with zero and pole locations shown in Fig. 9 there exist three characteristic circles with parameters:

1) $p=0$, $r=1$, $\epsilon=1$ (circle of type I).

2) $p=2/3$, $r=\sqrt{5}/3$, $\epsilon=-1$ (circle of type II).

3) $p=3/2$, $r=\sqrt{5}/2$, $\epsilon=1$ (circle of type I).

References

[1] R. N. Biswas and E. S. Kuh, "Optimum synthesis of a class of multiple-loop feedback systems," *IEEE Trans. Circuit Theory*, vol. CT-18, pp. 582–587, Nov. 1971.

[2] K. R. Laker and M. S. Ghausi, "A comparison of active multiple-loop feedback techniques for realizing high-order bandpass filters," *IEEE Trans. Circuits Syst.*, vol. CAS-21, pp. 774–783, Nov. 1974.

[3] J. Tow, "Design and evaluation of shifted-companion form active filters," *Bell Syst. Tech. J.*, vol. 54, pp. 545–568, Mar. 1975.

[4] G. Szentirmai, "Systhesis of multiple feedback active filters," *Bell Syst. Tech. J.*, vol. 52, pp. 527–555, Apr. 1973.

[5] K. R. Laker and M. S. Ghausi, "Synthesis of a low-sensitivity multi-loop feedback active *RC* filter," *IEEE Trans. Circuits Syst.*, vol. CAS-21, pp. 252–259, Mar. 1974.

[6] J. Tow, "Some results on generalized follow-the-leader-feedback active filters," in *Proc. IEEE Int. Symp. Circuits and Systems*, (Phoenix, AZ) pp. 462–465, 1977.

[7] K. R. Laker and M. S. Ghausi, "Minimum sensitivity multiple-loop feedback bandpass active filters," in *Proc. IEEE Int. Symp. Circuits and Systems*, (Phoenix, AZ) pp. 458–461, 1977.

[8] K. R. Laker, M. S. Ghausi, and J. J. Kelly, "Minimum sensitivity leap-frog active filters," in *Proc. IEEE Int. Symp. Circuit Theory*, (San Francisco, CA) pp. 201–204, 1974.

[9] W. Schüssler, "Zur allgemeinen Theorie der Verzweigungsnetzwerke," *A.E.U.*, vol. 22, pp. 361–367, Aug. 1968.

[10] C. T. Chen, *Introduction to Linear System Theory*. New York: Holt, Rinehart and Winston, 1974.

[11] B. Styblińska, "Synthesis and optimization of multiloop feedback active filters," (in Polish), Ph.D. dissertation, Tech. Univ. of Warsaw, Warsaw, Poland, 1977.

[12] ——, "Sensitivity minimization in multiloop-feedback active filters with modification of the frequency response," in *Proc. Summer Symp. Circuit Theory*, (Prague, Czechoslovakia) pp. 290–294, 1977.

[13] R. M. Biernacki and B. Styblińska, "Synthesis of multiloop active filters with identical biquadratic blocks," *Acta Polytech.*, vol. III, 7, pp. 45–56, 1977.

[14] R. M. Biernacki and B. Styblińska, "General synthesis of active filters with identical biquadratic sections," in *Proc. European Conf. Circuit Theory and Design*, (Lausanne, Switzerland) pp. 63–67, 1978.

[15] R. M. Biernacki and B. Styblińska, "A new approach to synthesis of active filters in multiloop structures," *Bull. Acad. Pol. Sci.*, (in print).

[16] J. W. Bandler and D. Sinha, "FLOPT5—A program for minimax optimization using the accelerated least *p*th algorithm," Faculty of Engineering, McMaster University, Hamilton, Canada, Rep. SOC-218, 1978.

Radosław M. Biernacki was born in Warsaw, Poland, on June 4, 1945. He received the M.Sc. degree in electronics and the Ph.D. degree in technical sciences, both from the Technical University of Warsaw, Warsaw, Poland, in 1969 and 1976, respectively.

He became Research and Teaching Assistant in 1969 and Assistant Professor in 1976 in the Institute of Electronics Fundamentals, Technical University of Warsaw, Poland. Since 1978, he has been on leave with the Research Group on Simulation, Optimization and Control and with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, Ont., Canada, as a Post-Doctorate Fellow. His fields of interest in teaching and research are network theory and synthesis, active filters, optimization, computer-aided circuit design and, recently, fault analysis of analog circuits. This research was several times awarded prizes by the Rector of the Technical University of Warsaw. He also won a prize of the Minister of Science, Higher Education and Technology in Poland.

Bogumiła Styblińska received the M.Sc. in electronics and the Ph.D. degree in technical sciences in 1977, both from the Technical University of Warsaw, (Warsaw Polytechnik) Warsaw, Poland. Her Ph.D. dissertation concerned synthesis and optimization of multiloop active filters, was forwarded for a prize of the Minister of Science, Technology and Higher Education and was awarded by the Rector of the Technical University of Warsaw.

At present she is Assistant Professor at Institute of Electronics Fundamentals, Technical University of Warsaw. She is interested in network and systems theory, active *RC* and switched-capacitor filters, sensitivity analysis, and optimization.

Sparsity Considerations in Network Solution by Tearing

IBRAHIM N. HAJJ, MEMBER, IEEE

***Abstract*—Network solution by tearing consists of partitioning the network into subnetworks, solving each subnetwork separately, and then combining the subnetwork solutions to obtain the solution of the entire network. In this paper it is shown that all recently proposed sparse matrix algorithms for network solution by tearing belong to a set of algorithms which is derived by applying block matrix elimination to a partitioned system of network equations. The computational requirements of the algorithms are determined and compared. Equation sparsity is considered at all levels in the solution process. In particular, the structures of the equations at the subnetwork level as well as the interconnection level are analyzed in detail.**

I. Introduction

THE SOLUTION of networks by tearing is an approach by which part of a given network is torn away so that the remaining subnetwork or subnetworks can be analyzed independently. The solutions of the latter are

Manuscript received October 19, 1977; revised June 13, 1979. This work was supported in part by the National Research Council of Canada and in part by the Joint Services Electronics Program (U.S. Navy, U.S. Army, U.S. Air Force).

The author is with the Coordinated Science Laboratory, University of Illinois, Urbana-Champaigne, IL 61801.

0098-4094/80/0500-0357$00.75 ©1980 IEEE

then combined with that of the torn-away part in order to obtain the solution of the entire network. This approach, also known as "diakoptics" or piecewise analysis, was introduced by Kron [1] and expanded upon by many others to analyze power systems [2], electrical networks [3], and structural and other large-scale systems [4], to mention a few. Unfortunately, in its original form the method required the explicit computation of inverses of the submatrices that represented the individual subnetworks. As a result, any sparsity that might exist in the subnetwork equations could not be exploited. Recently, however, network analysis by tearing using *sparse matrix* solution techniques has been the subject of many investigations [5]–[15].

From the algebraic point of view tearing can be considered to be a partitioning of the network equations in a special way. Thus we shall use the terms "partitioning" and "tearing" interchangeably. Tearing is not restricted to linear networks. It can also be applied to the analysis of nonlinear resistive [10] as well as dynamic networks [12], [15], [16]. The use of tearing, however, has been questioned as to whether computations could be saved if sparse matrix techniques were instead employed to solve the network as a whole without tearing [11]. Nevertheless, there are a number of situations where tearing becomes advantageous or even necessary. This is true, for example, when a network is so large that its equations cannot be stored on an available computer even though sparse matrix solution techniques are being used. Tearing and overlay schemes thus become a necessity. Tearing techniques can also be employed in cases where the network consists of repetitive identical subnetworks so that the equations of only one such subnetwork need to be stored and solved. In addition, tearing allows parallel processing and the use of "latency" concepts [12] in finding the network solution.

There are many ways of tearing a network. In this paper our main concern is not with finding the most efficient way of tearing a given network, but rather with finding the most efficient way of solving the network equations once a partitioning pattern has been chosen. A number of sparse matrix algorithms have been proposed [5]–[11], [13]. The question is which one of these algorithms is more efficient than the others in a given case. In [18], George studied three algorithms for solving partitioned systems of equations. It turns out that all recently proposed tearing algorithms are included in these algorithms. In this paper we study these algorithms when they are applied to *network* solution by tearing. In addition, some variations on George's algorithms which were not included in [18] are derived and studied. It is shown that the structure of the subnetwork equations plays an important role in determining the computational complexity of the algorithms and in determining which algorithm is most efficient when applied to the solution of the subnetwork equations.

In the next section we review the formulation of the network equations in partitioned form and comment upon the structure of the various submatrices. In Section III we present the basic solution algorithms and their variations. The computational requirements of the algorithms are investigated in Section IV. The structure of the connection equation, which provides the link between the solution of the torn subnetworks and the solution of the entire network, is discussed in Section V. An example to illustrate the various solution algorithms is discussed in detail in Section VI.

II. Network Equations in Partitioned Form

Consider a linear network N which is to be analyzed by tearing. Let the network equations be formulated using the modified nodal approach [19]. Tearing can then be viewed as a problem of partitioning or ordering the network equations in a way such that the associated network equations have either a bordered-block-diagonal or bordered-block-triangular structure. In this paper we consider the case where the equations have a bordered-block-diagonal form. From the network point of view, tearing can be accomplished by either removing a set of branches [11] or a set of nodes [14]. If the tearing set consists of branches, such that there is no coupling neither among the torn subnetworks nor between the subnetworks and the tearing branches, the partitioned equations will have the following form [11]:

$$\left[\begin{array}{cccc|cc} \boldsymbol{M}_1 & & & & \boldsymbol{Y}_{1o} & \boldsymbol{A}_1 \\ & \boldsymbol{M}_2 & & & \boldsymbol{Y}_{2o} & \boldsymbol{A}_2 \\ & & \ddots & & \vdots & \vdots \\ & & & \boldsymbol{M}_k & \boldsymbol{Y}_{ko} & \boldsymbol{A}_k \\ \hline \boldsymbol{Y}_{o1}^T & \boldsymbol{Y}_{o2}^T & \cdots & \boldsymbol{Y}_{ok}^T & \boldsymbol{Y}_{oo} & \boldsymbol{A}_t \\ \boldsymbol{A}_1^T & \boldsymbol{A}_2 & \cdots & \boldsymbol{A}_k^T & \boldsymbol{A}_t^T & -\boldsymbol{Z}_t \end{array}\right] \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_k \\ \boldsymbol{v}_o \\ \boldsymbol{i}_t \end{bmatrix} = \begin{bmatrix} \boldsymbol{Y}_1 \\ \boldsymbol{y}_2 \\ \vdots \\ \boldsymbol{y}_k \\ \boldsymbol{j}_o \\ \boldsymbol{e}_t \end{bmatrix} \tag{1}$$

where $\boldsymbol{M}_i$ is an $m_i \times m_i$ (nonsingular) matrix, $\boldsymbol{x}_i$ an m_i-vector containing the node-to-global-datum voltages together with a subset of subnetwork branch currents and $\boldsymbol{y}_i$ is the vector of sources in $N_i \cdot \boldsymbol{Y}_{io}$ and $\boldsymbol{Y}_{oi}$ each contains *at most* one nonzero column. $\boldsymbol{A}_i$ is a topological matrix which contains exactly b_{ti} nonzero columns and n_{ti} nonzero rows, where b_{ti} is the number of tearing branches incident with N_i at n_{ti} nodes other than the local-datum node. $\boldsymbol{Y}_{oo}$ is a diagonal matrix and $\boldsymbol{Z}_t$ is the impedance matrix of the tearing branches; $\boldsymbol{v}_o$ is the vector of voltages at the local-datum nodes and $\boldsymbol{i}_t$ is the vector of currents through the tearing branches. The reason the local-datum nodes have been torn away and numbered last is that [11]: $\boldsymbol{Y}_{oo} - \Sigma_{i=1}^{k} \boldsymbol{Y}_{oi}^T \boldsymbol{M}_i^{-1} \boldsymbol{Y}_{io} = \boldsymbol{0}$.

If the tearing set consists of node voltages only, such that no coupling exists neither among the torn subnetworks nor between the subnetworks and the tearing nodes,[1] the network equations will have the following

[1] Note that as long as there is no coupling among the subnetworks themselves, it is possible to have coupling between the subnetworks and the tearing variables (v_o and i_t in the case of BT and v_c in the case of NT) and 1) and 2) would still have bordered-block-diagonal forms. This coupling, however, would add extra branches to the connection network which is discussed in Section V. For simplicity, it is assumed here that no coupling exists between the subnetworks and the tearing set.

TABLE I
Basic Factorization Procedures

F_1	F_2	F_3
$\begin{bmatrix} L & 0 \\ W^T & L_r \end{bmatrix}\begin{bmatrix} U & V \\ 0 & U_r \end{bmatrix}$	$\begin{bmatrix} LU & 0 \\ Q^T & L_r \end{bmatrix}\begin{bmatrix} I & \bar{V} \\ 0 & U_r \end{bmatrix}$	$\begin{bmatrix} I & 0 \\ \bar{W}^T & L_r \end{bmatrix}\begin{bmatrix} LU & P \\ 0 & U_r \end{bmatrix}$

form [14], [15]:

$$\left[\begin{array}{ccccc|c} M_1 & & & & & Y_{1c} \\ & M_2 & & & & Y_{2c} \\ & & \cdot & & & \cdot \\ & & & \cdot & & \cdot \\ & & & & \cdot & \cdot \\ & & & & M_k & Y_{kc} \\ \hline Y_{c1}^T & T_{c2}^T & & \cdots & Y_{ck}^T & Y_{cc} \end{array}\right]\begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_k \\ v_c \end{bmatrix}=\begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ \cdot \\ y_k \\ i_c \end{bmatrix} \tag{2}$$

where Y_{ic} and Y_{ci} each now contains exactly n_{ci} nonzero columns, where n_{ci} is the number of tearing nodes connected to subnetwork N_i; v_c is the set of voltages at the torn nodes. Note that in (2), if the entire network equation is nonsingular, $Y_{cc}-\sum_{i=1}^{k} Y_{ci}^T M_i^{-1} Y_{ic}$ will also be nonsingular. For easy reference we denote the partitioned form in (1) as branch-tearing (BT) and in (2) as node-tearing (NT). Also, the vertical and the horizontal border submatrices in (1) and (2) will be denoted by P_i and Q_i, respectively; i.e., $P_i=[Y_{io}\ A_i]$ or $= Y_{ic}$ and $Q_i=[Y_{oi}\ A_i]$ or $= Y_{ci}$.

III. Solution Algorithms

Let systems (1) and (2) be written in the form:

$$\begin{bmatrix} M & P \\ Q^T & R \end{bmatrix}\begin{bmatrix} x \\ z \end{bmatrix}=\begin{bmatrix} y \\ s \end{bmatrix}. \tag{3}$$

System (3) may be solved using one of three different factorizations which are denoted by $\mathcal{F}_1$, $\mathcal{F}_2$, and $\mathcal{F}_3$, respectively, [18], and given in Table I, where $V=L^{-1}P$, $\bar{V}=U^{-1}V$, $W^T=Q^TU^{-1}$, $\bar{W}^T=W^TL^{-1}$, and $I\equiv$identity matrix. In all three factorization procedures above

$$L_rU_r=R-Q^TU^{-1}L^{-1}P. \tag{4}$$

The difference lies in the order in which $Q^TU^{-1}L^{-1}P$ is computed. There are two variations in executing factorization $\mathcal{F}_2$ and $\mathcal{F}_3$ which are now given.

Factorization Variations

(1) If $\bar{V}$ is not required other than in the factorization procedure $\mathcal{F}_2$, then only those *rows* of $\bar{V}$ corresponding to the nonzero *columns* of Q^T are required in order to calculate $Q^T\bar{V}=Q^T(U^{-1}L^{-1}P)$; denote this variation in executing $\mathcal{F}_2$ as $\bar{\mathcal{F}}_2$.

(2) If $\bar{W}$ is not required other than in the factorization procedure $\mathcal{F}_3$, then only those *columns* of $\bar{W}^T$ corresponding to the nonzero *rows* of P are required in order to calculate $\bar{W}^TP=(Q^TU^{-1}L^{-1})P$; denote this variation in executing $\mathcal{F}_3$ as $\bar{\mathcal{F}}_3$.

TABLE II
Basic Substitution Procedures

Step	S_1	S_2	S_3
1	$La = y$	$La=y,\ Ub=a$	$y = y$
2	$\bar{y} = W^Ta$	$\bar{y} = Q^Tb$	$\bar{y} = \bar{W}^Ty$
3	$L_rU_rz = s-\bar{y}$	$L_rU_rz = s-\bar{y}$	$L_rU_rz = s-\bar{y}$
4	$z_1 = Vz$	$z_2 = \bar{V}z$	$z_3 = Pz$
5	$\hat{a} = a-z_1$	$x = b-z_2$	$\hat{y} = y-z_3$
6	$Ux = \hat{a}$	$x = x$	$L\hat{a}=\hat{y},\ Ux=\hat{a}$

TABLE III
Variations on Substitution Procedures

Step	Variations	Associated Factorization
S_1^* : 2	$\bar{y} = Q^T(U^{-1}a)$	$F_1,\ \bar{F}_2$
$\bar{S}_1^*$: 4	$z_1 = L^{-1}(Pz)$	$F_1,\ \bar{F}_3$
S_1^{**} : 2 4	$\bar{y} = Q^T(U^{-1}a)$ $z_1 = L^{-1}(Pz)$	$F_1,\ \bar{F}_2,\ \bar{F}_3$
S_2^* : 4	$z_2 = U^{-1}(Vz)$	$F_1,\ \bar{F}_2$
S_2^{**} : 4	$z_2 = U^{-1}L^{-1}(Pz)$	$F_1,\ \bar{F}_2,\ \bar{F}_3$
S_3^* : 2	$\bar{y} = W^T(L^{-1}y)$	$F_1,\ \bar{F}_3$
S_3^{**} : 2	$\bar{y} = Q^T(U^{-1}L^{-1}y)$	$F_1,\ \bar{F}_2,\ \bar{F}_3$

Associated with the three basic factorizations given in Table I are three different *substitution* procedures denoted by $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$ [18] and given in Table II. Note that Steps 1, 2, 4, 5, and 6 in Table II are performed at the subnetwork level while Step 3 is performed at the interconnection level. There are variations in the execution of the substitution procedures at the subnetwork level which are discussed below.

Substitution Variations

In all three substitution procedures Step 3 is identical, and Steps 1 and 6 combined look the same. The main difference lies in Steps 2 and 4. Variations in executing these steps are set out in Table III. As can be seen from the third column of this table, the variations can be used with more than one substitution procedure. It is important to note that in these variations $\bar{V}$ and $\bar{W}$ do not explicitly appear; they are only required in factorizations $\mathcal{F}_2$ and $\mathcal{F}_3$ where they are used to calculate $Q^T\bar{V}$ and $\bar{W}^TP$. Thus when substitution variations are used in conjunction with either $\mathcal{F}_2$ or $\mathcal{F}_3$, only those rows of $\bar{V}$ or $\bar{W}$ that correspond to the nonzero rows of Q or P are required and one only needs to use the factorization variations $\bar{\mathcal{F}}_2$ or $\bar{\mathcal{F}}_3$.

Remark

$\mathcal{F}_1+\mathcal{S}_1$ has been used in [6]–[8], [10], and [11], and as proved in [20] $\mathcal{F}_1+\mathcal{S}_1$ is equivalent to step-by-step solution of the equations. $\mathcal{F}_2+\mathcal{S}_2$ has been used in [5], [13],

[15], and [21]; $\bar{\mathfrak{F}}_2 + \mathbb{S}_2^{**}$ in [9]; $\mathbb{S}_1^{**}$ has been considered in [18] and [20]; $\mathbb{S}_2^{**}$ and $\mathbb{S}_3^{**}$ have been mentioned, but not investigated, in [18]. The remaining variations appear not to have been considered elsewhere.

IV. Computational Requirements

Before we calculate the computational requirements of the solution procedures, we first give some definitions and for easy reference some lemmas which were proved in [18] and [20].

Let the symbol $|.|$ denote the number of nonzero elements in a vector or a matrix. $M(.j)$ and $M(j.)$ the jth column and the jth row of a matrix M, respectively. For any triangular $m \times m$ matrix T, T^I is defined to be $T-I$ if T has a diagonal of ones; otherwise $T^I = T$. Consider a subset ρ of the rows of T, with the first (last) row being row r when T is upper (lower) triangular. Let $\hat{T}$ be an $r \times m$ submatrix which consists of the first r rows of T when T is lower triangular or the last $m-r+1$ rows when T is upper triangular. Define $\Pi(\rho)$ to be the set of indexes which is constructed as follows: Let $\Pi^o(\rho)$ be the set of indexes of the nonzero columns of the ρ rows of $\hat{T}$, $\Pi^1(\rho)$ the set of indexes of the nonzero columns of the $\Pi^o(\rho)$ rows of $\hat{T}$, $\Pi^{i+1}(\rho)$ the set of indexes of the nonzero columns of the $\pi^i(\rho)$ rows of $\hat{T}$, and so on until $\Pi^{k+1}(\rho) = \Pi^k(\rho) \equiv \Pi(\rho)$. We consider the number of long operations (multiplications and divisions) to be the basis for determining the computational complexity of an algorithm. In practice, however, other factors, such as storage, coding and programming complexities, should also be taken into consideration. Let $C(.)$ denote the amount of computation required by a given procedure.

Lemma 4.1 [20]

The number of operations required to factorize an $m \times m$ nonsingular matrix M into the product LU is

$$\alpha = \sum_{j=1}^{m} |L^I(.j)|\,|U^I(j.)|.$$

Lemma 4.2 [20]

If M is numerically symmetric, we can factorize $M = U^T D U$, where D is a diagonal matrix, and the number of operations required in the factorization process is

$$\alpha^s = \sum_{j=1}^{m} (|U^I(j.)|+3)|U^I(j.)|/2.$$

Lemma 4.3 [18]

Let A, B, and C be given matrices with $A = BC$. The number of operations required to compute A from B and C is

$$\theta(BC) = \sum_{j=1}^{n} |B(.j)|\,|C(j.)|$$

where n is the number of columns in B. If A is known to be numerically symmetric and $B(ij) \neq 0 \Rightarrow C(ij) \neq 0$, then the number of operations required to compute A satisfies

$$\theta^s(BC) \leqslant \sum_{j=1}^{n} |B(.j)|(|C(j.)|+1)/2.$$

If $A = B^T B$, then

$$\theta^s(B^T B) = \sum_{j=1}^{n} |B(j.)|(|B(j.)|+1)/2.$$

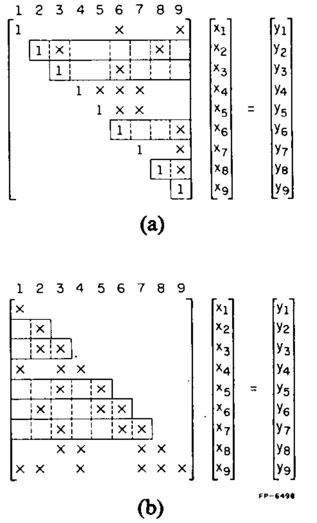


Fig. 1. (a) An upper triangular system of equations. (b) A lower triangular system of equations.

Lemma 4.4 [18]

Let T be an $m \times m$ nonsingular triangular matrix with $TX = Y$. The number of operations required to compute X from T and Y is

$$\theta(TX) = \sum_{j=1}^{m} |T^I(.j)|\,|X(j.)|.$$

In addition, the following lemma is proven here.

Lemma 4.5

Let T be a nonsingular $m \times m$ triangular matrix with $TX = Y$ where X and Y are $m \times m$ matrices. Let X_ρ be a subset of the rows of X and ρ the corresponding set of rows of T, with the first (last) row being row r when T is upper (lower) triangular. The number of operations required to compute X_ρ from T and Y is

$$\theta(TX_\rho) = \sum_{j \in \Pi(\rho)} \left(\sum_{k \in \Pi(\rho)} |T^I(kj)| \right) |X(j.)|.$$

Proof: $\Pi(\rho)$ consists of the set of indexes indicating those rows of X (and also of T) that are needed to compute X_ρ. By applying Lemma 4.4 the above result follows, but only if those rows and columns of T which correspond to $\Pi(\rho)$ are considered.

Examples on how to compute $\Pi(\rho)$ and $\theta(TX_\rho)$ in given situations follow. Consider the solution of $Tx = y$, where x and y are vectors and T is a 9×9 upper triangular matrix with ones on the diagonal, as shown in Fig. 1(a). Suppose $x_\rho = [x_2, x_3]$, then $\rho = [2,3]$, $\Pi^o(\rho) = [2,3,6,8]$, $\Pi^1(\rho) =$

$[2,3,6,8,9]=\Pi^2(\rho)=\Pi(\rho)$. Therefore, x_6, x_8, and x_9 should be determined before one could solve for x_2 and x_3. The number of operations required to compute x_2 and x_3 (and at the same time x_6, x_8, and x_9) is equal to the number of nonzero off-diagonal elements in rows $\Pi(\rho)$ of $\boldsymbol{T}$; or, equivalently, it is equal to the number of nonzero off-diagonal elements in columns $\Pi(\rho)$ of $\boldsymbol{T}$ whose indices belong to the set $\Pi(\rho)$; namely, $\theta(\boldsymbol{Tx}_\rho)=1+1+1+2=5$ operations. Note that if some of the components of $\boldsymbol{x}$ corresponding to $\Pi(\rho)$ are zero, $\theta(\boldsymbol{Tx}_\rho)$ would be further reduced, according to Lemma 4.5. Suppose that $\boldsymbol{T}$ is now a lower triangular matrix as shown in Fig. 1(b), and it is required to compute $\boldsymbol{x}_\rho=[x_6,x_7]$. It follows that $\rho=[7,6]$, $\Pi^o(\rho)=[7,6,5,3,2]=\Pi^1(\rho)=\Pi(\rho)$. Therefore, x_2, x_3, and x_5 should be determined before one could solve for x_6 and x_7. The number of operations required to solve for x_6 and x_7 is then $\theta(\boldsymbol{Tx}_\rho)=3+3+2+2+1=11$ operations.

Note that if $\boldsymbol{T}$ is upper triangular,

$$\theta(\boldsymbol{TX}_\rho)\leqslant\sum_{j=r}^{m}\left(\sum_{k=r}^{m}|\boldsymbol{T}^{I}(kj)|\right)|\boldsymbol{X}(j.)|$$

$$\leqslant\left(\sum_{j=r}^{m}|\boldsymbol{T}^{I}(j.)|\right)n.$$

If T is lower triangular,

$$\theta(\boldsymbol{TX}_\rho)\leqslant\sum_{j=1}^{r}\left(\sum_{k=1}^{r}|\boldsymbol{T}^{I}(kj)|\right)|\boldsymbol{X}(j.)|$$

$$\leqslant\left(\sum_{j=1}^{r}|\boldsymbol{T}^{I}(j.)|\right)n.$$

It should be noted that careful programming is necessary in order to implement the above lemmas. Such programs, however, are not difficult to develop [23].

The computational requirements of the solution procedures are now considered at both the subnetwork level, investigated below, and the interconnection level, which is dealt with in the next section.

The Subnetwork Level

Using Lemmas 4.1–4.5, the number of operations required to solve the subnetwork equations by each of the factorization and substitution procedures is given in Tables IV and V. Subscript i is dropped for the sake of clarity; superscript s refers to numerically symmetric matrices; $\bar{\boldsymbol{V}}_\rho$ and $\bar{\boldsymbol{W}}_\rho$ refer to the subsets of the rows of $\bar{\boldsymbol{V}}$ and $\bar{\boldsymbol{W}}$ which correspond to the nonzero rows of $\boldsymbol{Q}$ and $\boldsymbol{P}$, respectively; and $\boldsymbol{b}_\rho$ is the set of components of $\boldsymbol{b}$ which correspond to the nonzero rows of $\boldsymbol{Q}$.

Remarks

In Tables IV and V the number of operations given is that required for NT. If BT is used, the number of operations given in the tables could be reduced since the nonzero columns of $\boldsymbol{A}_i$ contain ±1 only and $\boldsymbol{Y}_{oi}^T\boldsymbol{M}_i^{-1}\boldsymbol{Y}_{io}=\boldsymbol{0}$. Furthermore, in BT, when two or more tearing branches are incident with the same node in a subnetwork, the corresponding columns in $\boldsymbol{A}_i$ are either equal to or the negative of each other. In this case the size of $\boldsymbol{A}_i$ in the factorization procedures can be reduced by temporarily eliminating the redundant columns of $\boldsymbol{A}_i$. For BT, then, the number of operations in Tables IV and V should be reduced as follows: 1) For $\mathcal{F}_2$, $\bar{\mathcal{F}}_2$, $\mathcal{F}_2^{s}$, and $\bar{\mathcal{F}}_2^{s}$ the number of operations is reduced by $|\boldsymbol{Y}_{oi}|$, 2) for $\mathcal{F}_3$, $\bar{\mathcal{F}}_3$, $\mathcal{F}_3^{s}$, and $\bar{\mathcal{F}}_3^{s}$ the number of operations is reduced by $|\boldsymbol{Y}_{io}|$, 3) in all entries, $\theta(\boldsymbol{Q}_i^T\bar{\boldsymbol{V}}_i)$, $\theta(\boldsymbol{P}_i^T\bar{\boldsymbol{W}}_i)$, $\theta(\boldsymbol{Q}_i^T\boldsymbol{b}_i)$, and $\theta(\boldsymbol{P}_i\boldsymbol{z})$ are reduced to $\theta(\boldsymbol{Y}_{oi}^T\bar{\boldsymbol{V}}_{2i})$, $\theta(\boldsymbol{Y}_{io}^T\bar{\boldsymbol{W}}_{2i})$, $\theta(\boldsymbol{Y}_{oi}^T\boldsymbol{b}_i)$, and $\theta(\boldsymbol{Y}_{io}^T\boldsymbol{z}(i))$, respectively, where $\bar{\boldsymbol{V}}_{2i}=\boldsymbol{M}_i^{-1}\boldsymbol{A}_i$, $\bar{\boldsymbol{W}}_{2i}^T=\boldsymbol{A}_i^T\boldsymbol{M}_i^{-1}$, and $z(i)$ is the ith component of $\boldsymbol{z}$. This, of course, does not mean that BT requires less computation than NT when solving the same system since the formulations in the two cases are different [15].

Note that $\mathcal{C}(\bar{\mathcal{F}}_2)$, $\mathcal{C}(\bar{\mathcal{F}}_3)$, $\mathcal{C}(\bar{\mathcal{F}}_2^{s})$, and $\mathcal{C}(\bar{\mathcal{F}}_3^{s})$ are less than $\mathcal{C}(\mathcal{F}_2)$, $\mathcal{C}(\mathcal{F}_3)$, $\mathcal{C}(\mathcal{F}_2^{s})$, and $\mathcal{C}(\mathcal{F}_3^{s})$, respectively; however, $\mathcal{F}_2$, $\mathcal{F}_3$, $\mathcal{F}_2^{s}$, and $\mathcal{F}_3^{s}$ are needed whenever $\mathcal{S}_2$ or $\mathcal{S}_3$ is to be used.

TABLE IV
NUMBER OF OPERATIONS REQUIRED IN THE VARIOUS FACTORIZATION PROCEDURES

Procedure	Number of Operations
F_1	$\alpha + \theta(LV) + \theta(U^TW) + \theta(W^TV)$
F_2	$\alpha + \theta(LV) + \theta(U\bar{V}) + \theta(Q^T\bar{V})$
F_3	$\alpha + \theta(U^TW) + \theta(L^T\bar{W}) + \theta(P^T\bar{W})$
$\bar{F}_2$	$\alpha + \theta(LV) + \theta(U\bar{V}_\rho) + \theta(Q^T\bar{V}_\rho)$
$\bar{F}_3$	$\alpha + \theta(U^TW) + \theta(L^T\bar{W}_\rho) + \theta(P^T\bar{W}_\rho)$
F_1^{s}	$\alpha^s + \theta(LV) + \theta^s(V^TV)$
F_2^{s}	$\alpha^s + \theta(LV) + \theta(U\bar{V}) + \theta^s(Q^T\bar{V})$
F_3^{s}	$\alpha^s + \theta(U^TW) + \theta(L^T\bar{W}) + \theta^s(P^T\bar{W})$
$\bar{F}_2^{s}$	$\alpha^s + \theta(LV) + \theta(U\bar{V}_\rho) + \theta^s(Q^T\bar{V}_\rho)$
$\bar{F}_3^{s}$	$\alpha^s + \theta(U^TW) + \theta(L^T\bar{W}_\rho) + \theta^s(P^T\bar{W}_\rho)$

TABLE V
NUMBER OF OPERATIONS REQUIRED IN THE VARIOUS SUBSTITUTION PROCEDURES

Procedure	Number of Operations
S_1	$\theta(La) + \theta(W^Ta) + \theta(Vz) + \theta(Ux)$
S_1^*	$\theta(La) + \theta(Ub_\rho) + \theta(Q^Tb_\rho) + \theta(Vz) + \theta(Ux)$
$\bar{S}_1^*$	$\theta(La) + \theta(W^Ta) + \theta(Pz) + \theta(Lz_1) + \theta(Ux)$
S_1^{**}	$\theta(La) + \theta(Ub_\rho) + \theta(Q^Tb_\rho) + \theta(Pz) + \theta(Lz_1) + \theta(Ux)$
S_2	$\theta(La) + \theta(Ub) + \theta(Q^Tb) + \theta(\bar{V}z)$
S_2^*	$\theta(La) + \theta(Ub) + \theta(Q^Tb) + \theta(Vz) + \theta(Uz_2)$
S_2^{**}	$\theta(La) + \theta(Ub) + \theta(Q^Tb) + \theta(Pz) + \theta(Lz_1) + \theta(Uz_2)$
S_3	$\theta(\bar{W}^Ty) + \theta(Pz) + \theta(L\hat{a}) + \theta(Ux)$
S_3^*	$\theta(La) + \theta(W^Ta) + \theta(Pz) + \theta(L\hat{a}) + \theta(Ux)$
S_3^{**}	$\theta(La) + \theta(Ub_\rho) + \theta(Q^Tb_\rho) + \theta(Pz) + \theta(L\hat{a}) + \theta(Ux)$

It is important to note that in any given case the choice of the most efficient algorithm depends on both the particular problem being solved and on the subnetwork equation ordering strategy. A particular ordering strategy may favor one algorithm over another. In the following, however, we assume that the same equation order is adopted in all the algorithms. In Tables IV and V it can be clearly seen that many entries in column two are common to many of the algorithms. Thus when comparing the computational requirements of the algorithms, it is only necessary to calculate and compare some of the entries in column two of these tables. However, with a knowledge of the structures of the various subnetwork matrices and vectors, some results can be deduced from the tables, which eliminates the need for comparing some of the algorithms. Since z depends on the solution of the interconnection equation, which is not known at the outset, it is assumed to be full. The nonzero components of z_1, z_2, and z_3, however, correspond to the nonzero rows of V, $\bar{V}$, and P, respectively. Let $\sigma = |L^I| + |U^I|$.

Fact 4.1: Suppose the subnetwork matrix is structurally symmetric and in $\mathcal{F}_2$ and $\mathcal{F}_3$ U and L, respectively, are chosen to have diagonals of ones. Then

1) $|P| = |Q|$, $|L^I| = |U^I|$, $|W| = |V|$, and $|\bar{W}| = |\bar{V}|$
2) $\mathcal{C}(\underline{\mathcal{F}}_3) = \mathcal{C}(\underline{\mathcal{F}}_2)$
3) $\mathcal{C}(\bar{\mathcal{F}}_3) = \mathcal{C}(\bar{\mathcal{F}}_2)$
4) $\mathcal{C}(\underline{\mathcal{F}}_3^*) = \mathcal{C}(\underline{\mathcal{F}}_2^*)$
5) $\mathcal{C}(\bar{\mathcal{F}}_3^*) = \mathcal{C}(\bar{\mathcal{F}}_2^*)$.

Fact 4.2: In all cases:

1) $|z_1| \leqslant |\hat{a}| \leqslant |x|$ (assuming no-cancellation in Step 5 of $\mathcal{S}_1$ in Tablę II)
2) $\mathcal{C}(\mathcal{S}_3^*) \geqslant \mathcal{C}(\bar{\mathcal{S}}_1^*)$
3) $\mathcal{C}(\mathcal{S}_3^{**}) \geqslant \mathcal{C}(\mathcal{S}_1^{**})$
4) If $\mathcal{C}(\mathcal{S}_1) < \mathcal{C}(\underline{\mathcal{S}}_1^*) \Leftrightarrow \mathcal{C}(\bar{\mathcal{S}}_1^*) < \mathcal{C}(\mathcal{S}_1^{**})$ and vice versa
5) If $\mathcal{C}(\mathcal{S}_1) < \mathcal{C}(\bar{\mathcal{S}}_1^*) \Leftrightarrow \mathcal{C}(\mathcal{S}_1^*\underline{)} < \mathcal{C}(\mathcal{S}_1^{**})$ and vice versa.

Fact 4.3: If all the rows of $\bar{V}$ are nonzero, then

1) z_2 is full
2) $\mathcal{C}(\mathcal{S}_2^*) \geqslant \mathcal{C}(\mathcal{S}_1^*)$
3) $\mathcal{C}(\mathcal{S}_2^{**}) \geqslant \mathcal{C}(\mathcal{S}_1^{**})$.

The condition that all the rows of $\bar{V}$ are nonzero is not uncommon; for example, if M has property $\mathcal{P}$ [18], the nonzero columns of $\bar{V}$ would be full.

Fact 4.4: Suppose y is full, then

1) a, b, $\bar{y}$, $\hat{a}$, $\hat{y}$, and x are also full
2) $\mathcal{C}(\mathcal{S}_1) = \sigma + |V| + |W|$
3) $\mathcal{C}(\underline{\mathcal{S}}_1^*) = \sigma + |V| + |Q| + \theta(Ub_\rho)$
4) $\mathcal{C}(\bar{\mathcal{S}}_1^*) = \sigma + |W| + |P| + \theta(Lz_1)$
5) $\mathcal{C}(\mathcal{S}_1^{**}) = \sigma + |P| + |\underline{Q}| + \theta(Ub_\rho) + \theta(Lz_1)$
6) $\mathcal{C}(\mathcal{S}_2) = \sigma + |Q| + |\bar{V}|$
7) $\mathcal{C}(\mathcal{S}_2^*) = \sigma + |Q| + |V| + \theta(Uz_2)$
8) $\mathcal{C}(\mathcal{S}_2^{**}) = \sigma + |P| + |\underline{Q}| + \theta(Lz_1) + \theta(Uz_2)$
9) $\mathcal{C}(\mathcal{S}_3) = \sigma + |P| + |\bar{W}|$
10) If the subnetwork matrix is structurally symmetric, then $\mathcal{C}(\mathcal{S}_3) = \mathcal{C}(\mathcal{S}_2)$.

Fact 4.5: If y is zero, then

1) $a = b = 0$; $\bar{y} = 0$; $\hat{a} = -z_1$; $x = -z_2$; $\hat{y} = -z_3$
2) $\mathcal{C}(\mathcal{S}_1) = |V| + \theta(Ux)$
3) $\mathcal{C}(\mathcal{S}_1^{**}) = |P| + \theta(Lz_1) + \theta(Ux)$
4) $\mathcal{C}(\mathcal{S}_2) = |\bar{V}|$
5) $\mathcal{C}(\underline{\mathcal{S}}_1^*) = \mathcal{C}(\mathcal{S}_2^*) = \mathcal{C}(\mathcal{S}_1)$
6) $\mathcal{C}(\bar{\mathcal{S}}_1^*) = \mathcal{C}(\mathcal{S}_3^*) = \mathcal{C}(\mathcal{S}_2^{**}) = \mathcal{C}(\mathcal{S}_1^{**})$.

V. The Interconnection Level

On the interconnection level the equation is formed according to (3) after the factorization of all the subnetwork equations has been completed. In BT the interconnection equation is of the following form [11]:

$$\begin{bmatrix} \mathbf{0} & G \\ H^T & \hat{Z}_t \end{bmatrix} \begin{bmatrix} v_o \\ i_t \end{bmatrix} = \begin{bmatrix} j \\ e \end{bmatrix} \tag{5}$$

where

$$\hat{Z}_t = -Z_t - \sum_{i=1}^{k} A_i^T M_i^{-1} A_i$$
$$= -Z_t - \sum_{i=1}^{k} F_i = -Z_t - F$$
$$G = A_t - \sum_{i=1}^{k} Y_{oi}^T M_i^{-1} A_i = A_t - \sum_{i=1}^{k} G_i$$
$$H^T = A_t^T - \sum_{i=1}^{k} A_i^T M_i^{-1} Y_{io} = A_t^T - \sum_{i=1}^{k} H_i^T$$

while in NT the interconnection equation has the following form:

$$\hat{Y}_c v_c = j_c \tag{6}$$

where

$$\hat{Y}_c = Y_c - \sum_{i=1}^{k} Y_{ci}^T M_i^{-1} Y_{ic}$$
$$= Y_c - \sum_{i=1}^{k} E_i = Y_c - E.$$

Note that the dimension of the interconnection equation in the case of BT is equal to the number of tearing branches plus the number of local-datum nodes, whereas in NT it is equal to the number of tearing nodes. This dimension will be large if the number of tearing branches or tearing nodes is large, therefore, any sparsity that might exist in the interconnection equation should be exploited. To be able to do this the structure of the interconnection equation should be determined *a priori* so that it could be coded at the outset for sparse matrix solution, before the subnetwork equations are solved.

Definition 5.1

Given a graph $\mathcal{G}$ with b branches and n nodes:

1) the *branch adjacency matrix* [9] of $\mathcal{G}$ is a $b \times b$ matrix whose ijth entry is nonzero if $i = j$ or if $i \neq j$ and branches i and j share a common node;

2) the *node adjacency matrix* of $\mathcal{G}$ is an $n \times n$ matrix whose ijth entry is nonzero if $i = j$ or if $i \neq j$ and nodes i and j are connected by a common branch.

Definition 5.2 [9]

A BT interconnection network N_b of a branch-torn network is a network formed by collapsing all torn sub-

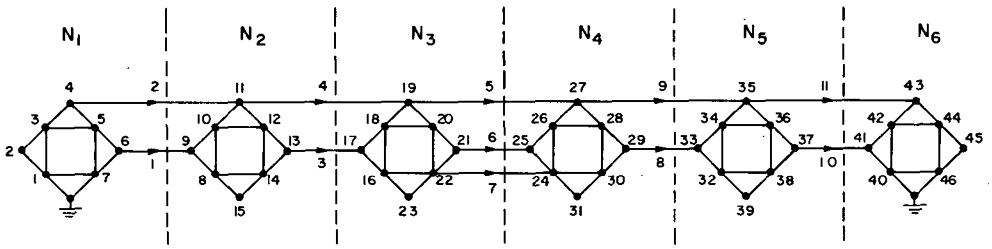


Fig. 2. Network N.

networks into a single node each, with the tearing branched becoming the branches of N_b.

Definition 5.3

A grounded BT interconnection network N_g of a branch-torn network is a network formed from N_b by connecting the terminals of those tearing branches incident with local-datum nodes to the ground rather than to the nodes formed by the collapse of the subnetworks.

Definition 5.4

An NT interconnection network N_n of a node-torn network is a network which is formed by removing all torn subnetworks and connecting any two tearing nodes which were connected by branches to the same subnetwork with a new branch, which in turn becomes a branch of N_n. Any branches connecting tearing nodes only also become branches of N_n.

Fact 5.1: In (5)

1) $\boldsymbol{F}$ has the structure of the branch adjacency matrix of N_g;

2) $\boldsymbol{G}$ and $\boldsymbol{H}$ have the same structure;

3) $\boldsymbol{G}$ has the structure of $\boldsymbol{A}_{nl}$, the incidence matrix of N_b with respect to n_l, where n_l is the set of nodes in N_b that was formed by the collapse of subnetworks having local-datum nodes.

Proof:

1) Consider $\boldsymbol{F}_i = \boldsymbol{A}_i^T \boldsymbol{M}_i^{-1} \boldsymbol{A}_i$; it is symmetric and its nonzero entries are located in positions corresponding to the nonzero columns of $\boldsymbol{A}_i$, which in turn correspond to the tearing branches incident with N_i at nodes other than the local-datum node. Since $\boldsymbol{F} = \sum_{i=1}^{k} \boldsymbol{F}_i$, 1) follows.

2) Consider $\boldsymbol{G}_i = \boldsymbol{Y}_{oi}^T \boldsymbol{M}_i^{-1} \boldsymbol{A}_i$; it has only one nonzero row, row i, the nonzero elements of which correspond to the nonzero columns of $\boldsymbol{A}_i$. Similarly, $\boldsymbol{H}_i^T = \boldsymbol{A}_i^T \boldsymbol{M}_i^{-1} \boldsymbol{Y}_{io}$ has only one nonzero column, column i, which has nonzero elements in positions corresponding to the nonzero columns of $\boldsymbol{A}_i$. Since $\boldsymbol{G} = \boldsymbol{A}_t - \sum_{i=1}^{k} \boldsymbol{G}_i$ and $\boldsymbol{H}^T = \boldsymbol{A}_t^T - \sum_{i=1}^{k} \boldsymbol{H}_i$, 2) follows.

3) $\boldsymbol{A}_t$ has nonzero columns in positions corresponding to the torn branches that are incident with local-datum nodes. Therefore, from 2) above, each row of $\boldsymbol{G}$ has nonzero components in positions corresponding to tearing branches incident with N_i at nodes including the local-datum node. Hence 3) follows.

Fact 5.2: Matrix $\boldsymbol{E}$ in (6) has the structure of the node adjacency matrix of N_n. The proof follows by applying the graph theoretic interpretation of Gaussian elimination [20].

Knowing the structure of $\boldsymbol{Z}_t$ or of $\boldsymbol{Y}_c$ and using the above facts, it is then possible to determine the structures of $\hat{\boldsymbol{Z}}_t$, $\boldsymbol{G}$, and $\boldsymbol{H}$ in (5) and of $\hat{\boldsymbol{Y}}_c$ in (6) without the necessity of knowing the internal structures of the torn subnetworks. Note that $\hat{\boldsymbol{Y}}_c$ in NT does not generally contain zeros on its diagonal, while in BT the matrix of the interconnection equation may contain zeros on part of its diagonal, as shown in (5), where the diagonal entries corresponding to the local-datum nodes are zeros. However, if the entire network has a solution, the interconnection matrix will be nonsingular. If the equation is reordered, then the zero diagonal elements will be filled at a certain stage in the factorization process. In [11] a method is proposed for solving the interconnection equation which is essentially equivalent to ordering the local-datum node voltages last and thus ensuring that the zero diagonal entries will be filled. Forcing some variables to be ordered last, however, may affect the sparsity of the entire factorized system of equations. An alternative approach which might save on computation is the inclusion of the local-datum node voltages as candidates for pivot selection as soon as their corresponding zero entries are filled. This ordering approach is illustrated in the example in the following section.

VI. Example

An example is given in this section to illustrate the differences in the various factorization and substitution procedures discussed in the previous sections. Consider the network shown in Fig. 2, which is torn into six subnetworks first by BT and then by NT. For the sake of simplicity it is assumed that no coupling exists among the branches and that nodal analysis can be applied at the subnetwork levels. Fig. 3(a) and (b) show the structures of the partitioned matrices of the entire network equations when BT and BT, respectively, are used. Note that in subnetwork N_2 in Fig. 3(a) column i_2 in the border submatrix is the negative of column i_4 because tearing branches 2 and 4 are incident with the same node in N_2 but directed opposite to each other with respect to the node. The same holds for N_5 where column i_{10} is the negative of column i_{11}.

In order to illustrate how to compare the computational requirements of the various algorithms, we choose the BT case and consider the solution of subnetwork N_3, which is

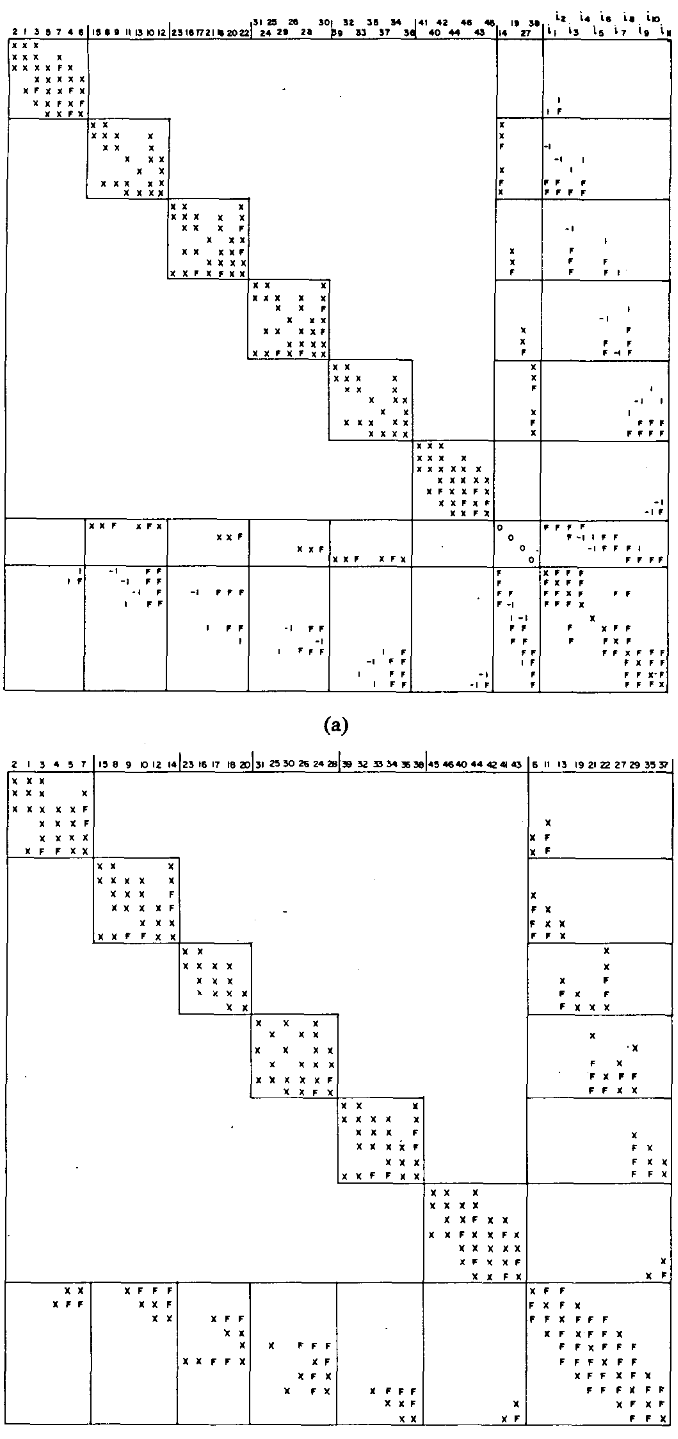


Fig. 3. Structures of the partitioned matrices of the entire network equations. (a) BT case. (b) NT case. F denotes a fill-in.

shown in Fig. 4(a). The structure of the subnetwork matrix is shown in Fig. 4(b), where node 19 is chosen as the local-datum node. The subnetwork equations are ordered so as to minimize the fillins in the entire subnetwork equations, not only in LU but also in V and W [15] and at the same time to try to increase as much as possible the number of zero leading rows in V (or in W) [22]. This approach minimizes $\sigma+\theta(LV)+\theta(U^TW)$ and favors $\mathfrak{F}_1$ and at the same time increasing the number of leading rows of V (or of W) favors $\bar{\mathfrak{F}}_2$ (or $\bar{\mathfrak{F}}_3$). Table VI shows the number of operations required to factorize the subnetwork equation while Table VII gives the number of operations required in the substitution procedures. Note that $\mathfrak{F}_3$ is not considered since the subnetwork matrix is structurally symmetric. In practice, in order to determine which algorithm is most efficient, it is not necessary to calculate the total number of operations required by each. First of all, it can be easily proven that all the nonzero columns of $\bar{V}$ are full. From Fact 4.3 we can then eliminate $\mathcal{S}_2^*$ and $\mathcal{S}_2^{**}$. If we consider y to be full, then from Fact 4.4 $\mathcal{C}(\mathcal{S}_3)=\mathcal{C}(\mathcal{S}_2)$. That leaves $\mathcal{S}_1$, $\mathcal{S}_1^*$, $\bar{\mathcal{S}}_1^*$, and $\mathcal{S}_1^{**}$, and $\mathcal{S}_2$ as candidates for the substitution procedure. Using Fact 4.4, we find $|Q|+\theta(Ub_\rho)<|W|<|P|+\theta(Lz_1)$; therefore, $\mathcal{C}(\mathcal{S}_1^*)<\mathcal{C}(\mathcal{S}_1)$, $\mathcal{C}(\mathcal{S}_1^{**})<\mathcal{C}(\bar{\mathcal{S}}_1^*)$, and $\mathcal{C}(\mathcal{S}_1^*)<\mathcal{C}(\mathcal{S}_1^{**})$. Comparing $\mathcal{S}_1^*$ to $\mathcal{S}_2$, we find $|V|+\theta(Ub_\rho)<|\bar{V}|$, therefore, $\mathcal{S}_1^*$ requires the least number of operations. The corresponding factorization procedures to be considered with $\mathcal{S}_1^*$ would be $\mathfrak{F}_1$ and $\bar{\mathfrak{F}}_2$ or $\mathfrak{F}_1^{\delta}$ and $\mathfrak{F}_2^{\delta}$. Using Table

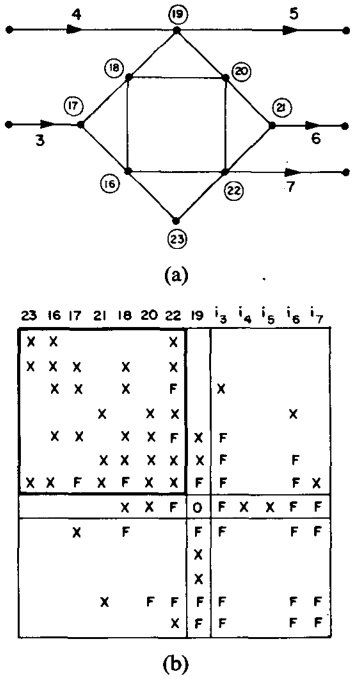


Fig. 4. (a) Subnetwork N_3. (b) Structure of the subnetwork matrix.

TABLE VI
NUMBER OF OPERATIONS REQUIRED IN FACTORIZING SUBNETWORK N_3 EQUATIONS USING VARIOUS FACTORIZATION PROCEDURES

Procedure	F_1	F_2	$\bar{F}_2$	F_1^{δ}	F_2^{δ}	$\bar{F}_2^{\delta}$
No. of Operations	99	114	94	71	107	87

TABLE VII
NUMBER OF OPERATIONS REQUIRED FOR SOLVING SUBNETWORK N_3 EQUATIONS USING VARIOUS SUBSTITUTION PROCEDURES

Procedure		S_1	S_1^*	$\bar{S}_1^*$	S_1^{**}	S_2	S_2^*	S_2^{**}	S_3
Number of	y full	53	51	56	54	61	56	59	61
Operations	$y = 0$	23	23	26	26	28	23	26	26

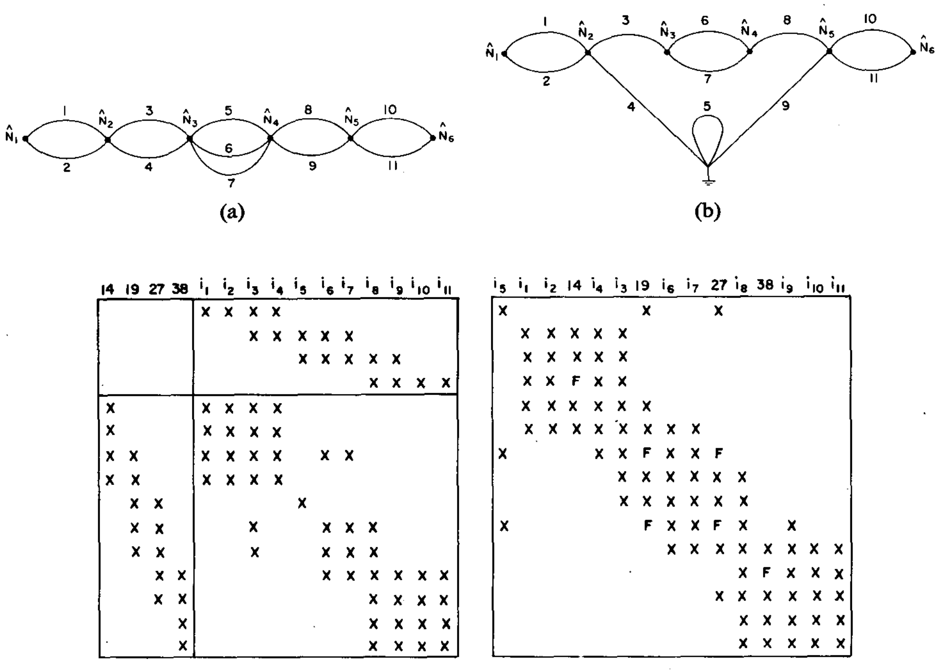

(a)
(b)
(c)
(d)
14 19 27 38 i_1 i_2 i_3 i_4 i_5 i_6 i_7 i_8 i_9 i_{10} i_{11}
i_5 i_1 i_2 14 i_4 i_3 19 i_6 i_7 27 i_8 38 i_9 i_{10} i_{11}

Fig. 5. BT interconnection networks. (a) N_b. (b) N_g; structure of the BT interconnection equation matrix. (c) Before reordering. (d) After reordering.

TABLE VIII
NUMBER OF OPERATIONS REQUIRED BY MOST EFFICIENT ALGORITHM WHEN SOLVING NETWORK EXAMPLE BY BT AND NT

Subnetwork		Procedures		Number of Operations	
		Nonsymm	Symm	Nonsymm	Symm
Branch-Tearing	N_1, N_6	$\bar F_2 + S_1^{**}$	$\bar F_2^\delta + S_1^{**}$	87	78
	N_2, N_5	$F_1 + S_1^*$	$F_1^\delta + S_1^*$	137	112
	N_3, N_4	$\bar F_2 + S_1^*$	$F_1^\delta + S_1^*$	145	122
	Inter. Net	$F_1 + S_1$	$F_1^\delta + S_1$	225	191
	Total			963	815
Node-Tearing	N_1	$F_1 + S_1$	$F_1^\delta + S_1$	90	78
	N_2, N_5	$\bar F_2 + S_1^*$	$F_1^\delta + S_1$	125	104
	N_3	$F_1 + S_1$	$F_1^\delta + S_1$	107	88
	N_4	$F_1 + S_1$	$F_1^\delta + S_1$	148	119
	N_6	$F_1 + S_1$	$F_1^\delta + S_1$	93	82
	Inter. Net	$F_1 + S_1$	$F_1^\delta + S_1$	118	104
	Total			806	679

IV we find that $\theta(U\bar{V}_\rho)+\theta(Q^T\bar{V}_\rho)<\theta(U^TW)+\theta(W^TV)$ and $\theta^s(V^TV)<\theta(U\bar{V}_\rho)+\theta^s(Q^T\bar{V}_\rho)$. Therefore, one could use $\mathcal{F}_2$ for the numerically unsymmetric case and $\mathcal{F}_1^*$ for the symmetric one.

Table VIII shows the number of operations required by the most efficient algorithms for solving each of the subnetworks and the interconnection equation when using BT or NT; y is assumed to be full. In BT the interconnection networks N_b and N_g are shown in Fig. 5(a) and (b), respectively. The local-datum nodes chosen are $\{14,19,27,38\}$. The structure of the corresponding interconnection equation matrix before and after reordering is shown in Fig. 5(c) and (d), respectively. In the case of NT the interconnection network is shown in Fig. 6, where nodes $\{6,11,13,19,21,22,27,29,35,37\}$ are chosen as the set of tearing nodes. The structure of the corresponding interconnection equation matrix is given as the lower right-hand submatrix of the matrix shown in Fig. 3(b).

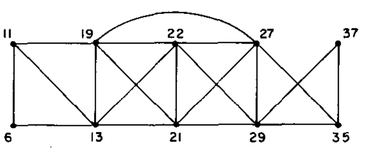


Fig. 6. NT interconnection network N_n.

Note that in the NT case the number of operations required to solve the entire network is less than that required by BT. However, there are cases where BT is more desirable. In the above example, for instance, if by removing the tearing branches the resulting torn subnetworks were identical, then the total amount of computation could be considerably reduced, while in the same example NT would not result in identical subnetworks.

VIII. DISCUSSION

In this paper we have shown that all recently proposed sparse matrix solution algorithms for network solution by tearing belong to a set of algorithms which have been derived by applying block matrix elimination to a partitioned system of network equations. These algorithms have previously been studied [18], [20]. In this paper we extended them to include some new variations, and then applied the algorithms to network solution by tearing in cases where the network matrix was assumed to have a bordered-block-diagonal structure; i.e., where the torn subnetworks are assumed not to have any coupling among each other. When this is the case parallel processing can be applied to the solution. As well, the algorithms apply to other partitioned systems of equations such as bordered-block-triangular systems; in this case, however, parallel processing is not as easily applicable.

By closely considering the structure of the partitioned network, sparsity can be exploited at both the subnetwork and the interconnection levels in the solution process. On the subnetwork level, provided there is no coupling between the subnetwork and the remainder of the tearing set, the number of nonzero columns in the border submatrix is equal to the subset of the tearing set incident with or connected to the subnetwork. The computational requirements of the algorithms are derived in such a way that a comparison between the computational efficiencies of the algorithms can easily be obtained if the same equation order is used in all the algorithms. This leads to the conclusion that it is always possible to find an algorithm which requires computation less than or equal to step-by-step solution without tearing, provided the same equation order is used. Of course, a different partitioning or blocking of the network equations or a different equation order may require more or less computation. On the subnetwork level the problem of selecting from all possible ordering strategies the one which requires the minimum number of operations needs further investigation.

On the interconnection level the structure of the interconnection equation was related to the branch and node adjacency matrices of the interconnection network which was defined for both BT and NT. Knowing the structure of the interconnection equation at the outset enables one to use sparse matrix techniques in its solution, especially when this equation is a large one.

In this paper we considered the tearing set to have been selected *a priori* according to certain criteria, such as creating identical subnetworks, and/or preserving the identities of some subnetworks. In general, the problem may be the selection of a tearing set which will result in the most efficient tearing algorithm. An algorithm suitable for NT has been proposed in [24]. A similar algorithm which is suitable for BT has not, to the author's knowledge, yet been published.

Acknowledgment

The author would like to thank the reviewers for their constructive comments and suggestions.

References

[1] G. Kron, *Diakoptics: The Piecewise Solution of Large Scale Systems*. London, England: MacDonald, 1963.

[2] H. H. Happ, *Diakoptics and Networks*. New York: Academic, 1971.

[3] F. H. Branin, "Computer methods of network analysis," *Proc. IEEE*, vol. 55, pp. 1787–1801, Nov. 1967.

[4] D. M. Himmelblau, ed. *Decomposition of Large Scale Problems*. Amsterdam, The Netherlands: North Holland, 1973.

[5] M. V. Bhat and H. K. Kesavan, "Diakoptic equations and sparsity," presented at *IEEE PES Summer Meeting*, (Anaheim, CA), July 1974.

[6] F. F. Wu, "Diakoptics and triangular factorization," presented at 1975 *Winter Power Meeting*.

[7] ——, "Diakoptic network analysis," in *IEEE* 1975 *Power Industry Computer Applications* (*PICA*), (New Orleans, LA), pp. 364–371, June 1975.

[8] F. H. Branin, "A sparse matrix modification of Kron's method of piecewise analysis," in *IEEE Proc. ISCAS*, (Boston, MA), pp. 383–386, Apr. 1975.

[9] F. L. Alvarado, D. K. Reitan, and M. Bahari-Kashani, "Sparsity in diakoptic algorithms," presented at *IEEE PES Summer Meeting*, (Portland, OR), July 1976.

[10] L. O. Chua and L. K. Chen, "Diakoptic and generalized hybrid analysis," *IEEE Trans. Circuits Syst.*, vol. CAS-23, pp. 694–705, Dec. 1976.

[11] F. F. Wu, "Solution of large-scale networks by tearing," *IEEE Trans. Circuits Syst.*, vol. CAS-23, pp. 706–713, Dec. 1976.

[12] N. B. Rabbat and H. V. Hsieh, "A latent macromodular approach to large-scale sparse networks," *IEEE Trans. Circuits Syst.*, vol. CAS-23, pp. 745–752, Dec. 1976.

[13] G. Guardabassi and A. Sangiovanni-Vincentelli, "A two levels algorithm for tearing," *IEEE Trans. Circuits Syst.*, vol. CAS-23, pp. 783–791, Dec. 1976.

[14] A. Sangiovanni-Vincentelli, L. K. Chen, and L. O. Chua, "A new tearing approach—node-tearing nodal analysis," in 1977 *IEEE Int. Symp. on Circuits and Systems*, (Phoenix, AZ), pp. 143–147, Apr. 1977.

[15] ——, "Node-tearing nodal analysis," Electronics Res. Lab., Univ. of California, Memo no. ERL-M582, Sept. 1976.

[16] K. U. Wang and T. Chao, "Diakoptics for large-scale nonlinear time-varying networks," in *IEEE Proc. ISCAS*, (Boston, MA), pp. 277–278, Apr. 1975.

[17] M. Chandrashekar and H. K. Kesavan, "Graph-theoretic state models for the piecewise analysis of large-scale electrical networks," *Circuit Theory and Appl.*, vol. 5, pp. 23–34, Jan. 1977.

[18] A. George, "On block elimination for sparse linear systems," *SIAM J. Numer. Anal.*, vol. 11, no. 3, pp. 585–603, June 1974.

[19] C. -W. Ho, A. E. Ruehli, and P. A. Brennan, "The modified nodal approach to network analysis," *IEEE Trans. Circuits Syst.*, vol. CAS-22, pp. 504–509, June 1975.

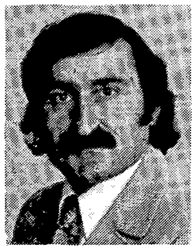
[20] J. R. Bunch and D. J. Rose, "Partitioning, tearing and modification of sparse linear systems," Harvard Univ. Rep. #27-72, Nov. 1972.

[21] I. N. Hajj, "Solution of interconnected subsystems," *Electron. Lett.*, vol. 13, no. 3, pp. 78–79, Feb. 1977.

[22] ——, "Diakoptics and block elimination," in *IEEE PES Summer Meeting*, (Los Angeles, CA), July 1978.

[23] K. Gopal, K. Singhal, and J. Vlach, "A method for solution sparse systems," presented at *Midwest Symp. on Circuit Theory*, (Milwaukee, WI), Aug. 1976.

[24] A. Singiovanni-Vincentelli, L. K. Chen, and L. O. Chua, "An efficient heuristic cluster algorithm for tearing large-scale networks," *IEEE Trans. Circuits Syst.*, vol. CAS-24, pp. 709–717, Dec. 1977.



Ibrahim N. Hajj was born in Lebanon. He received the B.E. degree from the American University of Beirut in 1964, the M.S. degree from the University of New Mexico, Albuquerque, in 1966, and the Ph.D. degree from the University of California, Berkeley, in 1970, all in electrical engineering.

From 1970 to 1973 and from 1975 to 1978 he was with the Department of Electrical Engineering at the University of Waterloo, Waterloo, Ont., Canada. From 1973 to 1975 he taught at the Lebanese University in Beirut, Lebanon, and at the same time worked as a consultant with engineering consulting firms operating in the Middle East. In December 1978, he joined the Coordinated Science Laboratory and the Department of Electrical Engineering, University of Illinois, Urbana, as an Assistant Professor. His present research interests include nonlinear network theory, modeling techniques, numerical methods for computer-aided simulation, and analysis of large-scale systems and circuits.