An efficient algorithm for the determination of certain bifurcation points

James P. Abbott (*)

ABSTRACT

Many practical problems require information about a branch of solutions of a system of nonlinear equations dependent upon a scalar parameter. We discuss some techniques for following such a branch through a turning point and describe an efficient method, with second order convergence, for finding the turning point. We also show that, if extra information is available about the solution branch, the method can be successfully applied to finding simple bifurcation points.

1. INTRODUCTION

In many physical problems it is necessary to solve a system of nonlinear equations of the form

$$f(x, \lambda) = 0, \tag{1.1}$$

$f : D \subset R^n \times R \to R^n$, where the solution vector $x(\lambda)$ is a simple, continuously differentiable arc in R^n dependent upon the scalar parameter λ. Problems of this kind occur in many branches of mathematics but most notably in the theory of elasticity (see, for example, [4], [10], [11], [22], and the references therein). It is necessary to find $x(\lambda)$ numerically for values of λ sufficient to define the curve $[x(\lambda), \lambda]$, which is called *a solution branch* of (1.1) in $R^n \times R$. This solution branch will often exhibit complex behaviour, but we note that $x(\lambda)$ satisfies the differential equation

$$\frac{dx}{d\lambda}(\lambda) = - J(x, \lambda)^{-1} d(x, \lambda), \tag{1.2}$$

where $J(x, \lambda) = \partial_x f(x, \lambda)$ and $d(x, \lambda) = \partial_\lambda f(x, \lambda)$.

We assume throughout that $f(x, \lambda)$ is twice continuously differentiable with respect to x and λ on D and then (1.2) follows by differentiating (1.1) with respect to λ. Thus, if $J[x(\lambda), \lambda]$ is non-singular, then $x(\lambda)$ is locally continuous. Points on $[x(\lambda), \lambda]$ at which $J(x, \lambda)$ is singular are called *critical points* (often bifurcation or singular points) and have received a large amount of attention in the literature (see the references cited above plus [3], [7], [8], [12], [13], [19], [20], [22], [23]).

It is the purpose of this paper to describe some efficient methods for the accurate determination of certain critical points. Firstly we consider the simplest type, which is a point, (x^*, λ^*), such that $J[x(\lambda), \lambda]$ is nonsingular for λ close to λ^* and where

$$\text{rank } [J(x^*, \lambda^*)] = n-1, \tag{1.3a}$$

$$\text{rank } [J(x^*, \lambda^*)\, d(x^*, \lambda^*)] = n. \tag{1.3b}$$

(x^*, λ^*) is called a *limit point*. If the solution branch $[x(\lambda), \lambda]$ through (x^*, λ^*) exists for all λ in an open neighbourhood of λ^*, then (x^*, λ^*) is called a *point of inflexion* otherwise it is a *turning point*. In structural problems, a turning point represents the boundary between stability and instability of the system.

Prior to discussing methods for finding a turning point we consider, in section 2, the problem of following a solution branch through a turning point. We describe a method which is similar to that developed by Riks [22] and Menzel and Schwetlick [13] but involves less work per step.

In section 3 we describe methods for the accurate determination of (x^*, λ^*). Simpson [23] described an iterative method which required, at each iteration, the solution of (1.1) for some λ and the estimation of the smallest eigenvalue of $J[x(\lambda), \lambda]$. His method converges linearly to (x^*, λ^*) and is suitable only for symmetric $J(x, \lambda)$. Here we describe methods which require less work per iteration, have quadratic convergence to (x^*, λ^*) and do not require $J(x, \lambda)$ to be symmetric.

A critical point, (x_B, λ_B), which is such that $J[x(\lambda), \lambda]$ is nonsingular for λ close to λ_B and where

$$\text{rank } [J(x_B, \lambda_B)] = n-1, \tag{1.4a}$$

$$\text{rank } [J(x_B, \lambda_B)\, d(x_B, \lambda_B)] = n-1. \tag{1.4b}$$

is called a *simple bifurcation point*. Given an additional condition at the second derivative of $f(x, \lambda)$, Crandall and Rabinowitz [7] have shown that, in a neighbourhood of (x_B, λ_B), the totality of solutions of (1.1) form two continuous curves intersecting only at (x_B, λ_B). In many applications it is necessary to follow one branch, often called the primary branch, and on detecting the presence of a secondary branch, to follow it (see Keller and Langford [12] and Rheinboldt [19], [20] for methods). In the case when the primary branch satisfies some symmetry relations it is often possible to generate a method which converges to (x_B, λ_B) with second order convergence and we

(*) J. P. Abbott, Department of Applied Mathematics and Theoretical Physics, Silver Street CAMBRIDGE CB3 9EW, England.

discuss this in section 4. The method also has the advantage of providing an approximation to the zero eigenvector of $J(x_B, \lambda_B)$, which is required by the methods in [12] and [20] for finding a point on the secondary branch.
Finally, in section 5, we describe some numerical experience with the methods.
We note that, in an attempt to be brief we have omitted much of the detail in the following sections, particularly the numerical aspects. The interested reader will find much of this detail in Abbott [2].

2. FOLLOWING TRAJECTORIES THROUGH TURNING POINTS

In this section we describe briefly the method due to Riks [22] and Menzel and Schwetlick [13] and our modification. In [13] the method was described as a means of extending the region of convergence of methods for the solution of nonlinear equations. It appears that such a method involves an unnecessary amount of work for that problem where the accurate determination of the solution trajectory is not required (see [1], [2] for details). However the approach is effective when following a solution branch past a turning point. Earlier methods for this problem, e.g. [4], [23], solved (1.1) by Newton's method for a sequence of values of λ, λ_i, i = 1, 2,... However, failure occurs when $[x(\lambda_i), \lambda_i]$ approaches a turning point. Once failure has occured the turning point can be passed by extrapolating over (x^*, λ^*) but the accuracy and efficiency of the method is impaired since $J(x, \lambda)$ is nearly singular close to (x^*, λ^*). Anselone and Moore [3] suggested changing the scalar variable to overcome these difficulties but considered only particular cases. Recently Riks [22] and Menzel and Schwetlick [13] have employed an idea essentially due to Davis [8] and make a change of variable which is applicable generally.
It will be convenient to write

$$y = \begin{bmatrix} x \\ \lambda \end{bmatrix},$$

or, more conveniently, $y = (x, \lambda)$, and to consider f as a mapping from $D \subset R^{n+1} \rightarrow R^n$. Then (1.1) becomes

$$f(y) = 0. \tag{2.1}$$

Define $y^* = (x^*, \lambda^*)$ and H(y) by
$H(y) = [J(y)\ d(y)]$
then, from (1.3), rank $[H(y^*)] = n$. In fact, it follows from our assumptions that, for any y satisfying (2.1) in a neighbourhood of y^*,

$$\text{rank } [H(y)] = n. \tag{2.2}$$

The technique described by Riks, Menzel and Schwetlick is to add, at each iteration, an auxiliary equation to (2.1). They choose a function $\beta(y)$, $\beta : D \subset R^{n+1} \rightarrow R$, such that the solution of

$$g(y) = \begin{bmatrix} f(y) \\ \beta(y) \end{bmatrix} = 0 \tag{2.3}$$

is well defined and is a required point on the solution branch. Let G(y) denote the Jacobian of g(y). Suppose $\hat{y}$ is a known solution of (2.1) and we wish to find a new point on the solution branch. We can define the branch in R^{n+1} by y(s), where s represents the arc length, and let $\hat{y} = y(\hat{s})$. Now $\beta(y)$ is defined simply as

$$\beta(y) = b^T(y-\hat{y}) - \sigma \tag{2.4}$$

for some b and σ. Denoting the derivative of y(s) with respect to s by $\dot{y}(s)$, Riks, Menzel and Schwetlick make the choice
$b = \dot{y}(\hat{s})$.
Note that $\dot{y}(\hat{s})$ is a unit vector tangent to the solution branch at $\hat{y}$ and is the unique solution, of unit length, of
$H(\hat{y})\ \dot{y}(\hat{s}) = 0$.
This choice of b actually maximises $|\text{Det } [G(\hat{y})]|$ over all possible choices of b of unit length. This follows from the following Lemma which is similar to a result in [22]. We omit the proof which is given in [2].

Lemma 1

Let G(y) be the Jacobian of g(y), defined in (2.3), with $\beta(y)$ defined by (2.4). Then

$$\text{Det } [G(\hat{y})] = \rho\, b^T \dot{y}(\hat{s})$$

where ρ is a non-zero constant independent of b.

An initial estimate, z, of the new point is found from
$z = \hat{y} + \sigma \dot{y}(\hat{s})$
and the system

$$\begin{bmatrix} f(y) \\ (y-\hat{y})^T\, \dot{y}(\hat{s}) \end{bmatrix} = \begin{bmatrix} 0 \\ \sigma \end{bmatrix} \tag{2.5}$$

is solved using Newton's method. That (2.5) has a well defined solution, for sufficiently small σ, follows from the nonsingularity of $G(\hat{y})$ and the implicit function theorem. The basic idea is expressed in Fig. 1 for the scalar case.
The equations (2.5) constitute n + 1 equations in n + 1 unknowns and, whilst work can be saved by noting that one equation is linear, we prefer to reduce the number of variables in a direct way. If $\beta(y)$ is chosen as

$$\beta(y) = e_r^T(y-\hat{y}) - \sigma,$$

for some r, σ, where e_r is the rth unit vector, then (2.5) becomes

$$f(y) = 0, \tag{2.6a}$$

$$y_r = \hat{y}_r + \sigma \tag{2.6b}$$

which, since y_r is known, constitute n equations in n unknowns (*). The index r is chosen so that the determinant of $G(\hat{y})$ is as large as possible at $\hat{y}$. When r = n+1 the method is one of incrementing λ as described above. Close to a turning point some other

(*) This idea has been developed independently by W.C. Rheinboldt in an, as yet, unpublished manuscript and by W. Kubicek [24].

element of y will be more suitable as the incremental variable. Since we have reduced the number of equations by one, the amount of work saved may be significant if n is small or if many points on the solution branch are required.

To choose the index r we note, from Lemma 1, that $\mathrm{Det}[G(\hat{y})] = \rho\, e_r^T \dot{y}(\hat{s})$ and we choose r to maximise $|e_j^T \dot{y}(\hat{s})|$, $j = 1, \ldots, n+1$. We note that the angle, θ_j, between the solution branch at $\hat{y}$ and the jth coordinate direction is given by

$$\cos\theta_j = e_j^T \dot{y}(\hat{s}).$$

Thus our choice of r gives the variable, y_r, whose coordinate direction makes the smallest angle with the solution branch. This is expressed in Fig. 2 for the scalar case.

In practice the initial estimate of the solution of (2.6) is taken as

$$z = \hat{y} + a\dot{y}(\hat{s}),$$

where $a = \sigma / e_r^T \dot{y}(\hat{s})$, which is the linear estimate of the solution. Then (2.6a) is solved by Newton's method.

Similar processes to these were also used in [11] and [17] but for problems with only 2 or 3 dimensions.

3. ACCURATE DETERMINATION OF A TURNING POINT

3.1. Introduction

Several methods, based upon interpolation, have been suggested for the accurate determination of a turning point, (x^*, λ^*), on a solution branch of (1.1). Notably Simpson [23] describes an iterative method which gives linear convergence to (x^*, λ^*) and which is suitable for problems with symmetric $J(x, \lambda)$. In this section we present some methods which, for less work per iteration, give second order convergence to (x^*, λ^*) and do not require $J(x, \lambda)$ to be symmetric.

We assume that a reasonable estimate, (x_o, λ_o), of (x^*, λ^*) is known from following a solution branch using a method from section 2. In many problems the value of $\Delta(\lambda) = \mathrm{Det}[J(x(\lambda), \lambda)]$ determines whether or not the system is stable and, as $\Delta(\lambda)$ changes sign, the branch passes through a turning point (in or out of a region of stability). When $\Delta(\lambda)$ can be easily evaluated it can be monitored to specify when two iterates straddle a turning point. But better than evaluating $\Delta(\lambda)$ is to note that

$$\Delta[\lambda(s)] = a(s)\,\dot{\lambda}(s)$$

where $a(s)$ is a non-zero function in the region of the turning point, (see [2] for a proof of this result). Thus $\Delta(\lambda)$ changes sign with $\dot{\lambda}(s)$ which is computed as a component of $\dot{y}(s)$ and so the sign of $\Delta(\lambda)$ can be monitored without extra computation.

To find the turning point we set up a set of equations which, in the region of interest, have a unique solution (x^*, λ^*). These are of the form

$$f(x, \lambda) = 0, \tag{3.1a}$$

$$\phi(x, \lambda) = 0, \tag{3.1b}$$

where $\phi : D \subset R^n \times R \to R^n$ is chosen so that

$$\phi(x, \lambda) = 0 \text{ iff } J(x, \lambda) \text{ is singular.} \tag{3.1c}$$

In section 3.3 we give some choices of $\phi(x, \lambda)$ which have proved successful in practice but they are characterised by being expensive to calculate. For this reason we describe, in section 3.2, a method suitable for this case.

3.2. A Newton Like Method

In this section we describe a method which we will use for solving (3.1). Since it may be of interest in other cases, we describe it in some generality and apply it to (3.1) in the next section. We consider the general problem of solving the nonlinear equations

$$q(z, \mu) = 0 \tag{3.2a}$$

and

$$\psi(z, \mu) = 0, \tag{3.2b}$$

$q : D \subset R^n \times R \to R^n$, $\psi : D \subset R^n \times R \to R$, where

$$\partial_z q(z, \mu) = Q(z, \mu) \tag{3.3}$$

is nonsingular in the region of a solution (z^*, μ^*) of (3.2). We assume that derivatives of $\psi(z, \mu)$ are not available and that $\psi(z, \mu)$ is expensive to evaluate. The method we describe is similar to those of Brown [6] and Brent [5] but is more suitable when $Q(z, \mu)$ is available analytically and when $Q(z, \mu)$ is large and sparse or easy to evaluate. We note that, for small problems, we have used Brent's method with success. (See [14] for an implementation and also [5] for Brent's comments on the suitability of his method for problems where the Jacobian is sparse).

Suppose (z_i, μ_i) is an approximation to (z^*, μ^*), then we linearise (3.2a) about (z_i, μ_i) and define the subspace $\mathcal{L}_i$ to be the space where this linearisation is zero. That is, $\mathcal{L}_i$ is the set of points (z, μ) such that

$$q(z_i, \mu_i) + [Q(z_i, \mu_i)\; u(z_i, \mu_i)] \begin{bmatrix} z - z_i \\ \mu - \mu_i \end{bmatrix} = 0,$$

where $u(z, \mu) = \partial_\mu q(z, \mu)$. Now, omitting the arguments (z_i, μ_i) and writing $q(z_i, \mu_i) = q_i$ etc., and assuming Q_i is nonsingular, $\mathcal{L}_i$ is defined by

$$\mathcal{L}_i = \{(z, \mu) \mid z = \hat{z}_{i+1} - Q_i^{-1} u_i (\mu - \mu_i)\}$$

where

$$\hat{z}_{i+1} = z_i - Q_i^{-1} q_i. \tag{3.4}$$

Now we define $\psi_i : D_i \subset R \to R$ as ψ, restricted to $\mathcal{L}_i$, by

$$\psi_i(\mu) = \psi[\hat{z}_{i+1} - Q_i^{-1} u_i (\mu - \mu_i), \mu], \tag{3.5}$$

where $D_i = \{\mu \mid [\hat{z}_{i+1} - Q_i^{-1} u_i (\mu - \mu_i), \mu] \in D\}$. Then we

can attempt to find a zero of $\psi(\mu)$ on $\mathcal{L}_i$ by linearising ψ_i and applying a Newton step. Since we can not evaluate $\frac{d\psi_i}{d\mu}(\mu_i)$, we approximate it by

$$\frac{d\psi_i}{d\mu}(\mu_i) \simeq \frac{\psi_i(\mu_i+\delta_i)-\psi_i(\mu_i)}{\delta_i} = \Delta_i, \tag{3.6}$$

for some $\delta_i \neq 0$, and generate the step

$$\mu_{i+1} = \mu_i - \frac{\psi_i(\mu_i)}{\Delta_i}. \tag{3.7}$$

Then z_{i+1} is given by

$$z_{i+1} = \hat{z}_{i+1} - Q_i^{-1}u_i(\mu_{i+1}-\mu_i). \tag{3.8}$$

The following theorem gives sufficient conditions for the sequence $\{(z_i, \mu_i)\}$ generated by (3.7) and (3.8), to converge, with second order convergence, to (z^*, μ^*).

Theorem 1

Suppose $q : D \subset R^n \times R \to R^n$ and $\psi : D \subset R^n \times R \to R$ are Frechet differentiable on D and their derivatives satisfy a Lipschitz condition on an open neighbourhood S of the point (z^*, μ^*) which is a solution of (3.2). Suppose also that $Q(z, \mu)$, defined in (3.3) has a bounded inverse in S and that the inverse of

$$R(z, \mu) = \begin{bmatrix} Q(z, \mu) & u(z, \mu) \\ \partial_z\psi(z, \mu)^T & \partial_\mu\psi(z, \mu) \end{bmatrix} \tag{3.9}$$

exists and is bounded on S, where $u(z, \mu) = \partial_\mu q(z, \mu)$. Then there exists an $\epsilon > 0$ such that, if

$$\left\| \begin{matrix} z_0-z^* \\ \mu_0-\mu^* \end{matrix} \right\| \leq \epsilon,$$

the sequence $\{(z_i, \mu_i)\}$ defined by (3.4) - (3.8), where δ_i is chosen as

$$\delta_i = \begin{cases} |\psi(z_i, \mu_i)|/[1+\|Q(z_i, \mu_i)^{-1}u(z_i, \mu_i)\|], \\ \text{if } \psi(z_i, \mu_i) \neq 0, & (3.10a) \\ \\ \text{sufficiently small otherwise,} & (3.10b) \end{cases}$$

converges to (z^*, μ^*) and the convergence is second order.

(Note that, in practice, to ensure that $\delta_i \neq 0$ we can choose $\delta_i = \tau$, where the stopping criterion is

$$\left\| \begin{matrix} z_{i+1} - z_i \\ \mu_{i+1} - \mu_i \end{matrix} \right\| \leq \tau, \text{ if (3.10a) gives a value less than } \tau.)$$

Proof

The proof is fairly straightforward but rather long and so will be omitted. See [2] for the complete proof.

3.3. Choices for $\phi(x, \lambda)$

The equations we wish to solve are given in (3.1) and, to apply the method of section 3.2, we must put them into a form which satisfies the conditions of Theorem 1. To do this we note that, from (2.2), rank $H(x, \lambda) = n$ in the region of a turning point, where

$$H(x, \lambda) = [J(x, \lambda)\ d(x, \lambda)]. \tag{3.11}$$

Thus $H(x, \lambda)$ has n linearly independent columns. It is convenient to define the $(n + 1) \times n$ matrices P_j, $j = 1, 2, \ldots, n + 1$ by

$$P_{n+1} = \begin{bmatrix} I_n \\ 0 \end{bmatrix}, \quad P_j = P_{n+1} + (\tilde{e}_{n+1} - \tilde{e}_j)\, e_j^T, \tag{3.12}$$

where I_n is the $n \times n$ unit matrix with columns $e_1, \ldots, e_n$ and $\tilde{e}_1, \ldots, \tilde{e}_{n+1}$ are the columns of I_{n+1}. Then, if we write

$$H = H(x, \lambda) = [h_1\ h_2\ \ldots\ h_n\ d] = [J(x, \lambda)\ d(x, \lambda)]$$

we have, for some r,

$$HP_r = [h_1\ h_2\ \ldots\ h_{r-1}\ d\ h_{r+1}\ \ldots\ h_n].$$

Thus we can choose r so that HP_r is nonsingular. It is shown in [2] that the best choice of r is actually the value chosen in section 2, because that choice of r maximises $\mathrm{Det}\,(HP_r)$. Now we define (z, μ) and (z^*, μ^*) by

$$z = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_{r-1} \\ \lambda \\ x_{r+1} \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} = P_r^T \begin{bmatrix} x \\ \lambda \end{bmatrix}, \quad \mu = x_r;\ z^* = P_r^T \begin{bmatrix} x^* \\ \lambda^* \end{bmatrix}, \quad \mu^* = x_r^*,$$

and $q(z, \mu)$ by

$$q(z, \mu) = f(x, \lambda).$$

Then $Q(z, \mu) = H(x, \lambda)\, P_r$, which is nonsingular in a neighbourhood of (z^*, μ^*) as required, and $u(z, \mu) = h_r(x, \lambda)$. Also we define $\psi(z, \mu)$ by

$$\psi(z, \mu) = \phi(x, \lambda).$$

If $\phi(x, \lambda)$ satisfies (3.1c), then it follows (see [2]) that $R(z, \mu)$ defined in (3.9) is singular at (z^*, μ^*) if and only if the solution branch at (x^*, λ^*) is tangential to the surface on which $J(x, \lambda)$ is singular. This is not generally the case.

We now consider specific choices for $\phi(x, \lambda)$. From (3.1c) the obvious choice is

$$\phi_1(x, \lambda) = \mathrm{Det}\,[J(x, \lambda)].$$

This choice proved acceptable except in two cases. When $J(x, \lambda)$ is large and sparse, the evaluation of

$\phi_1(x, \lambda)$ may be inconvenient since it requires the factorisation of $J(x, \lambda)$ into matrices which are not necessarily sparse. Secondly, if Det $[J(x,\lambda)]$ is very small compared with $\|f(x,\lambda)\|$, then loss of significance occurs in the evaluation of $\psi_i(z,\mu)$ and therefore of Δ_i in (3.6), which adversely affects the convergence rate of the method. Despite these difficulties, this choice proved successful for several small problems, but we discuss two further choices which do not suffer the same disadvantages.

Define $\phi_2(x, \lambda)$ by

$$\phi_2(x, \lambda) = e_r^T[H(x, \lambda)P_r]^{-1} h_r(x, \lambda),$$

where r is the index described earlier, which guarantees that $H(x, \lambda)P_r$ is nonsingular in the neighbourhood of (x^*, λ^*) and $h_r(x, \lambda) = H(x, \lambda)\tilde{e}_r$.

We note that, from (3.11), and omitting the variables (x, λ) as arguments, we have

$$J = HP_{n+1}$$

and, from (3.12),

$$J = HP_r + H(\tilde{e}_r - \tilde{e}_{n+1})e_r^T,$$

$$= HP_r [I + (HP_r)^{-1} (h_r - d)e_r^T].$$

Using the identity $\mathrm{Det}(I + ab^T) = 1 + b^Ta$, we have

$$\mathrm{Det}\ J = \mathrm{Det}(HP_r) [1 + e_r^T(HP_r)^{-1} (h_r - d)]$$

and, using $(HP_r)^{-1} d = e_r$, we have

$$\mathrm{Det}\ J = \mathrm{Det}\ (HP_r)\ e_r^T(HP_r)^{-1} h_r.$$

This shows that $\phi_2(x, \lambda)$ is a suitable choice since $\phi_2(x, \lambda) = 0$ iff $J(x, \lambda)$ is singular.

Its evaluation requires the solution of a system of linear equations and so it is suitable in the case when $H(x, \lambda)$ is sparse.

Our final choice for $\phi(x, \lambda)$ is given by defining $\phi_3(x, \lambda)$ by the relation

$$J(x, \lambda)\, v(x, \lambda) = \phi_3(x, \lambda)w, \qquad (3.13a)$$

$$s^T v(x, \lambda) = 1, \qquad (3.13b)$$

for some fixed s and w such that $\|s\| = \|w\| = 1$. This choice is an extension of the method of Osborne and Michaelson [15], [16] for the nonlinear eigenvalue problem in one variable. $\phi_3(x, \lambda)$ is well defined and continuous in a neighbourhood of (x^*, λ^*) provided that s and w have components in the directions u_r^* and u_l^*, respectively, where u_r^* and u_l^* are the right and left eigenvectors of $J(x^*, \lambda^*)$ corresponding to the zero eigenvalue. In practice, if w is a reasonable approximation to u_r^*, then the choice $s = e_k$, where k is chosen to maximise $|e_j^T w|$, $j = 1, \ldots, n$, is suitable. Also it is most efficient to change w, and therefore s, at each iteration, always using the best estimate of u_r^* for w.

$J(x_o,\lambda_o)^{-1}\, d(x_o,\lambda_o)$, which is the first n components of $\hat{y}$ at (x_o, λ_o) and so has already been calculated, is a good initial choice for w when suitably scaled.

To complete our description of the method for this choice of $\phi(x, \lambda)$, we define the subspace $\mathcal{L}_i$ as in section 3.2 and the matrix $M_i(\mu)$ by

$$M_i(\mu) = J[\hat{z}_{i+1} - Q_i^{-1}u_i(\mu-\mu_i), \mu]$$

where, for brevity, we are considering J to be a function of z and μ. Then if w_i is our current estimate of u_r^* and e_k is our current choice of s, then $\psi_i(\mu_i)$ is given by

$$M_i(\mu_i)\, v_{i+1} = \psi_i(\mu_i)w_i$$

and

$$e_k^T v_{i+1} = 1. \qquad (3.13b')$$

v_{i+1} is found by solving

$$M_i(\mu_i)v = w_i \qquad (3.14)$$

and scaling the solution to satisfy (3.13b') and also

$$\psi_i(\mu_i) = 1/e_k^T v.$$

This represents one step of inverse iteration and so v_{i+1} will be richer in u_r^* than w_i. Thus v_{i+1} is a better choice for w_{i+1} than w_i. It is shown by Osborne [17] that another efficient choice of w_{i+1} is v'_{i+1}, given by

$$v'_{i+1} = \frac{dM_i}{d\mu}(\mu_i)\, v_{i+1}.$$

We do not have the derivative of $M_i(\mu)$, but, in the estimation of $\frac{d\psi_i}{d\mu}(\mu)$ on $\mathcal{L}_i$, we also calculate $M_i(\mu_i+\delta_i)$ and so we can improve v_{i+1} by forming

$$v'_{i+1} = [M_i(\mu_i) - M_i(\mu_i+\delta_i)]v_{i+1}. \qquad (3.15)$$

Then we set w_{i+1} to be v_{i+1} or v'_{i+1} and scale it suitably. It is important to note that, as the process converges $\{M_i(\mu_i)\}$ approaches $J(x^*, \lambda^*)$, however, in the same way as inverse iteration, no difficulties arise when solving (3.14) due to $M_i(\mu_i)$ being nearly singular. All that is necessary is that care be taken in solving (3.14) so that the solution remains within machine bounds.

We conclude this section with three remarks.

Remark 1

For each choice of ϕ, an iteration requires the solution of four linear systems and gives second order convergence to the turning point. This compares favourably with the method described by Simpson [23].

Also we note that with ϕ_2 and ϕ_3 the work in solving these systems can be reduced as follows. If a direct method is to be employed for solving the linear equations then, when calculating $Q_i^{-1}u_i$ and $Q_i^{-1}q_i$ in (3.4) and (3.5), it is only necessary to decompose Q_i into its appropriate factors once. This saving cannot be made if an iterative method is being used to solve the linear systems. In this case, however, the calculation of $\psi_i(\mu_i+\delta_i)$ and $\psi_i(\mu_i)$ each require the solution of a linear system. Moreover, the solution of the first will provide an excellent estimate of the solution of the second. The result is that few iterations will be required for the second system.

Remark 2

The method of Osborne and Michaelson is just one of a class of methods for the nonlinear eigenvalue problem which could be applied to this problem. Some of these are discussed in [21].

Remark 3

In addition to the conditions on f already assumed, to satisfy the conditions of Theorem 1 it is necessary that the second derivatives of f satisfy a Lipschitz condition in a neighbourhood of (x^*, λ^*).

4. SIMPLE BIFURCATION POINTS

We point out, in this section, that the method of section 3 can sometimes be applied to finding simple bifurcation points. To find a point (x_B, λ_B) defined in (1.4) we can solve

$$f(x, \lambda) = 0, \tag{4.1a}$$

$$\phi(x, \lambda) = 0, \tag{4.1b}$$

with $\phi(x, \lambda)$ given by ϕ_1 or ϕ_3 from section 3. In this case, however, the resulting Jacobian is singular at the solution and so the method converges only linearly. However, it is often the case that, on a primary branch, we have independent information about the solution curve $x(\lambda)$. For example, in the problems discussed in section 5, noting the symmetry gives the required information. If x, on the solution branch, also satisfies

$t(x, \lambda) = 0$

$t : D \subset R^n \times R \to R^m$, $m < n$, then it will be possible to replace certain components of f by components of t in such a way that the resulting system has full rank at (x_B, λ_B). In the case when $J(x, \lambda)$ is factorised we can first apply the method to (4.1) and then convergence to (x_B, λ_B) is linear. In solving systems of the form $J(x_i, \lambda_i)v = b$ we factorise $J(x_i, \lambda_i)$ into

$PJ(x_i, \lambda_i) = LU$

where P is a permutation matrix and U is upper triangular and L is unit lower triangular. We extend the decomposition to form

$$\begin{bmatrix} PJ(x_i, \lambda_i) \\ T(x_i, \lambda_i) \end{bmatrix} = \begin{bmatrix} L \\ W \end{bmatrix} [U].$$

where $T(x, \lambda) = \partial_x t(x, \lambda)$. When a pivot in the decomposition of J, from the kth row say, becomes small compared with the elements of J, we replace that row by a row of $T(x, \lambda)$, the jth say, which maximises the pivot. We then continue with a new system, in which $f_k(x, \lambda)$ in (4.1a) is replaced by $t_j(x, \lambda)$. This new system satisfies the conditions of Theorem 1 and so we can attain rapid convergence to (x_B, λ_B).

It is particularly convenient to use $\phi(x, \lambda) = \phi_3(x, \lambda)$ from section 3.3 since, on converging to (x_B, λ_B), the current value of w_i gives a good approximation to the zero eigenvector of $J(x_B, \lambda_B)$ which is useful when looking for a point on the secondary branch. (See [12], [20].)

5. NUMERICAL RESULTS

We have applied the methods of sections 2, 3 and 4 to several problems with success and we describe two which have appeared in the literature. The trussed dome problem [10], which was also considered in [19], is a physical example of stability loss. The dome of Fig. 3, if subjected to vertical forces at the nodes 1, 2, ..., 7, deforms until it loses stability at a turning point. Here $x(\lambda)$ defines the position of the nodes. The force at node i was $\lambda\beta_i$, i = 1, ..., 7, where $\beta_1 = 10^{-4}$, $\beta_j = 2 \times 10^{-4}$, j = 2, ..., 7. Fig 4 shows the vertical displacement, ξ, of the central node for varying λ and the turning point was found to be at $\lambda^* = 9.074147...$,

when for example, $\xi^* = 0.7865549...$ With the choices of $\phi = \phi_2$ and ϕ_3 the algorithm displayed second order convergence to (x^*, λ^*). The choice of $\phi(x, \lambda) = \text{Det}\,[J(x, \lambda)]$ suffered from the loss of significance described in section 3.3. Typical values of the relevant functions in the region of (x^*, λ^*) where

$\|f(x, \lambda)\| = 10^{-5}$, $|\phi_1(x, \lambda)| = 10^{-37}$, $|\phi_2(x, \lambda)| = 10^{-1}$, $|\phi_3(x, \lambda)| = 10^{-4}$

and so the choice of $\phi_1(x, \lambda)$ was less effective than the other choices.

The second problem was described by Simpson [23] and is the solution of the boundary value problem

$$-\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} = \lambda e^u, \qquad (x, y) \in D,$$

$$u(x, y) = 0 \qquad (x, y) \in \partial D,$$

where D is the unit square. The problem was discretised using the 9-point box form of the Laplacian (See Fox [9]) on a uniform mesh of size h. The resulting system is of the form (1.1) where λ appears nonlinearly. If m = 1/h, the problem is of dimension $(m-1)^2$ and is sparse, so we used the iterative method of Paige and Saunders [18] to solve the linear systems. We used the choices $\phi_2(x, \lambda)$ and $\phi_3(x, \lambda)$ and both were successful.

Fig. 5 shows how u(0.5, 0.5) varies with λ (calculated with h = 1/12). We calculated the turning point on mesh sizes h = 1/16 and h = 1/24 and derived the results :

h = 1/16 : λ^* = 6.8080865..., u(0.5, 0.5) =1.3916567...
h = 1/24 : λ^* = 6.80811698..., u(0.5, 0.5) = 1.3916603...

with convergence, in each case, being attained to more than the figures shown. These results for λ^* should be more accurate than those given by Simpson.

Typically, the number of iterations were the same for $\phi_2(x, \lambda)$ and $\phi_3(x, \lambda)$ with the correction (3.15). Without this correction, on average, using $\phi_3(x, \lambda)$ cost about one extra iteration. But in all cases the second order convergence to the turning point was apparent.

The method of section 4 was applied to finding the simple bifurcation point which occurs in the trussed dome problem. The value of Det[J(x, λ)] was monitored along [x(λ), λ] to bracket (x_B, λ_B) and then the method of section 4 was applied with $\phi(x, \lambda)$ given by $\phi_1(x, \lambda)$ and $\phi_3(x, \lambda)$ and with several of the obvious symmetry relations. The methods were again successful and, on choosing an appropriate symmetry relation, the convergence to (x_B, λ_B) was second order. The bifurcation point was found to be at

λ_B = 4.341092788...

where, for example ξ_B = 0.1796179807... Note that when using $\phi_3(x, \lambda)$, the initial choice of $w_o = J(x_o, \lambda_o)^{-1}\, d(x_o, \lambda_o)$ is not suitable since, as in this example, w_o may have a very small component in the direction of the appropriate eigenvector. For the bifurcation point problem we have found choosing w_o^T = (1, 1...., 1) is acceptable.

ACKNOWLEDGEMENT

The work in this paper was undertaken at the Computer Centre, Australian National University and the author would like to thank members of that department, particularly R. P. Brent and M. R. Osborne, for several helpful discussions. The author is also grateful to Professor W. C. Rheinboldt for several comments, a preprint of [20] and for a FORTRAN subroutine for the trussed dome problem discussed in [19].

REFERENCES

1. ABBOTT, J. P. : "Methods for finding several solutions of simultaneous nonlinear equations", Proc. 7th. Australian Computer Conference, Perth, 1976, pp. 1014-1022.
2. ABBOTT, J. P. : Ph. D. Thesis, Australian National University, 1977.
3. ANSELONE, P. M. and MOORE, R. H. : "An extension of the Newton-Kantorovich method for solving nonlinear equations with application to elasticity", J. Math. Anal. Appl., 13 (1966), pp. 476-501.
4. BAUER, L.; REISS, E. L. and KELLER, H. B. : "Axisymmetric buckling of hollow spheres and hemispheres", Comm. Pure Appl. Math., 23 (1970), pp. 529-568.
5. BRENT, R. P. : "Some efficient algorithms for solving systems of nonlinear equations", SIAM J. Numer. Anal., 10 (1973), pp. 327-344.
6. BROWN, K. M. : "A quadratically convergent Newton-like method based on Gaussian elimination", SIAM J. Numer. Anal., 6 (1969) pp. 560-569.
7. CRANDALL, M. G. and RABINOWITZ, P. H. : "Bifurcation from simple eigenvalues", J. Functional Analysis, 8 (1971), pp. 321-340.
8. DAVIS, J. : "The solution of nonlinear operator equations with critical points", Tech. Rep. No. 25, Department of Mathematics, Oregon State University, Corvallis, Oregon, 1966.
9. FOX, L. : "Numerical solution of ordinary and partial differential equations", Pergamon Press, 1962, p. 261.
10. HANGAI, Y. and KAWAMATA, S. : "Analysis of geometrically nonlinear and stability problems by static perturbation method", Univ. of Tokyo, Report of the Inst. of Ind. Science, Vol. 22, 5, No 143, 1973.
11. KELLER, H. B. and WOLFE, A. W. : "On the nonunique equilibrium states and buckling mechanism of spherical shells", J. Soc. Ind. Appl. Math., 13, (1965), pp. 674-705.
12. KELLER, H.B. and LANGFORD W. F. : "Iterations, perturbations and multiplicities for nonlinear bifurcation problems", Arch. Rat. Mech. Anal., 48 (1972), pp. 83-108.
13. MENZEL, R. and SCHWETLICK, H. : "Zur Behandlung von Singularitäten bei Einbettungsalgorithmen", manuscript, Mathematics Dept., Dresden Technical University, 1975.
14. MORÉ, J. J. and COSNARD, M. Y. : "Numerical comparison of three nonlinear equation solvers", Rep. TM-286, Argonne Nat. Laboratory, 1976.
15. OSBORNE, M. R. and MICHAELSON, S. : "The numerical solution of eigenvalue problems in which the eigenvalue appears nonlinearly, with an application to differential equations", Computer J., 7 (1964), pp. 66-71.
16. OSBORNE, M. R. : "A new method for the solution of eigenvalue problems", Computer J., 7 (1964), pp. 228-232.
17. OSBORNE, M. R. : "Numerical methods for hydrodynamic stability problems", SIAM J. Appl. Math., 15 (1967), pp. 539-557.

18. PAIGE, C. C. and SAUNDERS, M. A. : "Solution of sparse indefinite systems of equations and least squares problems", Stanford report, STAN-CS-73-399, Nov. 1973.

19. RHEINBOLDT, W. C. : "Numerical continuation methods for finite element applications",Tech. Rep. TR-454, Computer Science Center, University of Maryland, 1976.

20. RHEINBOLDT, W. C. : "A note on bifurcation iteration", manuscript, Computer Science Centre, University of Maryland, 1976.

21. RUHE, A. : "Algorithms for the nonlinear eigenvalue problem", SIAM J. Numer. Anal., 10 (1973), pp. 674-689.

22. RIKS, E. : "The application of Newton's method to the problem of elastic stability", J. Appl. Mech., Dec. 1972, pp. 1060-1065.

23. SIMPSON, R. B. : "A method for the numerical determination of bifurcation states of nonlinear systems of equations", SIAM J. Numer. Anal., 12 (1975), pp. 439-451.

24. KUBICEK, M. : "Dependence of solution of nonlinear systems on a parameter", Algorithm 502, CACM-TOMS, 2 (1976), pp. 98-107.

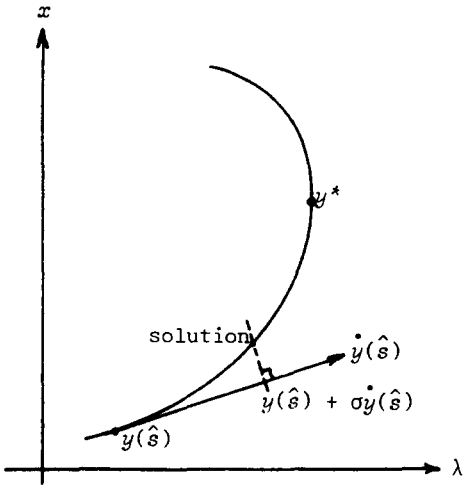

Fig. 1. One step with b = ẏ (ŝ).

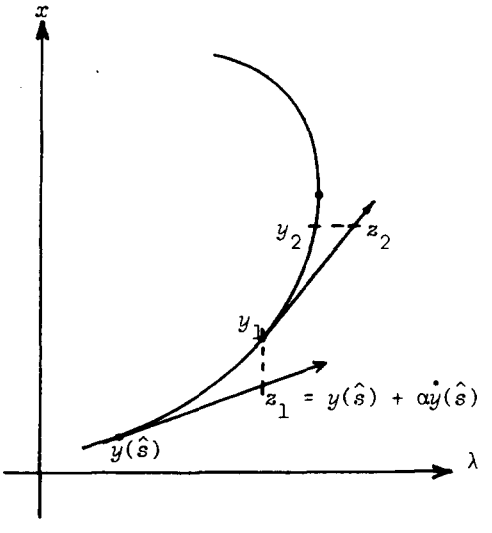

Fig. 2. Two steps, with b = e_2 then b = e_1. (z_1, z_2 are initial estimates of y_1, y_2)

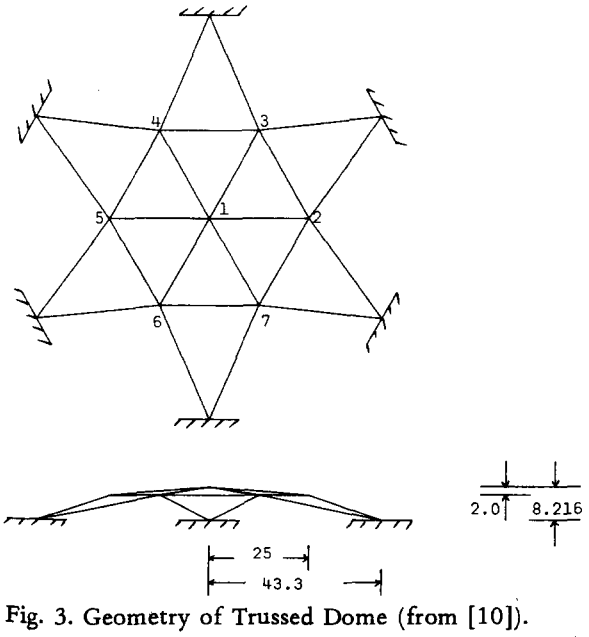

Fig. 3. Geometry of Trussed Dome (from [10]).

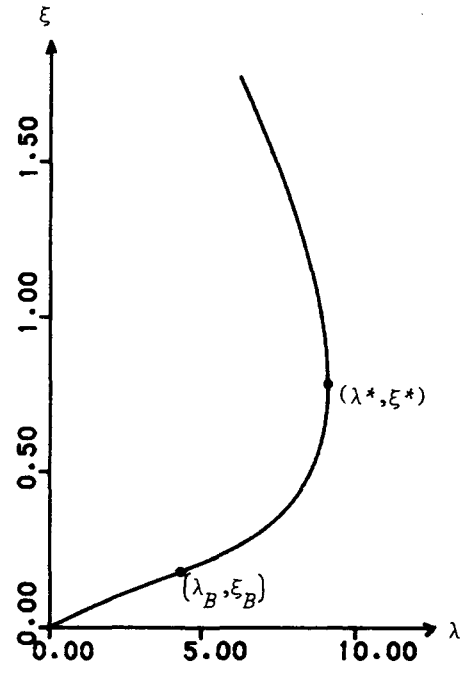

Fig. 4. Vertical displacement of central node (ξ) vs. λ.

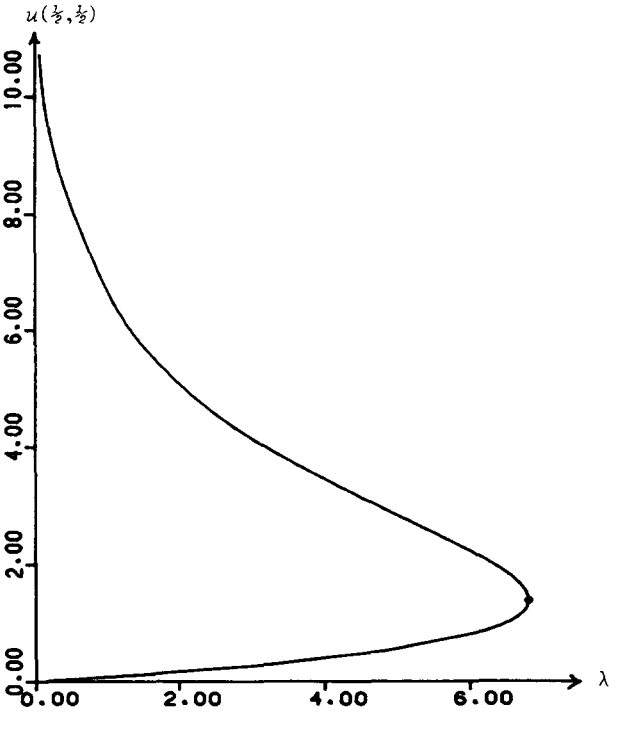

Fig. 5. u (1/2, 1/2) vs. λ.