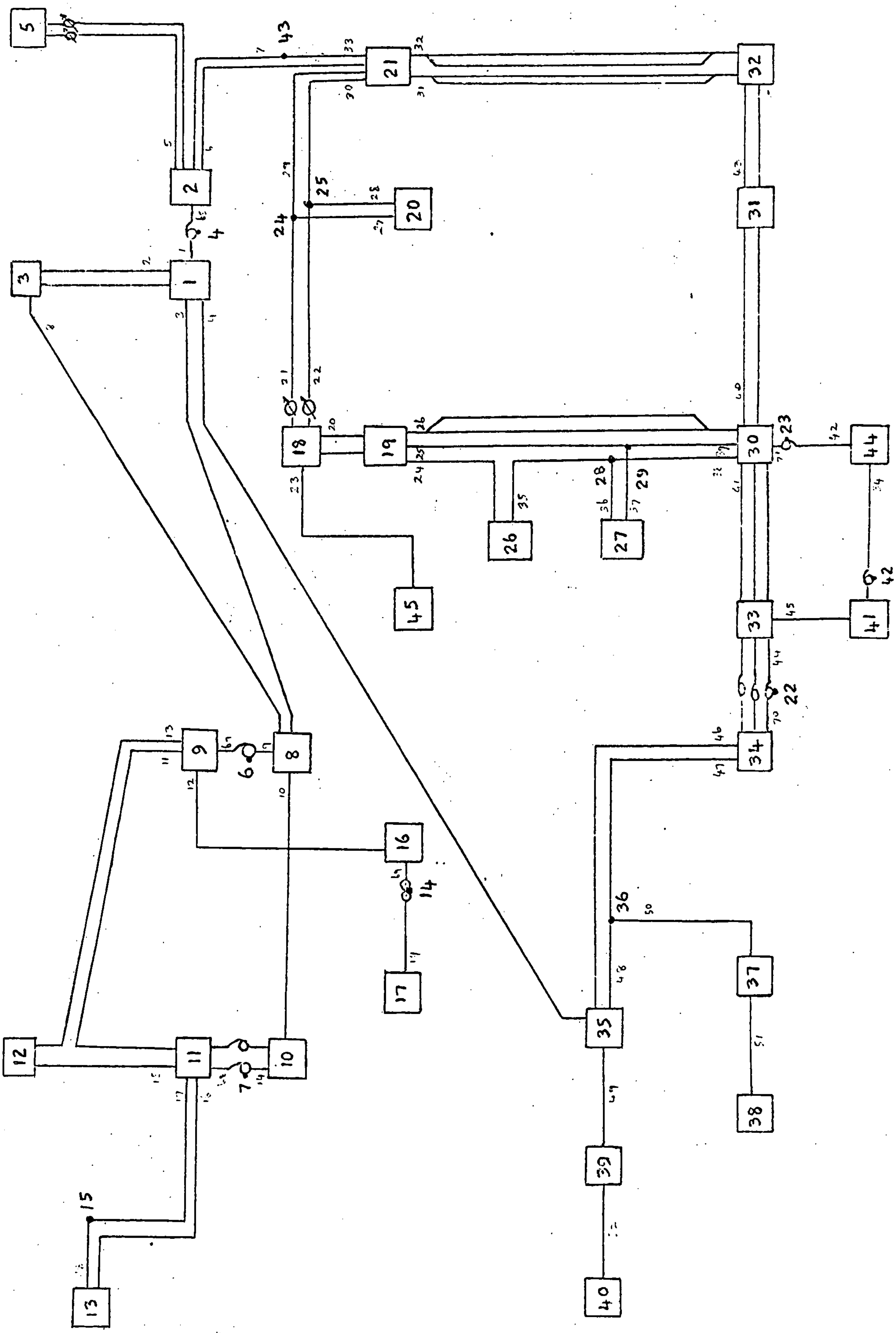
```
                          kp:=e:=e+1;  bus[e]:=tr;
116:                      at[kp,jp]:= if tr=j then -1 else 1
                      end tr forming bus list and incidence matrix;
                  tr:=tp+jp*(jp+1) div 2;  x:=reimped[tr]; y:=imimped[tr];   goto 112; comment adding self admittance terms;
1112:     sw:= 10;  jp:= 0;
1113:     jp:= jp+1;
                  for kp:=1 step 1 until h do
                      begin x:=y:=0.0;  k:=kp*(kp-1) div 2;
                          for tr:=1 step 1 until h do
                              begin if tr=kp then goto 118;
                                  d:=at[jp,tr];  if d=0 then goto 118;
                                  gr:=tp+( if tr>kp then tr*(tr-1) div 2+kp else k+tr);
                                  x:=x+( if d=1 then reimped[gr] else -reimped[gr]);
                                  y:=y+( if d=1 then imimped[gr] else -imimped[gr]);
118:                          end;
                          reped[kp]:=x;  imped[kp]:=y
                      end kp; fp:= bus[jp];  kp:= jp-1;
1114:             kp:= kp+1; x:=y:=0.0;
                          for tr:=1 step 1 until h do
                              begin d:=at[kp,tr];  if d=0 then goto 119;
                                  x:=x+( if d=-1 then reped[tr] else -reped[tr]);
                                  y:=y+( if d=-1 then imped[tr] else -imped[tr]);
119:                          end sign changed so that positive value added to admittance matrix;
                          g:=bus[kp];
                          if jp=kp then
                              begin busmat[g,1]:=busmat[g,1]-x;  busmat[g,2]:=busmat[g,2]-y
                              end else
                              begin  j:= if fp>g then g else fp;
                                  k:= if fp>g then fp else g;  goto 113
                              end;
1110:     if kp<e then goto 1114 else if jp<e then goto 1113 else goto 115;
1111: end admat;
          sameline; digits(3); aligned(1,4);
          start; C5:=C6:=C7:=0;
          if p>0 then mutual; C8:=C7+C7;
          admat;
          organise(n); I:=1; outstring(str,I);
      begin comment CBS:; integer array column[1:C4]; comment CBS:; real array remat,imat[1:C4];
          comment insert procedures lineflow and output;

procedure lineflow(E,F,D,J,X,Y);
value E,F,J;  integer E,F,D,J;  real X,Y;
comment uses global variables mutind,muline,reimped,imimped,connec,reped,imped,busind and procedure impedance;
      begin integer d,e,f,g,i,j,k,q,r,pi,pk;  real u,v,x,y;  switch ss:=111;
          d:=mutind[E,1]; e:=mutind[E,2];
          comment d,e number and location of lines in group;  j:=0;
      111:  j:=j+1;  if muline[e+j]≠F then goto 111;  comment j is position of line in group;
          X:=Y:=0.0;  pi:=mutind[E,3];  pk:=j*(j-1) div 2;
          for i:=1 step 1 until d do
              begin q:=muline[e+i];
                  f:=connec[q,1];  g:=connec[q,2];
                  q:= if f=0 then 1000 else busind[f];
                  r:=busind[g];
                  for k:=q,r do if k<D then
                      begin impedance(J,k,D,2,column,remat,imat);  D:=k
                      end;
                  u:=reped[r]-( if f=0 then 0.0 else reped[q]);
                  v:=imped[r]-( if f=0 then 0.0 else imped[q]);
```

```
                    q:= if i>j then i*(i-1) div 2+j else pk+i;
                    x:=reimped[pi+q]; y:=imimped[pi+q];
                    X:=X+u*x-v*y;  Y:=Y+u*y+v*x
                end i
end lineflow;

procedure output;
comment uses global variables a,m,n,C2,linnum,linind,admitt,connec,busind,order,reped,imped,column,
        remat,imat and procedures renumber, impedance and lineflow;
    begin comment segment[1]; integer d,e,f,g,i,j,k,q,r,t,lh,bh,sw,lines,bus,L,M;  real u,v,w,x,y,z,u1,v1,x1,y1;  boolean b;
        read lines,bus;
        if bus>0 then print punch(2),£1  £q?equivalent circuit value admittance£u??,bus,bus*(bus+1) div 2,£  1  1  3  3?;
            begin integer array buses,lin[0:lines],row[0:bus];
                switch ss:=111,112,113,114,115,116,117,118,119,1110;
                for i:=1 step 1 until lines do read buses[i],lin[i];
                for i:=1 step 1 until bus do read row[i];
                for i:=1 step 1 until bus do
                    begin d:=row[i];
                        k:=f:= if i=1 then busind[d] else d;
                        for j:=i+1 step 1 until bus do
                            begin d:=row[j];
                                g:= if i=1 then busind[d] else d;
                                if g<f then
                                    begin f:=g;  q:=j
                                    end else if i=1 then row[j]:=g
                            end;
                        if k≠f then
                            begin row[i]:=f;  row[q]:=k
                            end else if i=1 then row[i]:=f
                    end sorting re-ordered bus numbers,lowest number first;
                renumber(lines,lin);
                lh:=0;  bh:=M:=1; print ££l2??;
                for i:=1 step 1 until n do
                    begin j:=busind[i]; if j=1000 then goto 119;  sw:=4;
                        if M=1 then goto 112;
                        if L<52 then goto 113;
                    111: top of form;
                    112: print ££s60?PAGE?,M;  M:=M+1;
        print ££l2?  BUS        LINE   FROM BUS   TO BUS      CURRENT            IMPEDANCE            FAULT MVA?;
                        L:= if M=2 then 5 else 3;
                    113: print ££l2??,i;  b:= true;  goto ss[sw];
                    114: impedance(j,j,C2,1,column,remat,imat);  d:=j;
                        u:=reped[j];  v:=imped[j];
                        z:=u*u+v*v;  x:=u/z;  y:=v/z;  sw:=5; k:= 0;
                115:    k:= k+1; if k≤m then goto 117;
                        if lines=lh then goto 1110;  sw:= 6;  e:= 0;
                116:    e:= e+1; if e> lines then goto 1110;  if i ≠ buses [e] then goto 116;
                        lh:= lh+1;  k:= lin[e];  goto 117;
                1110:   goto if bh> bus then 119 else 118;
                    117: f:=connec[k,1];
                        if sw=5 and f>i then goto 115;  g:=connec[k,2];
                        if sw=5 and f≠i and g≠i then goto 115;
                        q:=linind[k];
                        if q=0 then
                            begin q:= if f=0 then 1000 else busind[f];
                                r:=busind[g];
                                for t:=q,r do if t<d then
                                    begin impedance(j,t,d,2,column,remat,imat);  d:=t
```

```
                                        end;
                                    u1:=reped[r]-( if f=0 then 0.0 else reped[q]);
                                    v1:=imped[r]-( if f=0 then 0.0 else imped[q]);
                                    z:=admitt[k,1]; w:=admitt[k,2];
                                    x1:=u1*z-v1*w;  y1:=v1*z+u1*w
                                end else lineflow( q,k,d,j,x1,y1);
                            if not b then print ££1?    ?;
                            print prefix(£      ?),linnum[k],f,g;
                            print £    ?,x1,special(2),y1,£J?;
                            if b then
                                begin print £    ?,u,special(2),v,£J?;
                                      print £    ?,aligned(4,1),x*a,special(2),y*a,£J?
                                end;
                            L:=L+( if b then 2 else 1);
                            if L>56 then
                                begin if sw=6 and lines-lh>4 then goto 111
                                end;
                            if b then b:= false;  goto ss[sw];
                        118: if row[bh]=j then
                                begin for k:=bh step 1 until bus do
                                        begin d:=row[k];  f:=order[d];
                                            print punch(2),prefix(££1??),i,f,scaled(9),reped[d],imped[d]
                                        end;  bh:=bh+1
                                end printing terms for equivalent circuit;
                    119:  end i;
                    if bus>0 then print punch(2),£END?
                end block
        end output;
            factor(n,column,remat,imat);
            output
        end
        end
    end short circuit;
```



£ZERO SEQUENCE 1968 MAXIMUM PLANT SAMPLE REDUCED NETWORK?

MVA base 100.0
No of lines 77
No of mutuals 52
Highest bus 45

Line data	1	1	4	0.0	0.105	1.0
	2	1	3	16.748	91.349	220.0
	3	1	8	21.39	98.83	220.0
	4	1	35	24.73	98.53	220.0
	5	2	5	10.382	57.22	110.0
	6	2	21	20.53	71.78	110.0
	7	2	43	11.0	40.4	110.0
	8	3	8	14.13	91.4	220.0
	9	8	6	0.0	0.1	1.0
	10	8	10	12.43	59.24	220.0
	11	9	11	14.13	59.9	110.0
	12	9	16	13.23	25.24	110.0
	13	9	12	13.92	59.11	110.0
	14	10	7	0.0	0.055	1.0
	15	11	12	0.88	3.72	110.0
	16	11	13	23.68	79.0	110.0
	17	11	15	22.95	76.7	110.0
	18	13	15	0.706	2.36	110.0
	19	17	14	13.22	48.65	22.0
	20	18	19	0.086	0.349	110.0
	21	18	24	4.65	26.71	110.0
	22	18	25	4.65	26.71	110.0
	23	18	45	7.82	25.42	110.0
	24	19	26	19.18	72.63	110.0
	25	19	29	24.98	106.22	110.0
	26	19	30	25.005	111.636	110.0
	27	20	24	0.57	1.83	110.0
	28	20	25	0.57	1.83	110.0
	29	21	24	5.37	25.29	110.0
	30	21	25	5.37	25.29	110.0
	31	21	32	25.636	121.645	110.0
	32	21	32	28.986	121.719	110.0
	33	21	43	8.05	26.13	110.0
	34	42	44	11.53	22.07	88.0
	35	26	28	6.62	37.34	110.0
	36	27	28	1.43	5.4	110.0
	37	27	29	1.43	5.4	110.0
	38	28	30	5.94	28.55	110.0
	39	29	30	5.94	28.55	110.0
	40	30	31	0.663	3.99	110.0
	41	30	33	0.425	2.435	110.0
	42	44	23	16.7	32.84	88.0
	43	31	32	0.573	3.025	110.0
	44	33	22	0.0	-0.003	1.0
	45	33	41	9.6	45.08	110.0
	46	34	35	22.72	108.49	220.0
	47	34	36	16.34	85.5	220.0
	48	35	36	6.38	22.99	220.0
	49	35	39	4.23	3.43	220.0
	50	36	37	1.05	6.04	220.0
	51	37	38	0.61	3.54	220.0
	52	39	40	4.45	3.62	220.0

	53	0	1	0.0	0.048	1.0	
	54	0	2	0.0026	0.0896	1.0	
	55	0	5	0.0	0.11	1.0	
	56	0	18	0.0	0.07	1.0	
	57	0	19	0.0	0.086	1.0	
	58	0	26	0.0	0.20	1.0	
	59	0	35	0.0	0.134	1.0	
	60	0	37	0.0	0.568	1.0	
	61	0	38	0.0	0.284	1.0	
	62	0	39	0.0	0.26	1.0	
	63	0	40	0.0	0.185	1.0	
	64	0	45	0.0	0.645	1.0	
	65	2	4	0.0	-0.0167	1.0	
	66	0	4	0.0	0.183	1.0	
	67	6	9	0.0	-0.0136	1.0	
	68	7	11	0.0	-0.0084	1.0	
	69	14	16	0.0	1.0	1.0	
	70	34	22	0.0	0.031	1.0	
	71	30	23	0.0	0.052	1.0	
	72	0	6	0.0	0.246	1.0	
	73	0	7	0.0	0.193	1.0	
	74	0	22	0.0	0.058	1.0	
	75	0	23	0.0	0.9	1.0	
	76	0	42	0.0	0.9	1.0	
	77	0	14	0.0	0.35	1.0	
mutual data	1	42	88.0	45	110.0	0.569	2.232
	2	35	110.0	25	110.0	3.744	23.763
	3	35	110.0	26	110.0	1.872	18.749
	4	35	110.0	24	110.0	-0.237	-0.59
	5	35	110.0	46	220.0	-2.654	-11.088
	6	35	110.0	47	220.0	-2.028	-9.159
	7	35	110.0	48	220.0	0.626	2.825
	8	25	110.0	24	110.0	7.158	42.922
	9	25	110.0	26	110.0	5.451	56.671
	10	25	110.0	46	220.0	-7.844	-28.972
	11	25	110.0	47	220.0	-5.977	-25.23
	12	25	110.0	48	220.0	1.867	7.87
	13	26	110.0	24	110.0	7.158	34.678
	14	26	110.0	38	110.0	5.353	26.161
	15	26	110.0	39	110.0	5.357	28.385
	16	26	110.0	46	220.0	-9.456	-32.631
	17	26	110.0	47	220.0	-7.206	-27.669
	18	26	110.0	48	220.0	2.25	8.64
	19	38	110.0	39	110.0	2.939	18.089
	20	38	110.0	41	110.0	-0.521	-2.42
	21	38	110.0	46	220.0	-1.612	-6.732
	22	38	110.0	47	220.0	-1.228	-5.539
	23	38	110.0	48	220.0	0.384	1.737
	24	39	110.0	41	110.0	-0.521	-2.64
	25	39	110.0	46	220.0	-1.612	-6.528
	26	39	110.0	47	220.0	-1.228	-5.18
	27	39	110.0	48	220.0	0.384	1.62
	28	36	110.0	37	110.0	0.545	3.459
	29	41	110.0	45	110.0	-0.095	-0.48
	30	41	110.0	26	110.0	-0.521	-3.3
	31	31	110.0	32	110.0	14.315	53.267
	32	4	220.0	29	110.0	2.654	10.752

33	4	220.0	30	110.0	2.654	12.88
34	4	220.0	21	110.0	-0.379	-1.536
35	4	220.0	22	110.0	-0.379	-1.84
36	27	110.0	28	110.0	0.19	0.96
37	23	110.0	29	110.0	-0.64	-3.24
38	7	110.0	6	110.0	4.171	22.84
39	7	110.0	4	220.0	4.171	16.965
40	11	110.0	15	110.0	-0.428	-2.705
41	16	110.0	17	110.0	7.74	48.85
42	16	110.0	18	110.0	-0.238	-1.504
43	6	110.0	33	110.0	-0.735	-0.359
44	6	110.0	4	220.0	4.835	17.61
45	33	110.0	4	220.0	-2.702	-10.26
46	5	110.0	2	220.0	4.55	39.744
47	24	110.0	46	220.0	-5.19	-19.386
48	24	110.0	47	220.0	-3.956	-17.853
49	24	110.0	48	220.0	1.234	5.58
50	46	220.0	47	220.0	9.483	49.64
51	46	220.0	48	220.0	-1.96	-15.5
52	11	110.0	13	110.0	6.806	42.97

Additional Element Current Distribution Factors 8 Equivalent Circuit 0

Elements	19	21	19	22	19	23	19	56
	8	4	8	1	8	2	8	53E

ZERO SEQUENCE 1968 MAXIMUM PLANT REDUCED NETWORK

PAGE 1

BUS	LINE	FROM BUS	TO BUS	CURRENT	IMPEDANCE	FAULT MVA
1	1	1	4	-0.1844+0.0053J	0.0011+0.0289J	129.6+3458.4J
	2	1	3	-0.0297-0.0006J		
	3	1	8	-0.0633-0.0057J		
	4	1	35	-0.1210-0.0215J		
	53	0	1	0.6015-0.0226J		
2	5	2	5	-0.0581-0.0072J	0.0012+0.0334J	103.4+2993.8J
	6	2	21	-0.0251-0.0052J		
	7	2	43	-0.0263-0.0055J		
	54	0	2	0.3724-0.0020J		
	65	2	4	-0.5181+0.0158J		
3	2	1	3	0.6362+0.0145J	0.0177+0.1115J	138.8 +874.9J
	8	3	8	-0.3638+0.0145J		
4	1	1	4	0.2118-0.0119J	0.0018+0.0317J	175.7+3140.0J
	65	2	4	0.6147+0.0216J		
	66	0	4	0.1735-0.0097J		
5	5	2	5	0.1931+0.0286J	0.0031+0.0888J	39.8+1125.3J
	55	0	5	0.8069-0.0286J		
6	9	6	8	-0.4187-0.0038J	0.0052+0.0955J	57.2+1044.0J
	67	6	9	-0.1931-0.0175J		
	72	0	6	0.3882-0.0213J		
7	14	7	10	-0.3240-0.0196J	0.0068+0.1026J	64.2 +970.6J
	68	7	11	-0.1444-0.0155J		
	73	0	7	0.5315-0.0351J		
8	3	1	8	0.3158+0.0246J	0.0084+0.0813J	125.6+1217.5J
	8	3	8	0.2001+0.0094J		
	9	6	8	0.2764-0.0278J		
	10	8	10	-0.2076+0.0063J		
	4	1	35	-0.0667-0.0169J		
	1	1	4	-0.1223-0.0056J		
	2	1	3	0.2001+0.0094J		
	53	0	1	0.3269+0.0115J		
9	11	9	11	-0.0588-0.0078J	0.0048+0.0868J	63.7+1149.2J
	12	9	16	-0.0556-0.0008J		
	13	9	12	-0.0497-0.0048J		
	67	6	9	0.8359-0.0135J		
10	10	8	10	0.4668+0.0361J	0.0098+0.1141J	75.1 +870.0J
	14	7	10	0.5332-0.0361J		

BUS	LINE	FROM BUS	TO BUS	CURRENT	IMPEDANCE	FAULT MVA
11	11	9	11	0.0707+0.0083J	0.0065+0.0965J	69.9+1031.6J
	15	11	12	-0.0598-0.0049J		
	16	11	13	-0.0000+0.0000J		
	17	11	15	-0.0000+0.0000J		
	68	7	11	0.8696-0.0132J		
12	13	9	12	0.1125+0.0042J	0.0127+0.1188J	89.1 +832.0J
	15	11	12	0.8875-0.0042J		
13	16	11	13	0.5003+0.0004J	0.1373+0.6311J	32.9 +151.3J
	18	13	15	-0.4997+0.0004J		
14	19	14	17	0.0000+0.0000J	0.0051+0.2761J	6.7 +362.0J
	69	14	16	-0.2110-0.0146J		
	77	0	14	0.7890-0.0146J		
15	17	11	15	0.5146-0.0004J	0.1353+0.6186J	33.7 +154.3J
	18	13	15	0.4854+0.0004J		
16	12	9	16	0.8140+0.0564J	0.0761+0.2510J	110.6 +364.8J
	69	14	16	0.1860-0.0564J		
17	19	14	17	1.0000+0.0000J	2.7365+1 ₁₀+01J	2.4 +9.0J
18	20	18	19	-0.4263-0.0004J	0.0010+0.0327J	95.7+3057.5J
	21	18	24	-0.0348-0.0063J		
	22	18	25	-0.0351-0.0063J		
	23	18	45	-0.0370-0.0016J		
	56	0	18	0.4668-0.0146J		
19	20	18	19	0.5518+0.0043J	0.0011+0.0330J	100.7+3023.3J
	24	19	26	-0.0251-0.0047J		
	25	19	29	-0.0128+0.0010J		
	26	19	30	-0.0261-0.0048J		
	57	0	19	0.3842-0.0128J		
	21	18	24	-0.0333-0.0063J		
	22	18	25	-0.0336-0.0064J		
	23	18	45	-0.0356-0.0019J		
	56	0	18	0.4492-0.0103J		
20	27	20	24	-0.4968+0.0006J	0.0199+0.1144J	147.9 +848.7J
	28	20	25	-0.5032-0.0006J		
21	6	2	21	0.1479+0.0065J	0.0264+0.1314J	147.1 +731.5J
	29	21	24	-0.2667+0.0091J		
	30	21	25	-0.2628+0.0094J		
	31	21	32	-0.0785+0.0003J		
	32	21	32	-0.0777-0.0034J		
	33	21	43	-0.1664-0.0088J		

PAGE 3

BUS	LINE	FROM BUS	TO BUS	CURRENT	IMPEDANCE	FAULT MVA
22	44	22	33	-0.1451-0.0159J	0.0018+0.0404J	110.0+2471.4J
	70	22	34	-0.1586-0.0151J		
	74	0	22	0.6963-0.0310J		
23	42	23	44	-0.0556-0.0096J	0.0049+0.0930J	56.0+1072.4J
	71	23	30	-0.8410+0.0042J		
	75	0	23	0.1033-0.0054J		
24	21	18	24	0.3674-0.0129J	0.0182+0.1063J	156.7 +914.3J
	27	20	24	0.4704+0.0075J		
	29	21	24	0.1622+0.0055J		
25	22	18	25	0.3665-0.0130J	0.0182+0.1062J	156.8 +915.1J
	28	20	25	0.4639+0.0063J		
	30	21	25	0.1696+0.0068J		
26	24	19	26	0.1426+0.0261J	0.0098+0.1183J	69.6 +839.6J
	35	26	28	-0.2659-0.0230J		
	58	0	26	0.5915-0.0491J		
27	36	27	28	-0.5445+0.0051J	0.0273+0.1669J	95.3 +583.4J
	37	27	29	-0.4555-0.0051J		
28	35	26	28	0.2022-0.0146J	0.0210+0.1355J	111.8 +720.8J
	36	27	28	0.3541+0.0172J		
	38	28	30	-0.4437+0.0026J		
29	25	19	29	0.0453-0.0109J	0.0226+0.1382J	115.5 +704.8J
	37	27	29	0.4432+0.0070J		
	39	29	30	-0.5115-0.0040J		
30	26	19	30	-0.0046+0.0002J	0.0047+0.0537J	162.1+1849.5J
	38	28	30	0.0256+0.0044J		
	39	29	30	0.0112+0.0020J		
	40	30	31	-0.0556-0.0075J		
	41	30	33	-0.8278+0.0127J		
	71	23	30	0.0845-0.0013J		
31	40	30	31	0.9116-0.0091J	0.0099+0.0818J	146.4+1204.8J
	43	31	32	-0.0884-0.0091J		
32	31	21	32	0.0571+0.0036J	0.0142+0.1017J	134.7 +964.9J
	32	21	32	0.0563+0.0062J		
	43	31	32	0.8866-0.0098J		
33	41	30	33	0.1334+0.0145J	0.0017+0.0382J	116.6+2611.3J
	44	22	33	0.8666-0.0145J		
	45	33	41	0.0000+0.0000J		

BUS	LINE	FROM BUS	TO BUS	CURRENT	IMPEDANCE	FAULT MVA
34	46	34	35	-0.1334-0.0160J	0.0032+0.0576J	95.2+1732.0J
	47	34	36	-0.1539-0.0129J		
	70	22	34	0.7127-0.0290J		
35	4	1	35	0.1742+0.0204J	0.0033+0.0351J	266.9+2827.7J
	46	34	35	0.0800-0.0016J		
	48	35	36	-0.1867-0.0042J		
	49	35	39	-0.2975-0.0016J		
	59	0	35	0.2616-0.0247J		
36	47	34	36	0.1852-0.0147J	0.0069+0.0516J	255.5+1905.4J
	48	35	36	0.5629+0.0451J		
	50	36	37	-0.2519+0.0304J		
37	50	36	37	0.7022+0.0358J	0.0071+0.0573J	211.8+1718.5J
	51	37	38	-0.1969+0.0234J		
	60	0	37	0.1009-0.0124J		
38	51	37	38	0.7829+0.0262J	0.0074+0.0617J	192.8+1598.3J
	61	0	38	0.2171-0.0262J		
39	49	35	39	0.6556+0.0459J	0.0061+0.0380J	413.9+2566.4J
	52	39	40	-0.1984+0.0223J		
	62	0	39	0.1461-0.0236J		
40	52	39	40	0.7677+0.0581J	0.0108+0.0430J	548.1+2189.7J
	63	0	40	0.2323-0.0581J		
41	45	33	41	1.0000+0.0000J	0.0807+0.4083J	46.6 +235.7J
42	34	42	44	-0.5036-0.1087J	0.0979+0.4468J	46.8 +213.6J
	76	0	42	0.4964-0.1087J		
43	7	2	43	0.5444+0.0089J	0.0484+0.1928J	122.5 +487.8J
	33	21	43	0.4556-0.0089J		
44	34	42	44	0.3190-0.0598J	0.1184+0.3691J	78.8 +245.7J
	42	23	44	0.6810+0.0598J		
45	23	18	45	0.7249+0.0534J	0.0345+0.1774J	105.5 +543.1J
	64	0	45	0.2751-0.0534J		

END

S. 10.

****
****

23/07/75

ALGOL LPRES

```
SEGMENT      AREA  1 SIZE 404
SEGMENT      AREA  1 SIZE 411
SEGMENT      AREA  1 SIZE 547
SEGMENT      AREA  1 SIZE 582
SEGMENT      AREA  1 SIZE 378
SEGMENT      AREA  1 SIZE 393
HEI 8G COMPLEX LOAD FLOW

FREE STORE     MS 2955   BS 5632
```

```
HEI 8g Complex Load Flow;

begin       comment calculates voltage and load or generation at each bus, power flow in lines, line losses
                    and line charging for a given network;
            integer I,J,J1,L,M,m,n,o,mo,mo10,m11,t1,t2,t3,t4,C1,C2,C3,C4,C5,C6,C7,C8,C9;
            real R,T,a;              integer array str[1:30];
            I:=1; instring(str,I);
            read C9,a,n,m,o,t1,t2,t3,t4;
            comment number of load flows, mva base, number of buses, lines, regulators, generators, loads, shunt
                    loads and special regulators;
            mo:= m+o;  mo10:= mo+10; m11:=m+11;
                  begin integer array spgen[1:t1,1:2], spreg[0:t4], spbus[0:t3],modn[1:20,1:2];
                        comment CBS:; integer array connec[1:mo10,1:2],parallel[1:m],
                        buslines,conlines[1:2*mo],colposn,rowposn,junct,order,linebus[1:n];
                        real array gen[1:t1,1:2], reg[0:t4,1:2], shunt[0:t3,1:2],value[1:20,1:3];
                        comment CBS:; real array load,puvolt,busmat[1:n,1:2],
                        nomvol[1:n],line[1:mo10,1:3],correction[1:2*n],diagonal[1:2*n,1:2];
                        switch s:= l1,l2;
                        comment insert procedures input,admat,organise,solve,current,corvol,page,
                                output,modify,adjust;
procedure input;
comment uses global variables nomvol,load,connec,line,junct,gen,reg,spreg,spbus,shunt,spgen,a,m,n,mo,mo10,
      t1,t2,t3,t4,C1;
begin comment segment[1];           integer e,f,h,i,j,k;                real x,y,z,g,v,w;
            h:=0;
            for i:=1 step 1 until mo do
                  begin read k,e,f,x,y,z,g;
                        if e=f or e<1 or f < 1 or e>n or f>n then
                              begin h:= h+1;    print ££l?data error?,k,e,f
                              end;
                        j:= if i>m then i+10 else i;
                        connec[j,1]:=if e>f then f else e; connec[j,2]:=if e>f then e else f;
                        w:= if abs(g-1.0)<0.1 then 1.0 else a/g/g;
                        v:= (x*x+y*y)*w;
                        line[j,1]:= if x=0.0 and y=0.0 then 0.0 else if i>m then x*z/v else x/v;
                        line[j,2]:= if x=0.0 and y=0.0 then 0.0 else if i>m then -y*z/v else -y/v;
                        line[j,3]:= if i>m or abs(g-1.0)<0.1 then z else z*₁₀-6/w
                  end reading line and regulator data, storing bus numbers in adjacent locations with lower
                      number first, per unit admittance for lines or tap times this for regs in first two locations,
                      per unit shunt susceptance or tap in location three;
            read j,x,g; junct[j]:= C1:= j;
            load[j,2]:= x/g;
            nomvol[j]:= g;
            for i:=2 step 1 until t1 do
                  begin read j,x,y,v,w,g; junct[j]:= j;
                        load[j,1]:= y/a;         load[j,2]:= x/g;
                        gen[i,1]:= v/a;          gen[i,2]:= w/a;
                        nomvol[j]:= g;
                        spgen[i,1]:= j;          spgen[i,2]:= 0
                  end reading generator data, per unit power and voltage in load , bus number in junct,
                      nominal voltage in nomvol, per unit reactive limits in gen , setting spgen;
            for i:=1 step 1 until t2 do
                  begin read j,x,y,g; junct[j]:= 500+j;     nomvol[j]:= g;
                        load[j,1]:= x/a;         load[j,2]:= y/a
                  end reading load data, per unit real and reactive power in load, nominal voltage in
                      nomvol, 500 plus bus number in junct;
            for i:=1 step 1 until t3 do
                  begin read j,x,y,g;
```

```
                spbus[i]:= j;
                w:= if abs(g-1.0)<0.1 then 1.0 else a/g/g;        v:= (x*x+y*y)*w;
                shunt[i,1]:= x/v;        shunt[i,2]:= -y/v
            end reading impedances at busses, per unit admittance in shunt, bus number in spbus;
        for i:=1 step 1 until t4 do
            begin read j,x,y;
                k:= j+m-490;             spreg[i]:= k;
                f:=connec[k,2]; g:= nomvol[f];
                reg[i,1]:= x/g;          reg[i,2]:= y/g
            end reading voltage limits on regulators, per unit limits in reg, setting spreg;
        if h>0 then stop
end input;

procedure admat;
comment uses global variables parallel,busmat,buslines,conlines,linebus,connec,line,spbus,shunt,
      m,n,mo,mo10,m11,t3,C2,C3;
begin comment segment[1];           integer i,j,k,d,e,f,g;  real x,y,z;
        switch ss:= l11,l12,l13,l14;
        for i:=1 step 1 until m do parallel[i]:= 0;
        i:=0;      d:= m+1;     C3:= mo;     C2:= m;
l11:    i:= i+1;
        if i>m then goto l12;
        if parallel[i]=0 then
            begin f:=connec[i,1]; g:=connec[i,2];
                x:= y:=z:= 0.0;          k:= 0;
                for j:= i+1 step 1 until m do
                begin if f=connec[j,1] and g=connec[j,2] then
                    begin k:= k+1;    parallel[j]:= d;
                        x:= x +line[j,1];       y:= y +line[j,2]; z:=z+line[j,3]
                    end
                end;
                if k>0 then
                    begin connec[d,1]:=f; connec[d,2]:=g;
                        line[d,1]:=line[i,1]+x; line[d,2]:=line[i,2]+y; line[d,3]:=line[i,3]+z;
                        parallel[i]:=d;         d:=d+1;
                        C2:= C2+1;  3:= C3-k
                    end combining parallel lines and storing in locations after m, C2 gives m plus
                       the sets of parallel lines, C3 number of connections between busses
            end;
        goto l11;
l12:    for i:=1 step 1 until n do busmat[i,1]:= busmat[i,2]:= 0.0;
        i:=k:= d:= 0;
l13:    i:= i+1;
        e:=0;
        for j:= 1 step 1 until C2,m11 step 1 until mo10 do
            begin if j ≤m then begin if parallel[j]≠0 then goto l14 end; f:=connec[j,1]; g:=connec[j,2];
                if f=i or g=i then
                    begin d:= d+1; buslines[d]:=j; conlines[d]:= if f=i then g else f;            e:=e+1;
                        x:=line[j,1]; y:=line[j,2]; z:=line[j,3];
                        busmat[i,1]:= busmat[i,1]+(if j≤C2 then x else if f=i then x*z else x/z);
                        busmat[i,2]:= busmat[i,2] + (if  j≤C2 then y else if  f=i then y*z else y/z);
                        if j≤C2 then busmat[i,2]:=busmat[i,2]+z
                    end forming diagonal terms of admittance matrix in busmat, listing lines
                       in order of bus connections and number of connections to each bus;
l14:            end; linebus[i]:=e;
        if e=0 then
            begin print ££l?no connections to bus?,i; k:=k+1
            end;
```

```
            if i<n then goto l13;
            for i:=1 step 1 until t3 do
                  begin j:= spbus[i];
                        busmat[j,1]:= busmat[j,1] + shunt[i,1];
                        busmat[j,2]:= busmat[j,2] + shunt[i,2]
      end adding admittances at busses to diagonal terms of matrix;
            if k>0 then stop
end admat;

procedure organise;

comment uses global variables linebus,conlines, C1,C4,C5,C6,n,mo setting up the global arrays
            colposn,rowposn,order;
begin comment segment[1];         integer d,e,f,g,h,i,j,k,q,ltcol,ltrow,rkci,rich,cpi,ropi,cpch;
            integer array colterm,rowterm,finterm,busind[1:n],row,column[1:3*mo10];
            switch ss:= l11,l12,l13,l14,l15;
            k:=1; f:=0;
            for i:=1 step 1 until n do
                  begin d:= f+1;      f:= f +linebus[i];
                        if i=C1 then goto l12;
                        j:=0;         colposn[i]:= rowposn[i]:= k;
                        for q:= d step 1 until f do
                              begin g:=conlines[q];
                                    if g=C1 then goto l11;
                                    column[k]:= row[k]:=g;
                                    k:= k+1;      j:= j+1;
                        l11:  end;
                        if j=0 then
                              begin print £slack bus is the only bus connected to bus?,i; stop
                              end;
                        colterm[i]:= rowterm[i]:=finterm[i]:=j;
                        busind[i]:=0;
            l12:  end column lists row number of terms in each column, row lists column number of
                       terms in each row, colterm and colposn give number of terms and location in column,
                       rowterm and rowposn in row, finterm stores maximum value of rowterm;
            busind[C1]:=1;
            ltcol:=ltrow:=k-1;
            j:=d:=1;
l13:        i:=0;
l14:        i:=i+1;
            if i>n then
                  begin d:=d+1;      goto l13
                  end;
            if busind[i]=1 then goto l14;
            if colterm[i]≠d then goto l14;
            order[j]:=i; busind[i]:=1;
            cpi:=colposn[i];         f:= cpi+d-1;
            for k:=cpi step 1 until f do
                  begin rkci:=column[k];
                        comment row number of term in column i which is to be eliminated;
                        ropi:= rowposn[i];
                        g:= ropi + rowterm[i]-1;
                        for h:= ropi step 1 until g do
                              begin rich:= row[h];
                                    comment column number of term in row i;
                                    if rkci= rich then goto l15;
                                    cpch:= colposn[rich];
                                    e:= cpch + colterm[rich]-1;
```

```
                              for q:= cpch step 1 until e do if rkci= column[q] then goto 115;
                              comment a new term to be added in column rich and row rkci;
                              colterm[rich]:= colterm[rich]+1;
                              for q:= rich+1 step 1 until n do if q≠C1 then
                              colposn[q]:= colposn[q]+1;
                              e:= e+2;    ltcol:= ltcol+1;
                              for q:= ltcol step -1 until e do column[q]:= column[q-1];
                              column[e-1]:= rkci;
                              if rich <i then
                                   begin f:= f+1;    k:= k+1
                                   end;
                              rowterm[rkci]:= e:= rowterm[rkci]+1;
                              if e>finterm[rkci] then finterm[rkci]:=e;
                              for q:= rkci+1 step 1 until n do if q ≠ C1 then
                              rowposn[q]:= rowposn[q]+1;
                              e:=e+ rowposn[rkci];    ltrow:= ltrow+1;
                              for q:= ltrow step -1 until e do row[q]:= row[q-1];
                              row[e-1]:= rich;
                              if rkci<i then
                                   begin g:= g+1;    h:= h+1
                                   end;
                    115:  end adding new term and modifying column,colposn,colterm,
                              row,rowposn,rowterm,finterm;
                    rowterm[rkci]:= rowterm[rkci]-1;
                    for q:= rkci+1 step 1 until n do if q≠C1 then rowposn[q]:= rowposn[q]-1;
                    ltrow:= ltrow-1;        g:= 0;
                    for q:= rowposn[rkci] step 1 until ltrow do
                         begin if row[q]=i then g:=1;
                              if g=1 then row[q]:= row[q+1]
                         end elimination of column i from row rkci
               end;
          j:= j+1;
          if j≠n then
               begin ltcol:= ltcol-d;
                    for q:=colposn[i] step 1 until ltcol do column[q]:= column[q+d];
                    for q:= i+1 step 1 until n do if busind[q]=0 then
                    colposn[q]:= colposn[q]-d;
                    goto 114
               end column i terms removed from column;
          C6:= j:=k:=1;
          for i:=1 step 1 until n do if i≠C1 then
               begin colposn[i]:= j;        rowposn[i]:= k;
                    j:=j+colterm[i];
                    g:= finterm[i];         k:= k+g;
                    if g>C6 then C6:= g
               end setting maximum values in colposn and rowposn;
          C4:=j-1;    C5:=k-1
end organise;

procedure solve (CTM,CMN,RTM,RW,ave,blw);
comment uses global variables colposn, rowposn,C6,order,correction,diagonal,n Eliminates
          pairs of columns in the prescribed order using the extra space in ave,blw, CTM,CMN,RTM and RW
          to avoid shifting terms;
integer array CTM,CMN,RTM,RW; real array ave,blw;
     begin integer d,e,f,g,h,i,j,k,p,q,n1,rt,rp,rw,ct,cp,u,uw;
          real x,y,z,z1,z2,cor,diag;    integer array  loc,coll[1:C6];
          real array  usterms, wtterms[1:C6]; switch ss:=111,112,113,114,115;
          d:=1; n1:=n-1;
```

```
111:  i:=order[d];           u:=0;
      rt:=RTM[i]; if rt=0 then goto 113;      comment number of terms in row i;
      rp:= rowposn[i];        f:=0;
112:  j:= RW[rp+f];
      cp:=colposn [j];  g:=cp+CTM[j]-1;
      for k:=cp step 1 until g do if CMN[k]=i then
           begin  f:=f+1;    loc[f]:=k; coll[f]:=j;
                if f = rt then goto 113  else goto 112
           end storing the locations and columns  of row i off diagonal terms;
113:  f:= if u=0 then i else i+n;
      x:= if u=0 then diagonal[f,1] else diagonal[f,2];
      if u=0 then
           begin diag:=diagonal[f,2]/x; diagonal[f,2]:=diag
           end;
      cor:=correction[f]/x; correction[f]:=cor;
      comment division of correction and semi-diagonal terms by diagonal pivot;
      for j:=1 step 1 until rt do
           begin k:= loc[j];
                for g:=1,2 do
                    begin y:=(if u=0 then ave[k,g] else blw[k,g])/x;
                          if u=0 then ave[k,g]:=y else blw[k,g]:=y;
                          if g=1 then usterms[j]:=y  else wtterms[j]:=y
                    end
           end division of  off-diagonal row i terms by diagonal pivot and storing;
      f:= if u=0 then i+n else i;
      y:= if u=0 then diagonal[f,1] else diag;
      for j:=1 step 1 until rt do
           begin  k:=loc[j];
                 for g:=1,2 do
                      begin z:= y*(if g=1 then usterms[j] else wtterms[j]);
                           if u=0 then blw[k,g]:=blw[k,g]-z else ave[k,g]:=ave[k,g]-z
                      end
           end modifying off-diagonal terms in corresponding row;
      if u=0 then diagonal[f,2]:=diagonal[f,2]-y*diag;
      correction[f]:=correction[f]-y*cor;
      comment corresponding semi-diagonal term is eliminated;
      uw:=e:=0; f:=if u=0 then 1 else 2;
      cp:=colposn[i];     ct:=CTM[i];
114:  g:=cp+e;     rw:=CMN[g];
      comment term in row rw and column i;
      y:=if uw=0 then ave[g,f] else blw[g,f];
      if u=0 then
           begin if uw=0 then ave[g,2]:=ave[g,2]-y*diag else blw[g,2]:=blw[g,2]-y*diag
           end  modifying  term in column i+n;
      g:= if uw=0 then rw else rw+n; correction[g]:=correction[g]-y*cor;
      comment modifying correction terms;
      for j:=1 step 1 until rt do
           begin h:=coll[j];
                z1:=y* usterms[j];    z2:=y*wtterms[j];
                if h=rw then
                    begin diagonal[g,1]:=diagonal[g,1]-z1; diagonal[g,2]:=diagonal[g,2]-z2;  goto 115
                    end otherwise search for a term in row rw and column h;
                p:= colposn[h]; rp:=CTM[h]; k:=p+rp-1;
                for q:=p step 1 until k do if CMN[q]= rw then
                    begin if uw=0 then
                                begin ave[q,1]:=ave[q,1]-z1; ave[q,2]:=ave[q,2]-z2
                                end else
                                begin blw[q,1]:=blw[q,1]-z1; blw[q,2]:=blw[q,2]-z2
```

```
                    end;    goto 115
                  end otherwise add a new term in row rw and column h;
                k:=k+1;
                if uw=0 then
                    begin ave[k,1]:=-z1; ave[k,2]:=-z2
                    end else
                    begin blw[k,1]:=-z1; blw[k,2]:=-z2
                    end;
                        if uw=1 then
                            begin CTM[h]:=rp+1; CMN[k]:=rw; k:=RTM[rw]; RTM[rw]:=k+1;
                                RW[k+ rowposn[rw]]:=h
                            end;
        115:  end elimination of term in row rw and column i;
        if u=1 and uw =1 and d≠n1 then
            begin rp:= rowposn[rw];     k:=RTM[rw];
                RTM[rw]:=k-1;    k:=k+rp -2;     g:=0;
                for q:=rp step 1 until k do
                    begin if RW[q]=i then g:=1;
                        if g=1 then RW[q]:= RW[q+1]
                    end elimination of i from row rw
            end;
        uw:=uw+1;   if uw =1 then goto 114;
        e:= e+1;  if e<ct then
                    begin uw:=0;      goto 114
                    end;
        u:=u+1;     if u=1 then goto 113;
        d:=d+1;     if d<n then goto 111
end solve;
procedure current(X,Y,U,V);
comment used for calculating current at a bus and requires global variables R,T;
value  X,Y,U,V;        real X,Y,U,V;
begin       R:= R+X*U-Y*V;
            T:= T+X*V+Y*U
end current;

procedure corvol;
comment sets up correction equations and corrects bus voltages, requires
        procedures current and solve,uses global variables colposn,rowposn, n,R,T,C4,C5,junct,load,busmat,correction,
          diagonal,puvolt,line,linebus,buslines,conlines;
begin       integer d,e,f,g,i,j,k,q,bus;
            real u,v,x,y,r,t,w,z; comment CBS:; real array above,below[1:C4,1:2];
            integer array rowterm, colterm[1:n], column[1:C4], row[1:C5];       switch ss:= 111, 112;
            f:=0; for i:=1 step 1 until n do colterm[i]:=0;
            for i:=1 step 1 until n do
                begin d:=f+1;     f:= f+linebus[i];
                    if i=C1 then goto 112;
                    bus:=junct[i];          comment if bus > 500 then load otherwise generator;
                    u:=puvolt[i,1]; v:=puvolt[i,2];
                    x:=busmat[i,1]; y:=busmat[i,2];
                    R:= T:= 0.0;   current(x,y,u,v);
                    w:= -u*y+v*x;  z:= u*x+v*y;
                    for q:= d step 1 until f do
                        begin j:=buslines[q]; g:=conlines[q];
                            r:=-line[j,1]; t:=-line[j,2];
                            current(r,t,puvolt[g,1],puvolt[g,2]);
                            if g=C1 then goto 111;
                            x:= -u*t+v*r;  y:= u*r+v*t;
                            e:= colterm[g];
```

```
                j:= colposn[g]+e;  k:= rowposn[g]+e;
                        colterm[g]:= e+1;
                        column[j]:= row[k]:=i;
                        below[j,1]:=y; below[j,2]:=x;
                        above[j,1]:=if bus>500 then x else 0.0;
                        above[j,2]:=if bus>500 then -y else 0.0;
                111:  end;  e:= n+i;
                correction[e]:=load[i,1]-u*R-v*T;
                diagonal[e,1]:=R+z; diagonal[e,2]:=T+w; x:=load[i,2];
                if bus>500 then
                    begin correction[i]:=x+u*T-v*R;
                        diagonal[i,1]:=-T+w; diagonal[i,2]:=R-z
                    end else
                    begin correction[i]:=x*x-u*u-v*v;
                        diagonal[i,1]:=2.0*u; diagonal[i,2]:=2.0*v
                    end calculation of terms in row i;
        112:  end setting up correction equations;
        for i:=1 step 1 until n do rowterm[i]:= colterm[i];
        solve(colterm,column,rowterm,row,above,below);
        for i:=1 step 1 until n do if i≠C1 then
            begin puvolt[i,1]:=puvolt[i,1]+correction[i];
                puvolt[i,2]:=puvolt[i,2]+correction[i+n]
            end correcting voltages at each bus
end corvol;

        procedure page (u,v);
        comment prints page numbers and headings, uses global variables L,M;
        value u,v;  integer u,v;
            begin  switch ss:=111,112,113,114,115;
                goto ss[u];
111:            top of form;
112:            print ££s60?PAGE?,M,££12??;
                M:=M+1;  goto ss[v];
113: print £  BUS        MW         MVAR          VOLTAGE              KV         PER UNIT    DEGREES   MVAR LINE CHARGING?;
                 if t 3>0 then print £  SHUNT MW MVAR?; goto 115;
114: print £LINE NO   BUS      MW        MVAR         MVA         BUS       MW          MVAR         MVA       MW LOSS    MVAR LOSS      TAP?;
115:            L:=0
            end page;
        procedure output;
        comment uses global variables a,m,n,mo10,m11,t3,R,T,L,C2,nomvol,busmat,line,linebus,buslines,
                conlines,puvolt,connec,spbus,shunt,requires procedures page and current;
            begin comment segment[1]; integer d,e,f,g,i,j,k,r,q;  real u,v,w,x,y,z,z1,z2,z3,z4,sum1,sum2,sum3,linch;
                real array volt[1:n]; print ££12??; r:=5;
                page(2,3);  aligned(4,2);  sum1:=sum2:=sum3:=linch:=0.0 ;  f:=k:=0;
                for i:=1 step 1 until n do
                    begin r:=r+1; if r=6 then
                            begin r:=1; if L = 9 then page(1,3); L:=L+1; print ££1??
                            end;
                        u:=puvolt[i,1]; v:=puvolt[i,2];
                        volt[i]:=w:=u*u+v*v;
                        R:=T:=z3:=0.0;  current(busmat[i,1],busmat[i,2],u,v);
                        d:=f+1;  f:=f+linebus[i];
                        for q:=d step 1 until f do
                            begin e:=buslines[q]; g:=conlines[q];
                                current(-line[e,1],-line[e,2],puvolt[g,1],puvolt[g,2]);
                                if e≤C2 then z3:=z3+line[e,3]
                            end calculation of current and susceptance at bus i;
                        z2:=z3*w*a;  linch:=linch+z2;
```

```
                    x:=a*(u*R+v*T);  y:=a*(v*R-u*T);
                    if x>1.0 then
                          begin sum1:=sum1+x;  sum2:=sum2+y;
                                sum3:=sum3+sqrt(x*x+y*y)
                          end adding generation;
                    z:= if u>0.0 then arctan(v/u) else if abs(v)>0.0 then
                    sign(v)*1.57079633-arctan(u/v) else 3.14159265;
                    w:=sqrt(w);  z1:=nomvol[i];
                    print ££l??,i,£    ?,x,£    ?,y,£    ?,u*z1,special(2),v*z1,£j?;
                    print prefix(£    ?),w*z1,aligned(2,4),w,z*57.2957795,aligned(3,2),z2;
                    if k < t3 then
                          begin for j:=1 step 1 until t3 do if spbus[j]=i then
                                      begin k:=k+1; z:=volt[i]*a;
                                            print aligned(3,2),prefix(£    ?),-shunt[j,1]*z,shunt[j,2]*z
                                      end
                          end
                end printing generation,loads,bus voltages, angles, line charging and shunt loads;
          print ££l2?TOTAL GENERATION=?,freepoint(4),sum1,£ MW  ?,sum2,£ MVAr  ?,sum3,£ MVA?;
          r:=5;  sum1:=sum2:=0.0;  aligned(3,2);
          for i:=1 step 1 until m,m11 step 1 until mo10 do
                    begin w:=line[i,2]; if w≠0.0 then
                begin r:=r+1; if r=6 then
                          begin r:=1; if L=9 or i=1 then page(1,4); L:=L+1; print ££l??
                          end;
                    f:=connec[i,1]; g:=connec[i,2];  if i>m then  sum3:=line[i,3];
                    u:=puvolt[f,1]; v:=puvolt[f,2]; z:=line[i,1];
                    x:=puvolt[g,1]; y:=puvolt[g,2];
                    z1:=u*x+v*y;  z2:=u*y-v*x;
                    u:=z2*z;  v:=z2*w;  x:=z1-( if i>m then volt[f]*sum3 else volt[f]);
                    y:=z1-( if i>m then volt[g]/sum3 else volt[g]);
                    z1:=(x*z-v)*a;  z3:=(y*z+v)*a;
                    z2:=(-x*w-u)*a;  z4:=(-y*w+u)*a;
                    x:=sqrt(z1*z1+z2*z2); if z1<0.0 then x:=-x;
                    y:=sqrt(z3*z3+z4*z4); if z3<0.0 then y:=-y;
                    z:=-z1-z3;  w:=-z2-z4;  sum1:=sum1+z;  sum2:=sum2+w;
                    if i>m then print ££l??,490+i-m else print ££l??,i;
                    print prefix(£    ?),f,z1,z2,x,g,z3,z4,y,aligned(2,3),z,w;
                    if i>m then print £    ?,aligned(1,3),sum3
                end printing line flows and losses
                    end;
          print ££l2?TOTAL LINE LOSS=?,freepoint(4),sum1,£ MW  ?,sum2,£ MVAr  ?,£TOTAL LINE CHARGING=?,linch,£ MVAr?
      end output;
procedure modify;
comment uses global variables a,C7,C8,value,modn,reads and stores network modifications;
begin comment segment[1]; integer e,f,q,j,i,k; real g,r,v,w,x,y,z;
procedure change(u);
comment uses global variables modn,value,connec,line,parallel,busmat,load,C8.  If u=1 restores original
        network, if u=2 sets up new network and stores old values;
value u;  integer u;
begin       for i:=1 step 1 until C8 do
                begin j:= modn[i,2];
                      if modn[i,1]=2 then
                            begin if u=2 then
                                        begin x:=load[j,1]; y:=load[j,2]
                                        end;
                                  load[j,1]:=value[i,1]; load[j,2]:=value[i,2];
                                  if u=2 then
                                        begin value[i,1]:=x;  value[i,2]:= y
```

```
                                    end
                                    end changing loads else
                                    begin e:=connec[j,1]; f:=connec[j,2];
                                          x:= value[i,1];   r:=line[j,1];
                                          y:= value[i,2];   v:=line[j,2];
                                          z:= value[i,3];   w:=line[j,3];
                                          line[j,1]:=x; line[j,2]:=y; line[j,3]:=z; k:=parallel[j];
                                          if k>0 then
                                                begin line[k,1]:=line[k,1]+x-r; line[k,2]:=line[k,2]+y-v; line[k,3]:=line[k,3]+z-w
                                                end;
                                          if u=2 then
                                                begin value[i,1]:=r;  value[i,2]:=v;  value[i,3]:=w
                                                end;
                                          for q:= e,f do
                                                begin busmat[q,1]:=busmat[q,1]+x-r; busmat[q,2]:=busmat[q,2]+y-v+z-w
                                                end
                                    end changing lines
                             end
end change;
              if C7>1 then change(1);
              read C8;
              for i:=1 step 1 until C8 do
                    begin read k,e,modn[i,2],x,y;
                          if e=1 then read z,g;  modn[i,1]:= e;
                          if e=2 then
                                begin value[i,1]:=x/a;  value[i,2]:=y/a
                                end storing new loads else
                                begin w:= if abs(g-1.0)<0.1  then 1.0 else a/g/g;
                                      v:= (x*x+y*y)*w;
                                      value[i,1]:= if x=0.0 and y=0.0 then 0.0 else x/v;
                                      value[i,2]:= if x=0.0 and y=0.0 then 0.0 else -y/v;
                                      value[i,3]:= if abs(g-1.0)<0.1 then z else z*₁₀-6/w
                                end storing new line data
                    end reading and storing modifications;
              change(2)
end modify;
procedure adjust;
comment uses global variables t1,t4,R,T,J,J1,spgen,gen,spreg,reg,busmat,load,junct,linebus,buslines,conlines,
              puvolt,line,connec and requires procedure current declared first;
begin comment segment[1];          integer d,e,f,g,h,i,j,k,q;          real r,u,v,w,x,y,z,x1,y1;
              h:=0;
              for i:=2 step 1 until t1 do if spgen[i,2]=0 then
                    begin j:=spgen[i,1];           d:= j-1;
                          u:=puvolt[j,1]; v:=puvolt[j,2];       R:= T:= 0.0;
                          current(busmat[j,1],busmat[j,2],u,v);            f:= 0;
                          for q:=1 step 1 until d do f:= f+linebus[q];
                          d:= f+1;    f:= f+linebus[j];
                          for q:= d step 1 until f do
                                begin e:=buslines[q]; g:=conlines[q];
                                      current(-line[e,1],-line[e,2],puvolt[g,1],puvolt[g,2])
                                end calculation of current at bus j;
                          w:= v*R-u*T;  u:= gen[i,1];  v:= gen[i,2];
                          if u>w or v<w then
                                begin if u>w then
                                            begin spgen[i,2]:=1;  gen[i,1]:=load[j,2];  load[j,2]:=u
                                            end else
                                            begin spgen[i,2]:= 2;  gen[i,2]:=load[j,2]; load[j,2]:=v
                                            end;
```

```
                    junct[j]:=500+j;  h:= h+1
                         end changing generator to load if either MVA limit exceeded
               end;
          for i:=1 step 1 until t4 do
               begin j:= spreg[i];  k:=connec[j,2];
                    u:=puvolt[k,1]; v:=puvolt[k,2]; z:=line[j,3];
                    w:= sqrt(u*u+v*v);  u:= reg[i,1];  v:= reg[i,2];
                    if ( u>w and abs(z-1.15)>0.0055) or ( v<w and abs(z-0.9)>0.0055) then
                         begin x:=line[j,1]; y:=line[j,2];
                              if J1=2 then
                                   begin x1:= entier(100.0*(if u>w then w-u else v-w))/100.0;
                                        v:= z + (if u>w then -x1 else x1);
                                        if v>1.1555 then v:= v-entier(100.0*(v-1.145))/100.0;
                                        if v<0.8945 then v:= v + entier(100.0*(0.905 - v))/100.0
                                   end else v:= z+ (if u>w then 0.01 else -0.01);
                              r:= v/z;  x1:= x*r;  y1:= y*r;
                              f:=connec[j,1]; line[j,1]:=x1; line[j,2]:=y1; line[j,3]:=v;
                              busmat[f,1]:=busmat[f,1]-x*z+x1*v; busmat[f,2]:=busmat[f,2]-y*z+y1*v;
                              h:= h+1
                         end changing tap, line and bmat if either voltage limit exceeded
               end;
          if h≠0 then J:=2
end adjust;
                    sameline; digits(3);
                    input;     admat;
                    for I:=1 step 1 until n do
                         begin puvolt[I,1]:= if junct[I]>500 then 1.0 else load[I,2]; puvolt[I,2]:=0.0
                         end;
                    organise; noflo;
                    J:=J1:=C7:=0;
          l1:       I:=M:=1; outstring(str,I);
          l2:       J:=J+1; J1:=J1+1;  corvol;
                    if J>1 and ([illegible]1>1 or t4>0) then adjust;
                    if J<5 then goto l2; C7:= C7+1;  output;
                    if C7<C9 then
                         begin top of form;
                              I:=1; instring(str,I);
                              modify; J:=2;          goto l1
                         end
               end
end load flow;
```

'LOAD FLOW FOR SAMPLE NETWORK'

Number of studies	1
mva base	100.0
number of busbars	9
number of lines	12
number of regulators	1
number of generators	2
number of loads	7
number of shunts	0
number of tap changers	0

line	connection busbar	busbar	resistance	reactance	total shunt susceptance	nominal voltage
1	5	2	1.25	3.4	42.0	110.0
2	2	1	1.34	3.85	48.0	110.0
3	1	6	10.36	18.25	230.0	110.0
4	1	7	29.34	40.68	134.0	110.0
5	7	6	7.2	44.3	15.0	110.0
6	7	8	0.18	0.69	5.0	110.0
7	7	3	11.25	19.72	252.0	110.0
8	6	3	18.21	45.09	102.0	110.0
9	8	9	0.0	0.04	0.0	1.0
10	9	4	4.83	19.37	233.0	220.0
11	1	7	29.34	40.68	134.0	110.0
12	7	6	7.2	44.3	15.0	110.0
regulator					tap	
501	3	5	0.0	2.45	1.08	110.0

generator busbar	voltage magnitude	MW	MVAr limits minimum	maximum	nominal voltage
1	110.0				110.0
3	110.0	212.0	80.0	110.0	110.0

load busbar	MW	MVAr	nominal voltage
2	32.0	0.0	110.0
4	122.0	48.0	220.0
5	0.0	0.0	110.0
6	-31.0	-10.0	110.0
7	-229.0	-104.0	110.0
8	0.0	0.0	110.0
9	0.0	0.0	220.0 E

LOAD FLOW FOR SAMPLE NETWORK

PAGE 1

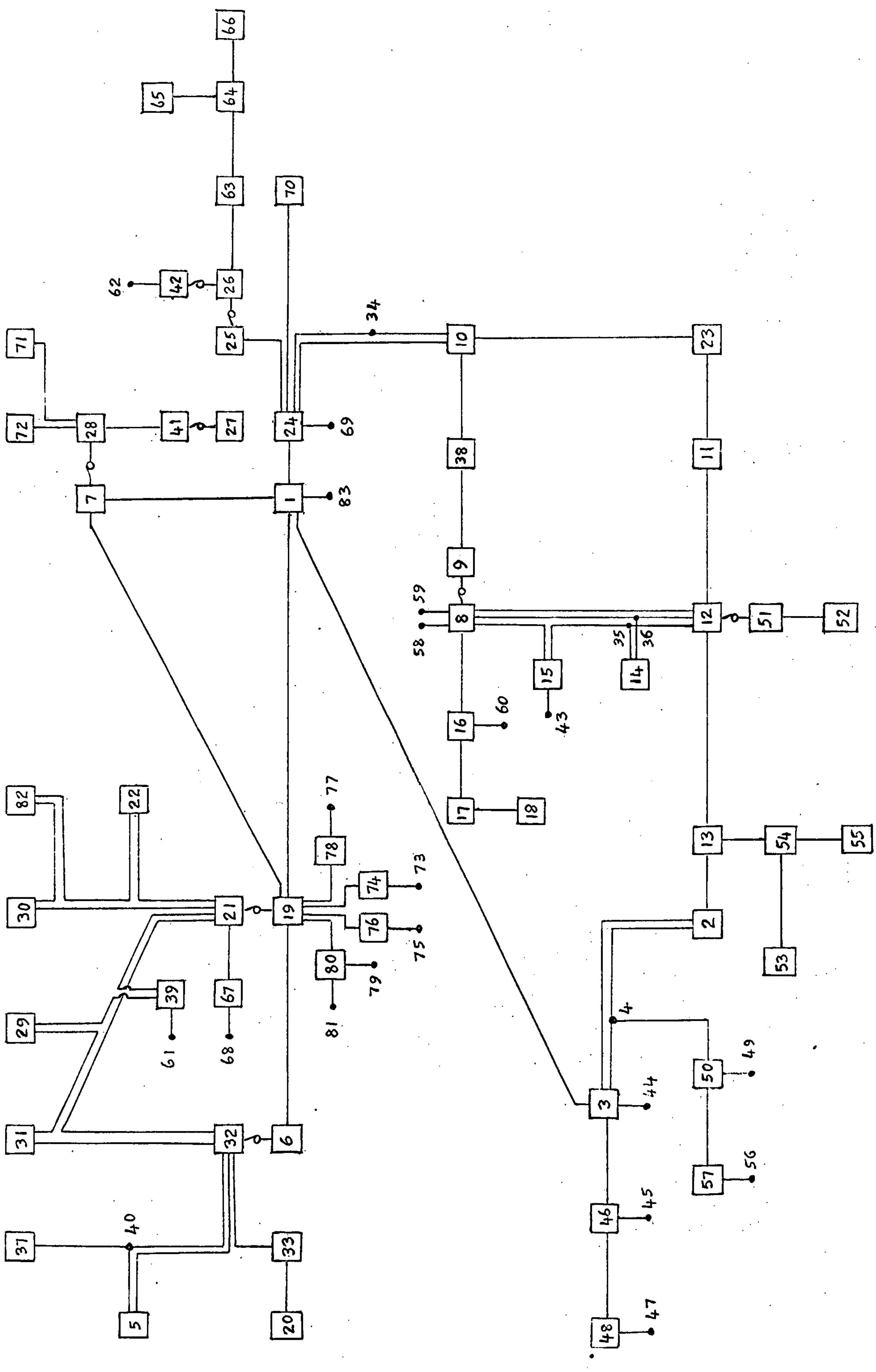
BUS	MW	MVAR	VOLTAGE		KV	PER UNIT	DEGREES	MVAR LINE CHARGING
1	-90.39	-19.04	110.00	+0.00J	110.00	1.0000	0.0000	6.61
2	32.00	0.00	113.89	+4.30J	113.97	1.0361	2.1622	1.17
3	212.00	90.87	109.64	+8.91J	110.00	1.0000	4.6469	4.28
4	122.00	48.00	214.78	+17.30J	215.47	0.9794	4.6059	10.82
5	-0.00	-0.00	117.06	+7.10J	117.28	1.0662	3.4731	0.58
6	-31.00	-10.00	104.51	-1.08J	104.51	0.9501	-0.5909	3.95
7	-229.00	-104.00	100.52	-2.88J	100.57	0.9142	-1.6427	5.61
8	-0.00	0.00	101.12	-2.17J	101.14	0.9194	-1.2298	0.05
9	0.00	-0.00	207.56	+7.04J	207.67	0.9440	1.9429	10.05

TOTAL GENERATION= 366.0 MW 138.9 MVAR 393.8 MVA

PAGE 2

LINE NO	BUS	MW	MVAR	MVA	BUS	MW	MVAR	MVA	MW LOSS	MVAR LOSS	TAP
1	2	114.79	67.61	133.22	5	-116.50	-72.26	-137.09	1.708	4.646	
2	1	144.08	60.99	156.46	2	-146.79	-68.78	-162.11	2.711	7.789	
3	1	-19.13	-22.25	-29.34	6	18.39	20.95	27.88	0.737	1.298	
4	1	-17.28	-13.16	-21.72	7	16.14	11.57	19.86	1.144	1.586	
5	6	-5.72	-8.42	-10.18	7	5.66	8.00	9.80	0.068	0.421	
6	7	119.79	51.95	130.57	8	-120.09	-53.11	-131.31	0.303	1.163	
7	3	-70.47	-15.80	-72.22	7	65.62	7.30	66.03	4.849	8.500	
8	3	-25.03	-4.34	-25.40	6	24.06	1.94	24.14	0.971	2.405	
9	8	120.09	53.06	131.29	9	-120.09	-61.21	-134.79	0.000	8.156	
10	4	-122.00	-58.82	-135.44	9	120.09	51.16	130.54	1.908	7.653	
11	1	-17.28	-13.16	-21.72	7	16.14	11.57	19.86	1.144	1.586	
12	6	-5.72	-8.42	-10.18	7	5.66	8.00	9.80	0.068	0.421	
501	3	-116.50	-75.01	-138.56	5	116.50	71.68	136.79	0.000	3.333	1.080

TOTAL LINE LOSS= 15.61 MW 48.96 MVAR TOTAL LINE CHARGING= 43.12 MVAR
END


66
65
64
63
70
62
42
26
34
25
10
23
71
72
28
41
27
24
69
38
11
7
1
83
9
59
58
8
12
51
52
35
36
15
14
60
43
16
17
18
82
22
77
78
13
54
55
73
74
30
21
19
76
2
75
80
79
53
39
67
81
4
29
61
68
49
50
44
3
31
32
6
56
57
45
46
40
37
33
5
20
47
48

£SYSTEM LOAD FLOW 1973 LOADINGS?

number of studies	1
mva base	100.0
number of busbars	83
number of lines	86
number of regulators	8
number of generators	1
number of loads	82
number of shunts	0
number of tap changes	0

line	connection busbar	busbar	resistance	reactance	total shunt susceptance	nominal voltage
1	10	23	6.41	11.12	523.0	110.0
2	12	36	3.0	8.18	27.0	110.0
3	8	36	14.0	30.7	99.0	110.0
4	8	12	11.16	20.26	245.0	110.0
5	9	38	1.18	3.36	42.0	110.0
6	8	15	11.78	21.56	70.0	110.0
7	15	35	2.61	11.3	37.0	110.0
8	12	13	0.13	0.49	17.0	110.0
9	11	12	0.16	0.66	10.0	110.0
10	2	13	0.0009	0.028	0.0	1.0
11	2	3	8.24	33.84	116.0	220.0
12	1	3	7.93	34.44	108.0	220.0
13	3	4	1.96	8.53	27.0	220.0
14	4	2	6.28	25.85	89.0	220.0
15	1	19	7.71	33.49	105.0	220.0
16	1	7	4.14	16.39	242.0	220.0
17	10	24	13.25	22.0	65.0	110.0
18	1	24	0.0029	0.089	0.0	1.0
19	24	25	3.41	9.11	125.0	110.0
20	41	28	0.175	1.1	2.0	110.0
21	21	32	7.3	18.0	61.0	110.0
22	6	19	4.57	19.8	62.0	220.0
23	5	32	15.13	22.2	68.0	110.0
24	20	33	3.8	9.52	36.0	110.0
25	11	23	0.197	0.528	7.0	110.0
26	33	32	9.1	24.0	82.0	110.0
27	10	34	5.37	8.25	23.0	110.0
28	14	36	0.5	13.97	3.0	110.0
29	14	35	0.5	13.97	3.0	110.0
30	8	16	5.25	8.08	23.0	110.0
31	16	17	19.22	29.03	85.0	110.0
32	17	18	10.03	15.4	43.0	110.0
33	21	22	3.43	7.24	22.0	110.0
34	24	34	6.89	12.49	37.0	110.0
35	29	39	2.75	6.7	21.0	110.0
36	21	39	1.24	3.05	11.0	110.0
37	31	32	0.45	1.11	4.0	110.0
38	29	31	4.03	9.9	32.0	110.0
39	22	82	3.8	11.3	78.0	110.0
40	7	19	4.21	26.4	95.0	220.0
41	12	35	3.0	8.18	27.0	110.0
42	21	30	1.25	6.0	22.0	110.0
43	5	40	0.47	0.69	2.0	110.0
44	32	40	14.7	21.5	66.0	110.0
45	40	37	6.04	8.9	27.0	110.0
46	10	38	1.35	3.88	48.0	110.0

47	3	44	0.0069	0.135	0.0	1.0
48	15	43	0.0081	0.197	0.0	1.0
49	46	48	0.61	2.66	9.0	220.0
50	3	46	0.58	2.52	8.0	220.0
51	50	57	0.21	1.17	4.0	220.0
52	4	50	0.36	1.99	6.0	220.0
53	54	55	5.91	8.3	28.0	110.0
54	53	54	0.61	0.89	3.0	110.0
55	13	54	4.93	12.58	46.0	110.0
56	51	52	13.3	10.58	28.0	88.0
57	56	57	0.0144	0.286	0.0	1.0
58	49	50	0.0246	0.568	0.0	1.0
59	45	46	0.0152	0.26	0.0	1.0
60	47	48	0.0098	0.185	0.0	1.0
61	39	61	0.00897	0.211	0.0	1.0
62	63	64	30.54	28.73	156.0	88.0
63	64	65	3.58	2.82	15.0	88.0
64	64	66	11.02	10.06	54.0	88.0
65	42	62	0.006	0.096	0.0	1.0
66	26	63	0.0	0.145	0.0	1.0
67	8	58	0.0054	0.0666	0.0	1.0
68	8	59	0.0061	0.092	0.0	1.0
69	16	60	0.052	0.647	0.0	1.0
70	21	67	0.26	1.71	7.0	110.0
71	67	68	0.00448	0.105	0.0	1.0
72	24	69	0.00212	0.0815	0.0	1.0
73	24	70	18.56	33.26	158.0	110.0
74	28	71	0.722	1.25	4.0	110.0
75	28	72	0.05	0.18	2.0	110.0
76	19	74	1.79	9.26	33.0	220.0
77	19	76	1.75	9.07	32.0	220.0
78	19	78	2.68	13.98	44.0	220.0
79	80	81	0.008	0.21	0.0	1.0
80	80	79	0.036	0.273	0.0	1.0
81	73	74	0.0156	0.324	0.0	1.0
82	75	76	0.006	0.11	0.0	1.0
83	77	78	0.008	0.162	0.0	1.0
84	19	80	3.42	18.15	62.0	220.0
85	1	83	0.00232	0.0875	0.0	1.0
86	30	82	1.86	4.84	34.0	110.0
regulator					tap	
501	8	9	0.0665	2.42	1.03	110.0
502	25	26	0.072	2.62	0.96	110.0
503	6	32	0.201	5.7	1.0	110.0
504	19	21	0.27	10.5	0.98	110.0
505	7	28	0.145	5.05	0.978	110.0
506	26	42	0.81	13.4	0.98	110.0
507	27	41	0.005	0.103	1.07	1.0
508	12	51	0.0024	0.0206	1.03	1.0
generator	voltage magnitude		nominal voltage			
83	16.05		14.70			
load	MW	MVAR				
1	0.0	0.0	220.0			
2	-42.0	-21.0	220.0			
3	0.0	0.0	220.0			
4	0.0	0.0	220.0			
5	-14.0	-7.0	110.0			
6	0.0	0.0	220.0			

7	-76.0	-39.0	220.0
8	0.0	0.0	110.0
9	0.0	0.0	110.0
10	-1.5	-0.6	110.0
11	-165.0	-82.0	110.0
12	-48.0	-24.0	110.0
13	0.0	0.0	110.0
14	-53.0	-16.0	110.0
15	0.0	0.0	110.0
16	0.0	0.0	110.0
17	-16.0	-2.0	110.0
18	-25.0	-4.0	110.0
19	0.0	0.0	220.0
20	-28.0	-14.0	110.0
21	0.0	0.0	110.0
22	-20.0	-10.0	110.0
23	-34.0	-17.0	110.0
24	-3.0	-1.0	110.0
25	0.0	0.0	110.0
26	0.0	0.0	110.0
27	114.0	70.0	14.0
28	-12.0	-6.0	110.0
29	-5.0	-2.0	110.0
30	-32.0	-16.0	110.0
31	-30.0	-15.0	110.0
32	-30.0	-15.0	110.0
33	0.0	0.0	110.0
34	-6.0	-2.0	110.0
35	0.0	0.0	110.0
36	0.0	0.0	110.0
37	-10.0	-4.0	110.0
38	30.0	10.0	110.0
39	0.0	0.0	110.0
40	0.0	0.0	110.0
41	0.0	0.0	110.0
42	-96.0	-30.0	22.0
43	39.0	12.0	10.52
44	83.0	40.0	10.5
45	38.0	16.0	10.5
46	0.0	0.0	220.0
47	48.0	20.0	10.5
48	0.0	0.0	220.0
49	21.0	2.0	10.5
50	0.0	0.0	220.0
51	0.0	0.0	88.0
52	-17.0	-7.0	88.0
53	-6.0	-2.0	110.0
54	0.0	0.0	110.0
55	-14.0	-6.0	110.0
56	28.0	12.0	10.5
57	0.0	0.0	220.0
58	125.0	20.0	10.76
59	90.0	12.0	10.76
60	9.0	4.0	10.76
61	26.0	16.0	10.2
62	80.0	48.0	10.75
63	0.0	0.0	88.0

64	0.0	0.0	88.0
65	-4.0	-2.0	88.0
66	-5.0	-2.0	88.0
67	0.0	0.0	110.0
68	60.0	40.0	10.2
69	50.0	15.0	14.7
70	-13.0	-6.0	88.0
71	-50.0	-25.0	110.0
72	-118.0	-58.0	110.0
73	30.0	10.0	10.3
74	0.0	0.0	220.0
75	80.0	30.0	10.1
76	0.0	0.0	220.0
77	50.0	20.0	10.3
78	0.0	0.0	220.0
79	6.0	0.0	21.1
80	0.0	0.0	220.0
81	44.0	15.0	10.1
82	-18.0	-6.0	110.0E

SYSTEM LOAD FLOW 1973 LOADINGS

PAGE 1

BUS	MW	MVAR	VOLTAGE		KV	PER UNIT	DEGREES	MVAR LINE CHARGING
1	-0.00	0.00	233.48	+4.66J	233.53	1.0615	1.1443	24.81
2	-41.99	-21.00	217.73	+5.64J	217.81	0.9900	1.4840	9.73
3	0.00	0.00	229.02	+14.86J	229.50	1.0432	3.7133	13.64
4	0.00	0.00	226.95	+13.87J	227.37	1.0335	3.4967	6.31
5	-14.00	-7.00	111.36	-3.68J	111.42	1.0129	-1.8912	0.87
6	0.00	0.00	231.85	+2.88J	231.87	1.0540	0.7125	3.33
7	-76.00	-39.00	230.66	+0.26J	230.66	1.0485	0.0642	17.93
8	-0.00	-0.00	111.04	+10.30J	111.51	1.0138	5.3022	5.43
9	0.00	0.00	114.53	+9.38J	114.91	1.0446	4.6825	0.55
10	-1.50	-0.60	113.11	+4.58J	113.20	1.0291	2.3204	8.44
11	-164.97	-81.98	104.47	-1.59J	104.48	0.9498	-0.8721	0.19
12	-47.99	-23.99	105.04	-0.92J	105.04	0.9549	-0.5018	3.60
13	-0.00	-0.00	105.58	-0.62J	105.58	0.9598	-0.3342	0.70
14	-52.99	-15.99	104.25	-2.83J	104.29	0.9481	-1.5562	0.07
15	0.00	0.00	108.35	+6.92J	108.57	0.9870	3.6535	1.26
16	-0.00	-0.00	109.03	+7.77J	109.31	0.9937	4.0742	1.29
17	-16.00	-2.00	99.02	-3.09J	99.07	0.9006	-1.7883	1.26
18	-24.99	-3.99	95.58	-6.51J	95.80	0.8709	-3.8943	0.39
19	-0.00	-0.00	236.30	+8.24J	236.44	1.0747	1.9969	24.21
20	-28.00	-14.00	105.94	-8.65J	106.30	0.9663	-4.6663	0.41
21	0.00	-0.00	116.24	+1.00J	116.24	1.0567	0.4916	1.66
22	-20.00	-10.00	114.79	-0.50J	114.79	1.0436	-0.2517	1.32
23	-33.99	-17.00	104.67	-1.42J	104.68	0.9516	-0.7762	5.81
24	-3.00	-1.00	115.93	+2.89J	115.97	1.0543	1.4272	5.18
25	-0.00	0.00	116.00	+0.55J	116.00	1.0545	0.2709	1.68
26	0.00	-0.00	111.55	-0.09J	111.55	1.0141	-0.0451	0.00
27	113.99	70.00	14.10	+1.24J	14.16	1.0112	5.0164	0.00
28	-12.00	-6.00	110.97	-2.84J	111.01	1.0091	-1.4656	0.10
29	-5.00	-2.00	115.21	-0.43J	115.21	1.0473	-0.2135	0.70
30	-32.00	-16.00	114.76	-0.96J	114.77	1.0433	-0.4794	0.74
31	-30.00	-15.00	113.78	-2.03J	113.80	1.0345	-1.0230	0.47
32	-30.00	-15.00	113.89	-1.98J	113.90	1.0355	-0.9965	3.65
33	0.00	0.00	108.32	-6.81J	108.53	0.9867	-3.5990	1.39
34	-6.00	-2.00	114.03	+3.61J	114.08	1.0371	1.8112	0.78
35	-0.00	0.00	105.82	+1.12J	105.83	0.9621	0.6038	0.75
36	-0.00	0.00	105.26	+0.17J	105.26	0.9569	0.0951	1.43
37	-10.00	-4.00	110.51	-4.24J	110.59	1.0054	-2.1990	0.33
38	30.00	10.00	114.18	+7.59J	114.43	1.0403	3.8013	1.18
39	0.00	-0.00	116.37	+0.91J	116.37	1.0579	0.4494	0.43
40	0.00	-0.00	111.38	-3.67J	111.44	1.0131	-1.8883	1.18
41	-0.00	-0.00	111.70	-1.82J	111.71	1.0156	-0.9340	0.02
42	-95.99	-29.99	22.08	-0.44J	22.09	1.0040	-1.1327	0.00
43	38.99	12.00	10.53	+1.48J	10.63	1.0107	8.0142	0.00
44	82.99	40.00	11.33	+1.84J	11.48	1.0930	9.2128	0.00
45	38.00	16.00	11.26	+1.74J	11.40	1.0855	8.7988	0.00

BUS	MW	MVAR	VOLTAGE	KV	PER UNIT	DEGREES	MVAR LINE CHARGING
46	-0.00	-0.00	229.50 +15.76J	230.04	1.0456	3.9287	0.90
47	48.00	20.00	11.24 +1.67J	11.37	1.0825	8.4492	0.00
48	-0.00	-0.00	229.77 +16.29J	230.35	1.0470	4.0554	0.48
49	21.00	2.00	10.80 +1.89J	10.96	1.0438	9.9167	0.00
50	-0.00	-0.00	227.09 +14.29J	227.54	1.0343	3.6002	0.52
51	-0.00	0.00	86.38 -1.07J	86.38	0.9816	-0.7074	0.21
52	-17.00	-7.00	82.75 -2.06J	82.78	0.9406	-1.4230	0.19
53	-6.00	-2.00	103.66 -2.71J	103.70	0.9427	-1.4996	0.03
54	0.00	-0.00	103.71 -2.68J	103.75	0.9432	-1.4775	0.83
55	-14.00	-6.00	102.42 -3.44J	102.48	0.9316	-1.9220	0.29
56	28.00	12.00	11.11 +1.50J	11.21	1.0679	7.7001	0.00
57	0.00	0.00	227.16 +14.43J	227.61	1.0346	3.6340	0.21
58	124.98	20.01	10.92 +1.89J	11.08	1.0301	9.8146	0.00
59	89.99	12.01	10.89 +1.88J	11.05	1.0267	9.8239	0.00
60	9.00	4.00	10.91 +1.39J	11.00	1.0221	7.2427	0.00
61	26.00	16.00	11.10 +0.60J	11.12	1.0899	3.1049	0.00
62	79.99	48.00	11.27 +0.57J	11.29	1.0500	2.8877	0.00
63	-0.00	-0.00	89.09 -1.25J	89.10	1.0125	-0.8067	1.24
64	0.00	0.00	85.11 -3.46J	85.18	0.9679	-2.3271	1.63
65	-4.00	-2.00	84.88 -3.50J	84.95	0.9653	-2.3628	0.11
66	-5.00	-2.00	84.25 -3.81J	84.33	0.9583	-2.5863	0.38
67	0.00	-0.00	116.88 +1.80J	116.89	1.0627	0.8838	0.10
68	60.00	40.00	11.21 +0.76J	11.24	1.1018	3.8804	0.00
69	50.00	15.00	15.64 +0.95J	15.67	1.0661	3.4887	0.00
70	-13.00	-6.00	90.08 -0.23J	90.08	1.0237	-0.1445	2.00
71	-50.00	-25.00	110.35 -3.22J	110.40	1.0036	-1.6736	0.05
72	-117.99	-57.99	110.82 -3.00J	110.86	1.0078	-1.5510	0.02
73	30.00	10.00	11.32 +1.36J	11.41	1.1074	6.8603	0.00
74	0.00	-0.00	236.84 +9.36J	237.03	1.0774	2.2628	1.85
75	80.00	30.00	11.16 +1.33J	11.24	1.1129	6.7932	0.00
76	-0.00	0.00	237.72 +11.16J	237.98	1.0817	2.6873	1.81
77	50.00	20.00	11.39 +1.28J	11.46	1.1126	6.4355	0.00
78	-0.00	0.00	237.83 +11.03J	238.09	1.0822	2.6546	2.49
79	6.00	0.00	22.84 +1.46J	22.89	1.0849	3.6645	0.00
80	0.00	-0.00	237.96 +11.91J	238.26	1.0830	2.8657	3.52
81	44.00	15.00	11.14 +1.41J	11.22	1.1114	7.2111	0.00
82	-18.00	-6.00	114.44 -1.30J	114.45	1.0404	-0.6518	1.47
83	-25.42	38.79	16.05 +0.00J	16.05	1.0918	0.0000	0.00

TOTAL GENERATION= 1051 MW 412.0 MVAR 1142 MVA

LINE NO	BUS	MW	MVAR	MVA	BUS	MW	MVAR	MVA	MW LOSS	MVAR LOSS	TAP
1	10	-81.42	-41.38	-91.33	23	77.25	34.15	84.46	4.173	7.239	
2	12	13.29	-2.16	13.46	36	-13.34	2.02	-13.49	0.049	0.134	
3	8	-37.90	-7.02	-38.54	36	36.23	3.35	36.38	1.672	3.667	
4	8	-61.16	-4.90	-61.36	12	57.79	-1.24	57.80	3.379	6.134	
5	9	-58.86	3.76	-58.98	38	58.55	-4.65	58.74	0.311	0.885	
6	8	-18.95	-5.11	-19.62	15	18.58	4.44	19.11	0.365	0.668	
7	15	-57.44	-14.49	-59.24	35	56.67	11.13	57.75	0.777	3.365	
8	12	90.30	90.81	128.07	13	-90.50	-91.54	-128.72	0.193	0.728	
9	11	121.77	58.95	135.28	12	-122.03	-60.06	-136.01	0.268	1.106	
10	2	-111.05	-105.07	-152.88	13	110.83	98.39	148.21	0.215	6.676	
11	2	71.29	56.80	91.15	3	-72.73	-62.72	-96.04	1.443	5.926	
12	1	59.93	-42.68	73.57	3	-60.71	39.26	-72.30	0.787	3.418	
13	3	-34.48	-49.44	-60.28	4	34.35	48.85	59.72	0.135	0.588	
14	2	81.75	59.55	101.14	4	-83.11	-65.12	-105.58	1.354	5.574	
15	1	27.70	13.75	30.92	19	-27.83	-14.34	-31.31	0.135	0.587	
16	1	-68.08	-24.26	-72.28	7	67.68	22.69	71.39	0.397	1.570	
17	10	-0.56	14.52	-14.53	24	0.34	-14.88	14.88	0.218	0.362	
18	1	5.92	-8.83	10.63	24	-5.92	8.74	-10.56	0.003	0.089	
19	24	-26.11	9.85	-27.91	25	25.91	-10.38	27.92	0.197	0.528	
20	28	113.01	52.89	124.77	41	-113.23	-54.28	-125.57	0.221	1.390	
21	21	-21.75	-6.52	-22.70	32	21.47	5.84	22.25	0.279	0.687	
22	6	70.50	36.56	79.42	19	-71.04	-38.88	-80.98	0.536	2.323	
23	5	11.87	4.32	12.63	32	-12.06	-4.61	-12.91	0.194	0.285	
24	20	28.00	13.59	31.12	33	-28.32	-14.41	-31.78	0.326	0.816	
25	11	43.21	22.84	48.87	23	-43.25	-22.96	-48.97	0.043	0.116	
26	32	-29.07	-15.00	-32.71	33	28.32	13.02	31.17	0.751	1.980	
27	10	-4.26	14.83	-15.43	34	4.16	-14.98	15.55	0.100	0.153	
28	14	22.86	6.08	23.66	36	-22.89	-6.80	-23.88	0.026	0.719	
29	14	30.13	9.85	31.70	35	-30.18	-11.14	-32.17	0.046	1.291	
30	8	-36.79	-6.86	-37.42	16	36.19	5.95	36.68	0.591	0.910	
31	16	-45.14	-10.64	-46.38	17	41.68	5.41	42.03	3.460	5.226	
32	17	-25.69	-4.67	-26.11	18	24.99	3.60	25.25	0.697	1.070	
33	21	-28.59	-9.89	-30.25	22	28.36	9.40	29.88	0.232	0.490	
34	24	-1.98	-16.46	-16.57	34	1.84	16.20	16.30	0.141	0.255	
35	29	26.81	8.92	28.26	39	-26.98	-9.32	-28.54	0.165	0.403	
36	21	-1.05	5.45	-5.55	39	1.05	-5.45	5.55	0.003	0.007	
37	31	8.35	7.31	11.09	32	-8.35	-7.32	-11.10	0.004	0.011	
38	29	-21.81	-7.62	-23.11	31	21.65	7.23	22.82	0.162	0.398	
39	22	-8.36	-0.72	-8.39	82	8.34	0.66	8.37	0.020	0.060	
40	7	75.62	37.28	84.31	19	-76.18	-40.80	-86.42	0.562	3.527	
41	12	26.30	0.23	26.30	35	-26.49	-0.74	-26.50	0.188	0.513	
42	21	-41.87	-20.14	-46.46	30	41.67	19.18	45.87	0.200	0.959	
43	5	2.13	1.81	2.79	40	-2.13	-1.81	-2.79	0.000	0.000	
44	32	-12.38	-4.67	-13.24	40	12.18	4.38	12.95	0.198	0.290	
45	37	10.00	3.67	10.65	40	-10.06	-3.75	-10.73	0.056	0.083	

PAGE 4

LINE NO	BUS	MW	MVAR	MVA	BUS	MW	MVAR	MVA	MW LOSS	MVAR LOSS	TAP
46	10	87.74	4.19	87.84	38	-88.55	-6.53	-88.79	0.813	2.336	
47	3	82.50	30.41	87.93	44	-82.99	-40.00	-92.13	0.490	9.591	
48	15	38.86	8.79	39.84	43	-38.99	-12.00	-40.80	0.132	3.210	
49	46	47.74	16.08	50.38	48	-47.77	-16.21	-50.44	0.029	0.128	
50	3	85.43	28.84	90.17	46	-85.52	-29.23	-90.38	0.090	0.389	
51	50	27.88	9.86	29.57	57	-27.88	-9.88	-29.58	0.004	0.020	
52	4	48.76	9.96	49.76	50	-48.77	-10.06	-49.80	0.017	0.095	
53	54	-14.13	-5.88	-15.30	55	14.00	5.70	15.12	0.129	0.181	
54	53	6.00	1.97	6.31	54	-6.00	-1.97	-6.32	0.002	0.003	
55	13	-20.34	-7.56	-21.69	54	20.13	7.03	21.32	0.208	0.531	
56	51	-17.65	-7.32	-19.11	52	17.00	6.81	18.31	0.651	0.518	
57	56	-28.00	-12.00	-30.46	57	27.88	9.67	29.51	0.117	2.327	
58	49	-21.00	-2.00	-21.09	50	20.90	-0.32	20.90	0.100	2.319	
59	45	-38.00	-16.00	-41.23	46	37.78	12.25	39.71	0.219	3.751	
60	47	-48.00	-20.00	-52.00	48	47.77	15.73	50.29	0.226	4.268	
61	39	25.93	14.34	29.63	61	-26.00	-16.00	-30.53	0.070	1.655	
62	63	-9.41	-2.26	-9.68	64	9.05	1.92	9.25	0.360	0.339	
63	64	-4.01	-1.90	-4.44	65	4.00	1.89	4.42	0.010	0.008	
64	64	-5.04	-1.65	-5.31	66	5.00	1.62	5.25	0.043	0.039	
65	42	79.52	40.42	89.20	62	-79.99	-48.00	-93.29	0.474	7.578	
66	26	-9.41	-1.15	-9.48	63	9.41	1.02	9.47	0.000	0.127	
67	8	124.17	9.95	124.57	58	-124.98	-20.01	-126.57	0.815	10.055	
68	8	89.51	4.81	89.64	59	-89.99	-12.01	-90.78	0.477	7.193	
69	16	8.95	3.40	9.57	60	-9.00	-4.00	-9.85	0.048	0.601	
70	21	59.71	34.99	69.21	67	-59.80	-35.60	-69.60	0.092	0.606	
71	67	59.80	35.50	69.55	68	-60.00	-40.00	-72.11	0.192	4.498	
72	24	49.95	13.04	51.62	69	-50.00	-15.00	-52.20	0.051	1.954	
73	24	-13.27	-4.48	-14.01	70	13.00	4.00	13.60	0.271	0.485	
74	28	-50.18	-25.27	-56.19	71	50.00	24.95	55.88	0.185	0.320	
75	28	-118.06	-58.22	-131.64	72	117.99	57.97	131.46	0.071	0.254	
76	19	29.84	9.05	31.18	74	-29.87	-9.21	-31.26	0.031	0.161	
77	19	79.43	24.21	83.03	76	-79.64	-25.33	-83.57	0.216	1.119	
78	19	49.68	18.00	52.84	78	-49.81	-18.70	-53.20	0.134	0.698	
79	80	43.86	11.33	45.30	81	-44.00	-15.00	-46.48	0.140	3.674	
80	79	-6.00	-0.00	-6.00	80	5.99	-0.08	5.99	0.011	0.083	
81	73	-30.00	-10.00	-31.62	74	29.87	7.36	30.76	0.127	2.642	
82	75	-80.00	-30.00	-85.44	76	79.64	23.52	83.04	0.354	6.483	
83	77	-50.00	-20.00	-53.85	78	49.81	16.20	52.38	0.187	3.795	
84	19	49.68	13.90	51.59	80	-49.85	-14.76	-51.99	0.163	0.864	
85	1	-25.46	37.21	-45.09	83	25.42	-38.79	46.38	0.042	1.579	
86	30	-9.67	-3.92	-10.44	82	9.66	3.88	10.41	0.015	0.040	
501	8	-58.88	3.68	-59.00	9	58.86	-4.32	59.02	0.018	0.639	1.030
502	25	-25.91	8.70	-27.33	26	25.91	-8.86	27.38	0.004	0.158	0.960
503	6	-70.50	-39.89	-81.01	32	70.40	37.11	79.58	0.098	2.783	1.000
504	19	-33.57	4.65	-33.89	21	33.55	-5.55	34.01	0.023	0.899	0.980

LINE NO	BUS	MW	MVAR	MVA	BUS	MW	MVAR	MVA	MW LOSS	MVAR LOSS	TAP
505	7	-67.31	-38.90	-77.74	28	67.24	36.50	76.51	0.069	2.399	0.978
506	26	-16.50	10.01	-19.29	42	16.47	-10.42	19.49	0.025	0.417	0.980
507	27	-113.99	-70.00	-133.77	41	113.23	54.25	125.56	0.764	15.743	1.070
508	12	-17.66	-7.19	-19.06	51	17.65	7.11	19.03	0.009	0.077	1.030

TOTAL LINE LOSS= 34.09 MW 182.8 MVAR TOTAL LINE CHARGING= 173.6 MVAR
END