A Coding based Approach to Load Flow Analysis using Krylov Subspace Methods for well conditioned systems

Dibyendu Chowdhury, Souvik Singha

***Abstract*—In this work, we propose to apply the conjugate gradient algorithm to the sparse systems; we encounter these in the system admittance matrices, and we will search for a numerical solution to this system using the locally optimal steepest descent method. The system admittance matrices for an IEEE 30-bus or 57-bus system(s) are too large to be handled by direct methods like the Cholesky decomposition method. Hence, we will make use of the flexible preconditioned conjugate-gradient method, which makes use of sophisticated preconditioners, leading to variable preconditioning that change between successive iterations. The Polak–Ribière formula, a highly efficient preconditioner, is applied to the system, to yield drastic improvements in convergence.**

Our experimental results include a comparison of the Krylov subspace method with traditional methods, assuming the IEEE five-busbar, seven-line reference system as the common basis for all load-flow analysis. The system base quantities are VA_{base}= 100 MVA and V_{base}= 132 kV. The results show an overall better assurance of convergence for all general systems, a lesser dependence on starting voltage profiles assumption and a robustness and efficiency of computation for well-conditioned systems.

***Keywords*- Krylov subspace methods, conjugate gradient algorithm, preconditioners, Polak–Ribière formula, assured convergence.**

I. INTRODUCTION

Power system load flow analysis mainly utilizes the Gauss-Seidel method, the Newton-Raphson method, and the Fast Decoupled Load Flow method. All these stationary iterative algorithms assure convergence for a limited class of well-conditioned matrices, and require a good enough estimate of nodal voltages at all system busbars under consideration, to provide assured convergence. The Krylov subspace methods are widely generalized in their approach, and work by forming an orthogonal basis of the sequence of successive matrix powers times the initial residual (the Krylov sequence). The prototypical method in this class is the conjugate gradient method (CG).

The Krylov subspace power flow (KSPF) method presented in this paper uses a newer, very successful approach-the Krylov subspace methodology-developed in applied linear algebra for the iterative solution of large, sparse systems of linear equations[8,10]. The method has been adapted to nonlinear equations and used for the solution of the power flow problem with the method of steepest descent[1,2,7]. Our work takes into account a test system, albeit well-conditioned, and compares an existing method to the proposed Krylov method of calculations [6,7,9].

In numerically challenging applications such as these, sophisticated preconditioners are used, which might lead to variable preconditioning that change between successive iterations. The bus admittance matrix Y is symmetric, as stated earlier and positive definite for all systems under consideration. Hence, the stage is set for the Krylov subspace method to be applied.[1,3,7]

Manuscript received December 09, 2011.

Dibyendu Chowdhury, Department of Electronics & Communication Engineering, Haldia Institute of technology, Haldia, India, (e-mail: dibyendu.chow@gamil.com).

Souvik Singha, Department of Computer Science & Informatics, Bengal Institute of Technology & Management, Santiniketan, India, (e-mail: singha.souvik @gamil.com).

II. CALCULATION PROCEDURE

The conjugate gradient method has two modes of application:-

1. The direct method of application (where Y matrix is not too large)
2. The iterative method of application (where Y matrix is quite large)

A. *The Direct Method Of Application*

We say that two non-zero vectors *u* and *v* are conjugate (with respect to *A*) if,

$$u^T Av = 0 \tag{1}$$

Since **A** is symmetric and positive definite, the left-hand side defines an inner product [13],

$$\langle u, v \rangle_A := \langle Au, v \rangle = \langle u, A^T v \rangle = \langle u, Av \rangle = u^T Av \tag{2}$$

So, two vectors are conjugate if they are orthogonal with respect to this inner product[1,2,4]. Being conjugate is a symmetric relation: if **u** is conjugate to **v**, then **v** is conjugate to **u**.

Suppose that $\{\mathbf{p}_k\}$ is a sequence of *n* mutually conjugate directions [13]. Then the $\mathbf{p}_k$ form a basis of $\mathbf{R}^n$, so we can expand the solution $\mathbf{x}_*$ of $\mathbf{Ax} = \mathbf{b}$ in this basis:

$$x_* = \sum_{i=1}^{n} \alpha_i p_i \tag{3}$$

The coefficients are given by

$$b = Ax_* = \sum_{i=1}^{n} \alpha_i p_i \tag{4}$$

$$p_k^T b = p_k^T Ax_* = \sum_{i=1}^{n} \alpha_i p_k^T Ap_i = \alpha_k p_k^T Ap_k$$

(Because $\forall i \neq k, p_i, p_k$ are mutually conjugate)

$$\alpha_k = \frac{p_k^T b}{p_k^T A p_k} = \frac{\langle p_k, b \rangle}{\langle p_k, p_k \rangle_A} = \frac{\langle p_k, b \rangle}{\|p_k\|_A^2} \qquad (5)$$

This result is perhaps most transparent by considering the inner product defined above.
This gives the following method for solving the equation **Ax** = **b**. We first find a sequence of *n* conjugate directions and then we compute the coefficients [1,5,2] α_k.

B. *The Iterative Method Of Application*

If we choose the conjugate vectors $\mathbf{p}_k$ carefully, then we may not need all of them to obtain a good approximation to the solution $\mathbf{x}_*$.
We denote the initial guess for $\mathbf{x}_*$ by $\mathbf{x}_0$. We can assume without loss of generality that $\mathbf{x}_0 = 0$. Starting with $\mathbf{x}_0$, we search for the solution and in each iteration we need a metric to tell us whether we are closer to the solution $\mathbf{x}_*$ (that is unknown to us). This metric comes from the fact that the solution $\mathbf{x}_*$ is also the unique minimizer of the following quadratic function [1,7,9]; so if f(**x**) becomes smaller in an iteration it means that we are closer to $\mathbf{x}_*$.

$$f(x) = \frac{1}{2} x^T A x - x^T b, x \in R^n \qquad (6)$$

This suggests taking the first basis vector $\mathbf{p}_1$ to be the negative of the gradient of *f* at **x** = $\mathbf{x}_0$. This gradient equals $\mathbf{Ax}_0 - \mathbf{b}$. Since $\mathbf{x}_0 = \mathbf{0}$, this means we take $\mathbf{p}_1 = \mathbf{b}$**[7,9]**. The other vectors in the basis will be conjugating to the gradient, hence the name conjugate gradient method.
Let $\mathbf{r}_k$ be the residual at the *k*th step:

$$r_k = b - A x_k \qquad (7)$$

Because $\mathbf{r}_k$ is the negative gradient of *f* at **x** = $\mathbf{x}_k$, so now using the gradient descent method would be to move in the direction $\mathbf{r}_k$. Here, we insist that the directions $\mathbf{p}_k$ be conjugate to each other. We also require the next search direction is built out of the current residue and all previous search directions, which is reasonable enough in practice.
The conjugation constraint is an orthonormal-type constraint and hence the algorithm bears resemblance to Gram-Schmidt orthonormalization.[1,7]
This gives the following expression:

$$p_{k+1} = r_k - \sum_{i \le k} \frac{p_i^T A r_k}{p_i^T A p_i} p_i \qquad (8)$$

The next optimal location is given by:

$$x_{k+1} = x_k + \alpha_{k+1} p_{k+1}$$

With

$$\alpha_{k+1} = \frac{p_{k+1}^T r_k}{p_{k+1}^T A p_{k+1}}$$

In this work, we propose our Krylov subspace algorithm incorporated in the load flow solution.

III.OUR APPROACH

The load flow system is given by:-
I = Y.V, where Y= the nodal bus admittance matrix.
V= the nodal busbar voltage matrix.
Thus,

Y.V=I is analogous to the linear system A.x=B, and so we can start our calculations by calculating the residual vector matrix R_0 associated with the voltage matrix V.

This residual vector is computed from the formula, R_0 = I – Y.V_0, where V_0 is the initial guess matrix on nodal busbar voltage matrix.

Let us start with the initial guess: - (for $\mathbf{V_0}$ matrix)

$\mathbf{V_0}$= $\mathbf{I_0}$=

1.000000000 +j0.0000000	1.000000000 +j0.0000000
0.988841 -j0.010351	-0.400000 +j0.20000
0.991808 -j0.007102	-0.25000000 +j0.150000
0.981054 -j0.017084	-0.40000000 +j0.2000000
0.989051 -j0.012369	-0.5000000 +j0.200000

The initial guess matrix can be obtained from the iterative load flow equations, namely:-

1) $$V_k^{(i+1)} = \left(\frac{1}{Y_{kk}}\right) \times \left\{ \frac{S_k^*}{V_k^{*(i)}} - \sum_{\substack{j=1 \\ j \neq k}}^{n} Y_{kj} V_j^{(i)} \right\}$$

2) $$\sum_{\substack{j=1 \\ j \neq k}}^{n} Y_{kj} V_j = \sum_{\substack{j=1 \\ j \neq k}}^{n} \left\{ (G_{kj} + jB_{kj})(E_j + jF_j) \right\}$$

R_0 = **I – Y.V_0**

0.6216+j0.1251
-0.2597+j0.0974
-0.3058+j0.132
-0.2298+j0.0802
-0.379+j0.1646

Thus, the residual matrix R_0 is evaluated.
Since this is the first iteration, we use the residual vector R_0 as our initial search direction $\boldsymbol{P_0}$.

We now compute the scalar α_0, and the result, obtained from previous relationships, as:-
α_0 = (0.66743 – j0.1365) / (0.9876 – j0.0098)

We can now compute X_1 using the formula-
$X_1 = X_0 + \alpha_0 P_0$.

This result completes the first iteration, the value of X_1 giving us an improved approximate solution to the system under consideration.
Successive iterations give us the desired result to the system solution, to the required degree of accuracy.

IV. CODING ALGORITHM

The resultant algorithm gives an explanation towards the conjugate gradient method. However, it requires storage of all the previous searching directions and residue vectors, and many matrix vector multiplications, thus could be computationally expensive. In practice, one modifies slightly the condition obtaining the last residue vector, not to minimize the metric following the search direction, but instead to make it orthogonal to the previous residue. Minimization of the metric along the search direction will be obtained automatically in this case.

$$r_0 := b - Ax_0$$
$$p_0 = r_0$$
$$k := 0$$

Repeat,

$$\alpha_k := \frac{r_k^T r_k}{p_k^T A p_k}$$
$$x_{k+1} := x_k + \alpha_k p_k$$
$$r_{k+1} := r_k - \alpha_k A p_k$$

If $\mathbf{r}_{k+1}$ is sufficiently small **then** exit loop **end if**

$$\beta_k := \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$$
$$p_{k+1} := r_{k+1} + \beta_k p_k$$
$$k := k + 1$$

End

Repeat

The result is x_{k+1}.

As previously stated, the Polak–Ribière formula needs to be incorporated within the existing framework to yield faster convergence of the Krylov Subspace method. The modified algorithm thus is presented below:-

$$r_0 := b - Ax_0$$
$$z_0 = M^{-1`} r_0$$
$$p_0 = z_0$$
$$k := 0$$

Repeat,

$$\alpha_k := \frac{r_k^T z_k}{p_k^T A p_k}$$
$$x_{k+1} := x_k + \alpha_k p_k$$
$$r_{k+1} := r_k - \alpha_k A p_k$$

If $\mathbf{r}_{k+1}$ is sufficiently small **then** exit loop **end if**

$$z_{k+1} = M^{-1`} r_{k+1}$$
$$\beta_k := \frac{z_{k+1}^T r_{k+1}}{z_k^T r_k}$$
$$p_{k+1} := z_{k+1} + \beta_k p_k$$
$$k := k + 1$$

End

Repeat

The result is x_{k+1}

These two algorithms, coupled together bring about drastic improvement in the load flow analysis of any ill-conditioned system.

V. RESULTS

The results show a marked improvement in the convergence time T taken for the iterations to converge. The time taken for the Krylov subspace method is found to be much lesser than the popular iterative method, the Gauss-Seidel method. The 2 tables presented below, gives us a clearer assumption about the nature of the degree of convergence, obtained via the locally optimal path of steepest descent incorporated in the Krylov subspace method, as compared to the Gauss-Seidel method.

Iteration	∂_2^0	$\lvert V_3 \rvert$	∂_3^0
0.	8.0000	0.7000	-16.0000
1.	-0.228	1.024	-8.210
2.	-0.226	1.028	-4.235
3.	-0.226	1.028	-4.233

The above results have been obtained via the Krylov Subspace algorithm.

Iteration	∂_2^0	$\lvert V_3 \rvert$	∂_3^0
0.	8.0000	0.7000	-16.0000
1.	-6.2340	0.8500	-12.0876
2.	-3.2786	0.8800	-10.0983
3.	-2.0986	0.9347	-8.9073
4.	-1.2563	0.9876	-6.2103
5.	-0.8792	1.0023	-4.2874
6.	-0.3567	1.0467	-4.3980
7.	-0.220	1.033	-4.2980
8.	-0.220	1.035	-4.2450

The above results have been obtained via the Gauss-Seidel algorithm.

The results clearly show a marked improvement of computational efficiency, when the Krylov subspace method is used over the Gauss-Seidel algorithm.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

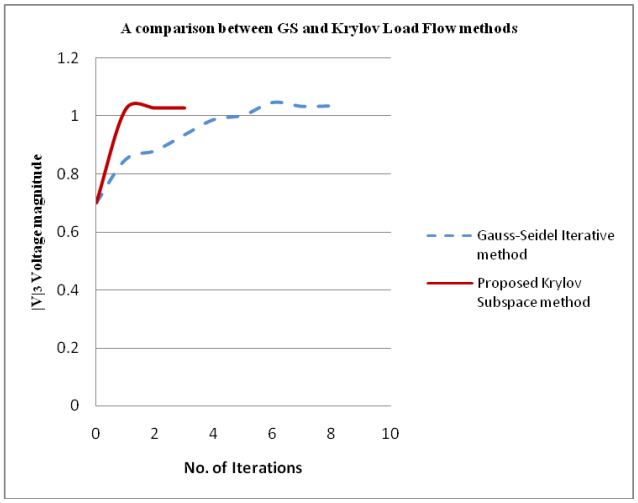

Figure 1: Convergence Analysis of the two methods

A graphical representation of the load flow convergence schema presented above has been illustrated for the sake of clarity of perception. Series 1 depicts the convergence of the proposed Krylov subspace method, whilst Series 2 depicts the convergence of the Gauss-Seidel method.

VI. CONCLUSION

The recent developments in power system load flow analysis have directed attention away from iterative methods, and towards stochastic search methods. The general stochastic methods, namely, Genetic Algorithms and Simulated Annealing offer better computational complexity-storage space trade-off. Nevertheless, the Conjugate Gradient method utilizes the sparsity of the Nodal Admittance matrix to near perfection, thus keeping the number of iterations at the lowest possible level, whilst not consuming too much computer storage space. Pre-conditioning, using the Polak–Ribière formula, helps to reduce the condition number of most well conditioned system matrices. Thus, we conclude that the results show an overall better assurance of convergence for all general systems, a lesser dependence on starting voltage profiles assumption and a robustness and efficiency of computation for well-conditioned systems.

REFERENCES

1. James W. Daniel, " Convergence of the Conjugate Gradient Method with Computationally Convenient Modifications" Numerische Mathemati (1967), pp125–131.
2. J. Grainger and W. Stevenson, "Power System Analysis" McGraw-Hill, New York, 1994, ISBN 0-07-061293-5.
3. R. Fletcher and M. J. D. Powell, "A Rapidly Convergent Descent Method for Minimization" Computer Journal **6** (1963), pp163–168.
4. R. Fletcher and C. M. Reeves, "Function Minimization by Conjugate Gradients", Computer Journal **7**(1964), pp149–154.
5. L. Fox, H. D. Huskey, and J. H. Wilkinson, "Notes on the Solution of Algebraic Linear Simultaneous Equations*"*, Quarterly Journal of Mechanics and Applied Mathematics **1** (1948), pp149–173.
6. Jean Charles Gilbert and Jorge Nocedal, "Global Convergence Properties of Conjugate Gradient Methods for Optimization*"*, SIAM Journal on Optimization **2** (1992), no. 1,pp 21–42.
7. Gene H. Golub and Dianne P. O'Leary, "Some History of the Conjugate Gradient and Lanczos Algorithms*":1948–1976*, SIAM Review **31** (1989), no. 1, pp50–102.
8. Magnus R. Hestenes, "Iterative Methods for Solving Linear Equations", Journal of Optimization Theory And Applications **11** (1973), no. 4, pp323–334.
9. Web links: http://en.wikipedia.org
10. Lynn Powell, *Power System Load Flow Analysis*, McGraw Professional Engineering Series.
11. Onoda Pessanha, Jose Eduardo; Portugal, Carlos Enrique; and Saavedra, Osvaldo R. (2011) "An Efficient Reordering Technique for Incomplete Factorization GMRES Preconditioning: Application in Load Flow Problems," *International Journal of Emerging Electric Power Systems*: Vol. 12: Iss. 2, Article 4.

AUTHORS PROFILE

Mr. Dibyendu Chowdhury is working as an Assistant Professor in the department of Electronics & Communication Engineering, Haldia Institute of Technology, Haldia. He received his B.Sc and M.Sc in Electronics under Vidyasagar University in the year of 2004 and 2006 respectively. He received his M.Tech in Electronics & Communication Engineering under West Bengal University of Technology in the year of 2009. He also received **Gold Medal** (For M.Sc) from Vidyasagar University. His area of research interest includes VLSI and Embedded Systems. He is working towards his Ph.D. degree.

Mr. Souvik Singha is working as an Assistant Professor in the department of Computer Science & Informatics, Bengal Institute of Technology & Management, Santiniketan. He received his Diploma in Engineering (CST) under West Bengal Board of Technical Education in the year of 2002 and he received BE (CSE) under University of Burdwan in the year of 2005. He received his M.Tech in CSE under National Institute of Technology of Durgapur, Burdwan in the year of 2008. His area of research interest includes VLSI. He is working towards his Ph.D. degree.