© Journal of Computational and Applied Mathematics (2003)

Fast computational methods for locating fold points for the power flow equations*

Anwar Hussein† and Ke Chen
Department of Mathematical Sciences,
The University of Liverpool, Peach Street, Liverpool L69 7ZL,
United Kingdom.

Abstract

Voltage collapse in a power system can occur following a progressive decline in voltage magnitude at the system buses and, mathematically, this phenomenon is associated with a fold bifurcation point occurring in the nonlinear algebraic equations used to model the power system. In this paper we first discuss some of the methods used to speed up the process of detecting a fold bifurcation, focusing on designing test function methods to predict a performance index. We then discuss some iterative methods that can be used to improve the Newton iterations. In particular, we present new and efficient preconditioners of the two-level type for the Jacobian matrix. Numerical results are given using standard IEEE test bus systems.

Keywords: Power flow equations, Performance indices, Voltage collapse, Continuation method, Saddle-node bifurcation, Iterative methods, Sparse Preconditioners, Deflation.

1 Introduction

A power system is typically modelled, from Kirchhoff laws and power conservation, by a parameter dependent system of non-linear algebraic equations [7, 10]

$$\mathbf{f}(\mathbf{x}, \lambda) = \mathbf{0} \tag{1}$$

where $\mathbf{x} \in \mathbf{R}^n$ is the vector of state variables in voltage magnitudes and phase angles, $\lambda \in \mathbf{R}$ is a parameter (usually load change), and $\mathbf{f} : \mathbf{R}^n \times \mathbf{R} \to \mathbf{R}^n$ is a sufficiently smooth function.

Voltage collapse in a power system has been explained by bifurcation problems of the power flow equations used to model the system, particularly fold bifurcations [16] which occur when a stationary point loses stability and coalesces with an unstable point as a parameter is varied. The usual procedure to determine such points is to use a continuation method to trace the bifurcation manifold. A performance

*For correspondence, email: k.chen@liverpool.ac.uk, Web: http://www.liv.ac.uk/maths/applied
†Email: anwarh@liverpool.ac.uk

index can then be obtained which gives the distance from an operating point in the parameter space to a voltage collapse. Computationally, the first challenge is to predict the fold bifurcation point without carrying out too many nonlinear iterations and the second one is to speed up the linear system solvers within non-linear iterations. While it is difficult to solve such problems in a general setting, some cases can be solved satisfactorily for the power flow equations.

In this paper we first survey some recent methods of using test functions to obtain an accurate prediction for the parameter λ^* at a fold point starting from any current value λ_0. The computational speedup will be measured by reduction in the number of nonlinear parameter stepping steps. Then we identify a class of sparse linear systems that must be solved to utilise the above prediction methods, and consider some iterative methods for solving them at each Newton iteration. In particular, we concentrate on constructing efficient preconditioners of the two level type for the Jacobian matrix of equation (1).

2 Power Flow Equations and Computation of Fold Points

The power flow equations are given by

$$\mathbf{f}(\mathbf{x}, \lambda) = \begin{pmatrix} \Delta\mathbf{P}(\mathbf{x}, \lambda) \\ \Delta\mathbf{Q}(\mathbf{x}, \lambda) \end{pmatrix} = \mathbf{0}, \tag{2}$$

where $\mathbf{x} = (\delta^\top, \mathbf{v}^\top)^\top \in \mathbf{R}^n$ is the state vector representing the bus voltage angle and magnitude, and $\lambda \in \mathbf{R}$ is the bifurcation parameter and refers to the total power load increase [22]. Let m be the number of buses minus the reference bus. For simplicity denote the voltage angle between two buses j and k by $\delta_{jk} \equiv \delta_j - \delta_k$. At each bus j the power flow equations are given by

$$\begin{cases} \Delta P_j = P_{Gj} - P_{L_j} - \sum_{k=1}^{m+1} v_j v_k (G_{jk} \cos \delta_{jk} + B_{jk} \sin \delta_{jk}) - \lambda \alpha_j, \\ \Delta Q_j = Q_{Gj} - Q_{L_j} - \sum_{k=1}^{m+1} v_j v_k (G_{jk} \sin \delta_{jk} - B_{jk} \cos \delta_{jk}) - \lambda \beta_j, \end{cases} \tag{3}$$

where P_{L_j} and Q_{L_j} are the real and reactive power loads, P_{G_j} and Q_{G_j} are the real and reactive generation power at bus j, G_{ij} and B_{jk} are the real and imaginary parts of the admittance between buses j and k, α_j and β_j describe the load pattern increase at bus j for the real and reactive loads respectively; refer to [10, 22].

Our primary task is to locate the fold point $(\mathbf{x}, \lambda) = (\mathbf{x}^*, \lambda^*)$ at which the Jacobian matrix $\mathbf{J} = \mathbf{f_x}$ of $\mathbf{f}(\mathbf{x}, \lambda) = \mathbf{0}$ is singular (or one of its eigenvalues becomes zero i.e. $\mathbf{J}h = \mathbf{0}$ for some nonzero vector h). In particular, $\text{rank}(\mathbf{f_x}) = \text{n} - 1$ and $\mathbf{f}_\lambda \notin \text{range}(\mathbf{f_x})$. In power system control only the parameter λ^* is of importance. To locate the fold point, we compute the solution path of $\mathbf{f}(\mathbf{x}, \lambda) = \mathbf{0}$ as λ varies.

Starting from the current operating point $(\mathbf{x}_0, \lambda_0)$, to find the next solution $\mathbf{x} = \mathbf{x}_1$ at $\lambda = \lambda_1$ of

$\mathbf{f}(\mathbf{x},\lambda) = \mathbf{0}$, we use Newton's method for $k \geq 0$ (in order to find λ^* gradually):

$$\boxed{\text{solve}} \quad \mathbf{J}(\mathbf{x}_1^{(k)},\lambda_1)\Delta\mathbf{x}_1^{(k)} = -\mathbf{f}(\mathbf{x}_1^{(k)},\lambda_1), \qquad \boxed{\text{update}} \quad \mathbf{x}_1^{(k+1)} = \mathbf{x}_1^{(k)} + \Delta\mathbf{x}_1^{(k)}$$

where $\mathbf{x}_0$ is given, $\mathbf{x}_1^{(0)} = \mathbf{x}_0$, and $\mathbf{J} = \mathbf{f}_\mathbf{x}$ is the Jacobian matrix. Newton's method will converge quadratically from a sufficiently good initial guess, provided that the Jacobian is nonsingular at the solution or in other words away from the fold bifurcation. Near λ^* (when $\mathbf{J}$ will be close to singularity), we must use a method that has no convergence difficulties close to and at the bifurcation point. This is usually achieved by using a continuation method which has been extensively discussed in the literature [27, 33, 17].

The idea of a continuation method is to numerically trace the solution path of $\mathbf{f}(\mathbf{x},\lambda) = \mathbf{0}$ as the parameter λ varies until the fold bifurcation is located through some monitoring procedure. A popular example is the pseudo arc-length continuation which is due to Keller [27] that involves solving, by Newton's method, the following parameterised system of equations for $(\mathbf{x},\lambda)$ sufficiently close to $(\mathbf{x}_0,\lambda_0)$:

$$\begin{cases} \mathbf{f}(\mathbf{x},\lambda) = \mathbf{0}, \\ \dot{\mathbf{x}}_0^\top(\mathbf{x}-\mathbf{x}_0) + \dot{\lambda}_0(\lambda-\lambda_0) - \Delta s = 0, \end{cases} \tag{4}$$

where s is a parameter that describes the solution path of $\mathbf{f}(\mathbf{x},\lambda)$, and $(\dot{\mathbf{x}}_0,\dot{\lambda}_0)$ is the normalised tangent vector such that $\|\dot{x}_0\|^2 + \dot{\lambda}_0^2 = 1$. The step size Δs can be taken as a constant throughout the continuation process. In practice though, Δs control is based on the convergence of the Newton iteration (see [33, 5]). This continuation method has proved to be very reliable in locating the bifurcation parameter λ^* at a fold. However for a practical situation, the total number of nonlinear iterations used can be too large for the method to be useful.

Other mathematically sound methods include setting up and solving an extended $(2n+1)$ dimensional system [30, 33] in the so-called direct methods. These methods have very stringent requirements on the initial point $(\mathbf{x}_0,\lambda_0)$ and are not suitable for practical computations. In what follows we shall seek ways to speed up the nonlinear solution process before considering linear solvers.

3 Test Functions based Performance Index

Test functions, as smooth functions $\tau : \mathbf{R}^{n+1} \to \mathbf{R}$ defined along the solution path of $\mathbf{f}(\mathbf{x},\lambda) = \mathbf{0}$, are necessary tools for detecting bifurcation points during the process of continuation [33]. A test function satisfies $\tau(\mathbf{x}^*,\lambda^*) = 0$ when $(\mathbf{x},\lambda) = (\mathbf{x}^*,\lambda^*)$ is a bifurcation point.

The obviously qualified test function $\tau(\mathbf{x},\lambda) = \det(\mathbf{J}(\mathbf{x},\lambda))$ suffers from scaling problems [10, 21, 20, 5]. To alleviate these problems, the bordering method sets up an $(n+1)$-dimensional bordered system to compute the test function τ

$$\begin{pmatrix} \mathbf{J}(\mathbf{x},\lambda) & \mathbf{d} \\ \mathbf{c}^\top & 0 \end{pmatrix} \begin{pmatrix} \mathbf{w} \\ \tau \end{pmatrix} = \mathbf{M} \begin{pmatrix} \mathbf{w} \\ \tau \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ 1 \end{pmatrix}, \tag{5}$$

where $\mathbf{d}, \mathbf{c} \in \mathbf{R}^n$ are chosen such that the matrix $\mathbf{M}$ is non-singular for all regular solutions of $\mathbf{f}(\mathbf{x},\lambda) = \mathbf{0}$. Then by Cramer's rule, such a τ is clearly a scaled Jacobian (satisfying $\tau(\mathbf{x}^*, \lambda^*) = 0$)

$$\tau(\mathbf{x},\lambda) = \frac{\det(\mathbf{J})}{\det(\mathbf{M})}. \tag{6}$$

A related and mathematically equivalent test function is the tangent test function due to Abbott-Seydel [2, 1, 32]

$$\tau = \mathbf{e}_k^\top \mathbf{B}^{-1} \mathbf{J} \mathbf{e}_k. \tag{7}$$

Here, $\mathbf{B}$ is defined to be the matrix $\mathbf{J} = \mathbf{f}_\mathbf{x}$ with column k replaced with $\mathbf{b} = \mathbf{f}_\lambda$, i.e.

$$\mathbf{B} = \mathbf{J}(\mathbf{I}_n - \mathbf{e}_k \mathbf{e}_k^\top) + \mathbf{b}\mathbf{e}_k^\top, \tag{8}$$

where $\mathbf{I}_n$ is the $n \times n$ identity matrix, and $\mathbf{e}_k$ is the k-th unit vector. In [26], the Abbott-Seydel test function (7) was reformulated into

$$\tau_k(\mathbf{x},\lambda) = \frac{1}{\mathbf{e}_k^\top (d\mathbf{x}/d\lambda)}. \tag{9}$$

Furthermore, a formal link was made to the $(n+1)$ augmented system in (5) and also to the TVI method [7, 15] that has been in use by the power engineering community for many years. Refer to [26, 25].

As remarked, a mathematical as well as engineering challenge is to be able to predict λ^* without having to compute many nonlinear stepping steps. We now consider how to adapt the test function $\tau(\lambda)$ in (9) to yield an approximate performance index i.e., an approximation to parameter λ^*.

The essence of this work is to analyse the special analytical behaviour of functions $det(J)$, $det(B)$ and hence τ_k. It turns out that this is possible for power systems (2) and a reliable approximation to λ^* can be computed from the Newton-like formula at the current solution point (x_0, λ_0)

$$\lambda^* = \lambda_0 - \frac{1}{2} \frac{\tau(\lambda_0)}{d\tau/d\lambda(\lambda_0)}, \tag{10}$$

where a finite difference formula may be used to compute

$$\frac{d\tau}{d\lambda} = \frac{\tau(\lambda + \delta\lambda) - \tau(\lambda)}{\delta\lambda}, \tag{11}$$

for suitable $\delta\lambda > 0$. We remark that the reliability of (10) is only established well after it has been used for general test functions and found unreliable for some test functions [12, 32, 34].

Varied bus	Exact index	CPF	C steps	Predicted index using τ
4	5.257974	5.2539	55	5.3251
5	2.454823	2.4305	25	2.4531
6	5.814020	5.7499	58	5.8133
7	3.201959	3.1859	34	3.1984
8	5.245256	5.2247	54	5.2968
9	2.342621	2.3287	25	2.3514

Table 1: Comparison of the performance index (using the Abbott-Seydel test function τ) with the continuation power flow solution ('Exact index'). 'CPF' stands for continuation power flow and 'C steps' for 'CPF steps'. Note that only two CPF steps are needed to obtain the predicted index using τ.

We present some numerical results to illustrate the effectiveness of the the performance index (i.e. the predicted fold parameter λ^*). The IEEE 9 bus system consisting of 3 generators, 6 load buses and 9 lines is used to illustrate the application of the performance index. Equation (9) is implemented by computing two successive Newton solutions at $\lambda = \lambda_0$ and $\lambda = \lambda_0 + \delta\lambda$ for a suitably small $\delta\lambda$. $d\tau/d\lambda$ is then then found via the finite difference described in (11). A performance index is then evaluated using (10). For simplicity, we set $\lambda_0 = 0$. Table 1 shows, for single bus variations, the appropriate fold bifurcation parameter λ^* evaluated using the tangent test function method and the pseudo arc-length continuation method with a fixed step length of 0.1. These results are compared with the exact λ^*. Observe that the test function (9) provides a very good prediction for λ^* at a cost of two Newton steps.

In summary, formula (10) defines a fast nonlinear method, based on the Abbott-Seydel test function (7), for location of the bifurcation parameter λ^*. This method only requires two successive power solutions. We shall discuss how to speed up the linear system solvers within each nonlinear power solution for large networks.

4 Preconditioning methods for the power flow equations

To further speed up fast computational methods in §3, we propose to solve iteratively a set of linear systems involving the Jacobian of (3) which is nonsymmetric, indefinite and usually very sparse. We use Krylov methods, mainly the generalised minimal residual method (GMRES [31]), to solve (12). However, *for a nonlinear power system to be stable*, all eigenvalues of $\mathbf{A}$ must lie on the left plane (i.e. far away from the desirable value 1) and consequently convergence cannot be fast even if the iterative method converges at all. Therefore preconditioning is required. Such an ill-conditioned nature of the linear systems is thus typical of stability systems to which our proposed preconditioners are applicable.

We remark that, for our particular type of problems where a system is well behaved (i.e. far away from a bifurcation point), the fast decoupled approximation can be employed as a preconditioner [11] and this preconditioner is essentially a 2×2 block diagonal matrix.

In this section we will discuss a class of two level sparse preconditioners. The level one preconditioner is based on sparse approximate inverses (SPAI) (refer to [3, 4, 28, 23, 24]) whilst level two preconditioners are based on deflation techniques (refer to [9, 19, 6]).

4.1 Preconditioning with sparse approximate inverses

Let a sparse set of linear systems be denoted by

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{12}$$

where $\mathbf{A}$ is a real $n \times n$ nonsymmetric matrix which represents the Jacobian matrix of (3). We shall solve an equivalent system to find the solution of (12)

$$\mathbf{A}\mathbf{M}\mathbf{y} = \mathbf{b}, \qquad \mathbf{x} = \mathbf{M}\mathbf{y} \tag{13}$$

where $\mathbf{M}$ is a right preconditioner. The intention is to choose $\mathbf{M}$ in such a away that the preconditioned matrix $\mathbf{AM}$ is better conditioned and hence more amenable to fast solution by GMRES.

We propose to use a SPAI method to select $\mathbf{M}$ by solving the minimisation problem

$$\min_{\mathbf{M}} \|\mathbf{I} - \mathbf{A}\mathbf{M}\|_F = \min_{\mathbf{M}} \sum_{k=1}^{n} \|(\mathbf{A}\mathbf{M} - \mathbf{I})\mathbf{e}_k\|_2^2 = \sum_{k=1}^{n} \min_{m_k} \|\mathbf{A}\mathbf{m}_k - \mathbf{e}_k\|_2^2 \tag{14}$$

where $\|.\|_F$ is the Frobenius norm and the solution separates into n independent least square problems

$$\min_{m_k} \|\mathbf{A}\mathbf{m}_k - \mathbf{e}_k\|_2, \qquad k = 1, \ldots, n. \tag{15}$$

Here $\mathbf{M} = [m_1, \ldots, m_n]$ and $\mathbf{I} = [e_1, \ldots, e_n]$. As $\mathbf{A}$ is sparse, the above least square problem can be solved effectively and moreover these problems can be solved in parallel.

For many other sparse matrix applications, the adaptive pattern strategy for specifying $\mathbf{M}$ pattern as shown in [23] is quite effective but also expensive to implement. We now briefly discuss a priori strategies for specifying $\mathbf{M}$ pattern so that computing $\mathbf{M}$ is made less expensive. The theoretical basis for the so-called powers of sparcified matrices (PSM) methods comes from analytical expressions for $\mathbf{A}^{-1}$ in terms of $\mathbf{A}$. From the Cayley-Hamilton theorem, $\mathbf{A}$ satisfies the characteristic polynomial

$$\mathbf{A}^n + a_{n-1}\mathbf{A}^{n-1} + \cdots + a_1\mathbf{A} + a_0 I = 0. \tag{16}$$

Hence, if $\mathbf{A}$ is non-singular, $\mathbf{A}^{-1} \in \text{span}(I,\ \mathbf{A},\ \mathbf{A}^2,\ \ldots,\ \mathbf{A}^{n-1})$. Alternatively if $\|\mathbf{A}\| < 1$, we may also use the geometric series to expand $\mathbf{A}^{-1} = [I - (I - \mathbf{A})]^{-1}$ in terms of powers of $(I - \mathbf{A})$. These algebraic considerations may also be explained by using the graph theory [14] and have recently been explored in [8, 14, 24]. However, even for a sparcified matrix $\mathbf{A}_0$ (out of $\mathbf{A}$), the number of nonzeros in its high powers can grow quickly to approach n^2 so in practice only low powers of $\mathbf{A}_0$ are used. Use 'drop' to denote a sparcification process; then $\mathbf{A}_0 = drop(\mathbf{A})$ and powers are given by $\mathbf{A}_i = \mathbf{A}_0^{i+1}$ or, if less nonzeros are desired, $\mathbf{A}_i = drop(\mathbf{A}_{i-1}\mathbf{A}_0)$. Finally the sought pattern for $\mathbf{M}$ is taken from the graph of $\mathbf{A}_i$. In this paper we shall mainly use $i = 2$.

Remark 4.1. *No matter how the sparsity pattern of* $\mathbf{M}$ *is specified, the maximal number of nonzeros, nzmax, in columns of* $\mathbf{M}$ *must not exceed a certain integer nnz and* $nnz \ll n$. *This is because* $nnz = n$ *implies that computing* $\mathbf{M}$ *would be more expensive than solving (12) directly. Consequently in (15),* $\|\mathbf{A}m_k - e_k\| \leq \epsilon$ *(say for* $\epsilon = 0.1$*) cannot be satisfied for particular columns* k *in certain problems. This happens to be case for our power flow equations. We can further observe that the spectrum of the preconditioned matrix* $\mathbf{AM}$ *contains some small eigenvalues (see Figure 2). Our idea is to employ a level two preconditioner* M_2 *so that the new preconditioned matrix* $\mathbf{A}_2 = \mathbf{A}M_2$ *will not have small eigenvalues. Then we expect GMRES to converge faster for* $\mathbf{A}_2$ *than* $\mathbf{AM}$.

4.2 Two level preconditioners

We consider deflation techniques for matrix $\mathbf{AM}$, where $\mathbf{M}$ is the first level preconditioner based on SPAI. Deflation of a single eigenvalue is a well known idea which can be adapted for multiple eigenvalues. Briefly if u_1 is an unit eigenvector corresponding to eigenvalue $\lambda_1 = \lambda_1(A) \neq 0$, then the matrix

$$A_1 = AM_1 \qquad \text{with} \qquad M_1 = I + \sigma u_1 u_1^H \tag{17}$$

has all eigenvalues identical to A except one of them $\lambda_1(A_1) = \lambda_1 + \sigma\lambda_1$. Clearly $\lambda_1(A_1) = 0$ if $\sigma = -\lambda_1$, $\lambda_1(A_1) = 1$ if $\sigma = \lambda_1^{-1} - 1$, or $\lambda_1(A_1) = \lambda_1 + 1$ if $\sigma = \lambda_1^{-1}$. To prove this result, consider $A_1 u_1$ if A is symmetric; however we must use orthogonal complements to u_1 (in R^n) for the nonsymmetric case [19].

Relevant to this work is adaption of (17) for deflating multiple eigenvalues of matrix $\mathbf{AM}$. Define $V = [v_1, \ldots, v_k]$ with $v_1, \ldots, v_k$ corresponding to k smallest eigenvalues $\lambda_1, \ldots, \lambda_k$ of matrix $\mathbf{AM}$.

The generalisation is only straightforward for the hypothetical case of a symmetric $\mathbf{AM}$

$$\mathbf{A}_2 = \mathbf{A}M_2 \qquad \text{with} \qquad M_2 = \mathbf{M} + \mathbf{M}VD_0V^H \tag{18}$$

where $D_0 = D_{k\times k} = diag(\lambda_1^{-1}, \ldots, \lambda_k^{-1})$. Then from $V^H V = I_{k\times k}$ and $V^H U = 0_{k\times(n-k)}$, for $U =$

$[u_{k+1}, \ldots, u_n]$ with u_j's eigenvectors for other $(n-k)$ eigenvalues of $\mathbf{AM}$, we can prove that

$$\lambda_j(\mathbf{A}_2) = \begin{cases} \lambda_j(\mathbf{AM}) + 1, & j = 1, \cdots, k \\ \lambda_j(\mathbf{AM}), & j > k. \end{cases} \tag{19}$$

However, neither matrix $\mathbf{A}$ nor $\mathbf{AM}$ is in general symmetric. Hence, eigenvectors are *not* orthogonal to each other and one needs to construct an orthogonal basis in the subspace of the first k eigenvectors. We now consider two methods of deflating multiple eigenvalues in the nonsymmetric case.

4.2.1 Method 1

Starting from $\mathbf{AM}V = VD_1$, $D_1 = diag(\lambda_1, \ldots, \lambda_k)$, We propose to orthogonalise matrix $V = V_{n\times k}$ and decompose $V = UR$ with $U = U_{n\times k}$ orthogonal and $R = R_{k\times k}$ upper triangular. Then we have obtained a reduced Schur decomposition $\mathbf{AM}U = UT$ with $T = T_{k\times k} = RD_1R^{-1}$ upper triangular. Clearly T and D_1 have the same eigenvalues.

To deflate the smallest k eigenvalues by shifting them by $+1$, we propose the following level two preconditioner

$$\mathbf{A}_2 = \mathbf{A}M_2 \qquad \text{with} \qquad M_2 = \mathbf{M} + \mathbf{M}UT^{-1}U^H. \tag{20}$$

From $U^HU = I_{k\times k}$ and $\mathbf{A}_2U = \mathbf{AM}U + \mathbf{AM}UT^{-1}U^HU = UT + UTT^{-1} = U(T+I)$, we see that $\lambda_j(\mathbf{A}_2) = \lambda_j(\mathbf{AM}) + 1$ for $j = 1, \ldots, k$. To prove that the remaining eigenvalues are identical, one may use an orthogonal complements idea relating to a upper triangular matrix (e.g. see [19]).

4.2.2 Method 2

Here we consider a dual orthogonalisation idea. As the preconditioned matrix $\mathbf{AM}$ is nonsymmetric, let $\mathbf{V}_r = [x_1, \ldots, x_k]$ be the matrix of the k right eigenvectors and $\mathbf{V}_l = [y_1, \ldots, y_k]$ be that of the k left eigenvectors, corresponding to the k smallest eigenvalues of $\mathbf{AM}$. Then $\mathbf{V}_l$ and $\mathbf{V}_r$ are orthogonal to each other [36, 9] i.e.

$$\mathbf{V}_l^H\mathbf{V}_r = \Lambda = diag(y_1^Hx_1, \cdots, y_k^Hx_k) \tag{21}$$

and $S = \mathbf{V}_l^HA\mathbf{V}_r = \mathbf{V}_l^H\mathbf{V}_rD_1 = \Lambda D_1$ is a diagonal matrix (although $\mathbf{V}_r^H\mathbf{V}_r \neq I$). Note that the above equation reduces to a familiar result for a symmetric case (as $\mathbf{V}_l = \mathbf{V}_r$).

To deflate the smallest k eigenvalues by shifting them by $+1$, we can define a second level two preconditioner (following [9])

$$\mathbf{A}_2 = \mathbf{A}M_2 \qquad \text{with} \qquad M_2 = \mathbf{M} + \mathbf{M}\mathbf{V}_rS^{-1}\mathbf{V}_l^H. \tag{22}$$

From $\mathbf{V}_l^H\mathbf{V}_r = S$ and $\mathbf{A}_2\mathbf{V}_r = \mathbf{AM}\mathbf{V}_r + \mathbf{AM}\mathbf{V}_rS^{-1}\mathbf{V}_l^H\mathbf{V}_r = \mathbf{V}_rD_1 + \mathbf{V}_rD_1S^{-1}\Lambda = \mathbf{V}_r(D_1+I)$, we see that $\lambda_j(\mathbf{A}_2) = \lambda_j(\mathbf{AM}) + 1$ for $j = 1, \ldots, k$. To prove that the remaining eigenvalues are identical, one may use the matrix of remaining eigenvectors using the dual orthogonality (e.g. see [9]).

Remark 4.2. *Finally, we comment on the implementation of the level two preconditioner M_2 for both Method 1 and Method 2. As M_2 is an approximate inverse type preconditioner, it is not necessary to form M_2 explicitly and one simply uses $M, T, S, U, \mathbf{V}_l, \mathbf{V}_r$ explicitly in matrix vector products. It should be remarked that it is also possible to estimate quantities like $T, S, U, \mathbf{V}_l, \mathbf{V}_r$ using Ritz or harmonic Ritz vectors [29]. We will not pursue this route further as not all of the deflated k eigenvalues (using Ritz vectors) are the smallest eigenvalues of matrix AM.*

5 Numerical Experiments

We illustrate the performance of preconditioned GMRES for five cases of power systems (the number of nonzeros in $\mathbf{A}$ and $\mathbf{M}$, nnz($\mathbf{A}$) and nnz($\mathbf{M}$), will be given in Table 2):

1. IEEE 300: 231 load buses and 68 generator buses. $\mathbf{A}$ nonsymmetric, $n = 530$ and nz=3736;
2. IEEE 662: 589 load buses and 72 generator buses. $\mathbf{A}$ nonsymmetric and $n = 1250$;
3. NYISO 3357: 2810 load buses and 546 generator buses[1]. $\mathbf{A}$ nonsymmetric and $n = 6166$.
4. GEMAT 11: 2400 bus system (from Western USA) as available in Matrix Market[2]. $\mathbf{A}$ nonsymmetric (initial Jacobian) and $n = 4929$.
5. GEMAT 12: 2400 bus system (from Western USA) as available in Matrix Market. $\mathbf{A}$ nonsymmetric (later Jacobian in a power solution) and $n = 4929$.

Note that the cases of no preconditioning, simple banded or ILU(0) type preconditioners have also been tested and found to give no or extremely slow convergence.

We compare the performance of the preconditioned matrices $\mathbf{AM}$ and $\mathbf{A}M_2$. As the performance of Methods 1 and 2 is similar, we only show the results from Method 1. The left half of Figure 1 shows the Jacobian matrix for the IEEE 300 bus system. The pattern of the matrix $\mathbf{M}$ on the right is obtained, in this case, from first performing a sparcification on $\mathbf{A}$ column-wise by setting $\alpha = mean(A_i - a_{ii}),\ i = 1, \ldots, n$. The choice of α is made to ensure a good sparsity pattern for $\mathbf{M}$. The left half of Figure 2 shows the clustered spectrum. Notice the trail of eigenvalues getting very close to the origin. We *remove* a few of these eigenvalues by using the deflation technique described in §4.2 with $k = 10$. This is shown in the right half of Figure 2. Notice how this effects the convergence performance of GMRES in the left plot of Figure 3. The effects of using the deflation technique were even more prominent when tested on the NYSIO 3357, GETMAT 11 and GEMAT 12 test systems (as can be seen from the right plot of Figure

[1]The authors wish to thank Professor Ian Dobson (University of Wisconsin-Madison, USA) for providing us the test data of this New York bus system.

[2]Publicly available from the web site *http://math.nist.gov/MatrixMarket*

System	Deflation	IEEE 300	IEEE 662	NYISO 3357	GEMAT11	GEMAT12
n=size($\mathbf{A}$)		530	1250	6166	4929	4929
nnz($\mathbf{A}$)		3736	8746	56680	33185	33111
nnz($\mathbf{M}$)		18464	33242	202530	114125	116872
$nzmax$		35	35	65	65	65
GMRES(∞)	No	36	40	283	180	X
	1	30	32	251	126	X
	10	14	20	142	77	X
	65	10	12	35	34	35
GMRES(20)	No	70	79	X	X	X
	1	42	36	X	X	X
	10	14	20	X	X	X
	65	10	12	38	46	54

Table 2: Convergence performance of GMRES for various deflation options (here "$nzmax$": the maximal number of nonzeros allowed in $\mathbf{M}$, "No": no deflation, "Deflation k": deflation of k smallest eigenvalues by Method 2 and "X": no convergence). The performance of Methods 1 and 2 is almost identical, only differing in computation.

3 for NYSIO 3357). For the GEMAT 11 and 12 cases, we have used the scaling algorithm from [18] to scale $\mathbf{A}$ before using the two-level methods. The results for all test systems are summarised in Table 2 and show the number of iterations required for GMRES(∞) and GMRES(20) with $tol = 10^{-5}$. Each test system is tested with and without sparcification. In all cases, performance of a method can be further enhanced if $nzmax$ is increased and there is a need to address the right balance of size $nzmax$ and the number of deflated smallest eigenvalues.

Clearly one observes that the sparse approximate inverse based preconditioners are effective for our linear systems arising from power equations and the deflation technique can further speed up the iterations. Alternative methods of and further improvements to the implementation of the deflation idea will be future work. Other further experiments may include re-use of a preconditioner from a previous step for remaining Newton iterates.

6 Conclusion

In this paper, we first surveyed methods for estimating the fold bifurcation parameter for the power flow equations by a test function method using two Newton power flows, without the need of computing to the point using many nonlinear continuation steps. We then showed how to improve the speed at each Newton step for large power system networks using iterative methods, in this case GMRES. We introduced a class of two-level preconditioners based on sparse approximate inverses with a priori pattern selection and deflation techniques.

References

[1] J.P. Abbott. *Numerical continuation methods for nonlinear equations and bifurcation problems.* PhD thesis, Australian National University, 1977.

[2] J.P. Abbott. An efficient algorithm for the determination of certain bifurcation points. *J. Comput. Appl. Math*, 4(1):19–27, 1978.

[3] M. W. Benson and P. O. Frederickson, *Iterative solution of large sparse linear systems arising in certain multidimensional approximation problems*, Utilitas Math., 22 (1982), pp.127-140.

[4] M. Benzi, J. C. Haws and M. Tuma, *Preconditioning highly indefinite and nonsymmetric matrices*, SIAM J. Sci. Comput., 22 (2000), pp.1333-1353.

[5] W. J. Beyn, A. Champneys, E. Doedel, W. Govaerts, Yu. A. Kuznetsov, and B. Sandstede. Numerical continuation and computation of normal forms. In B. Fiedler, editor, *Handbook of Dynamical Systems*, volume 2, pages 149–219. Elsevier Science, North-Holland, 2002.

[6] K. Burrage and J. Erhel, On the performance of various adaptive preconditioned GMRES strategies. Numer. Lin. Alg. Applics., 5:101-121, 1998.

[7] C. A. Canizares, A.Z. de Souza, and V.H. Quintana. Comparison of performance indices for detection of proximity to voltage collapse. *IEEE Transactions on Power Systems*, 11(3):1441–1450, August 1996.

[8] B. Carpentieri, I. S. Duff and L. Giraud, *Some sparse pattern selection strategies for robust Frobenius norm minimization preconditioners in electromagnetism*, Numer. Lin. Alg. Appl., 7 (2000), pp.667-685. (also RAL-TR-2000-009, http://www.numerical.rl.ac.uk/reports/)

[9] B. Carpentieri, I.S. Duff, and L. Giraud. A class of spectral two-level preconditioners. *SIAM J. Sci. Comp.*, to appear, 2003. Technical Report TR/PA/02/55, CERFACS, Toulouse, France, 2002.

[10] K. Chen, A. Hussein, and H. Wan. On a class of new and practical performance indices for approximation of fold bifurcations of nonlinear power flow equations. *J. Comput. Appl. Math*, 140:119-141, 2002.

[11] H.D. Chiang, , and A.J. Fleuck. Solving the nonlinear power flow equations with an inexact Newton method using GMRES. *IEEE Transactions on Power Systems*, 13(2):267–273, May 1998.

[12] H.D. Chiang and R. Jean-Jumeau. Toward a practical performance index for predicting voltage collapse in electric power systems. *IEEE Transactions on Power Systems*, 10(2):584–592, May 1995.

[13] H.D. Chiang, C.S. Wang, and A.J. Fleuck. Look-ahead voltage and load margin contingency selection functions for large-scale power systems. *IEEE Transactions on Power Systems*, 12(1):173–180, February 1997.

[14] E. Chow. A priori sparsity patterns for parallel sparse approximate inverse preconditioners. *SIAM J. Sci. Comp.*, 21:1804–1822, 2000.

[15] A. Z. de Souza, C. A. Canizares, and V.H. Quintana. New techniques to speed up voltage collapse computations using tangent vectors. *IEEE Transactions on Power Systems*, 12(3):1380–1387, August 1997.

[16] I. Dobson, H. D. Chiang, J. S. Thorp, and L. Fekih-Ahmed. A model for voltage collapse in power systems. In *Proc. of the 27th IEEE conference on decision and control*, pages 2104–2109, Austin, TX, Dec. 1988.

[17] E. Doedel. *Lecture notes on Numerical Analysis of Bifurcation Problems.* Hamburg, March 1997; from ftp://ftp.cs.concordia.ca/pub/doedel/doc/hamburg.ps.Z

[18] I. S. Duff and J. Koster, *The design and use of algorithms for permuting large entries to the diagonal of sparse matrices*, SIAM J. Matr. Anal. Appl., 20 (1999), pp.889-901. (also RAL-TR-1999-030, http://www.numerical.rl.ac.uk/reports/)

[19] J. Erhel, K. Burrage and B. Pohl. Restarted GMRES preconditioned by deflation. *J. Comput. Appl. Math*, 69:303-318, 1996.

[20] W. Govaerts. *Numerical Methods for Bifurcations of Dynamical Equilibria.* SIAM, Philadelphia, 2000.

[21] W. J. F. Govaerts. Bordered matrices and singularities of large nonlinear systems. *Int. J. Bifurcation and Chaos*, 5:243–250, 1995.

[22] J.J. Grainger and W.D. Stevenson. *Power System Analysis.* McGraw-Hill, 1994.

[23] M. J. Grote and T. Huckle. Parallel preconditioning with sparse approximate inverses. *SIAM J. Sci. Comp.*, 18(3):838–853, 1997.

[24] T. Huckle. Approximate sparsity patterns for the inverse of a matrix and preconditioning. *Appl. Numer. Math.*, 30:291–303, 1999.

[25] A. Hussein, Fast computational methods for power flow bifurcation problems, *Ph.D thesis*, University of Liverpool, UK, 2003.

[26] A. Hussein, K. Chen, and H. Wan. On adapting test function methods for fast detection of fold bifurcations in power systems. *Int. J. Bifurcation and Chaos*, 12:179-186, 2001.

[27] H.B. Keller. *Numerical Methods in Bifurcation Problems.* Springer-Verlag, 1987.

[28] Kolotilina L. Y., Nikishin A. A. and Yeremin A. Y., *Factorized sparse approximate inverse preconditioners IV: simple approaches to rising efficiency*, Numer. Lin. Alg. Applcs., 6 (1999), pp.515-531.

[29] R. B. Morgan, GMRES with deflated restarting, *SIAM J. Sci. Comp.*, 24(1):20–37, 2002.

[30] G. Moore and A. Spence. The calculation of turning points of nonlinear equations. *SIAM J. Numer. Anal.*, 17:567–576, 1980.

[31] Y. Saad and M.H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Comp.*, 7(3):856–869, 1986.

[32] R. Seydel. Numerical computation of branch points in non-linear equations. *Numerische Mathematik*, 33:339–352, 1979.

[33] R. Seydel. *Practical Bifurcation and Stability Analysis. From Equilibrium to Chaos*, volume 5 of *Interdisciplinary Applied Mathematics.* Springer-Verlag, New York, 2nd edition, 1994.

[34] R. Seydel. Assessing voltage collapse. *Latin American Applied Research*, 31(3):171–176, 2001.

[35] B. Stott and O. Alsac. Fast decoupled load flow. *IEEE Transactions on Power Apparatus and Systems*, 93:859–869, 1974.

[36] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Oxford University Press, UK, 1965.

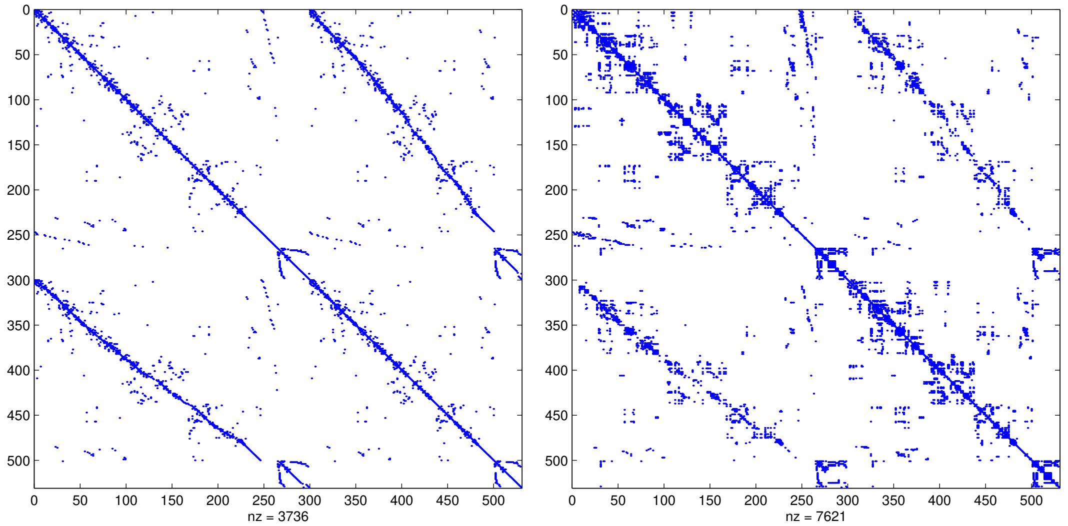

Figure 1: Matrix structure for the IEEE 300. The left half of the figure shows the original Jacobian $\mathbf{A}$ while the right half of the figure shows the sparcified approximate inverse $\mathbf{M}$.

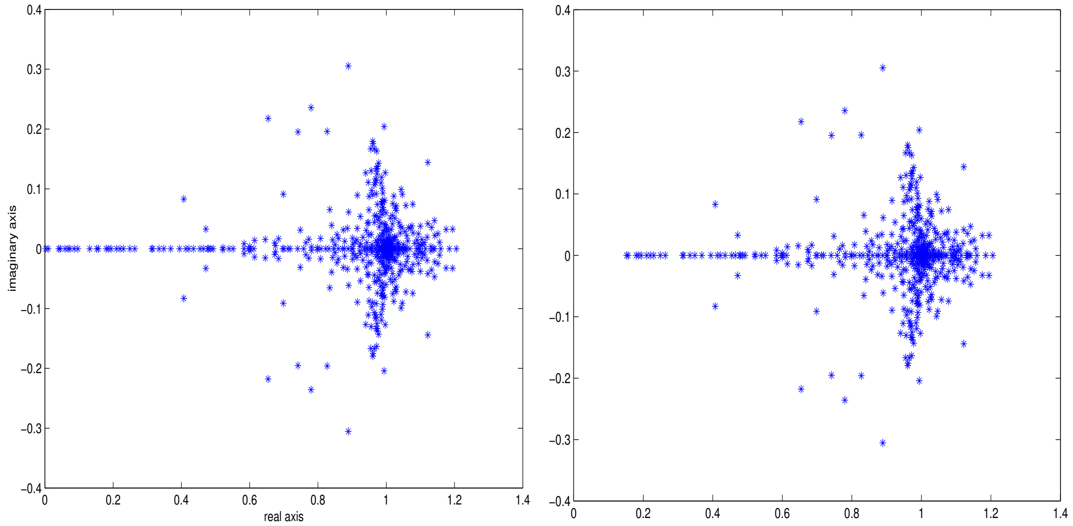

Figure 2: Spectrum of the preconditioned IEEE 300 Jacobian. The left half of the figure is for $\mathbf{AM}$ while the right half of the figure is for $\mathbf{A}M_2$ with $k = 10$ (Method 2).

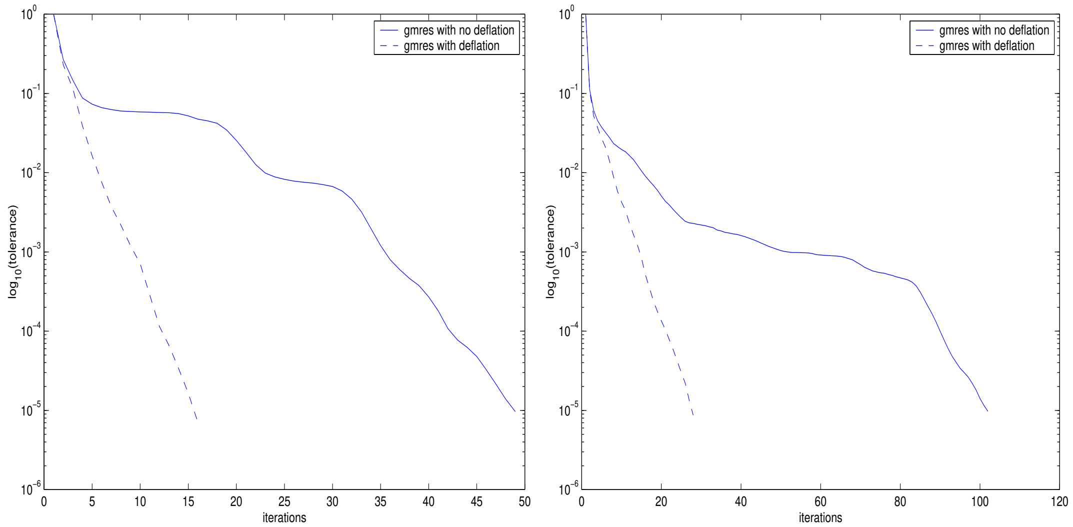

Figure 3: Performance of GMRES for IEEE 300 (left) and for NYSIO 3357 (right plot).