HIERARCHICAL DECOMPOSITION APPROACH TO D.C. POWER FLOW SOLUTION

J.A. Starzyk

Electrical & Computer Eng. Dept.
Ohio University
Athens, OH 45701

S.C. Rastogi*

Corporate Planning
Western Farmers Electric Cooperative
Anadarko, OK 73005

ABSTRACT

This paper discusses an algorithm to solve the d.c. power flow problem. In this method large power systems are hierarchically decomposed into small manageable subnetworks (blocks) using node decomposition. These blocks are solved separately and are interconnected to obtain the solution for the subnetworks on the higher level of hierarchy. Finally the solution for the original network is obtained after all hierarchy has been worked through.

I. INTRODUCTION

Power flow solutions of a large system have always been limited because of its memory and computational requirements. To achieve load flow solution with limited computational resources, following aspect of the problem were considered:

(1) Sparsity of the admittance matrix of the power system.
(2) Decomposition (tearing) of power systems network in smaller subnetworks (blocks) to suit computational requirement.
(3) Utilization of local effects under contingency condition.

Sparsity of the admittance matrix of power system network was exploited by storing nonzero elements only and thus avoiding zeros in mathematical operations [1]. Later network tearing [decomposition] led to simplicity, but simultaneous analysis of decomposed subnetwork was not possible [2]. A heirarchical decomposition approach discussed in [3] could analyze large power system network effectively by solving decomposed subnetworks simultaneously. This approach would further be viable when contingencies are simulated during operational and planning studies of the power system.

In this paper heirarchical decomposition approach has been illustrated to solve d.c. load flow problem and further extended to solve contingency load flow problem. This extention uses the fact (3) mentioned above and therefore only locally affected subnetworks are solved and solutions of the other subnetworks are retained from the base case - resulting in computional savings.

In the following sections of the paper, firstly brief description of the hierarchical decomposition of network is discussed. Then load flow equations and their solution is described. Later test result of the algorithm is discussed.

*Formerly with the Electrical and Computer Engineering Department, Ohio University, Athens, Ohio 45701

II. HIERARCHICAL DECOMPOSITION

Matrix decomposition is based on its Coats signal flow graph representation [4] in which a square matrix $A=[a_{ij}]$ is represented by a graph with n nodes and k edges, where k is the number of nonzero coefficients in A. In a Coates graph edge directed from node x_j to node x_i has a weight equal to a_{ij}. Coats graph of a coefficient matrix

$$A = \begin{bmatrix} 1 & 2 & 0 \\ 3 & 4 & 5 \\ 6 & 0 & 7 \end{bmatrix}$$

is shown in Fig. 1.

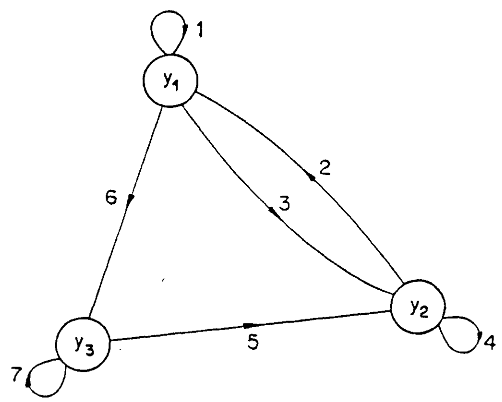

Fig. 1 Coates graph of the matrix A of Example 1.

A signal flow graph is decomposed through its nodes into two subgraphs (subnetworks). Each of these subgraphs can be decomposed further down to a sufficiently small size. This kind of graph decomposition is called hierarchical decomposition. The structure of hierarchical decomposition can be illustrated by a decomposition tree. Nodes of the tree correspond to subgraphs obtained on different levels of decomposition. If a subgraph G_j was obtained during decomposition of subgraph G_i then there is an edge from the node corresponding to G_i to the node corresponding to G_j. Fig. 3 shows the decomposition tree corresponding to Fig. 2.

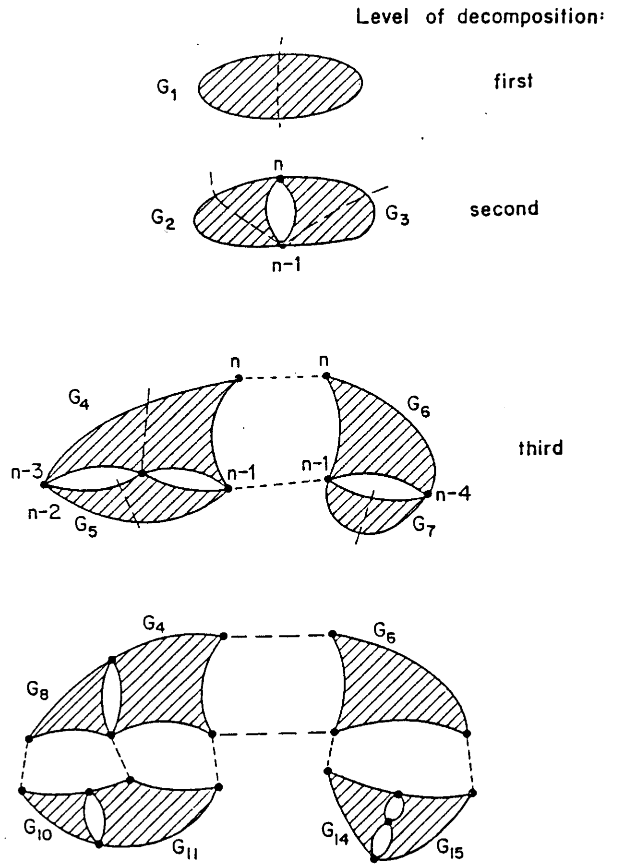

Fig. 2 Three level hierarchical decomposition.

In the decomposition tree we have one initial node - the one which is the starting point of edges. Terminal nodes are those which are only the end points of edges. All nodes that are not terminal nodes are middle nodes. Subgraphs associated with terminal nodes are called proper blocks. We limit ourselves to bisection as the only graph partition so that every middle node has exactly two descendants. If m is the index of a middle subgraph then two of its descendants have indices 2m and 2m+1, respectively. This way of numbering the graphs makes the analysis of interconnections easier.

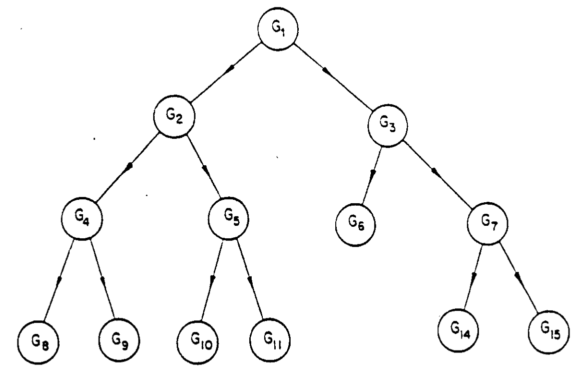

Fig. 3 Decomposition tree for Fig. 2.

Matrix Reordering: After all subgraphs have been numbered according to the structure of the decomposition tree, the nodes of a graph are renumbered consecutively in descending order starting from the partition nodes of the graph G_1, then the partition nodes of graph G_2 and G_3 up to the last partition and then the internal nodes of the proper blocks. After the renumbering, the numbers associated with the graph nodes are called original indices of the nodes. Such renumbering corresponds to reordering the coefficient matrix. An example for the matrix partitioned according to Fig. 2 is shown in Fig. 4. There is a strict correspondence between the set of partition nodes of a middle graph and submatrices of the reordered coefficient matrix. These submatrices are called interconnection matrices as they represent interconnection of two subsystems.

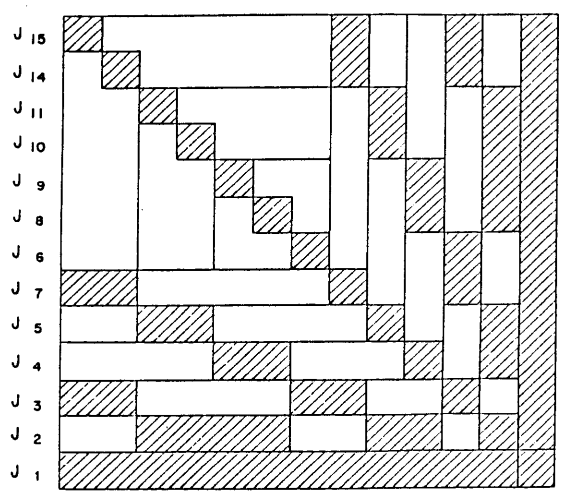

Fig. 4 Nonzero pattern of a reordered coefficient matrix.

III. POWER FLOW EQUATIONS

The power flow equations of a power system under steady state condition are obtained by equating the power injected into each node by source or load and power transmitted from the node via transmission network. Thus at ith node,

$$P_i = P_{Gi}+P_{Li}+P_{Ti}=V_i(G_{ik}\cos\theta_{ik}+B_{ik}\sin\theta ik) \qquad (1.a)$$

$$Q_i = Q_{Gi}+Q_{Li}+Q_{Ti}=V_i(G_{ik}\sin\theta_{ik}-B_{ik}\cos\theta_{ik}) \qquad (1.b)$$

where

P_{Gi}, P_{Li}, P_{Ti}	real powers injected into the ith node by the generator, load and ties connected to it,
P_i	net real power injected into the ith node,
Q_{Gi}, Q_{Li}, Q_{Ti}	reactive powers injected into the ith node by the generator, load and ties connected to it,
Q_i	net reactive power injected into the ith node,
$V_i\angle\theta i$	complex voltage at the ith node,
G_{ik}, B_{ik}	real and imaginary parts of Y_{ik}, the elements in ith row & jth column of the network admittance matrix,
N	number of nodes in the network,
θ_{ik}	$= \theta_i - \theta_k$.

It is interesting to note that for the power systems, G_{ik} are small, θ_{ik} rarely exceeds 20^0 and the node/bus voltages rarely deviate by more than 10% from their nominal values.

If we make approximations $G_{ik}\simeq 0$, $\sin O_{ik}\simeq\theta_{ik}$, and $\cos\theta i_k \simeq 1$ equations (1.b) reduces to the decoupled reactive power flow equations

$$Q_i = -\sum_{k=1}^{N} B_{ik}\, V_i\, V_k$$

Further, in normal operation the per unit voltage magnitudes V_i are close to 1. Therefore, approximating $V_i = 1$, equation (1.a) becomes the <u>d.c. load flow equation</u>.

$$P_i = \sum_{k=1}^{N} B_{ik}\,(\theta_i - \theta_k)$$

If the resistance is much smaller than the reactance then

$$P_i = \sum_{k=1}^{N} 1/X_{ik} * (\theta_i - \theta_k)$$

Writing this in matrix form,

$$\begin{bmatrix} P_1 \\ P_2 \\ \cdot \\ \cdot \\ P_n \end{bmatrix} = \begin{bmatrix} 1/X_{ik} \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \cdot \\ \cdot \\ \theta_n \end{bmatrix} \qquad (2)$$

This equation can be solved for θ and the resulting power flow on the line can be obtained by

$$P_{ik} = 1/X_{ik} * (\theta_i - \theta_k) \qquad (3)$$

The power flow equations are very sparse and mildly nonlinear. Using Newton's method and treating node voltage magnitude and angle as unknown, each iteration invloves solving a linearized version of (1), namely.

$$\begin{bmatrix} H & N \\ J & L \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta V/V \end{bmatrix} = \begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} \qquad (4)$$

Where H, N, J, and L constitute the Jacobian of (1), $\Delta\theta$ and ΔV are vectors of corrections to the estimates for node voltages and magnitudes and ΔP, ΔQ are the mismatches between the respective powers injected into and removed from the nodes. In practice N and J are much smaller than H and L. Therefore (4) may be approximated by,

$$\begin{bmatrix} \Delta P/V \end{bmatrix} = \begin{bmatrix} B' \end{bmatrix}\begin{bmatrix} \Delta\theta \end{bmatrix} \qquad (5)$$

and

$$\begin{bmatrix} \Delta Q/V \end{bmatrix} = \begin{bmatrix} B'' \end{bmatrix}\begin{bmatrix} \Delta V \end{bmatrix} \qquad (6)$$

Where B' and B" are constant matrices. The solution of a.c. load flow equations is obtained by iterating (5) and (6) until mismatch satisfy specified tolerance by the user. This method is called decoupled load flow method.

IV. SOLUTION OF N LINEAR EQUATIONS

The set of n linear equations,

$$A\,Y = b \qquad (7)$$

can be solved effectively, if A is a sparse matrix, by using bifactorization method [5]. The inverse of A can be expressed by a multiple product of 2n factor matrices.

$$A^{-1} = R^{(1)}R^{(2)}..R^{(n)}L^{(n)}..L^{(2)}L^{(1)} \qquad (8)$$

In order to find L and R the following sequence of intermediate matrices is introduced:

$$A^{(0)} = A$$

$$A^{(1)} = L^{(1)}A^{(0)}R^{(1)}$$

$$\cdot \qquad \cdot$$

$$A^{(j)} = L^{(j)}A^{(j-1)}R^{(j)}$$

$$\cdot \qquad \cdot$$

$$A^{(n)} = L^{(n)}A^{(n-1)}R^{(n)} = I$$

Where reduced matrix $A^{(j)}$ has elements defined by the following equations:

$$a^{(j)}_{jj} = 1\ ; \quad a^{(j)}_{ij} = a^{(j)}_{jk} = 0$$

$$a^{(j)}_{ik} = a^{(j-1)}_{ik} - \frac{a^{(j-1)}_{ij} \cdot a^{(j-1)}_{jk}}{a^{(j-1)}_{jj}}$$

j being the pivotal index and for i, k = (j+1),......,n.

The left-hand factor matrices $L^{(j)}$ are very sparse and differ from the unity matrix in column j only.

$$L^{(j)} = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & l^{(j)}_{j,j} & & \\ & & \cdot & 1 & \\ & & l^{(j)}_{n,j} & & 1 \end{bmatrix}$$

Where $l^{(j)}_{jj} = 1/\,a^{(j-1)}_{jj}$

and $l^{(j)}_{ij} = -\,a^{(j-1)}_{ij}/a^{(j-1)}_{jj}$,

$i = (j+1),\ldots,n$

The right hand factor matrices $R^{(j)}$ are also very sparse and differ from the unity matrix in row j only.

$$R^{(j)} = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & \ddots & & & \\ & & & 1 & r^{(j)}_{j,j+1} \cdot & r^{(j)}_{j,n} \\ & & & & 1 & \vdots \\ & & & & & 1 \end{bmatrix}$$

Where $r^{(j)}_{jk} = - a^{(j-1)}_{jk}/a^{(j-1)}_{jj}$ $k = (j+1), \ldots, n$

For symmetric matrix A, structures of left (L) and right (R) factor matrices are the same and can therefore be stored effectively in the matrix F as follows,

$$F = \begin{bmatrix} \diagdown & R \\ L & \diagdown \end{bmatrix}$$

Solution to equation (7) can be written as

$$y = R^{(1)} R^{(2)} \ldots R^{(n)} L^{(n)} \ldots L^{(2)} L^{(1)} b \qquad (9)$$

Let

$$z = \begin{bmatrix} z_1 \\ z_2 \\ \cdot \\ \cdot \\ z_n \end{bmatrix} = L^{(n)} \ldots L^{(2)} L^{(1)} b \qquad (10)$$

Where, individual values of $z_1, z_2, \ldots z_n$, can be stored. If the coefficients in only one proper block have been altered (for example in G_{14} of Fig.3) then according to above formula and the nonzero pattern of Fig. 4, we have to calculate new values of z_{14}, z_7, z_3 and z_1, to obtain the new value of z. From equations (9) and (10) we have,

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ y_n \end{bmatrix} = R^{(1)} R^{(2)} \ldots R^{(n)} z$$

If we are interested in the solution for any submatrix, then only those elements of y which corresponds to submatrix specified are to be calculated.

V. HIERARCHICAL DECOMPOSITION ALGORITHM

Following are the steps to solve the power flow equation.

1. Decompose the power system network hierarchically.
2. Renumber the nodes.
3. Form the $[1/X_{ik}]$ matrix for d.c. solution or B' and B" for a.c. solution.
4. Solve equation (2) for d.c. solution, or equations (5) and (6) for a.c. solution using method described in Section IV.
5. Obtain d.c. power flow by using equation (3) and for a.c. solution go to step 6.
6. Check power mismatch. If within specified tolerance limit then determine power flow else update equations (5) and (6) and go to step 4.

VI. TEST EXAMPLE

The data below is a listing of input data to the six bus sample power system used to test the method discussed in the paper.

NO. OF BUSES = 6 NO. OF LINES 11
NO. OF GENS. = 3 SWING BUS NO. 1

LINE DATA

FROM	TO	R	X	BCAP
1	2	0.1000	0.2000	0.0200
1	4	0.0500	0.2000	0.0200
1	5	0.0000	0.3000	0.0300
2	3	0.5000	0.1500	0.0300
2	4	0.0500	0.1000	0.0100
2	5	0.1000	0.3000	0.0200
2	6	0.0700	0.2000	0.0250
3	5	0.1200	0.2600	0.0250
3	6	0.0200	0.1000	0.0100
4	5	0.2000	0.4000	0.0400
5	6	0.1000	0.3000	0.0300

BUS DATA

BUS NO.	GEN PU-MW	VOLTAGE	P LOAD	Q LOAD
1	0.00	1.050	0.00	0.00
2	1.00	1.050	0.00	0.00
3	1.00	1.070	0.00	0.00
4	0.00	1.000	1.00	1.00
5	0.00	1.000	1.00	1.00
6	0.00	1.000	1.00	1.00

The system description and test results of the above example are shown in Fig. 5. Power flows

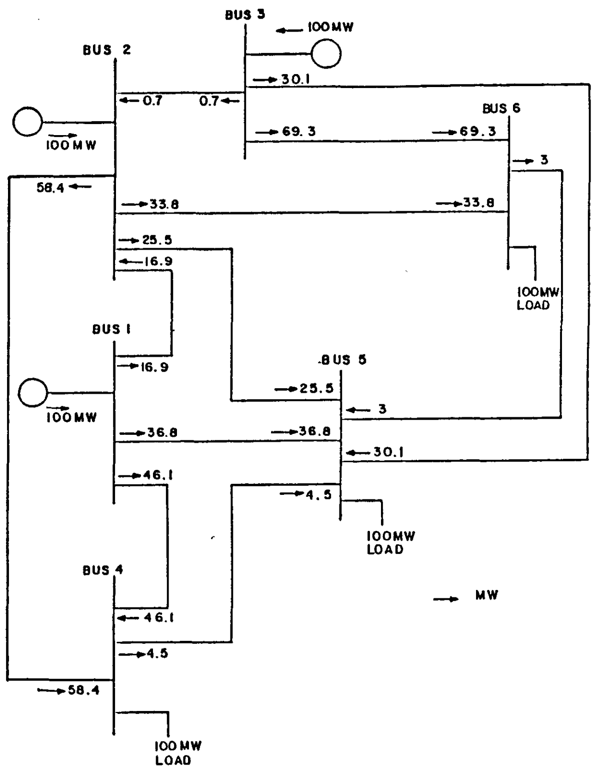

Fig. 5 six bus network base case d.c. load flow thus obtained are satisfactory and encouraging. Although CPU time for the present example was not recorded and therefore, can not be compared, however, it is expected that for larger networks and for contingency cases evaluation, the presented algorithm will show substantial computational savings.

VII. CONCLUSION

A hierarchical decomposition approach for solving a d.c. load flow for a large power system has been presented. First, analysis of proper blocks is performed and then subnetworks are combined in a hierarchical manner joining two subnetworks at any time. Finally, solution for the original network is obtained after all hierarchy has been worked through.

The algorithm discussed was tested on several moderate size power systems. The results obtained were encouraging, showing feasibility of this approach in power system analysis. Similar approach can be developed for a.c. power flow solution as well.

Such an algorithm is suitable for large power systems, since computational time can be reduced by solving the subnetworks in parallel. The most important application of this approach is in evaluation of contingency cases required in planning and operational studies of the power system. Since, during a contingency there is a change in coefficients of a proper block and complete analysis of the whole system is not required and thus is gained further savings in computational effort.

VIII. REFERENCES

1. N. Sato and W. F. Tinney, "Techniques for Exploiting the Sparsity of Network Admittance Matrix", IEEE Trans. Power Apparatus and Systems, Volume 82, pp. 944-950, 1963.

2. H. H. Happ, "Multi-level Tearing and Applications", IEEE Trans. Power Apparatus and Systems, Volume 92, pp. 725-733, 1973.

3. H. Gupta, J. W. Bandler, J.A. Starzyk, J. Sharma, "A Hierarchical Decomposition Approach For Network Analysis", Proc. ISCAS. Rome, Italy, May 1982.

4. W. K. Chen, Applied Graph Theory - Graphs and Electrical Networks, North-Holland, London, 1975.

5. K. Zollenkopf, "Bifactorization - Basic Computational Algorithm and Programming Techniques", in Large Sparse Sets of Linear Equations. J. K. Reid, Editor, Academic Press - New York, pp. 75-96, 1971.

6. J. A. Starzyk and J. W. Bandler, CSDSLE - A Fortran Package for the Solutions of Sparse Decomposed Systems of Linear Equations, Report Electrical and Computer Engineering, McMaster University, Hamilton, Canada, 1983.