

# INVERSE ITERATION FOR PURELY IMAGINARY EIGENVALUES WITH APPLICATION TO THE DETECTION OF HOPF BIFURCATIONS IN LARGE SCALE PROBLEMS

KARL MEERBERGEN AND ALASTAIR SPENCE

**Abstract.** The detection of a Hopf bifurcation in a large scale dynamical system that depends on a physical parameter often consists of computing the right-most eigenvalues of a sequence of large sparse eigenvalue problems. Guckenheimer et. al. (SINUM, 34, (1997) pp. 1-21) proposed a method that computes a value of the parameter that corresponds to a Hopf point without actually computing right-most eigenvalues. This method utilises a certain sum of Kronecker products and involves the solution of matrices of squared dimension, which is impractical for large scale applications. However, if good starting guesses are available for the parameter and the purely imaginary eigenvalue at the Hopf point, then efficient algorithms are available. In this paper, we propose a method for obtaining such good starting guesses, based on finding purely imaginary eigenvalues of a two-parameter eigenvalue problem (possibly arising after a linearisation process). The problem is formulated as an inexact inverse iteration method that requires the solution of a sequence of Lyapunov equations with low rank right hand sides. It is this last fact that makes the method feasible for large systems. The power of our method is tested on four numerical examples.

**1. Introduction.** This paper introduces a numerical procedure for the determination of the smallest  $|\lambda|$  for which the large, sparse eigenvalue problem

$$(1.1) \quad (A + \lambda B)x = \mu Mx,$$

has a pair of purely imaginary  $\mu$ 's. Here  $\lambda$  is a physical parameter and  $\mu$  is the eigenvalue of the generalised eigenvalue problem (1.1).

Our work is motivated by the bifurcation analysis of the non-linear dynamical system

$$\frac{du}{dt} = f(u, \lambda) \quad , \quad u(0) = u_0,$$

where  $f$  is an operator in  $(\mathbf{R}^n, \mathbf{R}) \mapsto \mathbf{R}^n$  with  $n$  large. Such analysis includes the computation of bifurcation diagrams, and more particularly, the stability analysis and detection of Hopf bifurcations, which lead to the birth of periodic solutions (see, for example, [46]). In many situations, for example, in nonlinear finite element discretizations, we have a dynamical system of the form

$$M \frac{du}{dt} = f(u, \lambda),$$

where  $M$  is a large sparse symmetric positive definite mass matrix. In the case of steady state solutions, i.e.  $du/dt = 0$ , often the values of  $\lambda$  are sought for which the solution  $u$  loses stability. In a linearized stability analysis, the steady state is said to be stable when the eigenvalues  $\mu$  of

$$(1.2) \quad J(\lambda)x = \mu Mx,$$

have strictly negative real parts, with  $J(\lambda)$  denoting the Jacobian matrix evaluated at the steady state  $u(\lambda)$ , namely,  $J(\lambda) = \frac{\partial f}{\partial u}(u(\lambda), \lambda)$ . Values of  $\lambda$  where eigenvalues of (1.2) cross the imaginary axis indicate a transition from a stable to unstable regime. When stability is lost due to a real eigenvalue  $\mu$  passing through zero there are many techniques available to determine the critical value of  $\lambda$ , see, for example [15]. In

contrast, at a Hopf bifurcation on a path of stable steady states, (1.2) has two purely imaginary eigenvalues and their detection is a particularly difficult task for large scale dynamical systems. On the other hand, if close starting values for  $\lambda$  and  $\mu$  are known then there are good methods, usually based on Newton's Method, for their accurate determination, see, for example, [15]. The contribution of this paper is the determination of good starting values with which to initiate the Hopf calculation.

Perhaps the most straightforward method to detect Hopf points is to monitor the right-most eigenvalues of (1.2) for a discrete set of  $\lambda$ 's. This requires the solution of an eigenvalue problem for each selected  $\lambda$ , which can be very expensive, especially when the system size is large. The solution of large scale eigenvalue problems in this context has been studied intensively the last fifteen years. We refer to [3] for an overview of eigenvalue solvers, and [34] for an overview on methods for computing right-most eigenvalues. The next most obvious method is the shift-and-invert Arnoldi method [40] [45] with zero shift. This is an attractive approach since a matrix factorization of  $J(\lambda)$  may well be available from the steady state computation. If a matrix factorization is not feasible, the JDQZ method [10], inexact rational Krylov [31] or Arnoldi's method with inexact shift-and-invert can be used. All these methods are quite reliable for computing eigenvalues near a known point or target, but sometimes fail to compute the right-most eigenvalue. This is likely to be the case when there are many eigenvalues lying near zero but the eigenvalues that cross the imaginary axis have large unknown imaginary part. This is precisely the case in the stability of double-diffusive convection, see Chapter 8 of [50] for the theory and [6] for a numerical discussion. Reference [9] warns for wrong stability assignments being made using standard eigenvalue tracking methods. If necessary, an expensive validation phase can be employed to ensure that the right-most eigenvalue is indeed computed [35] [33].

For small scale problems Guckenheimer and co-workers, [19], [20], introduced a novel technique to detect Hopf bifurcations based on the use of the bialternate product of  $J(\lambda)$ , defined as  $(J(\lambda) \otimes I + I \otimes J(\lambda))/2$ , which is an  $n^2 \times n^2$  matrix that has a pair of zero eigenvalues when  $J(\lambda)$  has a pair of purely imaginary eigenvalues. This approach was also used in [30], and [16], and expounded further in [15, §§4.4–4.5]. This construction forms the first theoretical step in our method, but we emphasize that we do not compute with the Kronecker product forms. Related ideas are used by [18] to estimate the distance to uncontrollability. There the task of determining when two  $2n \times 2n$  matrix pencils have a common purely imaginary eigenvalue is converted to that of finding a real eigenvalue of a  $2n^2 \times 2n^2$  matrix. This is accomplished using inverse iteration, where each matrix solve is carried out by solving a Sylvester equation. In [18] the applications are low dimensional and the Sylvester equations are solved using a variant of the Bartels-Stewart algorithm. Similar ideas have been discussed for the computation of the critical delay of delay differential equations, however, without exploiting the low rank structure of the eigenvectors [29] [36], which makes this work infeasible for large scale problems.

In this paper we build on the idea in [19], [20], and develop a method applicable to large scale problems. As in [18] we use an inverse iteration approach to solve the resulting Lyapunov equations, but we extend this idea in several aspects. In particular we present a detailed analysis of

1. an equivalence theory of three eigenvalue problems, and
2. the identification of structure of the eigenvectors.

Also, we discuss in detail the following computational issues:

1. the use of large sparse Lyapunov solvers in an inexact inverse iteration setting,

and

2. the exploitation of low rank and symmetry properties arising from the structure of the eigenvectors.

We consider the situation where  $J(\lambda)$  has the form

$$J(\lambda) = (A + \lambda B) .$$

In the bifurcation setting,  $A$  and  $B \in \mathbf{R}^{n \times n}$  arise from a linearization of  $f(u, \lambda)$  as follows: Assume that  $u(0)$  is a known steady state at  $\lambda = 0$ , then  $J(0)$  is known and  $J(\lambda) \approx J(0) + \lambda J'(0)$ , which we write as  $A + \lambda B$ . Here,  $A$  and  $B$  are usually large, sparse, nonsymmetric matrices and  $B$  can be singular. In this paper, we assume the following situation:  $\lambda = 0$  corresponds to a stable steady state solution, i.e. all eigenvalues of  $Ax = \mu Mx$  lie in the stable half plane. The goal is to compute the smallest  $|\lambda|$  for which the eigenvalue problem (1.1) has purely imaginary  $\mu$ 's. Generically, the  $\mu$ 's will be continuous functions of  $\lambda$  and the first  $\lambda$  for which there are  $\mu$ 's on the imaginary axis (including the case when  $\mu = 0$ ), will approximate the value of  $\lambda$  that corresponds to a transition from a stable to unstable steady state or to a Hopf point. A validation phase as in [33] [35] would no longer be required in this case. In this paper, we assume that  $A$  can be factorized by a sparse direct method or an accurate and fast iterative solver is available. The assumption of efficient factorization by a sparse direct method holds for a large class of large scale applications.

First, note that the bialternate product of  $(A + \lambda B)$  has the form (dropping the factor of  $\frac{1}{2}$ )

$$(1.3) \quad (A + \lambda B) \otimes M + M \otimes (A + \lambda B).$$

If (1.1) has purely imaginary eigenvalues then (1.3) has a double eigenvalue zero. Mathematically, this translates into the following  $n^2 \times n^2$  linear eigenvalue problem

$$(1.4) \quad (A \otimes M + M \otimes A)z + \lambda(B \otimes M + M \otimes B)z = 0 ,$$

whose solution gives the values of  $\lambda$  for which (1.1) has purely imaginary eigenvalues  $\mu$ . Although this is a nice mathematical property, it should only be used in this form for problems of small size, as in [19]. To produce a method that is applicable to large scale problems we exploit the well-known equivalence between equations involving the Kronecker product and Lyapunov equations. This shows that the eigenvalue problem (1.4) is equivalent to the eigenvalue problem

$$(1.5) \quad MZA^T + AZM^T + \lambda(MZB^T + BZM^T) = 0 ,$$

where  $\text{vec}(Z) = z$ , with  $z$  as in (1.4). Here  $Z$  is an unknown  $n \times n$  matrix, which we also call an 'eigenvector' of (1.5). A new element in this paper is our discussion of the rank of  $Z$  and its symmetry properties. A clear understanding of these two aspects turns out to be key in obtaining an efficient algorithm for large scale problems.

For the sake of completeness, we mention the connection with the two-parameter eigenvalue problem as discussed in [2], where the link with Kronecker (tensor) product formulations is made. The problem that we discuss in this paper can also be written in the form

$$(1.6) \quad Ax + \lambda Bx - \mu Mx = 0$$

$$(1.7) \quad A\bar{x} + \lambda B\bar{x} + \mu M\bar{x} = 0$$

where the second equation is the complex conjugate of the first and we have assumed that  $\mu$  is purely imaginary. Following Atkinson, [2], the pair (1.6), (1.7) is said to be nonsingular if the matrix

$$(1.8) \quad \Delta_0 := B \otimes M + M \otimes B$$

is nonsingular. Atkinson also shows that systems like (1.6), (1.7) induce a natural symmetry on  $\mathbf{C}^{n^2}$ , a feature that we shall return to in §2. Such problems can be solved with the Jacobi-Davidson method [24] [25], which seeks  $(\lambda, \mu)$  pairs near a target point  $(\sigma, \tau)$ , often with  $\tau = 0$ , since a good starting guess for  $\mu$  typically won't be known. However, for the detection of a Hopf bifurcation this is a potentially dangerous strategy and a situation that we want to avoid. This paper shows an alternative approach, where a starting guess for  $\mu$  isn't required, but where the computation of  $\mu$  is a byproduct of the method.

The plan of the paper is as follows. We first show properties of the eigenvalue problem (1.4) in §2. We show connections between (1.1) and (1.4) and prove that the  $\lambda$  of interest is a simple eigenvalue of (1.4) restricted to an appropriate subspace. In §3, we present an inverse iteration method for solving (1.4). Inverse iteration is an obvious choice, since we want to compute the  $\lambda$  nearest zero. A starting guess of  $\mu$  is not required, but the final value of  $\mu$  can be computed as a byproduct once  $\lambda$  and an eigenvector are known. Equation (1.4) is written as a Lyapunov-like equation, where the eigenvectors are matrices of low rank. The fact that we work with low rank matrices utilizes the efficiency of traditional solvers for large scale Lyapunov equations with low rank right-hand sides. We chose a Lyapunov solver using Arnoldi's method in our numerical tests. This method can be viewed as an inexact inverse iteration method and we can use convergence results for simple eigenvalues in the case that the solution  $\lambda$  corresponds to a Hopf bifurcation. Numerical examples are given in §4 and support the theory in this paper.

**2. Three eigenvalue problems.** In this section, we discuss the properties of the  $n^2 \times n^2$  generalised eigenvalue problem (1.4) and describe the relationship between its solutions and the solutions of (1.1). This relationship also holds between (1.1) and (1.5) because of the equivalence between (1.5) and (1.4). In the next section we use the results derived here to derive an efficient inverse iteration-type algorithm to compute the required solution of (1.5).

It is convenient to introduce the  $n^2 \times n^2$  matrices:

$$\begin{aligned} \Delta_0 &= B \otimes M + M \otimes B \\ \Delta_1 &= A \otimes M + M \otimes A \end{aligned}$$

and rewrite (1.4) in simpler form as

$$(2.1) \quad (\Delta_1 + \lambda \Delta_0)z = 0 .$$

The following theorem relates the solutions of (2.1) and (1.5) with those of (1.1).

**THEOREM 2.1.** *For a given real  $\lambda$ , let  $(\mu_j, x_j)$  be an eigenpair of (1.1). Then*

1. *if  $\mu_1 = 0$  is a simple eigenvalue and there are no other eigenvalues on the imaginary axis, then  $\lambda$  is a simple eigenvalue of (2.1) with eigenvector  $z = x_1 \otimes x_1$ ;*
2. *if  $\mu_{1,2} = \pm\beta i \in \mathbf{I}$  are two simple purely imaginary eigenvalues and there are no other eigenvalues on the imaginary axis, then  $\lambda$  is a double eigenvalue with eigenvector  $z = \xi_1 x_1 \otimes \bar{x}_1 + \xi_2 \bar{x}_1 \otimes x_1$  for any  $\xi_1, \xi_2 \in \mathbf{C}$ ;*

3. if  $\mu_{1,2} = \pm\alpha \in \mathbf{R}$  are two simple real eigenvalues and there are no other eigenvalues on the imaginary axis, then  $\lambda$  is a double eigenvalue with eigenvector  $z = \xi_1 x_1 \otimes x_2 + \xi_2 x_2 \otimes x_1$  for any  $\xi_1, \xi_2 \in \mathbf{C}$ .

*Proof.* If  $\mu_1 = 0$ , then substituting  $z = x_1 \otimes x_1$  into (2.1) gives

$$(\Delta_1 + \lambda\Delta_0)x_1 \otimes x_1 = (A + \lambda B)x_1 \otimes Mx_1 + Mx_1 \otimes (A + \lambda B)x_1 = 0 ,$$

which proves item 1. Items 2 and 3 follow from that  $(A + \lambda B)x_1 = \mu Mx_1$  and  $(A + \lambda B)x_2 = -\mu Mx_2$ , where  $\mu = \mu_1 = -\mu_2$ . We can write

$$\begin{aligned} (\Delta_1 + \lambda\Delta_0)x_1 \otimes x_2 &= (A + \lambda B)x_1 \otimes Mx_2 + Mx_1 \otimes (A + \lambda B)x_2 \\ &= (A + \lambda B + \mu M)x_1 \otimes Mx_2 + Mx_1 \otimes (A + \lambda B - \mu M)x_2 \\ &= 0 \otimes Mx_2 + Mx_1 \otimes 0 = 0 . \end{aligned}$$

□

We now show that the converse of this theorem holds if  $M$  is nonsingular.

**THEOREM 2.2.** *Let  $(\lambda, z)$  with  $\lambda$  real be an eigenpair of (2.1) and  $M$  be nonsingular. Then there are corresponding solutions of (1.1) as follows:*

1. *If  $\lambda$  is simple, then  $z = \xi x_1 \otimes x_1$  where  $(\mu_1 = 0, x_1)$  is an eigenpair of (1.1), and  $\xi \in \mathbf{C}$ .*
2. *If  $\lambda$  is a double eigenvalue, then there are  $\mu \in \mathbf{C}$  and  $x_1$  and  $x_2 \in \mathbf{C}^n$  so that  $(\mu, x_1)$  and  $(-\mu, x_2)$  are simple eigenpairs of (1.1) and there are  $\xi_1$  and  $\xi_2 \in \mathbf{C}$  so that  $z = \xi_1 x_1 \otimes x_2 + \xi_2 x_2 \otimes x_1$ .*

*Proof.* Let  $A + \lambda B = MX\Gamma X^{-1}$  be a Jordan canonical form associated with (1.1), where  $X = [x_1, \dots, x_n]$  and  $\mu_1, \dots, \mu_n$  are the main diagonal elements of  $\Gamma$ , then (2.1) is equivalent to

$$(MX \otimes MX)(\Gamma \otimes I + I \otimes \Gamma)(X^{-1} \otimes X^{-1})z = 0 .$$

Since  $MX \otimes MX$  and  $X^{-1} \otimes X^{-1}$  have full rank,

$$\Gamma \otimes I + I \otimes \Gamma$$

has at least one zero main diagonal element. The main diagonal elements of  $\Gamma \otimes I + I \otimes \Gamma$  are  $\mu_j + \mu_i$  where  $\mu_j$  are eigenvalues of (1.1). Since  $\Gamma \otimes I + I \otimes \Gamma$  is upper triangular,  $\mu_j + \mu_i$  are eigenvalues of  $(A + \lambda B) \otimes M + M \otimes (A + \lambda B)$  with associated eigenvectors  $z = x_i \otimes x_j$  and  $z = x_j \otimes x_i$ .

If  $\lambda$  is a double eigenvalue, the two eigenvalues correspond to  $\mu_1 + \mu_2 = 0$  and  $\mu_2 + \mu_1 = 0$ . Since  $A, B$  and  $M$  are real, we must have that  $\mu_{1,2} = \pm\alpha$  or  $\mu_{1,2} = \pm\beta i$  (possibly zero). The eigenvectors of the double eigenvalue zero must have the form  $\xi_1 x_1 \otimes x_2 + \xi_2 x_2 \otimes x_1$ . If  $\mu_{1,2} = \pm\beta i$ , then  $x_2 = \bar{x}_1$ .

If  $\lambda$  is a simple eigenvalue,  $\mu_i + \mu_j$  can only be simple and zero when  $i = j = 1$ , and  $\mu_1 = 0$ . The associated eigenvector is  $z = \xi x \otimes x$  where  $x$  is the corresponding eigenvector of (1.1).

This proves the theorem. □

Theorems 2.1 and 2.2 show the correspondence between purely imaginary eigenpairs of (1.1) and solutions of (2.1) (equivalently (1.5)). The next theorem provides the key to our understanding of the structure of  $Z$ , the eigenvector of (1.5).

**THEOREM 2.3.** *Assume that  $\lambda$  is a real eigenvalue of (2.1), and the conditions of Theorem 2.1 are satisfied. Then the solutions of (1.5) satisfy the following results:*

1. if  $\mu = 0$  is simple,  $Z$  is a symmetric matrix of rank 1, i.e.  $Z = xx^T$ , which is unique up to a scalar factor;
2. if  $\mu = \pm\beta i$ , there is a real symmetric eigenvector of rank two, namely,  $Z = xx^* + \bar{x}x^T$ , which is unique up to a scalar factor and is semi-definite, and a unique skew-symmetric eigenvector of rank two, namely,  $Z = xx^* - \bar{x}x^T$ .
3. if  $\mu = \pm\alpha$  is real, there is a symmetric eigenvector of rank two  $Z = x_1x_2^T + x_2x_1^T$ , which is unique up to a scalar factor and is indefinite, and a unique skew-symmetric eigenvector of rank two, namely,  $Z = x_1x_2^T - x_2x_1^T$ .

*Proof.* The proof for  $\mu = 0$  follows from  $z = x \otimes x$  and so  $Z = xx^T$ . For  $\mu$  real or purely imaginary, we know that the eigenvectors take the form

$$Z = \xi_1 x_1 x_2^T + \xi_2 x_2 x_1^T ,$$

where  $x_1$  and  $x_2$  are not parallel. If  $\xi_1$  or  $\xi_2$  is zero,  $Z$  has rank one. We now prove that for cases 2 and 3, a symmetric  $Z$  has rank two. From  $Z - Z^T = 0$  we derive  $(\xi_1 - \xi_2)x_1x_2^T + (\xi_2 - \xi_1)x_2x_1^T = 0$  so that  $\xi_1 = \xi_2$ . This implies that for a symmetric  $Z$ , we always have a rank two matrix. Note that for case 2,  $x_1 = x$  and  $x_2 = \bar{x}$ .

We can rewrite

$$(2.2) \quad Z = x_1x_2^T + x_2x_1^T$$

$$(2.3) \quad = (x_1 + x_2)(x_1 + x_2)^T - (x_1 - x_2)(x_1 - x_2)^T ,$$

If  $x_1$  and  $x_2$  are real,  $Z$  has one positive semi-definite term and one negative semi-definite term, so  $Z$  has one positive and one negative eigenvalue (and  $n - 2$  zero eigenvalues). If  $x_2 = \bar{x}_1$ , then  $Z = xx^* + \bar{x}x^T$  is the sum of two complex Hermitian but positive semi-definite terms, which concludes the proof of the theorem.  $\square$

The observation of Theorem 2.3 is very significant for our work. From the theorem we may conclude that for the three cases considered in Theorems 2.1, 2.2 and 2.3,  $\lambda$  is a simple eigenvalue when (2.1) is restricted to the subspace of symmetric eigenvectors  $Z$ . This observation is important, since theory for eigenvalue solvers is significantly easier for simple than for double eigenvalues. The above straightforward observations about symmetry provide an example of the more general symmetry induced by decomposable tensors discussed in §5.9 of [2].

The solution of (2.1) restricted to the symmetric eigenvector space is related to, but distinct from, the method discussed in [20] and [15, §§4.4–4.5], where it is suggested to solve (2.1) restricted to the anti-symmetric eigenvector space. For this approach the advantage is that a simple zero  $\mu$  cannot produce an eigenvalue  $\lambda$  in this space, since its corresponding eigenvector,  $Z$ , is always symmetric (Theorem 2.3, case 1). Hence, solving (2.1) restricted to the anti-symmetric eigenvector space avoids the computation of a simple zero  $\mu$ . In contrast, we have chosen the restriction to symmetric eigenvectors, since the inverse iteration method is then related to the symmetric solution of a Lyapunov equation, which is a rather well-known problem.

A natural representation of a low rank symmetric matrix is its truncated eigen-decomposition. In practice, we will write  $Z$  as  $Z = VDV^T$  where  $D$  is a diagonal matrix and  $V^TV = I$ . For a rank two  $Z$ ,  $D$  is a  $2 \times 2$  matrix and  $V$  is  $n \times 2$ .

Once  $Z$  and  $\lambda$  are computed, the computation of  $\mu$  readily follows from the solution of the 2-dimensional generalised eigenvalue problem

$$(2.4) \quad V^T(A + \lambda B)Vy = \mu V^T M V y .$$

Indeed, since

$$Z = xx^* + \bar{x}x^T = VDV^T ,$$

$x$  and  $\bar{x}$  lie in the column space of  $V$ .

When  $\Delta_0$ , defined by (1.8), is singular,  $\lambda = \infty$  is an eigenvalue with the nullspace of  $\Delta_0$  as eigenspace. Note that in case of a singular  $B$ ,  $\mu$  can take any finite value where the eigenvectors lie in the common nullspace. Here is a simple example.

EXAMPLE 1. *Let*

$$A = \begin{bmatrix} 2 & -1 & \\ 1 & 2 & \\ & & 3 \end{bmatrix} \quad B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad M = I .$$

The eigenvalues of the pencil  $\Delta_0 + \lambda\Delta_1$  are  $-2$  (double eigenvalue),  $-2 \pm i$  (simple eigenvalues),  $-5 \pm i$  (double eigenvalues), and  $\infty$ . For  $\lambda = -2$ , we have  $\mu = \pm i$  (which indicates a Hopf point), For the simple eigenvalues  $\lambda = -2 \pm i$ , we have  $\mu = 0$ . Complex  $\lambda$  have no physical meaning and are discarded. With the infinite  $\lambda$ , we can associate any finite  $\mu$  with eigenvector  $e_3 \otimes e_3$ .

Eigenvectors associated with  $\lambda = -2$  are  $(1, -i, 0) \otimes (1, i, 0)$  and  $(1, i, 0) \otimes (1, -i, 0)$ .

**3. Inverse iteration algorithm.** In this section, we describe the inverse iteration method [14, §7.6.1] [39, §4.2.2] to find the solution of (1.4). We rewrite the method for solving (1.5), since we intend to work with small rank matrices. We introduce an additional projection step to force the eigenvector iterates to have rank  $r = 1$  or 2. At the end of the section, we discuss the block Arnoldi method that we use for the numerical examples and make a note on the accuracy required by the Lyapunov solver.

**3.1. Inverse iteration.** A simple inverse iteration algorithm for (2.1) is as follows:

ALGORITHM 3.1 (Inverse iteration for (2.1)).

1. Given  $z_1 \in \mathbf{R}^{n^2}$  with  $\|z_1\|_2 = 1$ .
2. For  $j = 1, 2, \dots$ 
  - 2.1. Compute the eigenvalue approximation

$$\lambda_j = -z_j^T \Delta_1 z_j / z_j^T \Delta_0 z_j .$$

- 2.2. If  $\lambda_j$  is accurate enough, then stop.
- 2.3. Solve  $\Delta_1 y_j = \Delta_0 z_j$  for  $y_j$ .
- 2.4. Normalize:  $z_{j+1} = y_j / \|y_j\|_2$ .

The stopping criterion in Step 2.2 usually consists of checking if the residual norm  $\|\Delta_1 z_j + \lambda_j \Delta_0 z_j\|_2$  is below a prescribed tolerance.

This algorithm is impractical since it works with vectors of dimension  $n^2$ . We therefore rewrite the algorithm for the Lyapunov form of the eigenvalue problem given by (1.5):

ALGORITHM 3.2 (Inverse iteration for (1.5)).

1. Given  $Z_1 = V_1 D_1 V_1^T$  with  $V_1 \in \mathbf{R}^{n \times 1}$ ,  $\|V_1\|_2 = 1$ , and  $\|D_1\|_F = 1$ .
2. For  $j = 1, 2, \dots$ 
  - 2.1. Compute the eigenvalue approximation

$$(3.1) \quad \lambda_j = -\frac{\text{trace}(\tilde{A}_j D_j \tilde{M}_j^T + \tilde{M}_j D_j \tilde{A}_j^T)}{\text{trace}(\tilde{B}_j D_j \tilde{M}_j^T + \tilde{M}_j D_j \tilde{B}_j^T)}$$



with

$$(3.2) \quad \tilde{A}_j = V_j^T A V_j \quad , \quad \tilde{B}_j = V_j^T B V_j \quad , \quad \tilde{M}_j = V_j^T M V_j .$$

- 2.2. If  $\lambda_j$  is accurate enough, then stop.
- 2.3. Solve

$$(3.3) \quad \begin{aligned} A Y_j M^T + M Y_j A^T &= F_j \\ F_j &= B Z_j M^T + M Z_j B^T \end{aligned}$$

in factored form:  $Y_j = V_{j+1} \tilde{D}_j V_{j+1}^T$  with  $\|V_{j+1}\|_2 = 1$ .

- 2.4. Normalize:  $D_{j+1} = \tilde{D}_j / \|\tilde{D}_j\|_F$ . (Note that  $Z_{j+1} = V_{j+1} D_{j+1} V_{j+1}^T$ .)

The Rayleigh quotient (3.1) can be derived by multiplying (1.5) on the left by  $V_j^T$  and the right by  $V_j$  and plugging in  $Z = V_j D_j V_j^T$ .

If  $F_j$  in (3.3) is symmetric,  $Y_j$  is also symmetric. Indeed,  $Y_j^T$  is a solution of (3.3) and since the eigenvalues of  $M^{-1}A$  all lie in the left half plane, the solution is unique. So,  $Y_j = Y_j^T$ . If  $Z_j$  is symmetric, then  $F_j$  is symmetric. By an induction argument on  $j$ , starting from symmetric  $Z_1$ , we can see that  $Z_j$  is symmetric for  $j \geq 1$ . Consequently, all iterates  $\{Z_j\}_{j \geq 1}$  are symmetric, and if the sequence converges, its limit is a symmetric eigenvector, as desired.

Direct methods for Lyapunov equations compute a full (rank)  $Y_j$  from (3.3) [4][13][23]. Such methods are too expensive for large scale problems, since they store the full matrix and have too high complexity. Methods for large scale problems approximate the solution by a rank  $r \ll n$  matrix  $Y_j$  in factored form,  $Y_j = V_{j+1} \tilde{D}_j V_{j+1}^T$ . Such methods can only be applied when a good low rank approximation exists. Fortunately, in many cases, the eigenvalues of the solution of Lyapunov equations with low rank right-hand sides arising from dynamical systems decay quickly [42][1]. Krylov type methods [44][27][28][47][43][48] typically build a large subspace that is then reduced to a small rank Lyapunov solution. ADI type methods [41] and the Smith method [49] are other classes of methods. We refer to [22] for an overview. Also see [21][26][37].

For efficient computations, we want  $F_j$  to be low rank. Once  $Z_j$  starts to converge to an eigenvector, we know that  $Z_j$  has  $r = 1$  or  $2$  eigenvalues with large absolute values and  $n - r$  with small absolute values, but that is not the case for the first few iterations. If we start with a rank one  $Z_1$ ,  $F_1$  has rank two, and, depending on the problem,  $Y_2$  (and hence  $Z_2$ ) can have large rank, which would lead to a large rank  $F_2$ . Although we mentioned that the eigenvalues of  $Y_2$  decay quickly, the rank required for a good approximation of  $Y_2$  may be rather large. Such large ranks may prohibit the efficient solution of (3.3). We therefore introduce the projection step described in the following section.

**3.2. Rank reduction by projection.** Ideally, we would like to work with  $r = 1$  or  $2$  rank iterates  $\{Z_j\}_{j \geq 1}$ . The eigenvalue iterate,  $\lambda_j$  is the Rayleigh quotient given by (3.1). A ‘better’ approximation can be found by projecting (1.4) on a subspace. If  $V_j \in \mathbf{R}^{n \times k}$ , then the projection of (1.5) (and similarly of (1.4) onto  $\text{Range}(V_j \otimes V_j)$ ) leads to the following order  $k$  problem:

$$(3.4) \quad \tilde{A}_j \tilde{Z}_j \tilde{M}_j^T + \tilde{M}_j \tilde{Z}_j \tilde{A}_j^T = \tilde{\lambda} (\tilde{B}_j \tilde{Z}_j \tilde{M}_j^T + \tilde{M}_j \tilde{Z}_j \tilde{B}_j^T)$$

where  $\tilde{A}_j$ ,  $\tilde{B}_j$ , and  $\tilde{M}_j$ , given by (3.2), are  $k \times k$  matrices. The eigenvector approximation for (1.5) associated with  $\tilde{\lambda}$  is  $V_j \tilde{Z}_j V_j^T$ . There are two advantages of this



projection step:

1. we expect to have a ‘better’ approximate eigenpair, as is usually the case with projection methods (e.g. Arnoldi’s method compared to the power method, Jacobi-Davidson compared to the Newton method),
2. and the approximate eigenvector  $Z_{j+1}$  has rank  $r = 1$  or  $2$ .

With this idea, we obtain the following algorithm:

ALGORITHM 3.3 (Inverse iteration for (1.5) with projection).

1. Given  $V_1$  with  $\|V_1\|_2 = 1$ .
2. For  $j = 1, 2, \dots$ 
  - 2.1. Compute (3.2) and solve (3.4) in factored form:  $\tilde{Z}_j = \tilde{V}_j D_j \tilde{V}_j^T$ .
  - 2.2. Let  $V_j = V_j \tilde{V}_j \in \mathbf{R}^{n \times r}$  with  $r = 1$  or  $2$ .
  - 2.3. If  $\lambda_j$  is accurate enough, then stop.
  - 2.4. Solve (3.3) in factored form  $Y_j = V_{j+1} \tilde{D}_j V_{j+1}^T$  with  $V_{j+1} \in \mathbf{R}^{n \times k}$ .

The ‘eigenvector’  $V_j D_j V_j^T$  is a rank one or rank two matrix. When  $\lambda_j$  is computed from (3.4), the associated  $\mu$ ’s can be computed using (2.4) with  $V = V_j$ . The residual norm

$$\|(A + \lambda_j B)V_j y - \mu M V_j y\|_2$$

is relatively cheap to compute (much cheaper than solving the Lyapunov equation). It is used for the stopping criterion in Step 2.3.

Note that in Step 1, we do not assume  $D_1$  is given and that the normalization step after the Lyapunov solve has been removed. The matrix  $\tilde{D}_j$  is not required for the computation of the eigenvector.

**3.3. A block Krylov Lyapunov solver.** For our numerical experiments, we used a block Krylov method. We explain the details here. It should be noted that any Lyapunov solver can be used in inverse iteration.

We first write (3.3) in the form

$$(3.5) \quad S Y_j + Y_j S^T = A^{-1} F_j A^{-T}$$

with  $S = A^{-1} M$ . We have chosen to invert  $A$  since this is closer to the inverse iteration method for the standard eigenvalue problem, and since the eigenvalues of  $M^{-1} A$  of large modulus are less important than those near zero. The eigenvalues with large modulus are usually related to discretization errors of the PDE and should not play a major role. Note that the right-hand side is stored in factored form as  $A^{-1} F_j A^{-T} = P_j C_j P_j^T$ . If  $V_j$  has rank  $r$ ,  $P_j$  has rank between  $r$  and  $2r$ .

Lyapunov equations with rank one right-hand sides can be solved by Arnoldi’s method. The Arnoldi method builds an orthogonal basis of the Krylov space for the matrix  $S$ :

$$\text{span}\{w_1, S w_1, S^2 w_1, \dots, S^{k-1} w_1\}$$

and a  $k \times k$  upper Hessenberg matrix  $H_k = W_k^T S W_k$ . Here is an outline of the Arnoldi algorithm:

ALGORITHM 3.4 (Arnoldi).

1. Given  $w_1$  with  $\|w_1\|_2 = 1$
2. For  $j = 1, \dots, k$ 
  - 2.1. Compute  $\tilde{w}_j = S w_j$ .

- 2.2. Compute  $h_j = W_j^T \tilde{w}_j \in \mathbf{R}^j$  and  $\hat{w}_j = \tilde{w}_j - W_j h_j$
- 2.3. Compute  $\beta_j = \|\hat{w}_j\|_2$  and define  $w_{j+1} = \hat{w}_j / \beta_j$
- 2.4. Let  $W_{j+1} = [W_j \ w_{j+1}]$

When we collect the matrices  $h_j$  and coefficients  $\beta_j$  in the upper Hessenberg matrix  $H_k \in \mathbf{R}^{k \times k}$ , i.e.

$$H_k = \begin{bmatrix} h_1 & \cdots & h_{k-1} & h_k \\ \beta_1 & \ddots & \vdots & \vdots \\ 0 & \ddots & \vdots & \vdots \\ & 0 & \beta_{k-1} & \vdots \end{bmatrix},$$

then we have

$$(3.6) \quad SW_k - W_k H_k = w_{k+1} \beta_k e_k^T$$

where  $e_k$  is the last column of the  $k \times k$  identity matrix.

Suppose we want to solve the Lyapunov equation

$$(3.7) \quad SY + YS^T = w_1 w_1^T.$$

Let  $W_k$  and  $H_k$  be computed by the Arnoldi method with starting vector  $w_1$ . We look for a solution of the form  $Y = W_k X W_k^T$ . First substituting this into (3.7), and then using (3.6), we have

$$\begin{aligned} (SW_k) X W_k^T + W_k X (SW_k)^T &= w_1 w_1^T \\ W_k (H_k X + X H_k^T - e_1 e_1^T) W_k^T &= -w_{k+1} \beta_k e_k^T X W_k^T - W_k X e_k \beta_k w_{k+1}^T. \end{aligned}$$

Multiplication on the left and the right by  $W_k^T$  and  $W_k$  respectively, produces the Lyapunov equation of order  $k$

$$H_k X + X H_k^T = e_1 e_1^T.$$

If  $k$  is much smaller than  $n$ , this is a cheap problem to solve.

For rank  $p$  right-hand sides, we use the block Arnoldi method with  $p$  starting vectors. In the block Arnoldi method, we build the Krylov space

$$\text{span}\{P_j, SP_j, \dots, S^{k/p-1} P_j\}$$

which is a subspace of dimension smaller than or equal to  $k$ . The extension to the block Arnoldi method is easy, although rather technical, and therefore omitted, see [43] for technical details.

**3.4. A note on the accuracy of the Lyapunov equation solver.** An issue that is also of importance for the inexact inverse iteration method for the standard eigenvalue problem  $Ax = \lambda x$ , is that the error on the iterative solve should decrease proportional to the residual norm of the approximate eigenpair. For standard eigenproblems, inverse iteration requires the solution of

$$(3.8) \quad Ay_j = x_j.$$

An Arnoldi based method with zero initial guess for  $y_j$  produces the Krylov space

$$(3.9) \quad \text{span}\{x_j, Ax_j, \dots, A^{k-1}x_j\} .$$

Let  $r_j = Ax_j - \lambda_j x_j$  be the residual of the approximate eigenpair of  $A$  at iteration  $j$ . For inexact inverse iteration to converge, we impose that  $\|x_j - Ay_j\|_2 \leq \tau \|r_j\|_2$  with  $\tau < 1$  [32]. Since at some point,  $y_j$  makes a small angle with  $x_j$ , it may be more appropriate to solve (3.8) with initial guess  $\lambda_j^{-1}x_j$ , so that  $y_j = \lambda_j^{-1}x_j + \tilde{y}$  with

$$(3.10) \quad \begin{aligned} A\tilde{y} &= \lambda_j^{-1}(\lambda_j x_j - Ax_j) \\ &= -\lambda_j^{-1}r_j . \end{aligned}$$

Because the norm of the right-hand side in (3.10) is proportional to  $\|r_j\|_2$ , the remaining iterations only have to reduce this norm further with a factor  $\tau$ . This relative reduction is usually attained with roughly the same number of iterations for all  $j$ . Since

$$\text{span}\{x_j, Ax_j, \dots, A^{k-1}x_j\} = \text{span}\{x_j, r_j, \dots, A^{k-2}r_j\} ,$$

$k-1$  iterations for (3.10) produce a  $k-1$  dimensional Krylov space which is a subspace of (3.9). This implies that  $k$  iterations for (3.8) produce  $\|x_j - Ay_j\|_2$  of the order of  $\tau \|r_j\|_2$ . See [32] for the details.

We have a similar situation for the Lyapunov solution in Algorithm 3.3. Recall that approximate eigenvectors of (1.5) have the form  $Z_j = x_1 x_2^T + x_2 x_1^T$  if  $\mu \neq 0$ . (For  $\mu = 0$  we can derive a similar result.) With  $Z_j = V_j D_j V_j^T$ , we have that  $\text{Range}(V_j) = \text{Range}([x_1, x_2])$ . Define the residuals

$$(3.11) \quad \begin{aligned} r_i &= A^{-1}(A + \lambda_j B - \mu_i M)x_i \\ &= x_i + \lambda_j T x_i - \mu_i S x_i \end{aligned}$$

for  $i = 1, 2$ , where  $T = A^{-1}B$ . As was mentioned before, the eigendecomposition of the right-hand side of (3.5) is  $P_j C_j P_j^T$  where

$$(3.12) \quad \begin{aligned} \text{Range}(P_j) &= \text{Range}(SV_j) + \text{Range}(TV_j) \\ &= \text{Range}([Sx_1, Sx_2]) + \text{Range}([Tx_1, Tx_2]) \\ &= \text{Range}([Sx_1, Sx_2]) + \text{Range}([x_1 - r_1, x_2 - r_2]) . \end{aligned}$$

The last term in (3.12) tends to  $\text{Range}([x_1, x_2]) = \text{Range}(V_j)$  as  $r_1$  and  $r_2$  tend to zero.

Now let, as for the standard eigenvalue problem,

$$(3.13) \quad Y_j = \tilde{Y}_j - \lambda_j^{-1}((x_1 - r_1)(x_2 - r_2)^T + (x_2 - r_2)(x_1 - r_1)^T)$$

where the second term in the right-hand side lies in  $\text{Range}(P_j)$ .

LEMMA 3.1.

$$S\tilde{Y}_j + \tilde{Y}_j S^T = O(\|r_1\|_2) + O(\|r_2\|_2) .$$

*Proof.* From (3.5) and (3.3), we have that

$$(3.14) \quad SY_j + Y_j S^T = S(x_1 x_2^T + x_2 x_1^T)T^T + T(x_1 x_2^T + x_2 x_1^T)S^T .$$

Using (3.13), we find that

$$(3.15) \quad \begin{aligned} S\tilde{Y}_j + \tilde{Y}_j S^T &= Sx_1(x_2^T T^T + \lambda_j^{-1} x_2^T) + Sx_2(x_1^T T^T + \lambda_j^{-1} x_1^T) \\ &\quad + (Tx_2 + \lambda_j^{-1} x_2)x_1^T S^T + (Tx_1 + \lambda_j^{-1} x_1)x_2^T S^T \\ &\quad + O(\|r_1\|_2) + O(\|r_2\|_2) . \end{aligned}$$

From (3.11), we have that

$$Tx_i + \lambda_j^{-1} x_i = \lambda_j^{-1} (r_i + \mu_i Sx_i) .$$

Substituting this into (3.15), we obtain

$$\begin{aligned} S\tilde{Y}_j + \tilde{Y}_j S^T &= \lambda_j^{-1} Sx_1(r_2^T + \mu_2 x_2^T S^T) + \lambda_j^{-1} Sx_2(r_1^T + \mu_1 x_1^T S^T) \\ &\quad + \lambda_j^{-1} (r_2 + S\mu_2 x_2)x_1^T S^T + \lambda_j^{-1} (r_1 + S\mu_1 x_1)x_2^T S^T \\ &\quad + O(\|r_1\|_2) + O(\|r_2\|_2) . \end{aligned}$$

Since  $\mu_1 + \mu_2 = 0$ , the terms in  $\mu_i$  cancel out and the proof of the lemma follows.  $\square$  Similar to the solution of (3.8), we see that the solution of (3.14) only requires a relative reduction of the residual norm by a factor  $\tau$ .

From the numerical examples, we will indeed see a reducing residual norm for the Lyapunov equation for a fixed number of Krylov steps.

**3.5. The solution of the projected eigenvalue problem.** Step 2.1 in Algorithm 3.3 consists of the solution of the small size projected problem. We could use the QZ method for the related Kronecker eigenvalue problem, but this has two disadvantages: first,  $k^2$  can be of the order of a thousand or more, which makes the QZ method relatively expensive. More important perhaps is that the obtained eigenvector has to be forced to be symmetric to ensure the computed eigenvalue is simple, and this requires additional manipulations. We therefore use Algorithm 3.2 with the Bartels and Stewart method [4] as direct Lyapunov solver.

To speed-up the iterations on the small scale problem, we use shifted inverse iteration, with shift  $\sigma$ , which is equal to zero for the iteration  $j = 1$ , and  $\lambda_{j-1}$  for  $j > 1$ :

$$(\tilde{A}_j + \sigma \tilde{B}_j) \tilde{Y}_j \tilde{M}_j^T + \tilde{M}_j \tilde{Y}_j (\tilde{A}_j + \sigma \tilde{B}_j)^T = \tilde{B}_j \tilde{Z}_j \tilde{M}_j^T + \tilde{M}_j \tilde{Z}_j \tilde{B}_j^T .$$

The projected problem is formed from the dominant eigenvectors of the solution of (3.3). In our numerical examples, we perform the projection with the entire Krylov space of dimension  $k$ , i.e. the Lyapunov solution is not truncated to lower rank.

**4. Numerical examples.** In this section we present three examples arising from discretized partial differential equations. We also describe computations on a model problem to compare our approach with shift-and-invert Arnoldi.

**4.1. The Olmstead model.** The mathematical model represents the flow of a layer of viscoelastic fluid heated from below [38] [17]. The equations are

$$\begin{aligned} \frac{\partial u}{\partial t} &= (1 - C) \frac{\partial^2 v}{\partial X^2} + C \frac{\partial^2 u}{\partial X^2} + Ru - u^3 \\ B \frac{\partial v}{\partial t} &= u - v \end{aligned}$$

TABLE 4.1  
Result for the Olmstead equation for  $R = 3$  and  $k = 20$

$j$	$\lambda_j$	$\mu$	$\ r_j\ _2$	$\gamma_j$
1	1.78806	-0.250361	$6.0 \cdot 10^{-1}$	$8.5 \cdot 10^{-05}$
2	1.76932	1.26259	$4.7 \cdot 10^{-1}$	$3.4 \cdot 10^{-10}$
3	1.56152	5.19272	$1.4 \cdot 10^0$	$1.8 \cdot 10^{-8}$
4	1.93286	-4.68848 <i>i</i>	$7.6 \cdot 10^{-1}$	$6.5 \cdot 10^{-9}$
5	1.44784	-4.18513 <i>i</i>	$4.5 \cdot 10^{-3}$	$9.2 \cdot 10^{-10}$
6	1.44784	-4.18513 <i>i</i>	$1.7 \cdot 10^{-6}$	$5.3 \cdot 10^{-12}$
7	1.44784	-4.18513 <i>i</i>	$3.8 \cdot 10^{-8}$	$5.4 \cdot 10^{-14}$
8	1.44784	-4.18513 <i>i</i>	$1.1 \cdot 10^{-7}$	$4.4 \cdot 10^{-14}$
9	1.44784	-4.18513 <i>i</i>	$3.7 \cdot 10^{-8}$	$2.7 \cdot 10^{-14}$

where  $u$  represents the speed of the fluid and  $v$  is related to viscoelastic forces. The boundary conditions are  $u(0) = u(1) = 0$  and  $v(0) = v(1) = 0$ . After discretization with central differences with grid-size  $h = 1/(n/2 + 1)$ , the equations may be written as  $dx/dt = f(x)$  with  $x = [u_1, v_1, u_2, v_2, \dots, u_{N/2}, v_{N/2}]^T$ . We consider the Jacobian  $A + \lambda B = \partial f / \partial x$  for  $n = 20,000$ ,  $B = 2$ ,  $C = 0.1$  and  $\lambda = R \in [0.6, 5]$ , evaluated in the trivial steady state solution.

We used Algorithm 3.3 to solve this problem with the following parameters. We discretized the problem around  $R = 3$ . We used  $k = 20$  Krylov vectors in the solution of the Lyapunov equation. Table 4.1 shows the norm of

$$r_j = Ax + \lambda Bx - \mu Mx ,$$

the residual norm of the Lyapunov equation (Step 2.4)

$$\gamma_j = \|SY_jT^T + TY_jS^T - P_jC_jP_j^T\|_F ,$$

versus the iteration count. Note that  $\lambda = 1.44784$  corresponds to  $R = 4.44784$ . The final value of  $\mu$  is  $\pm 4.18513i$ .

Figure 4.1 shows slow decay for the absolute values of the eigenvalues of  $Y_1$  (first iteration) and the gap between ‘large’ and ‘small’ eigenvalues of  $Y_9$ . The two dominant eigenvalues of  $Y_9$  appear to be positive, which confirms, following Theorem 2.2, that  $\mu$  is purely imaginary. Note that the solution of the Lyapunov equation in the first iteration,  $Y_1$ , can be approximated by a rank  $r = 13$  matrix with a relative error  $10^{-6}$ . If we would use Algorithm 3.2, a block Krylov method with block size 26 would have to be used. This is too impractical by far. The projection step in Algorithm 3.3 solves this difficulty.

**4.2. The 2D Brusselator model.** The trimolecular reaction scheme in a two-dimensional square reactor can be studied by the Brusselator model. For more details, see [8, Chapter 5].

In this example,  $\alpha$  and  $\beta$  are the concentrations of a continuous input of both reactants where the unknowns  $X$  and  $Y$  are the concentrations of the other components in the chemical reaction. Under certain conditions of the parameters, a steady state solution is reached for  $X \equiv \alpha$  and  $Y \equiv \beta/\alpha$ . The stability of the steady state can be analysed by first perturbing the solution into  $X = \alpha + x$  and  $Y = \beta/\alpha + y$  and then analysing the stability of the linear equations

$$\frac{\partial x}{\partial t} = (\beta - 1)x + \alpha^2 y + D_x \nabla^2 x$$

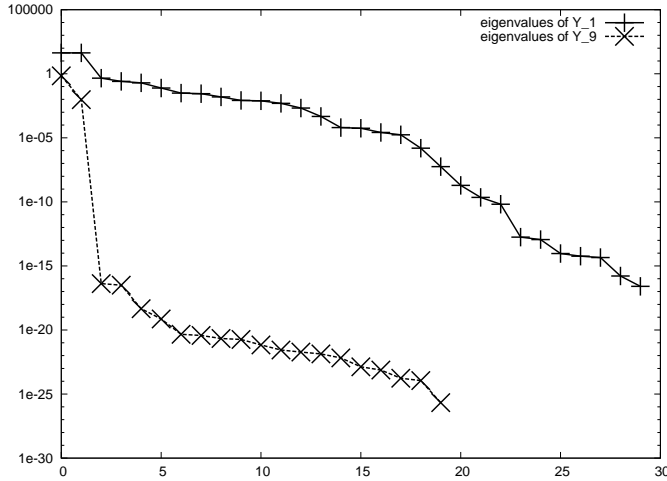


FIG. 4.1. Eigenvalues of  $Y_1$  and  $Y_9$  for the Olmstead equation

$$\frac{\partial y}{\partial t} = -\beta x - \alpha^2 y + D_y \nabla^2 y$$

where

$$\nabla^2 = \frac{\partial^2}{\partial r^2} + \frac{\partial^2}{\partial s^2},$$

and  $r$  and  $s$  are the spatial coordinates. The boundary conditions are

$$\begin{aligned} \frac{\partial x(0, s, t)}{\partial r} &= \frac{\partial x(r, 0, t)}{\partial s} = \frac{\partial x(L, s, t)}{\partial r} = \frac{\partial x(r, L, t)}{\partial s} = 0 \\ \frac{\partial y(0, s, t)}{\partial r} &= \frac{\partial y(r, 0, t)}{\partial s} = \frac{\partial y(L, s, t)}{\partial r} = \frac{\partial y(r, L, t)}{\partial s} = 0 \end{aligned}$$

The parameters are chosen as  $D_x = 1.6 \cdot 10^{-3}$ ,  $D_y = 8.0 \cdot 10^{-3}$ ,  $\alpha = 2.0$  and  $\beta = 4.6$ . Discretization by finite differences with gridsize  $h = L/(N + 1)$  leads to the linear system of ODE's  $\dot{u} = (A + \lambda B)u$  of dimension  $n = 2N^2$ . Note that for this problem  $B$  is again a singular matrix: it has eigenvalues 0 and 1. We have chosen  $L = 0.0798443$ . In our analysis, we chose  $\lambda = \beta$  as parameter and  $n = 20,000$ .

We used  $k = 20$  Arnoldi vectors, and a tolerance of  $\tau = 1.10^{-4}$ . The method converged in one iteration, i.e. after one iteration  $\lambda = 5$  was found with a residual norm  $\|(A + \lambda_j B - \mu M)x\|_2 \simeq 10^{-8}$ . The value of  $\lambda = 5$  corresponds to  $\beta = 5$  and  $\mu = \pm 2i$ .

**4.3. Heat equation.** The last example is the heat equation on a square grid with spatial coordinates  $(x, y)$  in  $[0, 1] \times [0, 1]$ :

$$\begin{aligned} \frac{\partial u(x, y)}{\partial t} + \nabla^2 u(x, y) &= 0 \quad \text{for } y > 0 \\ \frac{\partial u(x, 0)}{\partial t} + \nabla^2 u(x, 0) + \lambda \int_{x=0}^1 u(x, y) &= 0 \quad \text{for } y = 0. \end{aligned}$$

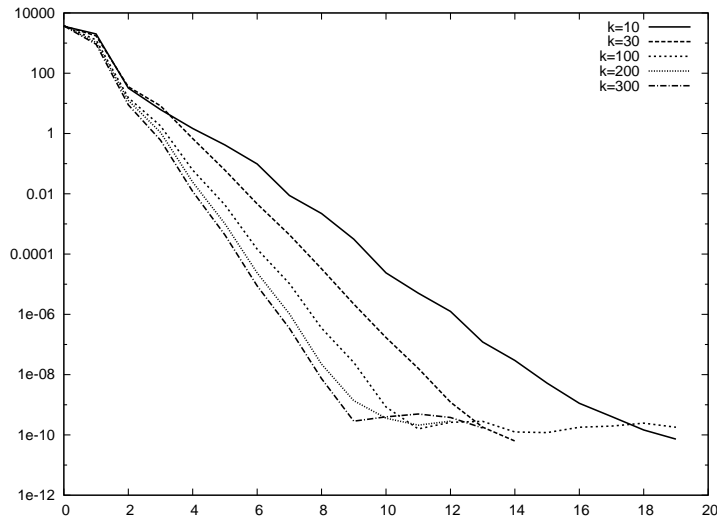


FIG. 4.2. Result for the 2D heat equation

The last term in the equation for  $y = 0$  is a controller that consists of the mean temperature in the  $x$  direction. We discretized the problem with 100 points in each direction in order to obtain an  $n = 10,000$  size problem.

We used Algorithm 3.3 with four values of the number of Arnoldi vectors,  $k$ . The residual norm  $\|(A + \lambda B - \mu M)x\|_2$  is plotted for the four values of  $k$  as a function of the iteration count  $j$  in Figure 4.2. All runs converge to the same value of  $\lambda = 4.67958$  and eigenvalue  $\mu = 0$ . The figure shows convergence behaviour that is typical for inexact inverse iteration: the more accurate the linear solves, the faster is the method, with a maximum convergence speed for the exact solves. There is a slight difference with standard eigenvalue problems: since we use the entire Krylov space in the projection in Algorithm 3.3, a larger  $k$  induces a larger search space so that eigenvalue approximates can be slightly more accurate.

**4.4. Comparison with shift-and-invert Arnoldi.** Here we compare the method introduced in this paper with the traditional shift-and-invert Arnoldi approach. We have constructed a problem to illustrate the points that we made in §1.

We generated  $n \times n$  matrices with  $n = 10000$ ,  $M = I$ , and where  $A$  and  $B$  are such that, for  $\lambda = 1$ ,  $A + \lambda B$  has eigenvalues  $-1, -2, \dots, -9998$  and the complex pair  $\pm 30i$ ; and for  $\lambda = 0$ ,  $A + \lambda B$  has eigenvalues  $-1, -2, \dots, -9998$  and the complex pair  $-30 \pm 30i$ . That means that the right-most eigenvalue for  $\lambda = 0$  is  $-1$ , which would correspond to a stable steady state in a dynamical systems setting, whereas for  $\lambda = 1$ , the rightmost eigenvalues are the purely imaginary pair  $\pm 30i$ . As  $\lambda$  varies from 0 to 1 the complex pair moves towards the imaginary axis. This construction simulates the physical situation in the double-diffusive convection example [50], [6] mentioned in the Introduction.

From earlier work [5] [12] [7] [34], we know that detecting right-most eigenvalues is not always an easy task. The reason is that most eigenvalue solvers search for eigenvalues near a target point, called a shift.

We compare the method from this paper with the shift-and-invert Arnoldi method. First we take the *exact* value of the parameter  $\lambda$  for which  $A + \lambda B$  has the purely



imaginary pair  $\pm 30i$ , that is we set  $\lambda = 1$ . We apply the shift-and-invert Arnoldi method with shift zero and a random starting vector to  $(A + \lambda B)^{-1}M$  which in this case is simply  $(A + B)^{-1}$ . After 10 Arnoldi steps there was a Ritz value at  $-1$  with residual norm  $1.3 \cdot 10^{-7}$ ; that is, the eigenvalue nearest the shift was calculated, not the eigenpair on the imaginary axis. In fact, there was no Ritz value near  $\pm 30i$ . However, after 20 Arnoldi steps the purely imaginary eigenvalue was found with residual norm  $3.6 \cdot 10^{-13}$ .

Next, we apply our method with  $\lambda = 0$ , that is we do not assume we know the value of  $\lambda$  where there is a purely imaginary eigenpair. We used a block Krylov space of dimension 10. Since we use the starting vector  $v \otimes v$  in inverse iteration, we have a block Krylov space of block size two, which means that 5 Arnoldi steps were performed. The eigenpair  $\mu = \pm 30i$  and the parameter value  $\lambda = 1$  were calculated with a residual norm of  $2 \cdot 10^{-13}$ . This is cheaper than shift-and-invert Arnoldi for one value of  $\lambda$ .

In a practical continuation setting using shift-and-invert Arnoldi, where the exact value of the parameter  $\lambda$  was not known beforehand, sparse factorizations of  $A + \lambda B$  would have to be carried out for several values of  $\lambda$  and then the consequent Krylov spaces would have to be constructed.

The shift-and-invert Arnoldi method can be used with a non-zero shift in order to improve its reliability [11] [34]. When we used the shift 10, i.e. we compute the eigenvalues of  $(A + B + 10I)^{-1}$ , then eigenvalue  $\mu = \pm 30i$  was found with a residual norm  $2.0 \cdot 10^{-7}$  after 20 iterations. This is still more expensive than our method.

**5. Conclusions.** We have described a method for the computation of good starting values for use in algorithms to compute Hopf bifurcations in large scale dynamical systems. This work was originally motivated by the use of the bialternate product in [19] and [15], which has the significant advantage that a good starting guess for an unknown complex eigenvalue is not required. The disadvantage of the bialternate product is that it is an  $n^2$  dimensional matrix, though the restriction to the anti-symmetric subspace reduces this to  $n(n-1)/2$ . We overcome this by a reformulation as an inexact inverse iteration algorithm that requires a sequence of  $n$ -dimensional Lyapunov-type equations, with the key feature that the right hand sides are of low rank. This results in an efficient procedure, provided the Lyapunov equations are solved accurately. Numerical results illustrate the power of the method.

The results of this paper could also be used for computing eigenvalues of  $Ax = \theta Mx$  nearest the imaginary axis. Shift-and-invert based methods are not always reliable when a rough idea of  $\text{Im}(\theta)$  is unknown. Our method could be applied for finding the smallest  $\lambda$  for which  $\theta = \lambda + i\mu$ ,  $\lambda, \mu \in \mathbf{R}$ , is an eigenvalue without computing  $\mu$ .

**Acknowledgement.** This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its author(s).

The authors also thank Bart Vandereycken and Marc Embree for discussions on the solution of Lyapunov equations.

#### REFERENCES

- [1] A. ANTOUNAS, D. SORENSSEN, AND Y. ZHOU, *On the decay rate of Hankel singular values and related issues*, Systems & Control Letters, 46 (2002), pp. 323–342.

- [2] F. ATKINSON, *Multiparameter eigenvalue problems*, Academic Press, New York, 1972.
- [3] Z. BAI, J. DEMMEL, J. DONGARRA, A. RUHE, AND H. VAN DER VORST, eds., *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia, PA, USA, 2000.
- [4] R. BARTELS AND G. STEWART, *Algorithm 432: Solution of the matrix equation  $ax + xb = c$* , Comm. ACM, 15 (1972), pp. 820–826.
- [5] K. CLIFFE, T. GARRATT, AND A. SPENCE, *Calculation of eigenvalues of the discretised Navier-Stokes and related equations*, in The mathematics of finite elements and applications VII MAFELAP, J. Whiteman, ed., Academic Press, 1990, pp. 479–486.
- [6] ———, *Eigenvalues of the discretized Navier-Stokes equation with application to the detection of Hopf bifurcations*, Advances in Computational Mathematics, 1 (1993), pp. 337–356.
- [7] ———, *Eigenvalues of block matrices arising from problems in fluid mechanics*, SIAM Journal on Matrix Analysis and Applications, 15 (1994), pp. 1310–1318.
- [8] B. DE DIER, *An investigation of a reaction-diffusion model exhibiting dissipative structures*, PhD thesis, K.U.Leuven, Leuven (Heverlee), Belgium, 1998.
- [9] J. DONGARRA, B. STRAUGHAN, AND D. WALKER, *Chebyshev tau-QZ algorithm methods for calculating spectra of hydrodynamic stability problems*, Applied Numerical Mathematics, 22 (1996), pp. 399–434.
- [10] D. FOKKEMA, G. SLEIPEN, AND H. VAN DER VORST, *Jacobi-Davidson style QR and QZ algorithms for the reduction of matrix pencils*, SIAM Journal on Scientific Computing, 20 (1999), pp. pp.94–125.
- [11] T. GARRATT, *The numerical detection of Hopf bifurcations in large systems arising in fluid mechanics*, PhD thesis, University of Bath, UK, 1991.
- [12] T. GARRATT, G. MOORE, AND A. SPENCE, *Two methods for the numerical detection of Hopf bifurcations*, in Bifurcation and Chaos : Analysis, Algorithms, Applications, R. Seydel, F. Schneider, and H. Troger, eds., Birkhäuser, 1991, pp. 119–123.
- [13] G. GOLUB, S. NASH, AND C. VAN LOAN, *A Hessenberg-Schur method for the problem  $AX + XB = C$* , IEEE Trans. Auto. Control, 24 (1979), pp. 909–913.
- [14] G. GOLUB AND C. VAN LOAN, *Matrix computations*, The Johns Hopkins University Press, 3rd ed., 1996.
- [15] W. GOVAERTS, *Numerical methods for bifurcations of dynamical equilibria*, SIAM, Philadelphia, PA, USA, 2000.
- [16] W. GOVAERTS, J. GUCKENHEIMER, AND A. Khibnik, *Defining functions for multiple Hopf bifurcations*, SIAM J Numer. Anal., 34 (1997), pp. 1269–1288.
- [17] M. GRINFELD, *Dynamics of a model equation in viscoelasticity*, in Proceedings Equadiff 91, Singapore, 1993, World Scientific.
- [18] M. GU, E. MENGI, M. OVERTON, J. XIA, AND J. ZHU, *Fast methods for estimating the distance to uncontrollability*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 477–502.
- [19] J. GUCKENHEIMER, S. GUERON, AND R. HARRIS-WARRICK, *Mapping the dynamics of a bursting neuron*, Philos. Trans. Roy. Soc. London Ser B, 341 (1993), pp. 345–359.
- [20] J. GUCKENHEIMER, M. MYERS, AND B. STURMFELS, *Computing Hopf bifurcations I*, SIAM J. Numer. Anal., 34 (1997), pp. 1–21.
- [21] T. GUDMUNDSSON AND A. LAUB, *Approximate solution of large sparse Lyapunov equations*, IEEE Trans. Auto. Control, 39 (1994), pp. 1110–1114.
- [22] S. GUGERCIN AND J.-R. LI, *Smith-type Methods for Balanced Truncation of Large Sparse Systems*, Springer-Verlag, Berlin, Heidelberg, 2005.
- [23] S. HAMMARLING, *Numerical solution of the stable, non-negative definite Lyapunov equation*, IMA Journal on Numerical Analysis, 2 (1982), pp. 303–323.
- [24] M. HOCHSTENBACH AND B. PLESTENJAK, *A Jacobi-Davidson type method for a right definite two-parameter eigenvalue problem*, SIAM Journal on Matrix Analysis and Applications, 24 (2002), pp. 392–410.
- [25] M. E. HOCHSTENBACH, T. KOSIR, AND B. PLESTENJAK, *A Jacobi-Davidson type method for the two-parameter eigenvalue problem*, SIAM J. Matrix Anal. Appl., 26 (2005), pp. 477–497.
- [26] A. S. HODEL, K. POOLLA, AND B. TENISON, *Numerical solution of the Lyapunov equation by approximate power iteration*, Linear Alg. Appl., 236 (1996), pp. 205–230.
- [27] D. HU AND L. REICHEL, *Krylov-subspace methods for the Sylvester equation*, Linear Alg. Appl., 172 (1992), pp. 283–313.
- [28] I. JAIMOUKHA AND E. KASENALLY, *Krylov subspace methods for solving large Lyapunov equation*, SIAM Journal on Numerical Analysis, 31 (1994), pp. 227–251.
- [29] E. JARLEBRING, *The spectrum of delay-differential equations: numerical methods, stability and perturbation*, PhD Thesis, Inst. Comp. Math, TU Braunschweig, Braunschweig, Germany, 2008.

- [30] Y. KUZNETSOV, *Elements of Applied Bifurcation Theory. 2nd ed.*, vol. 112 of Applied Math. Sci., Springer-Verlag, New York, 1998.
- [31] R. LEHOUCQ AND K. MEERBERGEN, *Using generalized Cayley transformations within an inexact rational Krylov sequence method*, SIAM Journal on Matrix Analysis and Applications, 20 (1998), pp. 131–148.
- [32] K. MEERBERGEN AND R. MORGAN, §11.2. *Inexact methods*, in Templates for the solution of algebraic eigenvalue problems: a practical guide, Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, eds., Philadelphia, PA, USA, 2000, SIAM.
- [33] K. MEERBERGEN AND D. ROOSE, *Finding eigenvalues to the right of a given line*, in Proceedings of the Second IMACS Symposium on Iterative Methods in Linear Algebra, P. Vassilevski and S. Margenov, eds., vol. 3 of IMACS Series in Computational and Applied Mathematics, Dept. of Computer Science, Rutgers University, Piscataway, NJ 08855, 1996, IMACS, pp. 402–412.
- [34] ———, *Matrix transformations for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems*, IMA Journal on Numerical Analysis, 16 (1996), pp. 297–346.
- [35] K. MEERBERGEN, A. SPENCE, AND D. ROOSE, *Shift-invert and Cayley transforms for detection of rightmost eigenvalues of nonsymmetric matrices*, BIT, 34 (1994), pp. 409–423.
- [36] W. MICHIELS AND S.-I. NICULESCU, *Stability and stabilization of time-delay systems. An eigenvalue based approach*, SIAM, Philadelphia, PA, USA, 2007.
- [37] R. NONG AND D. SORENSEN, *A parameter free ADI-like method for the numerical solution of large scale Lyapunov equations*, Technical Report TR09-16, Rice University, CAAM, Houston, Texas, 2009.
- [38] W. OLMSTEAD, W. DAVIS, S. ROSENBLAT, AND W. KATH, *Bifurcation with memory*, SIAM J. Appl. Math., 46 (1986), pp. 171–188.
- [39] B. PARLETT, *The symmetric eigenvalue problem*, Classics in Applied Mathematics, SIAM, Philadelphia, PA, USA, 1998.
- [40] B. PARLETT AND Y. SAAD, *Complex shift and invert strategies for real matrices*, Linear Alg. Appl., 88/89 (1987), pp. 575–595.
- [41] D. PEACEMAN AND H. RACHFORD, *The numerical solution of elliptic and parabolic differential equations*, J. Soc. Indust. Appl. Math., 3 (1955), pp. 28–41.
- [42] T. PENZL, *Eigenvalue decay bounds for solutions of Lyapunov equations: the symmetric case*, Systems Control Lett., 40 (2000), pp. 139–144.
- [43] M. ROBBÉ AND M. SADKANE, *Use of near-breakdowns in the block Arnoldi method for solving large Sylvester equations*, Applied Numerical Mathematics, 58 (2008), pp. 486–498.
- [44] Y. SAAD, *Numerical solution of large Lyapunov equations*, in Signal Processing, Scattering, Operator Theory, and Numerical Methods, M. Kaashoek, J. van Schuppen, and A. Ran, eds., Boston, MA, 1990, Birkhäuser, pp. 503–511.
- [45] ———, *Numerical methods for large eigenvalue problems*, Algorithms and Architectures for Advanced Scientific Computing, Manchester University Press, Manchester, UK, 1992.
- [46] R. SEYDEL, *From Equilibrium to Chaos. Practical Bifurcation and Stability Analysis, 2nd ed.*, Springer-Verlag, Berlin, 1994.
- [47] V. SIMONCINI, *A new iterative method for solving large-scale lyapunov matrix equations*, SIAM Journal on Scientific Computing, 29 (2007), pp. 1268–1288.
- [48] V. SIMONCINI AND V. DRUSKIN, *Convergence analysis of projection methods for the numerical solution of large Lyapunov equations*, SIAM Journal on Numerical Analysis, 47 (2009), pp. 828–843.
- [49] R. SMITH, *Matrix equation  $XA+BX = C$* , SIAM J. Appl. Math., 16 (1968), pp. 198–201.
- [50] J. TURNER, *Buoyancy effects in fluids*, C.U.P. Cambridge, 1973.