SIAM J. MATRIX ANAL. APPL.
Vol. 18, No. 2, pp. 312–333, April 1997
© 1997 Society for Industrial and Applied Mathematics
005

HOMOTOPY METHOD FOR THE LARGE, SPARSE, REAL NONSYMMETRIC EIGENVALUE PROBLEM*

S. H. LUI†, H. B. KELLER‡, AND T. W. C. KWOK§

This paper is dedicated to Gene H. Golub on the occasion of his 65th birthday.

Abstract. A homotopy method to compute the eigenpairs, i.e., the eigenvectors and eigenvalues, of a given real matrix A_1 is presented. From the eigenpairs of some real matrix A_0, the eigenpairs of

$$A(t) \equiv (1-t)A_0 + tA_1$$

are followed at successive "times" from $t = 0$ to $t = 1$ using continuation. At $t = 1$, the eigenpairs of the desired matrix A_1 are found. The following phenomena are present when following the eigenpairs of a general nonsymmetric matrix:

- bifurcation,
- ill conditioning due to nonorthogonal eigenvectors,
- jumping of eigenpaths.

These can present considerable computational difficulties. Since each eigenpair can be followed independently, this algorithm is ideal for concurrent computers. The homotopy method has the potential to compete with other algorithms for computing a few eigenvalues of large, sparse matrices. It may be a useful tool for determining the stability of a solution of a PDE. Some numerical results will be presented.

Key words. eigenvalues, homotopy, parallel computing, sparse matrices, bifurcation

AMS subject classifications. 65F15, 65H17

PII. S0895479894273900

1. Introduction.

Given a real $n \times n$ matrix A, we wish to find some or all of its eigenvalues and eigenvectors. That is, we seek $\lambda \in \mathbb{C}$ such that

$$Ax = \lambda x$$

holds for nontrivial $x \in \mathbb{C}^n$. We call (x, λ) an eigenpair.

The QR algorithm (see Golub and Van Loan [9]) is generally regarded as the best sequential method for computing the eigenpairs. Briefly, the QR algorithm uses a sequence of similarity transformations to reduce a matrix to upper Hessenberg form. It then applies a sequence of Givens rotations from the left and right to reduce the size of the subdiagonal elements. When these elements are sufficiently small, the diagonal elements are taken to be approximations to the eigenvalues of the matrix. If the matrix is large and sparse, the QR algorithm suffers two serious drawbacks. In the reduction to Hessenberg form, the matrix usually loses its sparsity. Hence the algorithm requires the explicit storage of the entire matrix. This may pose a problem if the matrix is so large that not all of its entries can be accommodated within the main memory of the computer. A second drawback is that it is inherently a sequential algorithm due to the fact that Givens rotations must be applied sequentially. Bai and Demmel [3]

* Received by the editors September 7, 1994; accepted for publication (in revised form) by R. Freund April 29, 1996.
http://www.siam.org/journals/simax/18-2/27390.html
† Hong Kong University of Science & Technology, Department of Mathematics, Clear Water Bay, Kowloon, Hong Kong (shlui@uxmail.ust.hk). The work of this author was supported in part by RGC grant DAG92/93.SC16.
‡ California Institute of Technology, 217-50, Pasadena, CA 91125 (hbk@ama.caltech.edu).
§ Hong Kong University of Science & Technology, Department of Mathematics, Clear Water Bay, Kowloon, Hong Kong (mawkwok@uxmail.ust.hk).

somewhat circumvented the second problem by performing a "block" version of the QR algorithm. This improved version seems to work well on vector machines.

We now describe a homotopy method to compute the eigenpairs of a given matrix A_1. From the eigenpairs of some real matrix A_0, we follow the eigenpairs of

$$A(t) \equiv (1-t)A_0 + tA_1$$

at successive times from $t = 0$ to $t = 1$ using continuation. At $t = 1$, we have the eigenpairs of the desired matrix A_1. We call the evolution of an eigenpair as a function of time an eigenpath.

When A_1 is a real symmetric tridiagonal matrix with nonzero off-diagonal elements, a very successful homotopy method is known (see Li and Li [16] and Li, Zhang, and Sun [21]). The following phenomena, while absent in the symmetric tridiagonal case, are present for the general case:

- bifurcation,
- ill conditioning due to nonorthogonal eigenvectors.

The first can present computational difficulties if not handled properly. The homotopy method does not produce the Schur decomposition. Instead, it evaluates the eigenvalues and eigenvectors and hence is subject to the difficulty of ill conditioning.

Since the eigenpairs can be followed independently, this algorithm is ideal for parallel computers. We are primarily concerned with the case of a large, sparse, real matrix. We assume that all the nonzero entries of the matrix can be stored in each node of a parallel computer with distributive memory. Furthermore, we assume that the associated linear systems can be solved quickly, say, in $O(n^2)$ time.

As a simple illustration, we consider 2×2 matrices where the matrix A_0 is diagonal and whose elements are the diagonal elements of A_1:

$$A_0 = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}, \quad A_1 = \begin{bmatrix} a & d \\ c & b \end{bmatrix}.$$

The eigenvalues of $A(t)$ are

$$\frac{a+b \pm \sqrt{(a-b)^2 + 4t^2cd}}{2}.$$

Assuming $a \neq b$, three different situations arise (see Figure 1). In the first case, the two eigenvalues never meet for all t in $[0,1]$. In the second case, there is a double eigenvalue at some time $t \in (0,1]$ with the eigenpaths remaining real throughout. In the third case, there is a bifurcation point with the eigenpaths becoming a complex conjugate pair to the right of the bifurcation point. Typically, this is how complex eigenpaths arise from real ones. (Whenever a quantity is said to be complex, we mean it has a nontrivial imaginary component.) The situation for higher-dimensional matrices is similar except that an eigenpath can have more than one bifurcation point and the reverse of case three described above can occur (i.e., a complex conjugate pair of eigenpaths occur to the left of the bifurcation point and two real eigenpaths to the right). See Figure 2 for the eigenpaths of a random 10×10 matrix.

We now give a synopsis of the rest of the paper. In section 2, the homotopy method along with complex bifurcations will be presented. We will discuss some different types of bifurcations that may arise and identify the generic kind. We will derive an upper bound on the number of bifurcation points of all the eigenpaths. The numerical algorithm will be discussed in section 3. We will describe how to deal with

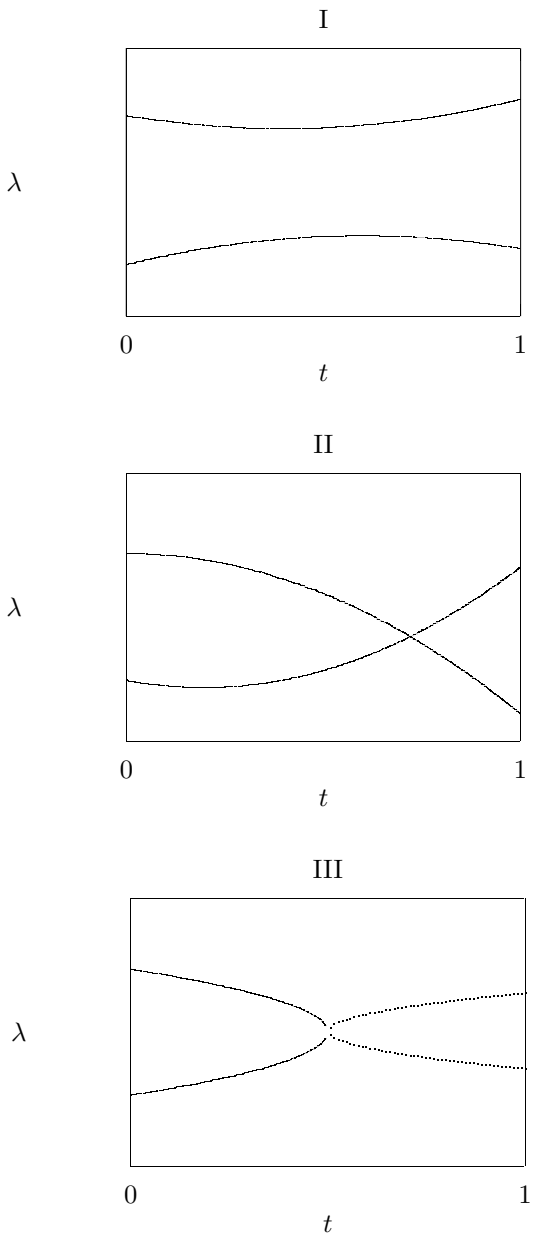

FIG. 1. *Eigenpaths of a* 2×2 *matrix. The dotted lines denote complex eigenpaths.*

bifurcations, how to choose the initial matrix, the selection of stepsizes etc. This will be followed by some numerical results. We will see that our homotopy method is impractical for dense matrices but has the potential to compete with other algorithms for finding a few eigenvalues of large, sparse matrices. Matrices of dimension 10^4 arising from the discretization of PDEs have been tested. In the final section, we recapitulate and suggest directions of further research.

Li, Zeng, and Cong [20] and Li and Zeng [19] have a very efficient homotopy

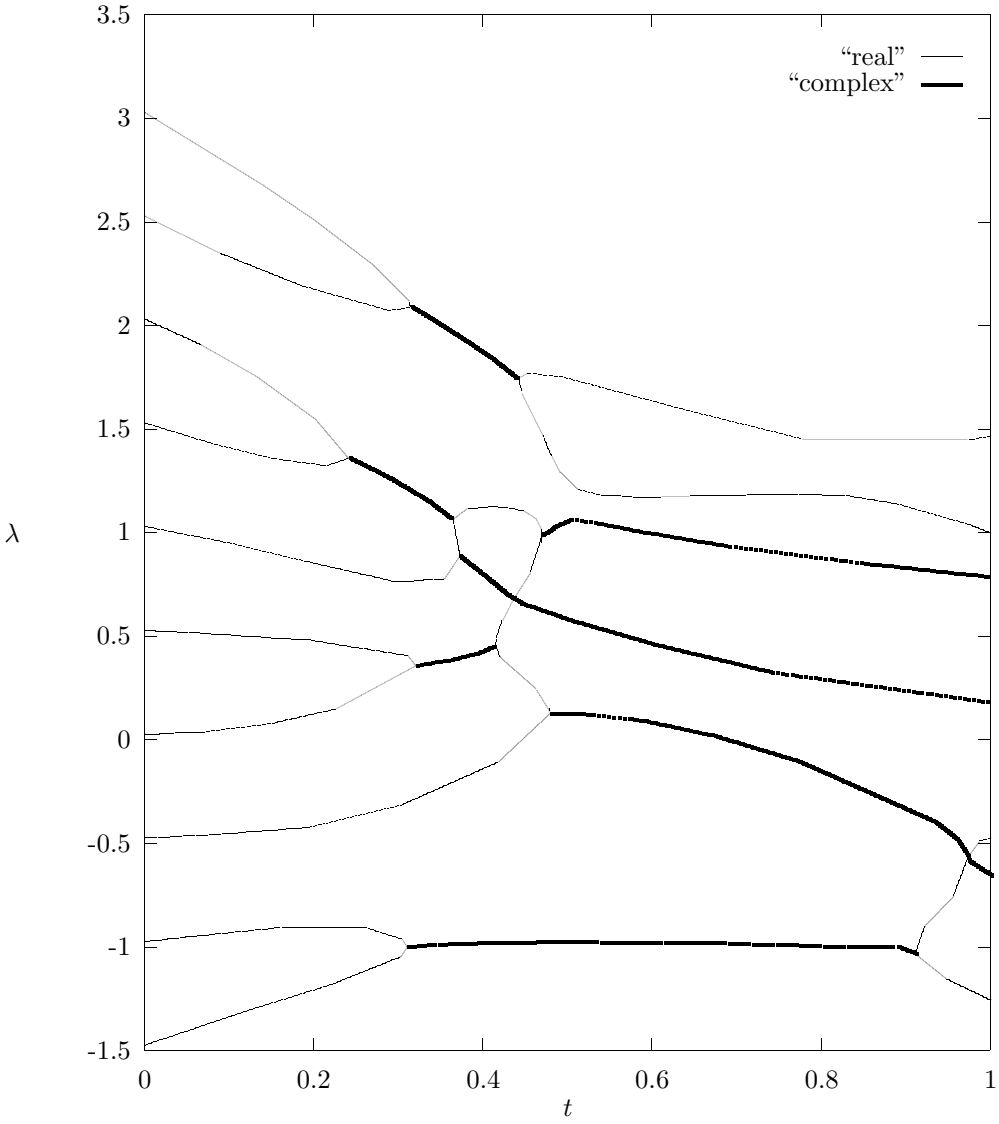

FIG. 2. *Eigenpaths of a random* 10×10 *matrix. Only one path of a complex conjugate pair of eigenpaths is shown.*

method for the dense matrix eigenvalue problem. For other approaches to the nonsymmetric eigenvalue problem see, for example, Cullum and Willoughby [5], Dongarra and Sidani [6], Saad [25], Shroff [28], Sorensen [29], Ruhe [24], and Bai, Day, and Ye [2]. The classic reference for the eigenvalue problem is the treatise by Wilkinson [30]. See also Saad [26] and Bai and Demmel [4] and the references therein.

Except for some of the numerical results, the work in this paper was completed by Lui [22]. In [20] Li, Zeng, and Cong proved Lemma A.1 (which they attribute to an unpublished work of Keller), which gives a necessary condition for a certain quantity $(\psi^*(G^0_{uu}\phi^2))$ to be nonzero. In this paper (Theorem 2), we give a necessary as well as

sufficient condition. Using analytic bifurcation theory, we identify the generic kinds of bifurcation which occur in following eigenpaths. We also give a bound on the number of bifurcation points in the eigenpaths. While the paper of Li, Zeng, and Cong addresses the dense eigenvalue problem, we address the complementary sparse case, although our algorithm has not had the same degree of success as theirs.

2. Homotopy method and complex bifurcation. In this section, we discuss some of the various phenomena that may arise on an eigenpath. Usually an eigenpath will be locally unique. That is, there are no other eigenpaths nearby. This can be characterized by a certain Jacobian being nonsingular. When this Jacobian is singular, bifurcation may occur. In other words, two or more eigenpaths may intersect at a point (u_0, t_0). Applying Henderson's work [10] on general analytic equations to our eigenvalue equations, we give a partial classification of some of the possible cases: simple quadratic fold, simple bifurcation point, simple cubic fold, and simple pitchfork bifurcation. We will show that the generic kind of bifurcation is the simple quadratic fold. In fact, the transition between real and complex eigenpaths (and vice versa) is via simple quadratic folds.

We first establish some notation. We use the superscripts ^{T} and * to denote the transpose and the complex conjugate transpose, respectively. The null and range spaces of a matrix are written as $\mathcal{N}()$ and $\mathcal{R}()$, respectively. The ith column of the identity matrix I is denoted by $\boldsymbol{e}_i$.

Given a real $n \times n$ matrix A_1, we form the homotopy

$$A(t) = (1-t)A_0 + tA_1, \quad 0 \le t \le 1, \tag{1}$$

where A_0 is a real matrix. We write the eigenvalue problem of $A(t)$ as

$$G(u,t) \equiv \begin{bmatrix} A(t)x - \lambda x \\ n(x) \end{bmatrix} = 0, \tag{2}$$

where u is the eigenpair (x, λ) of $A(t)$ and $n(x)$ is a normalization equation. In this paper, we take

$$n(x) = c^*x - 1,$$

where c is some fixed vector that is not orthogonal to x. The usual normalization $n(x) \equiv x^*x - 1$ is not differentiable, except at $x = 0$, and it only defines x up to a complex constant of magnitude one. We will always assume that every eigenvector x satisfies $c^*x \neq 0$; in section 3, we show how to choose c^*.

At this point, we make some remarks concerning the homotopy. It is known (Kato [12]) that the eigenvalues of $A(t)$ are analytic functions of t except at finitely many points where some eigenvalue may have an algebraic singularity. Away from these singularities, the eigenvectors *can be chosen* to be analytic functions of t. As we shall see, these singularities are typically encountered when an eigenvalue makes the transition from real to complex or vice versa.

Suppose an eigenpair u_0 is known at time t_0; i.e., $G(u_0, t_0) = 0$. We now describe how to obtain an eigenpair at a later time t_1. We must separate the discussion into different cases, depending on whether the Jacobian $G_u^0 \equiv G_u(u_0, t_0)$ is singular or not and on the nature of the singularity.

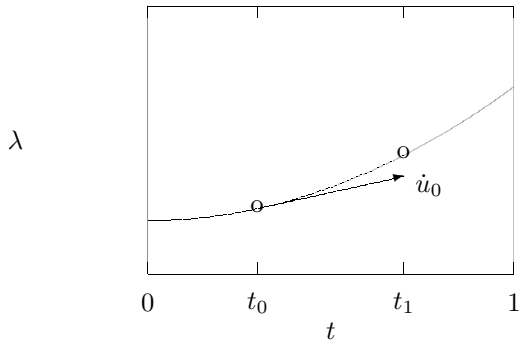

FIG. 3. *Euler–Newton continuation.*

2.1. Nonsingular Jacobian. When G_u^0 is nonsingular, the implicit function theorem tells us that locally about t_0 there is a unique solution $u(t)$ with $u(t_0) = u_0$. Differentiating (2) with respect to t and evaluating at t_0, we obtain

$$G_u^0 \dot{u}_0 + G_t^0 = 0,$$

where the dot denotes the t derivative and $G_t^0 \equiv G_t(u_0, t_0)$. Since G_u^0 is nonsingular, the above equation has a unique solution $\dot{u}_0$. To obtain the eigenpair at a later time t_1, we apply Newton's method to the equation $G(u, t_1) = 0$ with initial guess $u_0 + (t_1 - t_0)\dot{u}_0$. This is the Euler–Newton continuation method. The Euler step $(t_1 - t_0)\dot{u}_0$ is used to obtain the first Newton iterate (see Figure 3). Provided $t_1 - t_0$ is sufficiently small, the Newton iterates will converge quadratically to the eigenpair at t_1.

2.2. Singular Jacobian: Simple quadratic fold. Here we assume the eigenpair u_0 is real and

- G_u^0 has a one-dimensional null space spanned by, say, ϕ, and let ψ span the null space of $G_u^{0^T}$,
- $G_t^0 \notin \mathcal{R}(G_u^0)$,
- $a \equiv \psi^T(G_{uu}^0 \phi^2) \neq 0$.

Note that $G_{uu}^0 \phi^2$ is shorthand for $G_{uu}^0 \phi\phi$. The point (u_0, t_0) having the above properties is said to be a simple (real) quadratic fold point of equation (2). Pictorially, the real eigenpath is represented as the solid curve in Figure 4. Later, we will see that (1) λ_0 is an eigenvalue of $A(t_0)$ with algebraic multiplicity two and geometric multiplicity one and (2) $A'(t_0)x_0$ is not in the range of $[A(t_0) - \lambda_0 I, -x_0]$.

Since we can no longer use t to parametrize the solution, we employ the following pseudoarclength method due to Keller [13]. Augment (2) with the scalar equation

$$g(u, t, s) \equiv \phi^T \cdot (u - u_0) - (s - s_0) = 0.$$

This is the equation of a hyperplane whose unit normal is ϕ and is at a distance $s - s_0$ from u_0. Now define

$$F(u, t, s) \equiv \begin{bmatrix} G \\ g \end{bmatrix}. \tag{3}$$

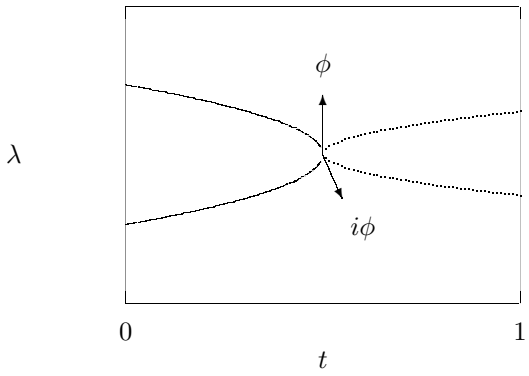

FIG. 4. *Complex conjugate pair of solutions on the opposite side of a simple real quadratic fold point. Dotted lines denote complex solutions.*

We immediately have $F(u_0, t_0, s_0) = 0$. It can be shown that the derivative of F with respect to (u, t) and evaluated at (u_0, t_0, s_0),

$$F^0_{(u,t)} = \begin{bmatrix} G^0_u & G^0_t \\ \phi^T & 0 \end{bmatrix}, \tag{4}$$

is nonsingular. Hence again by the implicit function theorem F has a locally unique solution $(u(s), t(s), s)$ with $u(s_0) = u_0$ and $t(s_0) = t_0$. In fact, the solution has the form

$$\begin{aligned} u(s) &= u_0 + \phi(s - s_0) + O(s - s_0)^2, \\ t(s) &= t_0 + \tau(s - s_0)^2 + O(s - s_0)^3, \end{aligned} \tag{5}$$

where

$$\tau = -\frac{1}{2}\frac{\psi^T(G^0_{uu}\phi^2)}{\psi^* G^0_t}.$$

From the definition of a simple quadratic fold, τ is well defined and nonzero. Note that $dt(s_0)/ds = 0$. We can apply the Euler–Newton continuation to the system $F = 0$ and follow the eigenpath around the fold point. Geometrically, the solution of $F = 0$ is the point at which the eigenpath punctures the hyperplane $g = 0$. Once around the fold point, t will begin to decrease. This is undesirable since our goal is to compute the eigenpair at $t = 1$. It turns out that a complex conjugate pair of eigenpaths will emerge to the right of the fold point. We now elaborate on this point.

Recall that a point $P_0 \equiv (u_0, t_0)$ is called a bifurcation point of the equation $G(u, t) = 0$ if in a neighborhood of P_0 there are at least two distinct branches of solutions $(u_1(s), t_1(s))$ and $(u_2(s), t_2(s))$ such that $u_i(s_0) = u_0$ and $t_i(s_0) = t_0$ for $i = 1, 2$. If at least one of these branches is complex, we will call P_0 a complex bifurcation point. When u_0 is real, (2) is a system of real equations. From the last paragraph, we know that locally about the point P_0 there is a unique path of *real* solutions. However, when considered as a system of equations over the complex numbers, Henderson and Keller [11] showed that P_0 is a complex bifurcation point with a complex conjugate pair of solutions on the opposite side of the real quadratic

fold (see Figure 4). Furthermore, the complex solutions have local expansions:

$$u(s) = u_0 + i\phi(s - s_0) + O(s - s_0)^2,$$
$$t(s) = t_0 - \tau(s - s_0)^2 + O(s - s_0)^3.$$

They are very similar in form to the real solution (5). Note that the tangent vector of the complex solution is a rotation of the tangent (ϕ) of the real solution. We can now use the Euler–Newton continuation with initial step in the direction $i\phi$ to find the complex eigenpairs at a later time.

The result of Henderson and Keller can be generalized to a complex quadratic fold point, i.e., $u_0 \in \mathbb{C}^{n+1}$, and satisfies the three properties outlined at the beginning of this section.

THEOREM 1 (Henderson [10]). *Let $G(u,t)$ be an analytic operator from $\mathbb{C}^{n+1} \times \mathbb{R}$ to $\mathbb{C}^{n+1}$. Let (u_0, t_0) be a simple quadratic fold point of $G(u,t) = 0$. Then in a small neighborhood of (u_0, t_0) there exist exactly two solution branches. They have the following expansions for small $|\epsilon|$:*

$$u_1(\epsilon) = u_0 + \epsilon e^{-i\alpha/2}\phi + O(\epsilon^2),$$
$$t_1(\epsilon) = t_0 - r\epsilon^2 + O(\epsilon^3),$$
$$u_2(\epsilon) = u_0 + i\epsilon e^{-i\alpha/2}\phi + O(\epsilon^2),$$
$$t_2(\epsilon) = t_0 + r\epsilon^2 + O(\epsilon^3),$$

where

$$re^{i\alpha} = \frac{\psi^*(G^0_{uu}\phi^2)}{2\psi^* G^0_t}.$$

2.3. Singular Jacobian: Simple quadratic bifurcation. Here, we assume the eigenpair u_0 is real and

- G^0_u has a one-dimensional null space spanned by, say, ϕ, and let ψ span the null space of $G^{0^T}_u$,
- $G^0_t \in \mathcal{R}(G^0_u)$,
- $a \neq 0$ and $b^2 - ac \neq 0$, where

$$a = \psi^T(G^0_{uu}\phi^2),$$
$$b = \psi^T(G^0_{uu}\phi\phi_0 + G^0_{ut}\phi),$$
$$c = \psi^T(G^0_{uu}\phi_0^2 + 2G^0_{ut}\phi_0),$$

and ϕ_0 is the unique solution of

$$G^0_u\phi_0 = -G^0_t \tag{6}$$

orthogonal to $\mathcal{N}(G^0_u)$.

The point (u_0, t_0) having the above properties is called a simple quadratic bifurcation point. In any small neighborhood of (u_0, t_0) there are exactly two distinct branches of solutions passing through the point (u_0, t_0) transcritically. If $b^2 - ac > 0$, then both branches are real. If $b^2 - ac < 0$, both branches are complex except at the point (u_0, t_0). See Henderson [10] for a more detailed discussion.

The tangent vectors of the two bifurcating branches can be computed and the Euler–Newton continuation can proceed as usual with these new directions. We will show that a simple quadratic bifurcation point is not likely to occur. Even if one existed, it would be transparent to a continuation method because it is highly unlikely that a numerical step would land exactly at the point.

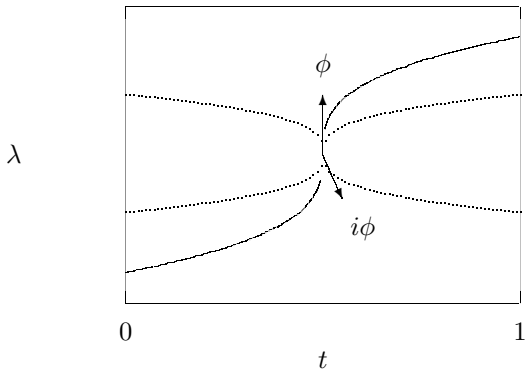

FIG. 5. *Cubic fold point.*

2.4. Singular Jacobian: Cubic fold point. Here, we assume the eigenpair u_0 is real and

- G_u^0 has a one-dimensional null space spanned by, say, ϕ, and let ψ span the null space of $G_u^{0^T}$,
- $G_t^0 \notin \mathcal{R}(G_u^0)$,
- $a \equiv \psi^T(G_{uu}^0\phi^2) = 0$,
- $\psi^T(G_{uu}^0\phi\phi_1) \neq 0$, where ϕ_1 is the unique solution of

$$G_u^0\phi_1 = -G_{uu}^0\phi^2 \tag{7}$$

  orthogonal to $\mathcal{N}(G_u^0)$.

The point (u_0, t_0) having the above properties is called a cubic fold point. It can be shown that (1) λ_0 is an eigenvalue of $A(t_0)$ with algebraic multiplicity three and geometric multiplicity one and (2) $A'(t_0)x_0$ is not in the range of $[A(t_0) - \lambda_0 I, -x_0]$. There is a unique branch of real solutions near (u_0, t_0) as well as a complex conjugate pair of solutions. See Figure 5. Cubic fold points are discussed, for example, in Yang and Keller [31] and Li and Wang [18]. Again, it will be seen that this case is not likely to occur in practice.

2.5. Singular Jacobian: Simple pitchfork bifurcation. Here, we assume the eigenpair u_0 is real and

- G_u^0 has a one-dimensional null space spanned by, say, ϕ, and let ψ span the null space of $G_u^{0^T}$,
- $G_t^0 \in \mathcal{R}(G_u^0)$,
- $a \equiv \psi^T(G_{uu}^0\phi^2) = 0$,
- $\psi^T(G_{uu}^0\phi\phi_1) \cdot \psi^T(G_{uu}^0\phi_0\phi + G_{ut}^0\phi) \neq 0$, where ϕ_0 and ϕ_1 were defined in (6) and (7).

The point (u_0, t_0) having the above properties is called a simple pitchfork bifurcation point. On one side of the point there are three real solutions. On the other side there is one real solution and a complex conjugate eigenpair. The situation is depicted in Figure 6. See Henderson [10] for a more detailed discussion.

2.6. Generic singular Jacobians. In the previous sections, we discussed four cases where the Jacobian G_u^0 has a one-dimensional null space. This list is of course not exhaustive. We will now see that of all the singularities only one, the simple quadratic fold, is likely to arise in the course of a calculation. The others are nongeneric.

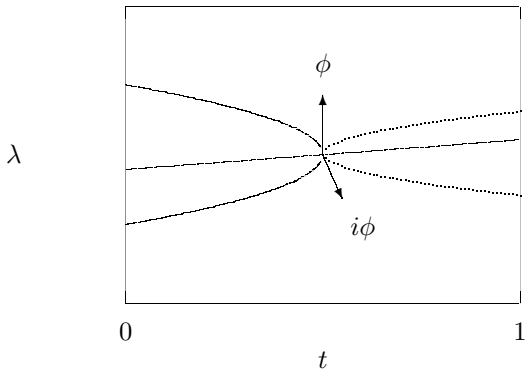

FIG. 6. *Simple pitchfork bifurcation.*

It is clear that of all the singular $n \times n$ matrices those with a one-dimensional null space are generic. Of the four cases considered, all but the first are nongeneric because they have nongeneric conditions $\psi^T(G^0_{uu}\phi^2) = 0$ and/or $G^0_t \in \mathcal{R}(G^0_u)$. The next result characterizes the generic singular Jacobian G^0_u.

THEOREM 2. *Let G be defined as in equation* (2). *Suppose for $(u_0, t_0) \in \mathbb{C}^{n+1} \times \mathbb{R}$, $G(u_0, t_0) = 0$ and G^0_u is singular with a one-dimensional null space. Let ϕ and ψ be spanning vectors for $\mathcal{N}(G^0_u)$ and $\mathcal{N}(G^{0^*}_u)$, respectively. Then $\psi^*(G^0_{uu}\phi^2) \neq 0$ iff λ_0 is an eigenvalue of $A^0 \equiv A(t_0)$ of algebraic multiplicity two and geometric multiplicity one.*

Proof. From (2), we obtain

$$G^0_u = \begin{bmatrix} A^0 - \lambda_0 I & -x_0 \\ c^* & 0 \end{bmatrix} \in \mathbb{C}^{n+1 \times n+1}.$$

Partition the null vectors as

$$\phi = \begin{bmatrix} h \\ \nu \end{bmatrix}, \quad \psi^* = [p^*, \mu],$$

where $h, p \in \mathbb{C}^n$ and $\nu, \mu \in \mathbb{C}$. By a direct calculation, we get

$$\psi^*(G^0_{uu}\phi^2) = -2\nu p^* h. \tag{8}$$

We rewrite the equation $\psi^* G^0_u = 0$, using the definitions of ψ^* and G^0_u, as

$$[p^*(A^0 - \lambda_0 I) + \mu c^*, \ -p^* x_0] = 0. \tag{9}$$

Taking the dot product of the first n components of the above vector with x_0, we obtain

$$p^*(A^0 - \lambda_0 I)x_0 + \mu c^* x_0 = 0.$$

Since $c^* x_0 = 1$,

$$\mu = 0. \tag{10}$$

The following two cases are the only possible ones in which $\dim \mathcal{N}(G^0_u) = 1$.

Case 1: λ_0 is an eigenvalue of A^0 with algebraic multiplicity $m \geq 2$ and geometric multiplicity one. Let

$$J \equiv Q^{-1}(A^0 - \lambda_0 I)Q = \left[\begin{array}{ccccc|c} 0 & 1 & & & & \\ & 0 & \ddots & & & \\ & & \ddots & 1 & & \\ & & & 0 & & \\ \hline & & & & & J_2 \end{array}\right] \tag{11}$$

be a Jordan form of $A^0 - \lambda_0 I$ where J_2 is nonsingular of dimension $n - m$ and x_0 is the first column of the matrix Q of principal (generalized) eigenvectors. Note that G_u^0 is similar to

$$\left[\begin{array}{cc} J & -\boldsymbol{e}_1 \\ c^*Q & 0 \end{array}\right].$$

Now from (9) and (10), we have

$$\begin{aligned} 0 &= p^*(A^0 - \lambda_0 I) \\ &= p^* Q J Q^{-1}. \end{aligned}$$

Let $y^* = p^*Q$. Then

$$y^* J = 0.$$

Thus from (11), we can take y^* to be $\boldsymbol{e}_m^*$.

From

$$G_u^0 \left[\begin{array}{c} h \\ \nu \end{array}\right] = 0,$$

we get

$$(A^0 - \lambda_0 I)h = \nu x_0. \tag{12}$$

Using (11) in the above, we obtain

$$QJQ^{-1}h = \nu x_0,$$

which implies that

$$Jw = \nu Q^{-1} x_0 = \nu \boldsymbol{e}_1,$$

where $w = Q^{-1}h$. From (11), we obtain the solutions $w = \alpha \boldsymbol{e}_1 + \nu \boldsymbol{e}_2$, where α is any complex number. Hence $y^* w = \nu \delta_{m2}$. Finally, from (8),

$$\begin{aligned} \psi^*(G_{uu}^0 \phi^2) &= -2\nu (p^*Q)(Q^{-1}h) \\ &= -2\nu y^* w \\ &= -2\nu^2 \delta_{m2}. \end{aligned}$$

Note that $\nu \neq 0$ since otherwise $w = \alpha \boldsymbol{e}_1$, which implies $h = \alpha x_0$. Since $c^*h = 0$ and $c^*x_0 = 1$, we must have $\alpha = 0$. We have reached a contradiction that ϕ is the zero vector. Hence $\psi^*(G_{uu}^0 \phi^2)$ is nonzero iff $m = 2$.

Case 2: λ_0 is an eigenvalue of A^0 with algebraic multiplicity $m \geq 2$ and geometric multiplicity two. Let

$$J \equiv Q^{-1}(A^0 - \lambda_0 I)Q = \left[\begin{array}{cc|c} J_1 & 0 & \\ 0 & J_2 & \\ \hline & & J_3 \end{array}\right] \tag{13}$$

be a Jordan form of $A^0 - \lambda_0 I$ where J_1 and J_2 are Jordan blocks of sizes m_1 and m_2, respectively, with $m_1 + m_2 = m$; J_3 is nonsingular and of dimension $n - m$; and x_0 is the first column of the matrix Q of principal eigenvectors. J_1 and J_2 have zeros on the diagonal. If J_1 is diagonal then, as before, we have from (12),

$$Jw = \nu \boldsymbol{e}_1,$$

where $w = Q^{-1}h$. From the form of J, it is clear that $\nu = 0$. Hence

$$\psi^*(G^0_{uu}\phi^2) = -2\nu p^* h = 0.$$

Finally, if J_1 is a nondiagonal Jordan block so that $m_1 > 1$, then J has at least two linearly independent left null vectors ($\boldsymbol{e}^*_{m_1}$ and $\boldsymbol{e}^*_m$). This implies that G^0_u has at least two linearly independent left null vectors ($[\boldsymbol{e}^*_{m_1}Q^{-1}, 0]$ and $[\boldsymbol{e}^*_m Q^{-1}, 0]$). (For example,

$$[\boldsymbol{e}^*_m Q^{-1}, 0]\, G^0_u = [\boldsymbol{e}^*_m Q^{-1}, 0] \left[\begin{array}{cc} Q & 0 \\ 0 & 1 \end{array}\right] \left[\begin{array}{cc} J & -\boldsymbol{e}_1 \\ c^*Q & 0 \end{array}\right] \left[\begin{array}{cc} Q^{-1} & 0 \\ 0 & 1 \end{array}\right] = 0$$

since $m > 1$ and $\boldsymbol{e}^*_m$ is a left null vector of J.) This contradicts the assumption that $\dim \mathcal{N}(G^0_u) = 1$.

Note that if λ_0 is an eigenvalue of A^0 of geometric multiplicity greater than two, it can be checked that the dimension of the null space of G^0_u is at least two. We have established the claim of the theorem. □

See also Li, Zeng, and Cong [20].

The fact that the generic case of a singular G^0_u occurs when λ_0 is an eigenvalue of A^0 of algebraic multiplicity two and geometric multiplicity one may seem surprising. We now attempt to give an intuitive explanation. Let X be the set of $n \times n$ matrices which have λ_0 as an eigenvalue of algebraic multiplicity two. Suppose A is a member of X. Now $A - \lambda_0 I$ can be similarly transformed to one of

$$\left[\begin{array}{cc|c} 0 & 1 & \\ 0 & 0 & \\ \hline & & J_1 \end{array}\right] \quad \text{or} \quad \left[\begin{array}{cc|c} 0 & 0 & \\ 0 & 0 & \\ \hline & & J_2 \end{array}\right],$$

where J_1 and J_2 are some nonsingular matrices. The rank of the left and right matrices are $n - 1$ and $n - 2$, respectively. Hence in the space X the matrix $A - \lambda_0 I$ with geometric multiplicity one (i.e., similar to the left matrix) is generic.

Using the notation of Theorem 2, we can show the following corollary.

COROLLARY 1. *Suppose $n \geq 4$ and $\mathcal{N}(G^0_u)$ is one dimensional. Then, generically, λ_0 is an eigenvalue of A^0 of algebraic multiplicity two and geometric multiplicity one and is real.*

Proof. Let X_r be the set of real $n \times n$ matrices with one real eigenvalue of algebraic multiplicity two and geometric multiplicity one and all other eigenvalues simple and let X_i be the set of real $n \times n$ matrices with one complex conjugate pair of eigenvalues

of algebraic multiplicity two and geometric multiplicity one and all other eigenvalues simple. Define $X = X_r \cup X_i$. From Theorem 2, the generic case of a one-dimensional $\mathcal{N}(G_u^0)$ implies that $A^0 \in X$. We now show that X_r is generic in X.

For each $A \in X_r$, we associate $V(A) \equiv (A, Y, \lambda, d_3, \ldots, d_n)$, where Y is a real $n \times n$ matrix, λ is the unique multiple eigenvalue of A, and $d_3, \ldots, d_n$ are real numbers. In the case in which all the eigenvalues of A are real, the columns of Y can be considered as the generalized eigenvectors of A and d_j as the eigenvalues. If A has a complex eigenvalue μ with eigenvector z, then we could take $\mu = d_3 + id_4$ and $z = y_3 + iy_4$, for example. (y_j denotes the jth column of Y.) Note that $(\overline{\mu}, \overline{z})$ is also an eigenvalue–eigenvector pair of A. The point is that the information contained in Y and d_j is enough to determine the eigenvalues and eigenvectors of A. In the case in which all eigenvalues are real, $V(A)$ must satisfy

$$AY = YJ, \quad J = \left[\begin{array}{cc|ccc} \lambda & 1 & & & \\ 0 & \lambda & & & \\ \hline & & d_3 & & \\ & & & \ddots & \\ & & & & d_n \end{array}\right].$$

If complex eigenvalues exist, the above must be appropriately modified. In addition, there are n normalization equations for the eigenvectors. Thus, $V(A)$ consists of $2n^2 + n - 1$ real variables which must satisfy $n^2 + n$ real polynomial equations and thus has $n^2 - 1$ degrees of freedom.[1]

For $A \in X_i$, let $V(A) \equiv (A, Y, \lambda_r, \lambda_i, d_5, \ldots, d_n)$, where $\lambda \equiv \lambda_r + i\lambda_i$ is the complex eigenvalue of A of algebraic multiplicity two. The Jordan form (in the case in which all other eigenvalues are real) is

$$J = \left[\begin{array}{cc|cc|ccc} \lambda & 1 & & & & & \\ 0 & \lambda & & & & & \\ \hline & & \overline{\lambda} & 1 & & & \\ & & 0 & \overline{\lambda} & & & \\ \hline & & & & d_5 & & \\ & & & & & \ddots & \\ & & & & & & d_n \end{array}\right].$$

Thus, $V(A)$ consists of $2n^2 + n - 2$ real variables and must also satisfy $n^2 + n$ real equations and thus it has $n^2 - 2$ degrees of freedom. Hence we see that X_r is generic.

We remark that the equations $AY = YJ$ and the normalization equations are linearly independent. If one normalization equation is omitted, then the length of some eigenvector is not uniquely determined. Also, if one of the real equations in $AY = YJ$ is omitted, then we may not have an eigenvalue–eigenvector pair. Also, in the above calculation we actually include matrices with eigenvalues of higher multiplicities and other multiple eigenvalues (besides λ). This is acceptable because they are nongeneric in X. □

At simple quadratic folds and simple quadratic bifurcation points the eigenvalue has algebraic multiplicity two and geometric multiplicity one. At both cubic fold and simple pitchfork bifurcation points the algebraic and geometric multiplicities are

[1] In the language of algebraic geometry, $V(A)$ is a variety and the degrees of freedom correspond to the dimension of the variety.

TABLE 1

Summary of some of the different types of points at a singular Jacobian G_u^0. With the exception of the quadratic fold, additional generic conditions must be satisfied for all.

	$\psi^* G_t^0 \neq 0$	$\psi^* G_t^0 = 0$
$\psi^* G_{uu}^0 \phi^2 \neq 0$	simple quadratic fold	simple quadratic bifurcation
$\psi^* G_{uu}^0 \phi^2 = 0$	simple cubic fold	simple pitchfork bifurcation

three and one, respectively. See Table 1. The Jacobian G_u^0 of course may have other types of nongeneric singularities. For example, the eigenvalue may have multiplicities three and two, respectively. However, these are nongeneric and unlikely to occur in practice.

The significance of the above theory is that in practice we encounter only simple real quadratic folds, and this is the route by which real eigenpaths become complex.

2.7. A bound on the number of bifurcation points. It is not difficult to show that at a real or complex bifurcation point of (2) the algebraic multiplicity of the eigenvalue of $A(t)$ is at least two. Let

$$p(t, \lambda) \equiv \det(A(t) - \lambda I).$$

Since $A(t)$ is linear in t, the above is a polynomial in (t, λ) of degree n. In fact, p can be written in the form

$$p(t, \lambda) = a_0(t) + a_1(t)\lambda + \cdots + a_n(t)\lambda^n, \tag{14}$$

where $a_i(t)$ is a polynomial in t of degree at most $n - i$ for $i = 0, \ldots, n$ and $a_n(t) = (-1)^n$. Define

$$q(t, \lambda) = \frac{\partial p(t, \lambda)}{\partial \lambda}.$$

From (14), it is easy to show that q is a polynomial of degree $n - 1$. At a bifurcation point (t, λ) we must have

$$p(t, \lambda) = q(t, \lambda) = 0.$$

This is a system of two polynomial equations of degrees n and $n - 1$ in two variables. By Bézout's theorem, it has at most $n(n - 1)$ roots. *Hence the eigenpaths collectively can have at most $n(n - 1)$ bifurcation points.*

We remark that some of these roots may have a complex time t and that some roots may lie outside the region of interest (i.e., $t \in [0, 1]$). In practice we usually see on the order of n bifurcation points.

3. Numerical algorithm. In this section, we describe the numerical implementation of the homotopy algorithm including choice of the initial matrix A_0, stepsize selection, and transition from real to complex eigenpairs and vice versa. For a more thorough treatment of some of these topics, see Keller [14] and Allgower and Georg [1].

Suppose that we have computed the eigenpairs at time t_0. The normalization equation for the eigenvector x at the new time is taken to be

$$x_0^* x - 1 = 0,$$

where x_0 is the eigenvector at time t_0. We always perform real arithmetic so that the pseudoarclength formulation (3) is written as an equivalent system of $2n + 4$ real equations whenever we are following a complex eigenpath.

3.1. Choice of initial matrix A_0. The constraint that the eigenpairs of A_0 be computable quickly severely limits the choice of A_0. Ideally, A_0 should be chosen so that the number of real and complex bifurcation points are minimized. This is because there is extra work involved in locating real fold points. In the example shown in Figure 2, A_0 is a diagonal matrix. By simply reordering the diagonal elements of this A_0 it is possible for the eigenpaths to have just three real fold points. This is the minimum possible because this A_1 has six complex eigenvalues. There are no "unnecessary" fold points. Another desirable property of A_0 is that the eigenpaths be well separated. This decreases the chance of the path-jumping phenomenon. However, it seems extremely difficult to choose a priori an initial matrix which has all of the above properties.

We tried three different kinds of initial matrices: real diagonal, real block diagonal with 2×2 diagonal blocks, and block upper triangular with 2×2 diagonal blocks. We now describe them in more detail.

The real diagonal initial matrix is defined as follows. Let a denote the trace of A_1 divided by n, the size of the matrix. This is the average value of the eigenvalues of A_1. Let ρ be the square root of the maximum of the Gerschgorin radii of A_1. Define the diagonal elements of A_0 as equally distributed points in $[a-\rho, a+\rho]$ in ascending order. There is no theoretical justification for this choice of A_0 except that the eigenvalues are initially simple and the eigenvectors are just the standard basis vectors. Without the square root in the definition of ρ, numerical experiments on random matrices show that the initial eigenvalue distribution is too spread out. An alternative is to simply use the diagonal part of A_1 as the initial matrix. One problem here is that this initial matrix may have multiple eigenvalues, leading to potential difficulties.

For a real diagonal initial matrix, the eigenpaths are real initially. As we shall see, the resultant homotopy usually has a large number of "unnecessary" fold points. As an attempt to remedy the situation, we tried initial matrices which have complex eigenvalues. One avenue is to try an A_0 which is real block diagonal with 2×2 diagonal blocks of the form

$$\begin{bmatrix} \alpha & \beta \\ -\beta & \alpha \end{bmatrix}.$$

The eigenvalues of this block are $\alpha \pm i\beta$. The pairs (α, β) are chosen as uniformly distributed in the square box in the complex plane with center at the point $a + 0i$ (the average of the eigenvalues of A_1) and width 2ρ, where ρ was defined in the above paragraph. Now the eigenpaths start out complex. Since the complex space is much bigger than the real space, there is less likelihood of two eigenpaths venturing close together (hence less chance of path jumping) and less possibility of encountering fold points.

The final kind of initial matrix we consider is block upper triangular with 2×2 diagonal blocks. The upper triangular part of the matrix is taken to be the upper

triangular part of A_1 and the 2×2 diagonal blocks are as defined above. We define the 2×2 diagonal blocks this way, instead of copying those of A_1, to avoid possible multiple eigenvalues in the beginning. The eigenpairs of this initial matrix can be found quickly. The motivation for this initial matrix is that it is closer to A_1 than previous initial matrices. A smaller $\|A_1 - A_0\|$ should lead to straighter eigenpaths and possibly fewer fold points.

Some very limited experiments with 100×100 random matrices confirm our observations. A diagonal initial matrix leads to many more fold points than the other two initial matrices. The third type of initial matrix performs marginally better than the second type.

3.2. Transition at real fold points. We first describe the transition from a real eigenpath to a complex one. When it detects that it is going backwards in time, then, generically, a real fold has been passed. By the theory of the last section, there must be a complex conjugate pair of solutions on the opposite side of the real fold. We first get a more accurate location of the fold point by using the secant method to approximate the point at which $dt/ds = 0$. (Recall that this is a necessary condition at a fold point.) With the augmented system, the Jacobian (4) is nonsingular, so there is no numerical difficulty in the task. We store the location of this fold point in a table for later reference. Using the tangent vector ϕ at the fold point, we solve problem (2) in complex space at a later time. This is done by carrying out the Euler–Newton continuation with the initial tangent $i\phi$, in accordance with the theory of Henderson and Keller.

When the partner of the above path comes from the other arm of the same fold, it checks that the fold point has been visited before and stops further computation. This way, only one path of a complex conjugate pair of eigenpaths is computed.

The reverse of the above situation also arises, although less frequently. That is, time decreases while advancing along a complex path. Generically, there must be a real fold on the opposite side of this complex path. Once the fold point has been located, we compute the real tangent vector ϕ. We then apply the Euler–Newton continuation in both the directions ϕ and $-\phi$. See Figure 7. Because the problem is being solved in real space, there is no chance of converging back to the complex solution. On a parallel computer, a node which became idle at another fold point can be invoked to carry out the computation along one of these directions. If we begin with k complex eigenpaths, we may end up with many more than k eigenpaths because of these complex-to-real bifurcations. Fortunately, in practice, at most a few more have been encountered.

3.3. Computing the tangent. Suppose two eigenpairs u_0 and u_1 have been found. We wish to compute the tangent vector at t_1. In formulation (3), we have

$$F_u^1 \dot{u}_1 + F_t^1 \dot{t}_1 = 0,$$

where the superscript ^{1} denotes the evaluation of the Jacobian at (u_1, t_1) and the dot denotes the s derivative. For a unit tangent, we require in addition that

$$\dot{u}_1^* \dot{u}_1 + \dot{t}_1^2 = 1. \tag{15}$$

Note that the above two equations define the tangent up to a sign. To ensure that we are always computing in the same direction, we further impose the condition

$$\Re(\dot{u}_0^* \dot{u}_1) + \dot{t}_0 \dot{t}_1 > 0.$$

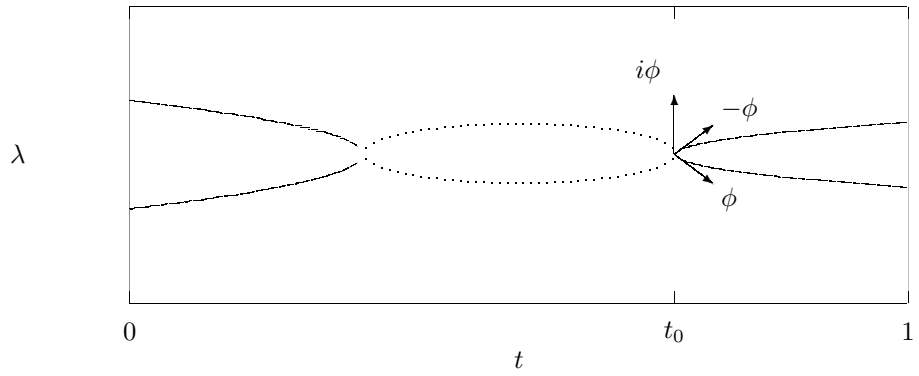

FIG. 7. *Transition from a complex solution to a real solution at a fold point. Dotted lines denote complex solutions.*

Because (15) is nonlinear, we instead solve the linear system (when u_0 is real)

$$\begin{bmatrix} F_u^1 & F_t^1 \\ \dot{u}_0^T & \dot{t}_0 \end{bmatrix} \begin{bmatrix} \phi \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

The tangent $(\dot{u}_1, \dot{t}_1)$ is obtained by normalizing the solution of the above system.

3.4. Selection of stepsize. Suppose we have the two eigenpairs u_0 and u_1. We obtain stepsize δs_2 for u_2 as follows:

$$\delta s_2 = \delta s_1 \ (\Re(\dot{u}_0^* \dot{u}_1) + \dot{t}_0 \dot{t}_1 + .5),$$

where δs_1 is the stepsize used to obtain u_1. The idea is that when the two previous tangents are parallel we increase the stepsize by 50%. If the tangents are perpendicular, we decrease the stepsize by a half. We use the above scheme until the time is close to one, at which time we solve the system $G(u, 1) = 0$.

Whenever a Newton iteration fails to converge after, say, six iterations, we restart it with a stepsize that is one-half of the original one.

3.5. Path jumping. Path jumping is a serious problem for the homotopy method. This is the phenomenon in which the Newton iteration converges to another eigenpath. This occurs when the stepsize is overly ambitious or the linear system involved in the solution of a Newton iterate has a large condition number. The latter situation arises whenever eigenvalues are poorly separated.

An elegant method of detecting path jumping is available when the matrix is symmetric tridiagonal with nonzero off-diagonal elements (Li and Rhee [17]). They employ the Sturm sequence property of symmetric matrices. Li and Zeng [19] can also detect path jumping in the case in which the eigenpath is real and the matrix is in Hessenberg form. However, no satisfactory procedure is known for general matrices. One inefficient way is to use the property that the sum of the eigenvalues of a matrix is equal to the trace of the matrix. Noting that

$$\mathrm{Tr}(A(t)) = \mathrm{Tr}(A_0) + t(\mathrm{Tr}(A_1 - A_0)),$$

almost all path jumps can be detected by comparing the sum of the computed eigenvalues and the above expression for the trace of $A(t)$. However, this does not tell us

which path has jumped, and hence it is necessary to recompute the last step for all eigenpaths. Other drawbacks include the necessity of synchronizing the computation of the eigenpaths and the fact that this method works only if all the eigenpaths are computed.

Our approach is perhaps the simplest, but certainly not the best. We keep track of the initial eigenvalue (at $t = 0$) of each eigenpath, and for each eigenpath that has been computed more than once (this is checked at $t = 1$) we repeat the entire calculation for those eigenpaths with a smaller stepsize.

3.6. Parallel aspects. The homotopy method is fully parallel because each eigenpath can be computed independently of the others. If the sparse matrix $A(t)$ can be stored in each node, then there is no communication overhead at all other than the trivial broadcast of the location of fold points.

3.7. Homotopy algorithm of Li, Zeng, and Cong. Li, Zeng, and Cong [20] use a different strategy in their homotopy algorithm. They first use Householder transformations to reduce the given matrix to a similar matrix A_1 in upper Hessenberg form. Their initial matrix A_0 is the same as A_1 except that one subdiagonal entry is set to zero. They use a divide-and-conquer strategy to obtain the eigenpairs of A_0. Because A_0 is very close to A_1, the eigenpaths will be nearly straight and path jumping is much less of a problem here. The performance of this method is very encouraging. However, it requires storage of the entire matrix plus large amounts of work storage. For another approach to finding the eigenvalues using homotopy, see Lenard [15].

4. Numerical results. We have done very limited testing on random matrices and matrices arising from the finite difference approximations of partial differential equations. The tests were performed on SUN Sparc workstations. In our code, we computed the eigenpairs one at a time. As mentioned already, in a parallel code each eigenpair can be assigned to a separate processor.

We use an initial $ds = .1$, a final tolerance of 10^{-12}, and an intermediate tolerance of 10^{-4}. The final tolerance means that the stopping criterion for the Newton iteration is that the norm of the Newton step is less than 10^{-12} at $t = 1$. Intermediate tolerance refers to the stopping criterion at $t < 1$. The criterion for stopping the iteration to locate a fold point is $|\dot{t}| < 10^{-3}$.

Empirically, we notice that the eigenpaths move in a relatively simple fashion as t progresses. That is, there are no wild oscillations. Thus, the homotopy method has the potential to find efficiently a few special eigenvalues, for example, those with the largest real part. Such eigenvalues are of interest in linear stability theory for partial differential equations. It is in this area that we believe the homotopy method will be most useful.

Our first set of test examples comes from the usual second-order finite difference discretization of the elliptic operator

$$\triangle + f(x,y)\frac{\partial}{\partial x} + g(x,y)\frac{\partial}{\partial y} + p(x,y) \tag{16}$$

on a rectangle of size 1×1.2 with homogeneous Dirichlet boundary condition. We choose the initial matrix as the discrete Laplacian whose eigenpairs are known. We make the following changes to the algorithm to account for the nature of the problem. Assuming a uniform mesh size h in both x and y, the modified equation for the tangent

TABLE 2
Execution times for five eigenpaths of matrices of various sizes corresponding to the discretizations of a PDE with different grid sizes.

size	238	696	1394	3510	10622
time (sec)	4	18	49	197	1619

is

$$h^2\dot{v}_0^*\dot{v} + \dot{\lambda}_0\dot{\lambda} + \dot{t}_0\dot{t} = 1.$$

Here, $\dot{v}$ denotes the s derivative of the eigenvector and the subscript denotes the corresponding quantity at the previous time t_0. The reason for the modification is that this approximates the underlying continuous equation

$$\int \dot{v}_0\dot{v}dxdy + \dot{\lambda}_0\dot{\lambda} + \dot{t}_0\dot{t} = 1.$$

Similarly, we employ the following pseudoarclength condition:

$$h^2\dot{v}_0^*(v - v_0) + \dot{\lambda}_0(\lambda - \lambda_0) + \dot{t}_0(t - t_0) - ds = 0.$$

For the numerical experiments, we take a uniform 95×114 grid leading to a matrix of dimension 10622. We follow the five paths whose initial eigenvalues are largest with the aim of computing the five eigenvalues of the PDE having the largest real parts. Our Fortran code uses GMRES [27] to solve each linear system. (An alternative is the QMR method of Freund and Nachtigal [8].) Here, only the nonzero entries of the matrices need be stored. The average number of time steps per eigenpath is 5 and the number of Newton iterations per step is 1. The program successfully computed the five eigenpairs with the five largest real parts in a number of examples that we tried. These computed paths all turned out to be real. Execution times for various choices of the coefficients of the PDE are between 27 and 28 minutes.

In Table 2, we give the execution times for computing five eigenpaths for the PDE with coefficients $f = e^x - 2y^2, g = y^2\cos(2x), p = 0$ for various grid sizes. The maximum dimension of the Krylov subspace, a parameter of GMRES, was set at 100 for all the test runs. Hence, the execution time for smaller matrices is more favorable than for larger matrices. The complexity is slightly less than $O(n^2)$.

We also tried a symmetric problem (with $f = g = 0$ and various choices of p). The execution times are between 16 and 22 minutes for matrices of size 10622.

We have not been able to devise a mechanism to guarantee that an eigenpath will end up (at $t = 1$) having an eigenvalue with the largest real part, even for the scalar PDE above. Problems which arise in practice (for example, in fluid mechanics) often involve systems of PDEs. It would be very difficult to obtain any theoretical result in this direction.

As our final illustration, we compute singular points of a parameter-dependent scalar PDE which arises in population biology. The PDE is

$$\triangle u + \alpha f(u) + \gamma u_x = 0$$

on a rectangular domain of sides of widths 1 and 1.2, and homogeneous Dirichlet boundary conditions are imposed. This is a population model for insects in a domain with a constant prevailing wind of strength γ in the $-x$ direction. Here, u represents

the population of the insects and α is a parameter depending on the birth rate and diffusion coefficient. The boundary conditions mean that the exterior of the domain is completely hostile to the insects. See Murray [23] for further details. We will only consider the Fisher model; i.e., $f(u) = u(1-u)$. The problem is to determine values of α for which the PDE becomes singular. Such points are of interest because bifurcation typically occurs there. These points are special because there solutions lose/gain stability. Singular points occur when the corresponding linearized eigenvalue problem (linearized about $u = 0$)

$$\triangle v + \alpha f'(0)v + \gamma v_x = \lambda v$$

has a zero eigenvalue. Hence the problem reduces to finding a zero eigenvalue of the matrix which arises from the discretization of the above equation.

Here is how the algorithm proceeds. Using the matrix which arises from the discretization of the Laplacian as the initial matrix, we use the homotopy algorithm to find the largest eigenvalue at $\alpha = 0$ (where all the eigenvalues are negative). We then follow this eigenpath at increasing values of α until the eigenvalue becomes positive. At that point, we use the secant method to locate the zero of the eigenvalue (as a function of α). For the eigenvalue problem at α_{i+1}, we use the corresponding matrix at α_i as the initial matrix.

Dividing the rectangle into a uniform 95×114 grid, we obtain a matrix of size 10622. For a wind strength $\gamma = 1$, the code computed the eigenvalue at $\alpha = 0, 5, 10, 15,$ and 20. Discovering that the eigenvalue becomes positive at the last value of α, it proceeded to compute the critical value $\alpha^* = 16.97\ldots$ in one step of the secant method. It found α^* with the eigenvalue at that point on the order of 10^{-12}. The entire procedure took 534 seconds, with the first eigenvalue solve at $\alpha = 0$ taking 416 seconds and the rest of the calculation taking about 120 seconds. This example illustrates the power of the homotopy method. When the initial matrix and the final matrix do not differ significantly, the eigenvalues can be found quite rapidly.

We have also tried the Lanczos code of Freund, Gutknecht, and Nachtigal [7] on problem (16) with a matrix of size 10622. With 500 Lanczos iterations, it computed the same five eigenpairs in about 280 seconds for each problem. This code is superior to our code in terms of both efficiency and robustness. However, it suffers the same problem as ours in that it cannot guarantee which eigenvalues it computed.

5. Conclusion. We have presented a homotopy method for computing the eigenpairs of a real matrix. Starting with a matrix with known eigenpairs, Euler–Newton continuation is used to advance the eigenpaths. A real eigenpath will remain real unless it encounters a real fold point. On the opposite side of this fold point, two complex conjugate eigenpairs emerge. The reverse situation in which two complex conjugate eigenpairs meeting at a real fold point with two real paths bifurcating to the right also occurs. By restricting the solutions in the real space, we have shown how to deal with these transitions without numerical difficulties.

The storage requirement is on the order of the number of nonzero elements of the matrix, and thus it is attractive for computing a few eigenpairs of a large, sparse matrix. This together with the fully parallel nature of the algorithm may make it a competitive method for the large, sparse nonsymmetric eigenvalue problem. However, several formidable obstacles must first be overcome. The path-jumping problem has already been mentioned. Another is the absence of a robust general-purpose iterative linear solver. GMRES had considerable convergence difficulties for general matrices. Even for the PDE examples that we tried, it encountered convergence problems when

computing interior eigenvalues. The homotopy method also has difficulty whenever eigenvalues are clustered together. This occurs even if the eigenvectors are orthonormal. The difficulty lies in the fact that eigenvectors cannot be computed accurately by a straightforward application of the inverse iteration (or Newton's method) if the corresponding eigenvalues are clustered together. One solution is to compute the clustered eigenvalues by subspace iteration. However, if the initial matrix is not well chosen, then it is possible that eigenvalues which are far apart initially at $t = 0$ drift together at some point $t \leq 1$. Choosing a good initial matrix for the homotopy which would minimize the number of bifurcation points and keep the eigenpaths well separated is another open problem. Finally, we would like to determine selected eigenvalues (for example, those with the largest real part) by following just one or two eigenpaths. The homotopy method seems to be a very efficient method for locating singular points of bifurcation problems.

The history of the homotopy method as a computational tool for the eigenvalue problem is rather short. We hope this work will stimulate further interest in this area.

Acknowledgment. We thank the referees for suggesting numerous improvements to the original draft.

REFERENCES

[1] E. L. Allgower and K. Georg, *Numerical Continuation Methods*, Springer-Verlag, New York, Berlin, 1990.

[2] Z. Bai, D. Day, and Q. Ye, *ABLE: An Adaptive Block Lanczos Method for Non-Hermitian Eigenvalue Problems*, Technical report, University of Kentucky, Lexington, KY, 1995.

[3] Z. Bai and J. Demmel, *On a block implementation of Hessenberg multishift QR iteration*, Internat. J. High Speed Comput., 1 (1989), pp. 97–112.

[4] Z. Bai and J. Demmel, *Design of a Parallel Nonsymmetric Eigenroutine Toolbox, Part* I, Technical report UCB/CSD-92-718, U. C. Berkeley, Berkeley, CA, 1992.

[5] J. Cullum and R. A. Willoughby, *A practical procedure for computing eigenvalues of large sparse nonsymmetric matrices*, in Large-Scale Eigenvalue Problems, Math. Stud. 127, J. Cullum and R. A. Willoughby, eds., North–Holland, Amsterdam, 1986.

[6] J. J. Dongarra and M. Sidani, *A parallel algorithm for the nonsymmetric eigenvalue problem*, SIAM J. Sci. Comput., 14 (1993), pp. 542–569.

[7] R. W. Freund, M. H. Gutknecht, and N. M. Nachtigal, *An implementation of the look-ahead Lanczos algorithm for non–Hermitian matrices*, SIAM J. Sci. Comput., 14 (1993), pp. 137–158.

[8] R. W. Freund and N. M. Nachtigal, *QMR: A quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math., 60 (1991), pp. 315–339.

[9] G. Golub and C. Van Loan, *Matrix Computations*, 2nd ed., The John Hopkins University Press, Baltimore, MD, 1989.

[10] M. E. Henderson, *Complex Bifurcation*, Ph.D. thesis, California Institute of Technology, Pasadena, CA, 1985.

[11] M. E. Henderson and H. B. Keller, *Complex bifurcation from real paths*, SIAM J. Appl. Math., 50 (1990), pp. 460–482.

[12] T. Kato, *A Short Introduction to Perturbation Theory for Linear Operators*, Springer-Verlag, New York, Berlin, 1982.

[13] H. B. Keller, *Numerical solutions of bifurcation and nonlinear eigenvalue problems*, in Applications of Bifurcation Theory, P. H. Rabinowitz, ed., Academic Press, New York, 1977.

[14] H. B. Keller, *Lectures on Numerical Methods in Bifurcation Problems*, Springer-Verlag, Berlin, 1987.

[15] C. T. Lenard, *A Homotopy Method for Eigenproblems*, Technical report, Australia National University, Canberra, 1990.

[16] K. Li and T. Y. Li, *An algorithm for symmetric tridiagonal eigenproblems: Divide and conquer with homotopy continuation*, SIAM J. Sci. Comput., 14 (1993), pp. 735–751.

[17] T. Y. Li and N. H. Rhee, *Homotopy algorithm for symmetric eigenvalue problems*, Numer. Math., 55 (1989), pp. 265–280.

[18] T. Y. Li and X. Wang, *Higher order turning points*, Appl. Math. & Comp., 64 (1994), pp. 155–166.
[19] T. Y. Li and Z. Zeng, *Homotopy-determinant algorithm for solving nonsymmetric eigenvalue problems*, Math. of Computation, 59 (1992), pp. 483–502.
[20] T. Y. Li, Z. Zeng, and L. Cong, *Solving eigenvalue problems of real nonsymmetric matrices with real homotopies*, SIAM J. Numer. Anal., 29 (1992), pp. 229–248.
[21] T. Y. Li, H. Zhang, and X. H. Sun, *Parallel homotopy algorithm for the symmetric tridiagonal eigenvalue problem*, SIAM J. Sci. Comput., 12 (1991), pp. 469–487.
[22] S. H. Lui, *Part* II: *Parallel Homotopy Method for the Real Nonsymmetric Eigenvalue Problem*, Ph.D. thesis, California Institute of Technology, Pasadena, CA, 1992.
[23] J. D. Murray, *Mathematical Biology*, Springer-Verlag, New York, Berlin, 1989.
[24] A. Ruhe, *The rational Krylov algorithm for nonsymmetric eigenvalue problems.* III: *Complex shifts for real matrices*, BIT, 34 (1994), pp. 165–176.
[25] Y. Saad, *Numerical solution of large nonsymmetric eigenvalue problems*, Computer Physics Comm., 53 (1989), pp. 71–90.
[26] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, Manchester University Press, 1992.
[27] Y. Saad and M. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856–869.
[28] G. Shroff, *A parallel algorithm for the eigenvalue and eigenvectors of a general complex matrix*, Numer. Math., 58 (1991), pp. 779–805.
[29] D. C. Sorensen, *Implicit application of polynomial filters in a k-step Arnoldi method*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 357–385.
[30] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.
[31] Z. H. Yang and H. B. Keller, *A direct method for computing higher order folds*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 351–361.