Title	A method for direct calculation of quadratic turning points
Author(s)	Yan, Z; Liu, Y; Wu, F; Ni, Y
Citation	IET Generation, Transmission and Distribution, 2004, v. 151 n. 1, p. 83-89
Issued Date	2004
URL	http://hdl.handle.net/10722/73586
Rights	Creative Commons: Attribution 3.0 Hong Kong License

Method for direct calculation of quadratic turning points

Z. Yan, Y. Liu, F. Wu and Y. Ni

Abstract: For a given one-parameter nonlinear system, the simplest bifurcation is the quadratic turning bifurcation where the Jacobian matrix becomes singular due to rank deficiency 1. To overcome the difficulty in solving the quadratic turning point caused by the singularity of the Jacobian matrix, the conventional Newton method can be applied to the so-called Moore-Spence determination system to solve for the quadratic turning point. However, the Moore-Spence system has much higher dimensions and causes much more complexity in factorisation of the extended Jacobian matrix. In the paper, by introducing an auxiliary variable and an auxiliary linear equation into Newton iterations in solving the Moore-Spence determination system, a matrix reduction technique can be worked out to solve the Moore-Spence extended equations much more efficiently. The high dimensions of the matrix can thus be reduced and the complexity involved in matrix factorisation can be reduced noticeably. The technique is proposed for general nonlinear systems. Formulation is derived for applying this technique to solving quadratic turning points, or say nose points, on load-flow solution curves of power systems. Computer tests on the IEEE 30-busbar system and a 2416-busbar East China power system are reported to show the effectiveness of the suggested technique.

1 Introduction

Many real world problems can be abstracted as nonlinear systems, in the form of either differential equations and/or algebraic equations. Free parameters may be presented in a nonlinear system in order to study the influence of parameter changes on the system behaviour. In analysing the overall behaviour of a nonlinear system containing free parameters, a commonly encountered phenomenon is bifurcation, where any small change of one parameter may induce completely different behaviours representing loss of structural stability of the system. Calculation of bifurcation points is of extreme importance in comprehensive understanding of the behaviours of the studied systems [1, 2].

The simplest bifurcation phenomenon is the occurrence of quadratic turning bifurcation attributed to nonlinear systems containing only one free parameter. The theoretical significance of calculating quadratic turning bifurcation points is in that it constitutes the basis for investigating more complicated bifurcation phenomena, and it is a bridge leading to studying more complex systems containing multiple parameters. This paper is devoted to the calculation of quadratic turning points (QTP) with applications to power systems.

A major difficulty in calculating turning points is that the Jacobian matrix of the nonlinear system becomes singular at these points, and, hence, the efficient Newton method cannot be applied directly. The current methods for computing QTPs can be roughly divided into two categories [1]: indirect methods and direct methods. Indirect methods detect the neighbouring points of a turning point based on certain kinds of test functions and then approximate the turning point through algebraic interpolation or extrapolation. An indirect method cannot reach the bifurcation point exactly and, hence, results in an approximate solution.

Direct methods treat a turning point as the solution of a so-called determination system, whose extended Jacobian matrix at the turning point is not singular, and thus the difficulty of the original Jacobian singularity is overcome. The Moore-Spence extended system is one of the most frequently used determining systems [1, 2, 3]. Once a neighbour point of the QTP has been reached, it can be solved accurately by applying Newton method to the corresponding determination system. Both the indirect and direct methods require certain kinds of initialisation that can be accomplished by calculating the regular points along the solution curve through usual continuation methods.

In power system voltage stability analysis, a typical parameter of interests is the system loading factor; while, in available transfer capability (ATC) calculation, a typical parameter of interests is the transfer power from sending area (one generator busbar or a group of generator busbars) to receiving area (one load busbar or a group of load busbars) [4–6]. Turning bifurcation points are frequently encountered in these studies and the QTP has been named as limit point, nose point, saddle-node point, maximum loadability point or point of collapse etc. in power system literatures [4–15]. Both direct and indirect methods have been applied to the QTP solution in power systems.

In [7], voltage instability phenomenon is observed in calculation of multiple power flow solutions while increasing loading factors. It is found that, with the increase of load, two equilibria of the power flow equations coalesce to

IEE Proceedings online no. 20030940

doi:10.1049/ip-gtd:20030940

Paper first received 10th February 2003 and in revised form 21st July 2003. Online publishing date: 4 November 2003

Z. Yan, F. Wu and Y. Ni are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong SAR, The People's Republic of China

Y. Liu is with the College of Electrical Power Engineering, South China University of Technology, Guangzhou, The People's Republic of China

each other at a critical point and then disappear after it. In [8–12], continuation power flow (CPF) methods are applied to systematically trace the power flow solution curve with respect to the changing parameter. The continuation method traces the solution curve by a series of predictor-corrector steps. Each step is embedded with a parameterisation procedure. In the CPF method, the number of equations is the same as the number of unknowns, and the Jacobian singularity can be overcome properly. In [9], a comprehensive production-stage package named CPFOLW, for tracing the power flow solution curve, is developed. CPF method is used together with the arc-length parameterisation technique. Although CPFLOW can automatically trace the power flow solution curve without failure at the turning point, it cannot calculate this point directly and is still an indirect method.

In [10–12], the QTP is named as point of collapse (POC) to reflect its physical nature in the voltage stability problem. The so-called Moore-Spence determination system (MSDS) is formed and an accurate POC is solved accordingly. The method is known as the POC method and can solve directly the extended MSDS to obtain the POC by the traditional Newton method. Special sparse matrix solver is required in implementation due to the high dimensions and the complexity of the extended Jacobian matrix.

The turning point calculation can also be formulated as an optimisation problem and be solved by optimisation methods [4, 13, 14, 15]. Similar to the POC and MSDS methods mentioned here, the optimisation method may encounter difficulty in identifying binding inequalities. Moreover, the information along the solution curve is lost because such a curve is never calculated in optimisation methods.

In this paper an efficient matrix reduction method is proposed to solve the MSDS efficiently. In the new method, the linear system derived from the Moore-Spence system can be decomposed into two smaller subsystems through introduction of one auxiliary variable and one auxiliary equation within the Newton iterations. The technique can be applied to general nonlinear systems to calculate QTPs. Formulas are presented to apply this technique to solving QTP, or say nose point, on load flow solution curves of power systems. Computer tests on the IEEE 30-busbar system and an East China 2416-busbar power system show the effectiveness of the suggested technique.

2 Mathematical fundamentals

2.1 Quadratic turning bifurcation point definition

Consider the following dynamical system [2]:

$$\dot{x} = f(x, \lambda), \; f: U \times R \to V \tag{1}$$

where $U \subset R^n$, $V \subset R^n$, and λ is a free parameter of particular interests. Let $(x(\lambda), \lambda)$ be a time-invariant solution curve, i.e. any point on this curve satisfies $f(x(\lambda), \lambda) = 0$.

Definition 1: (x, λ) is called a turning point, if there exist two nonzero vectors $\varphi \in U$, $\psi \in V$, such that

$$f(x, \lambda) = 0 \tag{2a}$$

$$N(f_x) = \text{span}\{\varphi, \; \varphi \neq 0\} \tag{2b}$$

$$R(f_x) = \{v \in V, \; \psi^T v = 0, \; \psi \neq 0\} \tag{2c}$$

(x, λ) is called a QTP if we also have

$$\psi^T f_\lambda \neq 0 \tag{2d}$$

$$\psi^T f_{xx} \varphi \varphi \neq 0 \tag{2e}$$

Furthermore, (x, λ) is called simple QTP, if (2*f*) is satisfied,

$$\psi^T \varphi \neq 0 \tag{2f}$$

where f_x and f_λ are the first derivatives of f with respect to x and λ, respectively; f_{xx} denotes the second derivative of f with respect to x; and φ and ψ are the right and left eigenvectors of the Jacobian matrix f_x, respectively. The condition (2*f*) means the zero eigenvalue of f_x is a simple eigenvalue.

2.2 Brief review on Moore-Spence determining system

The MSDS was developed in [3] and used to solve for QTPs with an exact mathematical formulation. From the viewpoint of numerical methods, the Jacobian matrix f_x becomes singular at the turning points. Hence, Newton methods cannot be employed to solve the algebraic equation $f(x, \lambda) = 0$ directly under ill conditions when the parameter λ is near its bifurcation value. For a real nonlinear system, such as a power system, a key task in calculating the solution curve with respect to parameter λ change is to overcome the singularity of the Jacobian at the turning point. An effective way for overcoming the singularity problem of the Jacobian matrix is to construct the Moore-Spence system defined at the turning point, i. e. to choose an arbitrary vector $l \in R^n$ with $l^T \varphi \neq 0$ and to construct the following extended system:

$$f(x, \lambda) = 0 \tag{3a}$$

$$f_x(x, \lambda) y = 0 \tag{3b}$$

$$l^T y = 1 \tag{3c}$$

where $y \in R^n$ is a right singular vector. This is a system of $2n+1$ equations with $2n+1$ unknowns x, λ, y. The following theorem depicts the property of (3).

Theorem 1: (x, λ) is a QTP, if and only if (x, λ, y) is a regular solution of (3).

Proof: The proof of this theorem can be found in [2, 3].

Remarks:

1. Theorem 1 shows that the following Jacobian:

$$M = \begin{pmatrix} f_x & f_\lambda & 0 \\ f_{xx} y & f_{x\lambda} y & f_x \\ 0 & 0 & l^T \end{pmatrix} \tag{4}$$

of (3) must be nonsingular at the QTP, even though the Jacobian matrix f_x is singular at that point. This means that the Moore-Spence system can overcome the singularity of f_x at the QTP. The '0' in (4) is of proper dimension, i. e. it may be a scalar, a proper zero row or column vector or a zero matrix, respectively, depending on its location in the matrix. And this notation will be adopted throughout this paper.

2. If any of the conditions (2*d*) or (2*e*) is not satisfied, then the Jacobian matrix M of (3) will become singular, and, hence, (3) cannot be solved by Newton method. Theorem 1 clearly asserts that any QTP can be solved from (3) by Newton method, once a suitable initial point is given, and that any solution solved from (3) by Newton method must be a QTP.

3. Equation (3) can be applied to solve for QTP where the Jacobian matrix f_x has rank deficiency 1 indicated in (3*b*). However, (3) cannot be used to compute the regular points in the solution curve where f_x has full rank, because (3*b*) conflicts with (3*c*).

4. In references [11] and [12], the same formulation has been used in the context of voltage stability analysis and the associated direct solution method is named as POC (point of collapse) method.

5. A set of equations with similar structure as (3) can be obtained by solving an optimisation problem built around the QTP. In this case, (3*b*) will be replaced, equivalently, by

$$w^T f_x(x, \lambda) = 0 \tag{3b'}$$

where the *w*-vector is a left singular vector and can be interpreted as the Lagrangian multiplier vector corresponding to (3*a*). This formulation may be more useful in practice, because it contains information for sensitivity analysis. The details in this aspect can be found in [4, 13, 15].

6. In a variant of Moore-Spence system, (3*c*) is replaced by

$$y^T y = 1 \tag{3c'}$$

Theorem 1 also holds for this system [2, 3].

2.3 Direct solution of the quadratic turning point by solving the Moore-Spence determination system

The MSDS provides a direct way to calculate the QTP, due to its sound and clear mathematical basis. When Newton method is employed to solve (3), we have to solve a set of linear equations at each Newton iteration step. This set of linear equations can be written as (for clarity of expression, the Newton iteration superscripts are omitted here):

$$\begin{pmatrix} f_x & f_\lambda & 0 \\ f_{xx}y & f_{x\lambda}y & f_x \\ 0 & 0 & l^T \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta\lambda \\ \Delta y \end{pmatrix} = -\begin{pmatrix} f(x,\lambda) \\ f_x y \\ 1 - l^T y \end{pmatrix} \tag{5}$$

It should be noted that the dimension of the coefficient matrix of (5) is $(2n+1)\times(2n+1)$. For a large-scale power system with N busbars, the dimension of this coefficient matrix is $(4N+1)\times(4N+1)$, with a solution that may be time consuming and may create additional complexity to the program. In this paper, a novel matrix reduction algorithm is proposed to solve (5) efficiently.

3 Novel matrix reduction method for solving MSDS

The key point of the proposed method is to expand (4) by introducing one auxiliary variable μ and adding one auxiliary equation of the following form to (5):

$$l^T \Delta x - \mu = 0 \tag{6}$$

Insert (6) into (5) as the second block row and expand the Jacobian matrix by one column corresponding to the auxiliary variable μ, we obtain the following equivalent equation:

$$\begin{pmatrix} f_x & f_\lambda & 0 & 0 \\ l^T & 0 & 0 & 0 \\ f_{xx}y & f_{x\lambda}y & f_x & f_\lambda \\ 0 & 0 & l^T & 0 \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta\lambda \\ \Delta y \\ \mu \end{pmatrix} - \begin{pmatrix} 0 \\ 1 \\ f_\lambda \\ 0 \end{pmatrix} \mu = -\begin{pmatrix} f(x,\lambda) \\ 0 \\ f_x y \\ 1 - l^T y \end{pmatrix} \tag{7}$$

The significant features of (7) as compared with (5) are that the upper right (2×2) block matrix is a zero matrix, and that the two (2×2) block diagonal matrices are equal to each other, which leads to the following efficient way to calculate the QTP:

Defining matrices A and B as

$$A = \begin{pmatrix} f_x & f_\lambda \\ l^T & 0 \end{pmatrix}, \; B = \begin{pmatrix} f_{xx}y & f_{x\lambda}y \\ 0 & 0 \end{pmatrix} \tag{8}$$

Then (7) can be decomposed into two equations:

$$A\begin{pmatrix} \Delta x \\ \Delta\lambda \end{pmatrix} - \begin{pmatrix} 0 \\ 1 \end{pmatrix}\mu = -\begin{pmatrix} f(x,\lambda) \\ 0 \end{pmatrix} \tag{9a}$$

and

$$A\begin{pmatrix} \Delta y \\ \mu \end{pmatrix} + B\begin{pmatrix} \Delta x \\ \Delta\lambda \end{pmatrix} - \begin{pmatrix} f_\lambda \\ 0 \end{pmatrix}\mu = -\begin{pmatrix} f_x y \\ 1 - l^T y \end{pmatrix} \tag{9b}$$

Equation (9) shows that if matrix A is nonsingular, then Δx and $\Delta\lambda$ can be obtained from (9*a*) once the auxiliary variable μ is available, and Δy can thus be calculated from (9*b*). Fortunately, the following proposition guarantees that matrix A is nonsingular. With this property, a straightforward way to calculate μ can be derived.

Proposition 1: Matrix A is nonsingular at the QTP.

Proof: See the Appendix, Section 9.

Now that matrix A is nonsingular, (9*a*) can then be rewritten as

$$\begin{pmatrix} \Delta x \\ \Delta\lambda \end{pmatrix} = \begin{pmatrix} r_x \\ r_\lambda \end{pmatrix} - \begin{pmatrix} s_x \\ s_\lambda \end{pmatrix}\mu \tag{10a}$$

where $r_x \in R^n$, $r_\lambda \in R$, $s_x \in R^n$ and $s_\lambda \in R$ are solutions of the following equations:

$$A\begin{pmatrix} r_x \\ r_\lambda \end{pmatrix} = -\begin{pmatrix} f(x,\lambda) \\ 0 \end{pmatrix} \tag{10b}$$

$$A\begin{pmatrix} s_x \\ s_\lambda \end{pmatrix} = -\begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{10c}$$

Substitute (10*a*) into (9*b*), we have

$$A\begin{pmatrix} \Delta y \\ \mu \end{pmatrix} + B\begin{pmatrix} r_x \\ r_\lambda \end{pmatrix} - B\begin{pmatrix} s_x \\ s_\lambda \end{pmatrix}\mu - \begin{pmatrix} f_\lambda \\ 0 \end{pmatrix}\mu = -\begin{pmatrix} f_x y \\ 1 - l^T y \end{pmatrix}$$

which can be rearranged as

$$\begin{pmatrix} \Delta y \\ \mu \end{pmatrix} = \begin{pmatrix} r_y \\ r_\mu \end{pmatrix} - \begin{pmatrix} s_y \\ s_\mu \end{pmatrix}\mu \tag{11a}$$

where $r_y \in R^n$, $r_\mu \in R$, $s_y \in R^n$ and $s_\mu \in R$ are solutions of the following equations:

$$A\begin{pmatrix} r_y \\ r_\mu \end{pmatrix} = -\begin{pmatrix} f_x y \\ 1 - l^T y \end{pmatrix} - B\begin{pmatrix} r_x \\ r_\lambda \end{pmatrix} \tag{11b}$$

$$A\begin{pmatrix} s_y \\ s_\mu \end{pmatrix} = -\begin{pmatrix} f_\lambda \\ 0 \end{pmatrix} - B\begin{pmatrix} s_x \\ s_\lambda \end{pmatrix} \tag{11c}$$

From the last equation of (11*a*) the auxiliary variable μ can be calculated as

$$\mu = \frac{r_\mu}{1 + s_\mu} \tag{12}$$

By substituting μ into (10*a*) and (11*a*) we finally get the solution of Δx, $\Delta\lambda$ and Δy. The aforementioned algorithm for each iteration of Newton's method can be implemented

through following procedure:

Step 1. Calculate $f(x, \lambda)$, f_x, f_λ, $f_{xx}y$, and $f_{x\lambda}y$ at initial point $(x^{(0)}, \lambda^{(0)}, y^{(0)})$ (Section 4 will discuss how to obtain the initial values in real application.)

Step 2. Form matrix A and B and factorise matrix A and store its factor table.

Step 3. Solve (10b) and (10c) for $\begin{pmatrix} r_x \\ r_\lambda \end{pmatrix}$ and $\begin{pmatrix} s_x \\ s_\lambda \end{pmatrix}$ by using the factor table of matrix A.

Step 4. Solve (11b) and (11c) for $\begin{pmatrix} r_y \\ r_\mu \end{pmatrix}$ and $\begin{pmatrix} s_y \\ s_\mu \end{pmatrix}$ by using the factor table of matrix A and solved $\begin{pmatrix} r_x \\ r_\lambda \end{pmatrix}$ and $\begin{pmatrix} s_x \\ s_\lambda \end{pmatrix}$.

Step 5. Calculate the auxiliary variable μ from (12).

Step 6. Substitute μ into (10a) and (11a) to calculate the current Newton increment step of Δx, $\Delta\lambda$, Δy.

Remarks:

1. The auxiliary variable μ and the auxiliary equation (6) are introduced in each Newton iteration, which does not alter the original MSDS (3). Their function is to facilitate the solution of the co-ordination equation in Newton iteration. Once we have finished using (10a) and (11a) to calculate $\begin{pmatrix} \Delta x \\ \Delta\lambda \end{pmatrix}$ and $\begin{pmatrix} \Delta y \\ \mu \end{pmatrix}$, μ will be discarded. The convergence of the Newton iterative process is checked by the original Moore-Spence system and is independent to the auxiliary equation.

2. In each Newton iteration, only the $(n+1) \times (n+1)$ matrix A is to be factorised, which is much smaller in size than the $(2n+1) \times (2n+1)$ Jacobian matrix of the original Moore-Spence system. CPU time can be significantly reduced for large-scale problems.

3. At each Newton iteration, the matrix A is only factorised for one time and its triangular factor tables are repeatedly called for four times in the forward and backward substitutions, for solving the four linear equations (10b), (10c), (11b) and (11c) that have matrix A as their coefficient matrix. The programming complexity has been simplified greatly and the triangular factorisation is extremely efficient and reliable.

4. The proposed technique is designed for general nonlinear systems to calculate the QTPs. The technique is particularly suitable to large-scale power systems, because the number of power flow equations is twice as many as the number of system busbars, and the latter might be in thousands.

5. Matrices $f_{xx}y$ and $f_{x\lambda}y$, that are relatively complicated in evaluation, do not appear in matrix A. Their influence on the solution of (5) is reflected in the right-hand side of equations (11b) and (11c), where only simple multiplication of sparse matrices with vectors is involved. This property removes the complexity incurred in direct factorisation of the coefficient matrix of (5).

4 Applications to power systems

4.1 MSDS model for QPTs on power flow solution curve

In power flow study and voltage stability analysis, a cluster of problems of increasing interests is to trace the power flow solution curve and find its QTP with respect to a specific parameter change. For a given power system containing N busbars, the power flow equation, corresponding to (3a), can be expressed as:

$$\begin{aligned} P(V, \theta) &= P^{(0)} + \lambda b_P \\ Q(V, \theta) &= Q^{(0)} + \lambda b_Q \end{aligned} \tag{13}$$

where P, $Q \in R^N$ are real and reactive power equations, expressed as functions of busbar voltage magnitudes ($V \in R^N$) and phase angles ($\theta \in R^N$). $P^{(0)}$, $Q^{(0)} \in R^N$ are real and reactive-power net injections in base case, b_P, $b_Q \in R^N$ are predefined change pattern for real and reactive power injections, λ is a scalar parameter reflecting the change along the predefined pattern.

Equation (13) is a unified form for various research purposes. The change pattern vectors b_P and b_Q can be constructed in various different ways to emphasise the research focus. The maximum value of λ in equation (13) is its bifurcation value at the QTP, where the eigenvector $y = (y_P^T \; y_Q^T)^T$, y_P, $y_Q \in R^N$, corresponding to the zero eigenvalue of the Jacobian matrix will satisfy

$$\begin{pmatrix} \frac{\partial P}{\partial \theta} & \frac{\partial P}{\partial V} \\ \frac{\partial Q}{\partial \theta} & \frac{\partial Q}{\partial V} \end{pmatrix} \begin{pmatrix} y_P \\ y_Q \end{pmatrix} = 0 \tag{14}$$

Equation (14) corresponds to (3b). To obtain a unique solution of the eigenvector, the following equation (corresponding to (3c)) is imposed:

$$l_P^T y_P + l_Q^T y_Q = 1 \tag{15}$$

where l_P, $l_Q \in R^N$ and $(l_P^T, l_Q^T)^T \neq 0$ (l_P and l_Q can be set as a vector composed by elements of 1). Equations (13)–(15) constitute the MSDS to calculate the QTP of power flow solutions.

A variant of (15) with the following form:

$$y_P^T y_P + y_Q^T y_Q = 1 \tag{15'}$$

has also been tested in the research. However, (15) is found to be more reliable than (15)′ and, hence, (15) is adopted in the prototype program.

4.2 Implementation

The technique proposed in Section 3 has been implemented in an existing Newton power flow program. The program has been further enhanced by incorporating a continuation method with natural parameterisation technique, so that (13) can be solved along the solution curve, when the Newton method can converge, and that an initial point for equations (13)–(15) can be obtained for QTP solution using the proposed method, once the Newton method diverges. In the program, the load is treated as a combination of constant power, constant current and constant impedance loads. The base-case power flow corresponds to $\lambda = 0$.

5 Simulation results and discussions

Test results for two systems, i. e. the IEEE 30-busbar system and an East China 2416-busbar system, are reported in this Section. The computer used is a Compaq Presario Mobile with 1200 MHz Intel Pentium-III CPU. The source code of the prototype program was written in Fortran and compiled by Microsoft Fortran PowerStation 4.0. In order to test the method, system constraint violations such as overflow of lines, undervoltage of busbars and generator reactive power limits are ignored. The test target is to verify the efficiency of the method for QTP solution. In all tests, we set $l_p = 1$ and $l_Q = 1$.

5.1 Simulation results of the IEEE 30-busbar system

The IEEE 30-busbar system is composed of 30 busbars and 41 branches. In this test, the generation at generator busbar 2 and the load at load busbar 30 are changed simultaneously for the following predefined pattern:

$$b_P(\text{busbar}-2)=10\,\text{MW},\ b_Q(\text{busbar}-2)=1.79\,\text{Mvar}$$
$$b_P(\text{busbar}-30)=-10\,\text{MW},\ b_Q(\text{busbar}-30)=-1.79\,\text{Mvar}$$

All the other elements of vectors b_P and b_Q are zeros. The step-by-step simulation begins from the base-case power flow with $\lambda=0$, until the power flow diverges at $\lambda=4.75$. The program automatically switches to calling the QTP subroutine to calculate the QTP. Table 1 lists the convergence process of λ_{max} in 5 iterations.

Table 1: Iteration process of λ_{max} at QTP

Iteration	0	1	2	3	4	5
λ_{max}	4.7500	4.9912	4.6919	4.6943	4.6933	4.6931

Table 2 lists major results of busbar voltage magnitudes and phase angles at three points during the continuation process: (1) the starting point (base case) of solution curve with $\lambda=0$; (2) the ending point of solution curve after λ reduced back to 0; (3) the QTP. The Newton power flow program converges very well in the whole computation process, including the ending point, where the voltage of busbar 30 drops to zero, with an exception at the turning point where the MSDS has to be used to get QTP due to a power flow Jacobian singularity. After the turning point is calculated, the program switches back to the step-by-step continuation power flow calculation with decreased λ until $\lambda=0$. The 8th and 9th columns of Table 2 are normalised right eigenvectors $y=(y_P^T\ y_Q^T)^T$ corresponding to the zero eigenvalue of the power flow Jacobian matrix at the QTP. Figures 1 and 2 show the voltage magnitude and phase angle curves of busbars 29 and 30 with respect to λ change, respectively.

The total CPU time for tracing the solution curve is 0.04 s for this case, most of which is used for calculating 40 points along the solution curve by the Newton power flow program, and the time for QTP solution is less than 0.01 s.

5.2 Simulation results of the East China 2416-busbar system

The East China system is composed of six areas, five provincial areas plus one large city (see Fig. 3). Area 1 is the

Table 2: Computational results for IEEE 30-busbar system

Busbar number	Results for $\lambda=0$				Results for $\lambda_{max}=4.6931$			
	Voltage at starting point		Voltage at ending point		Voltage at turning point		Eigenvector of zero eigenvalue	
	V	θ	V	θ	V	θ	y_P	y_Q
1	1.05	0	1.05	0	1.05	0	0	0
2	1.034	−2.73	1.034	−3.44	1.034	−2.63	0.0077	0
3	1.03	−4.65	1.016	−5.61	1.021	−5.75	0.0105	0.0089
4	1.025	−5.58	1.008	−6.75	1.015	−6.92	0.0129	0.0107
5	1.006	−8.99	1.006	−10.23	1.006	−9.91	0.0136	0
6	1.022	−6.47	1	−7.8	1.009	−8.21	0.0147	0.0133
7	1.007	−8.02	0.995	−9.31	1	−9.43	0.0141	0.0078
8	1.023	−6.47	1.023	−8.22	1.023	−8.58	0.0193	0
9	1.046	−8.11	1.017	−10.36	1.033	−10.76	0.025	0.0178
10	1.045	−9.97	0.997	−12.81	1.025	−13.13	0.0314	0.0293
11	1.091	−6.23	1.091	−8.43	1.091	−8.87	0.0244	0
12	1.054	−9.25	1.03	−11.92	1.044	−12.02	0.0294	0.0147
13	1.088	−8.06	1.088	−10.71	1.088	−10.82	0.0291	0
14	1.04	−10.14	1.007	−12.95	1.026	−13.11	0.0309	0.0205
15	1.036	−10.25	0.992	−12.97	1.017	−13.33	0.0302	0.0269
16	1.043	−9.83	1.009	−12.55	1.029	−12.75	0.03	0.0208
17	1.039	−10.14	0.996	−12.93	1.021	−13.23	0.0309	0.0266
18	1.027	−10.85	0.982	−13.61	1.008	−13.96	0.0306	0.0275
19	1.025	−11.01	0.98	−13.8	1.005	−14.14	0.0309	0.0278
20	1.029	−10.81	0.983	−13.61	1.01	−13.95	0.031	0.0282
21	1.033	−10.44	0.971	−13.44	1.006	−13.91	0.0335	0.0383
22	1.034	−10.43	0.966	−13.49	1.005	−14	0.0341	0.0413
23	1.027	−10.69	0.945	−13.65	0.99	−14.45	0.0336	0.0503
24	1.024	−10.92	0.89	−14.29	0.963	−15.67	0.0399	0.0817
25	1.026	−10.82	0.708	−15.89	0.883	−19.44	0.0742	0.1931
26	1.009	−11.23	0.696	−16.3	0.868	−19.85	0.0742	0.1898
27	1.036	−10.5	0.601	−17.45	0.844	−21.97	0.1123	0.2608
28	1.017	−6.88	0.954	−7.79	0.982	−9.18	0.0109	0.0393
29	1.017	−11.69	0.312	−17.68	0.698	−29.79	0.2138	0.4143
30	1.005	−12.55	0	−69.44	0.567	−41.95	0.5264	0.5507

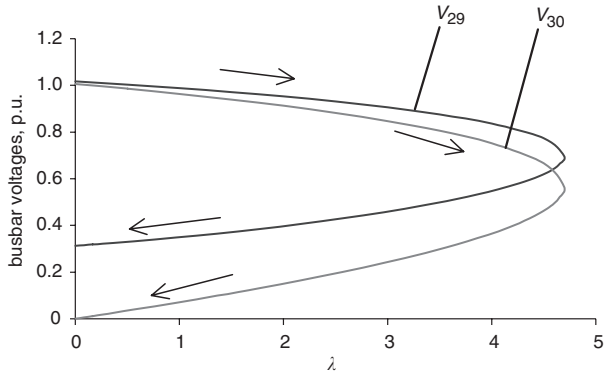

Fig. 1 *Voltage curves of busbars 29 and 30 for IEEE 30-busbar system*

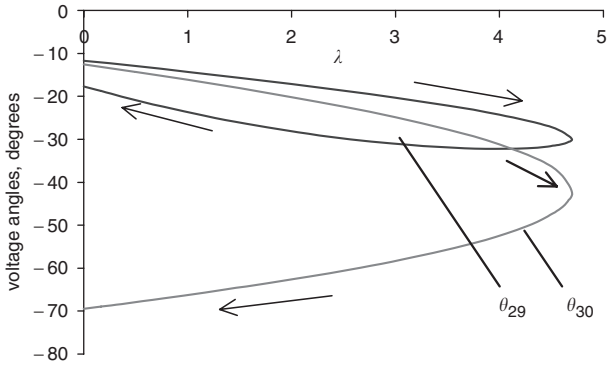

Fig. 2 *Phase angle curves of busbars 29 and 30 for IEEE 30-busbar system*

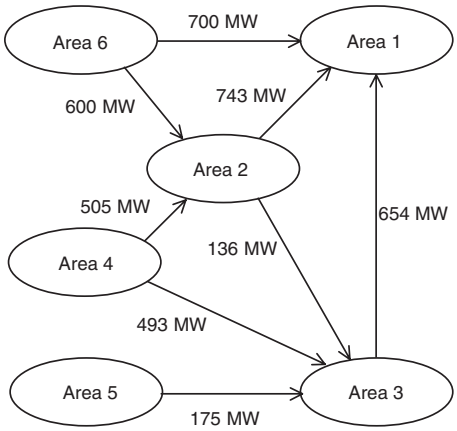

Fig. 3 *The six areas of the East China 2416-busbar system*

largest load centre in the entire system, and Area 2 is the next. Figure 3 shows the inter-area power exchanges at base load flow.

The computer test is to evaluate the maximum available transfer capability between Area 1 and Area 4 with other area generations and loads fixed to their base load flow values. A group of generators in Area 4 increase generations, while a group of loads in Area 1 increase demands simultaneously. At the QTP, the maximum parameter $\lambda_{max} = 9.8479$ is calculated using the proposed method. Figures 4 and 5 present the solution curves of voltages magnitudes and phase angles of two busbars. Similar to the IEEE 30-busbar test case, the Newton power flow program runs without any difficulty except at the turning point. The QTP is then calculated by resorting to the proposed new method.

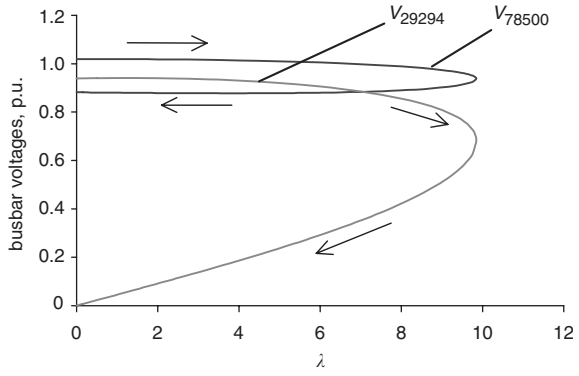

Fig. 4 *East China 2416-busbar system voltage curves of busbars 29294 and 78500*

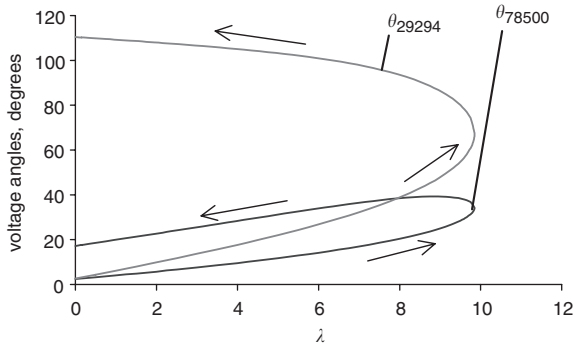

Fig. 5 *East China 2416-busbar system phase angle curves of busbars 29294 and 78500*

In the numerical simulation for this 2416-busbar system, most of the computational time is consumed in calculating the initial point of QTP. A good initial point is very important for convergence of QTP solution. In the current prototype program, the initial point is calculated by a continuation power flow method with natural parameterisation technique. The increment step size of the parameter λ will affect the total computational time significantly. However, once the initial point is obtained, the time used for calculating the QTP itself is very short. Table 3 lists the CPU time allocation for QTP calculation with different step sizes. In all tests, the load flow convergence tolerance is 10^{-5} per unit for maximum busbar power mismatch. From Table 3 it can be seen that the CPU time for QTP calculation is much smaller than that for initial point calculation and also it is not affected by the step size used in tracing the load flow curves. It can also be seen that a larger step size may save noticeable time for calculating the initial point. More efficient methods such as fast decoupled power flow should be used for initial point searching.

Table 3: CPU time for the East China 2416-busbar system

Step size, p.u.	0.01	0.1	0.5	1.0	2.0
CPU time for initial point, s	24.44	4.52	1.43	1.04	0.86
CPU time for QTP, s	0.16	0.16	0.16	0.16	0.16

5.3 Discussions

It should be pointed out that in all the tests conducted by the prototype program, constraints of line flows, voltage limits and generator reactive power limits are neglected. The tests are thus done with respect to simple cases, for given sets of power flow equations without inequalities. However,

in a real-world power system, these inequality constraints are always presented. Specifically, generator reactive power limits must be observed in real power system operation. Hence, the calculated QTPs without considering reactive power limits may not be realistic ones. More practical QTPs can be obtained by taking generator reactive power limits into account.

Another point worthy of attention is that more complexity can be introduced due to the presence of generator reactive power limits. For example, [4] and [15] described a case where a maximum loading point of the system may not necessarily occur at a singular point (Figure 10 of [4]). This type of specific cases needs particular and detailed consideration when the proposed technique will be implemented in a production-grade software package.

In all the tests, the QTPs can be calculated very quickly when proper initial points are available without ill condition. In more complex applications, for example when there are two free parameters and a large number of QTPs are to be solved, the proposed method will be better than the CPF method, because it can calculate QTP directly, hence it is more flexible and efficient.

6 Conclusions

In this paper, an efficient matrix reduction method is proposed to solve the MSDS defined at the QTPs of a general nonlinear system with one changeable parameter.

The method has sound mathematical foundation and can be easily implemented by slightly modifying an existing nonlinear equation solver. An extremely important feature of the method is that it eliminates the requirement of a specific sparse matrix solver that may degrade the efficiency of solving the MSDS and may increase the complexity of the program.

The method has been successfully applied to solve QTP in power flow solution curves. The computer tests on IEEE 30-busbar and East China 2416-busbar systems show that the proposed method is very efficient and computationally reliable. It is clear that, by including this method into an existing continuation power flow program, the QTP calculation can be more efficient and more reliable.

Further research and development is ongoing to apply the method to sensitivity analysis around a solved turning bifurcation point and to extend the method to systems consisting of two or more parameters.

7 Acknowledgment

This research is sponsored by the RGC of the Hong Kong Government (Grant number CityU 1/00C), the National Key Basic Research Project (973 Project) of the People's Republic of China (Grant numbers G1998020307, G1998020308), and the Natural Science Fund Team Project (number 003049) of Guangdong Province of China, to whom the authors would like to express their sincere acknowledgments.

8 References

1 Seydel, R.: 'Practical bifurcation and stability analysis—from equilibrium to chaos', (Springer-Verlag, New York, 1994)
2 Govaerts, W.: 'Numerical methods for bifurcations of dynamical equilibria', (SIAM, Philadelphia, 1999)
3 Moore, G., and Spence, A.: 'The calculation of turning points of nonlinear equations', *SIAM J. Numer. Anal.*, 1980, **17**, pp. 567–576
4 Van Cutsem, T.: 'Voltage instability: phenomena, countermeasures and analysis methods', *Proc. IEEE*, 2000, **88**, (2), pp. 208–227
5 Kundur, P.: 'Power system stability and control', (McGraw-Hill, New York, 1994)
6 Taylor, C.W.: 'Power system voltage stability', (McGraw-Hill, New York, 1994)
7 Tamura, Y., Mori, H., and Iwamoto, S.: 'Relationship between voltage stability and multiple load flow solutions in electric power systems', *IEEE Trans. Power Appar. Syst.*, 1983, **102**, (5), pp. 1115–1125
8 Ajjarapu, V., and Christy, C.: 'The continuation power flow: a tool for steady state voltage stability analysis', *IEEE Trans. Power Syst.*, 1992, **7**, (1), pp. 416–423
9 Chiang, H.D., Flueck, A.J., Shah, K.S., and Balu, N.: 'CPFLOW: A practical tool for tracing power system steady-state stationary behaviour due to load and generation variations', *IEEE Trans. Power Syst.*, 1995, **10**, (2), pp. 623–634
10 Ajjarapu, V.: 'Identification of steady-state voltage stability in power systems'. Proceedings IASTED International Conference on High Technology in the Power Industry, Phoenix AZ, 1–4 March 1988, pp. 244–247
11 Cañizares, C.A., Alvarado, F.L., DeMarco, C.L., Dobson, I., and Long, W.F.: 'Point of collapse methods applied to AC/DC power systems', *IEEE Trans. Power Syst.*, 1992, **7**, (2), pp. 673–683
12 Cañizares, C.A., and Alvarado, F.L.: 'Point of collapse and continuation methods for large AC/DC systems', *IEEE Trans. Power Syst.*, 1993, **8**, (1), pp. 1–8
13 Van Cutsem, T.: 'A method to compute reactive power margins with respect to voltage collapse', *IEEE Trans. Power Syst.*, 1991, **6**, pp. 145–156
14 Irisarri, G.D., Wang, X., Tong, J., and Mokhtari, S.: 'Maximum loadability of power systems using interior point non-linear optimisation method', *IEEE Trans. Power Syst.*, 1997, **12**, pp. 162–172
15 Van Cutsem, T., and Vournas, C.: 'Voltage stability of electric power systems' (Kluwer, Norwell, MA, 1998)

9 Appendix:

Proof of proposition 1

To prove that matrix A is nonsingular, it is sufficient to prove that the following equation:

$$\begin{pmatrix} f_x & f_\lambda \\ l^T & 0 \end{pmatrix} \begin{pmatrix} w \\ d \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \; w \in R^n, \; d \in R \tag{16}$$

has a unique zero solution. We will use conflict method to prove the proposition. Suppose that (16) has a nonzero solution, then we have the following two possible cases:

Case 1: Suppose $d \neq 0$. In this case the first block equation of (16) can be written as:

$$f_x w + f_\lambda d = 0 \tag{17}$$

this means

$$f_\lambda \in R(f_x) \tag{18}$$

and from (2c) we obtain

$$\psi^T f_\lambda = 0 \tag{19}$$

which contradicts (2d). Hence we certainly have $d = 0$.

Case 2: Suppose $w \neq 0$. Now that Case 1 asserts $d = 0$, then from (16) we have:

$$f_x w = 0 \tag{20}$$

$$l^T w = 0 \tag{21}$$

Because $\text{rank}(f_x) = n-1$, (20) means w is a right eigenvector of the zero eigenvalue of f_x. This obviously results in a contradiction between (21) and (3c). Therefore we certainly have $w = 0$.

The analysis on the two cases shows that (16) has a unique zero solution. The proposition is proven.