A PARALLEL PRECONDITIONED CONJUGATE GRADIENT ALGORITHM FOR FAST DECOUPLED LOAD FLOW

M. El Gammal
Department of Electrical Engineering,
Alexandria University, Alexandria, Egypt.

M. El Attar
Department of Engineering Mathematics,
Alexandria University, Alexandria, Egypt.

ABSTRACT

This paper presents a new parallel algorithm for the fast decoupled load flow(FDLF) problems. The algorithm is based on the parallel application of the iterative Preconditioned Conjugate Gradient(PCG) solver. The decomposition of the power system into subsytems is simple(no optimal numbering of buses is required). This is due to the fact that only non-zero elements are stored since the (PCG) method doesn't alter the zeros (within the band of the matrix) during solution. Thus no special techniques are required in order to reduce the band of the system matrix. The proposed method is such that only matrix problems within subsytems are solved. A numerical test example is given in which we have simulated a parallel architecture in order to examine the method in the solution of the power flow problem for the Unified Power System(UPS) of Egypt. The results indicate that the proposed method is powerfull and can be applied to larger systems.

INTRODUCTION

Load flow is the solution for the normal balanced three phase steady state operating conditions of an electric power system. The data obtained from load flow analysis are used for the studies of normal operating mode, contingency analysis, outage security assessment, as well as optimal dispatching and stability. The early digital methods for solving the load flow problems were based on the Gauss-Seidel method, however the method was slowly converging and fails to converge in ill-conditioned situations. In [1], Van Ness and Griffin has introduced a method based on the Newton-Raphson iterations. Since system planning studies and system operations may require multiple-case load flow solutions in some situations, the recent research efforts have been concentrated on the development of the decoupled Newton-Raphson methods. These methods are based on the fact that in any power transmission network operating in the steady state, the coupling between, the active powers P and bus voltage angles θ ,and, the reactive powers Q and bus voltage magnitudes V, is relatively weak. In contrary, there is a strong coupling between P and θ and between Q and V, therefore these methods solve the load flow problems by decoupling the P-θ and Q-V problems. The (FDLF) method [2-4], has gained considerable popularity in recent years for obtaining the power flow solutions both in the system planning and operation stages. It is characterized by a decoupling of the linearized load flow equations with constant load flow matrices. The basic assumptions are: first, the voltages are around their nominal values, second, the angle differences across the lines are small, and third, the system has low r/x ratios. However, In [5-7] some modifications to the FDLF method were suggested in order to handle systems with large r/x ratios. Today nearly all power utilities need to solve problems related to large networks, usually extending well beyond the boundaries of their own companies. Often the capacity and computing time of the system available limits the size and the number of sensitivities which can be analyzed for a given problem. Until recently, most increases in the speed of load flow computations come via the speed of vector processors. Now, the true potential for execution time improvement lies in concurrent multiprocessors. "A multiprocessor is a computer with two or more central processing units, each of which executes instructions independently of the others except when a processor needs to communicate or synchronize with one or more of the others" [8]. Recent studies [9-10], have adressed the parallel processing approach for power system analysis however most of them have applied direct methods for the solution of the linearized systems. Several studies [12-14] have shown that the conjugate gradient method with preconditioning is one of the best methods for solving sparse symmetric positive definite linear systems of equations. Specific implementations of this scheme together with the choice of the preconditioner become important factors when one considers machines which offer vector processing, concurrent processing or some combination of them.

In this paper, we introduce a parallel technique for the intensive parts of the FDLF computations. The first part, the construction and updating of the system matrices and vectors, is a purely local procedure and is ideally suited to parallel computations. The second part, the solution of the linearized equations, involves the simultaneous solution over the whole system. The power system is partitioned using a system decomposition which specifies the matrix structure to be used in the solution then a preconditioned conjugate gradient scheme is introduced. The proposed preconditioner is such that only matrix problems within subsystems are solved. A numerical test example is given in which we have simulated a parallel architecture in order to examine the method in the solution of the power flow problem for the UPS of Egypt. The efficiency and the speedup of the model were measured for two different cases. The results suggest that the present technique is powerfull for large size problems.

THE FDLF METHOD

The load flow problem can be defined as the calculation of of the real and reactive powers flowing in each line and the magnitude and phase angle of the voltage at each bus of a given transmission system for specified generation and load conditions. The FDLF model is obtained as

$$\frac{\Delta P}{V} = [B'] \Delta\theta \tag{1}$$

$$\frac{\Delta Q}{V} = [B''] \Delta V \tag{2}$$

where $\frac{\Delta P}{V}$ and $\frac{\Delta Q}{V}$ are the active and reactive power mismatches vectors respectively, divided by the voltage magnitude. ΔV , $\Delta\theta$ are the corrections vectors to the voltage magnitudes and phase angles respectively. [B'] and [B''] are the symmetric constant load flow matrices usually very sparse. As in [2],the elements of [B']-matrix are

$$b'_{i,j} = -b_{i,j} \qquad \forall i \neq j$$

$$b'_{i,i} = \sum_{j=1, j\neq i}^{n} b_{i,j} \qquad i = j$$

for all buses except the slack bus, and b's are the susceptances of the bus-admittance matrix elements. Also, the elements of [B'']-matrix are

$$b''_{i,j} = -b_{i,j}$$

for all load bus-bar types. However, these elements can be modified in order to include the second order terms of the Taylor expansion such that

$$b'_m(i,i) = b'_{ii}(1 + 2\Delta V_i/V_i) - \sum_{j=1, j\neq i}^{n} b'_{ij}(\Delta V_j/V_j)$$

$$b'_m(i,j) = b'_{ij}(1 + \Delta V_i/V_i + \Delta V_j/V_j)$$

$$b''_m(i,i) = b''_{ii}(1 + 2\Delta V_i/V_i)$$

$$b''_m(i,j) = b''_{ij}(1 + \Delta V_i/V_i + \Delta V_j/V_j)$$

THE PRECONDITIONED CONJUGATE GRADIENT METHOD

In order to solve the linear system

$$C x = d \tag{3}$$

by the conjugate gradient method, we choose a positive definite symmetric matrix **D** and write (3) as

$$D^{-1} C x = D^{-1} d \tag{4}$$

The preconditioner **D** should be spectrally close to **C** ; i.e. the eigenvalues of $\mathbf{D^{-1}C}$ are clustred arround unity and at the same time the computation of $\mathbf{D^{-1}d}$ is easy.

Applying the conjugate gradient method to (4), the solution procedure can be described as follows [12] :

$$x_0 = D^{-1} d$$

$$r_0 = d - C x_0$$

$$p_0 = D^{-1} r_0$$

then for k = 0, 1, 2, ..., until convergence

$$\alpha_k = \frac{r_k \cdot D^{-1} r_k}{p_k \cdot C p_k}$$

$$x_{k+1} = x_k + \alpha_k p_k$$

$$r_{k+1} = r_k - \alpha_k C p_k$$

$$\beta_k = \frac{r_{k+1} \cdot D^{-1} r_{k+1}}{r_k \cdot D^{-1} r_k}$$

$$p_{k+1} = D^{-1} r_{k+1} + \beta_k p_k$$

SYSTEM DECOMPOSITION

The approximating systems (1) and (2) are generated using local information of the power system geometry in the neighbourhood of each bus. Assume that the system Ω is divided by an interface G_{12} into two subsystems Ω_1, Ω_2. Within each subsystem the buses are ordered consecutively, then the coefficient matrix **C** in (3) can be partitioned as

$$C = \begin{bmatrix} C_{00}^{(1)} & \cdot & C_{01}^{(1)} \\ \cdot & C_{00}^{(2)} & C_{01}^{(2)} \\ C_{10}^{(1)} & C_{10}^{(2)} & C_{11}^{(1)} + C_{11}^{(2)} \end{bmatrix}$$

where the block-row i (i=1,2) corresponds to the equations at buses in the interior of subsystem Ω_i and the final block-row contains the equations for the buses on the interface G_{12}. The blocks $C_{00}^{(i)}$, $C_{01}^{(i)}$, $C_{10}^{(i)}$ can be constructed using only informations on Ω_i, the components of the remaining block $C_{11} = C_{11}^{(1)} + C_{11}^{(2)}$ involve the summation of terms from both sides of the interface, clearly this form of matrix and system partition can be extended to any number of subsystems. In general, if Ω is partitioned into p subsystems and the buses within the subsystems are numbered first, then we can write,

$$C = \begin{bmatrix} C_{00}^{(1)} & \cdot & \cdot & \cdot & \sum_{i=1}^{p} C_{01}^{(1,i)} \\ \cdot & C_{00}^{(2)} & \cdot & \cdot & \sum_{i=1}^{p} C_{01}^{(2,i)} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & C_{00}^{(p)} & \sum_{i=1}^{p} C_{01}^{(p,i)} \\ \sum_{i=1}^{p} C_{10}^{(1,i)} & \sum_{i=1}^{p} C_{10}^{(2,i)} & \cdot\ \cdot & \sum_{i=1}^{p} C_{10}^{(p,i)} & \sum_{i,j=1}^{p} C_{11}^{(i,j)} \end{bmatrix}$$

The splitting matrix D is chosen from computations within subsystems only, such that

$$D = \begin{bmatrix} C_{00}^{(1)} & \cdot & \cdot & \cdot & C_{01}^{(1,1)} \\ \cdot & C_{00}^{(2)} & \cdot & \cdot & C_{01}^{(2,2)} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & C_{00}^{(p)} & C_{01}^{(p,p)} \\ C_{10}^{(1,1)} & C_{10}^{(2,2)} & \cdot\ \cdot & C_{10}^{(p,p)} & \sum_{i=1}^{p} C_{11}^{(i,i)} \end{bmatrix}$$

this splitting matrix decouples completely into p subsystems preconditioners of the form

$$D^{(i)} = \begin{bmatrix} C_{00}^{(i)} & C_{01}^{(i,i)} \\ C_{10}^{(i,i)} & C_{11}^{(i,i)} \end{bmatrix}$$

As $D^{(i)}$ do not contain the computations for the whole of their boundary, they are non-singular.

Now suppose that we have a parallel architecture having p processors, then the geometry of the subsystem Ω_i is loaded on the processor i. The equations corresponding to all the buses in the interior of Ω_i can be computed on processor i without any interprocessor communication. Within each processor we construct $C^{(i)}$ and the local preconditioner $D^{(i)}$. Since we are going to apply an iterative method for solution, both $C^{(i)}$and $D^{(i)}$ are stored in compact forms. It is also clear, from the above partition, that when solving an equation of the form

$$Dx = y$$

it is possible to solve in parallel disjoint equations in the subsystems.

TEST IMPLEMENTATION AND NUMERICAL EXPERIMENT

The parallel preconditioned conjugate gradient technique described above has been tested for the solution of the power transmission network of the northern part of Egypt, Figure (1).

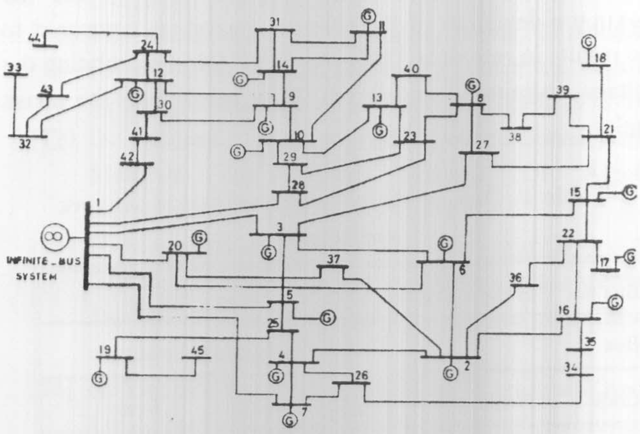

Figure 1. The power transmission network of northern Egypt.

The line-data was collected from reference [11]. The model problem consists of 45 buses , Cairo station was considered as the slack bus and the bus-loading specifications corresponding to the maximum load demand and 0.8944 power factor lagging ($\tan\Phi = 0.5$) of the other power plants and substations are given in Table 1.

The solution of the load flow problem was carried out using the proposed technique for two different cases, in the first case the system is divided into two subsystems while in the second case into four subsystems. In each case the buses are numbered carefully such that the internal buses are numbered before the interface buses, also within each subsystem the generator buses are numbered first.

In order to measure both the performance and the efficiency of the algorithm, we simulate parallel architecture on the regular single processor machine using the Microsoft FORTRAN professional development system (Version 5.1). For this simulation, we consider that the elapsed time for any parallel process is equal to the maximum time required to perform the process in the subsytems. The two major factors in testing the parallel technique are speed-up and efficiency. Speed-up equals the time taken by one processor to do the job divided by the time required for parallel solution. Efficiency is the speed-up per processor. The test example was solved on a 4 Mbyte IBM 386 and the results are given in table (2), where N_p is the number of parallel processors, N_n is the number of Newton-Raphson iterations, N_g is the average number of PCG iterations, T_r is the run time in seconds, S is the speed-up and E is the efficiency.

The active and reactive power mismatch during FDLF iterative solution was considered as 1.0 E-04 and the maximum deviation of the solution vector, as compared to the results in reference [11], was 0.5 E-04 in calculating the voltage magnitudes and 0.3 E-02 in calculating the phase angles.

Table 1. Data of the power transmission network considered.

Bus no. 1 is the slack bus V(1) = 1.06 + j 0.0				
Bus		Volt. mag.	Injected Power	
Code	Name		Act	React
2	C.E.	1.05	-.210	
3	C.N.	1.06	-.053	
4	C.S	1.05	+.058	
5	C.W.	1.06	+.158	
6	Heliop.	1.06	-.265	
7	New Teb.	1.05	-.146	
8	Talkha	1.07	+.130	
9	K.D.	1.11	+.179	
10	Dam.	1.11	+.076	
11	A.Kir	1.12	+.179	
12	Amria	1.10	+.013	

Bus no. 1 is the slack bus V(1) = 1.06 + j 0.0				
Bus		Volt. mag.	Injected Power	
Code	Name		Act	React
13	ATF.	1.10	+.270	
14	Abis	1.21	+.013	
15	ISM.	1.06	+.380	
16	Suez new	1.05	+.380	
17	Suez old	1.05	+.050	
18	N. Said	1.06	-.010	
19	Ayat	1.04	+.380	
20	Shoubra	1.06	+.380	
21	Ism. tr.		-.040	-.0200
22	Suez tr.		-.089	-.0445
23	Tan.		-.061	-.0305

Bus no. 1 is the slack bus V(1) = 1.06 + j 0.0				
Bus		Volt. mag.	Injected Power	
Code	Name		Act	React
24	DEK.		-.076	-.0380
25	Hadaba		-.061	-.0305
26	W.H		-.190	-.0950
27	ZAG.		-.134	-.0670
28	Menouf		-.176	-.0880
29	T.b.		-.070	-.0350
30	GH.		-.077	-.0390
31	Semouha		-.102	-.0510
32	Free zone		-.085	-.0425
33	A. City		-.050	-.0250
34	Sokhna		-.060	-.0300

Bus no. 1 is the slack bus V(1) = 1.06 + j 0.0				
Bus		Volt. mag.	Injected Power	
Code	Name		Act	React
35	Cement		-.028	-.0140
36	10th RAM.		-.078	-.0390
37	Sabtia		-.345	-.1730
38	MAN.		-.140	-.0700
39	KA.		-.058	-.0280
40	K. SH.		-.092	-.0460
41	TM.		-.112	-.0560
42	Sadat		-.125	-.0625
43	SUM.		-.040	-.0200
44	Steel comp.		-.120	-.0600
45	Faioum		-.070	-.0350

Table 2. Results of parallel solution of the UPS of Egypt.

N_p	N_n	N_g	T_r	S	E
1	20	28	121		
2	16	14	79	1.53	76.5%
4	10	6	35	3.46	86.5%

CONCLUSION

We have presented a parallel technique for the FDLF problems that is based on a preconditioned conjugate gradient algorithm. The proposed preconditioner requires only the solution of matrix problems within the subdomains and therefore it may be considered a promising local parallel solver for the future unified network of the middle east area. The proposed technique was tested by simulating a parallel architecture. The results suggest that the proposed technique is usefull for interconnected power systems.

REFERENCES

[1] J.E. Van Ness and J.H. Griffin, "Elimination Methods for Load Flow Studies", *Trans. Am. Inst. Electr Eng.*, Vol. 80, pp.299-304, 1961.

[2] B. Stott, "Review of Load-Flow Calculation Methods", *Proceedings of the IEEE, vol 62*, pp. 916-929, 1974.

[3] B. Stott and O. Alsac, "Fast Decoupled Load Flow", *IEEE Transaction PAS*, vol 93,pp. 859-869, 1974.

[4] A. Monticelli, A. Garcia and O.R. Saavedra, "Fast Decoupled Load Flow:Hypothesis, Derivatios and Testing", *IEEE Transaction on Power Sytems*, vol 5, No.4, pp. 1425-1431, 1990.

[5] R.A.M. Amerongen, "A General-Purpose Version of the Fast Decoupled Load Flow", *IEEE Transaction on Power Sytems*, vol 4, No.2, pp. 760-770, 1989.

[6] L. Wang, N. Xiang, S. Wang, B. Zhang and M. Huang, "Novel Decoupled Load Flow", *IEE Proceedings*,Vol. 137,No. 1, pp. 1-7, 1990.

[7] J. Nanda, V. Bapi Raju, P.R. Bijwe, M.L. Kothari and B.M. Joma, "New Findings of Convergence properties of Fast Decoupled Load Flow Algorithms", *IEE Proceedings*,Vol. 138,No. 3, pp. 218-220, 1991.

[8] D.A. Poplawski, "Parallel Computer Architectures", *Appl. Math. Comput.*, 20(1), pp. 41-51, 1986.

[9] T. Oyama, T. Kitahara and Y. Serizawa, "Parallel Processing for Power System Analysis using Band Matrix", *IEEE Transaction on Power Sytems*, vol 5, No.3, pp. 1010-1016, 1990.

[10] David C. Yu, "A New Parallel LU Decomposition Method", *IEEE Transaction on Power Sytems*, vol 5, No.1, pp. 303-310, 1990.

[11] The publications of "Reactive Power Compensation Project in the A.R.E. Network", which was organized by the governments of Finland and Egypt 1979-1985.

[12] O. Axelson, "A Survey Of Preconditioned Iterative Methods For Linear Systems Of Algebraic Equations",*BIT (1985)*, pp.166-187.

[13] U. Meier and A. Sameh, "The Behaviour Of Conjugate Gradient Algorithms On A Hierarchical Memory", *Journal of Computational and Applied Mathematics*, vol.24 (1988), pp.13-32.

[14] H. Van Der Vorst and K. Dekker, "Conjugate Gradient Type Methods And Preconditioning", *Journal of Computational and Applied Mathematics*, vol.24 (1988), pp.73-87.