An Arnoldi-frontal approach for the stability analysis of flows in a collapsible channel

Yujue Hao, Zongxi Cai, Steven Roper, Xiaoyu Luo*

School of Mathematics and Statistics
University of Glasgow
University Gardens, Glasgow G12 8QW, UK
Corresponding author: xiaoyu.luo@glasgow.ac.uk

In this paper, we have developed a new approach based on a combination of the Arnoldi and frontal methods, which is suitable for solving large sparse asymmetric and generalized complex eigenvalue problems. The new eigensolver seeks the most unstable eigen-solution in the Krylov subspace and makes use of the efficiency of the frontal solver developed for the finite element methods. The approach is used for a stability analysis of flows in a collapsible channel and is found to significantly improve the computational efficiency compared to the traditionally used QZ solver or a standard Arnoldi method. With the new approach, we are able to validate the previous results obtained either on a much coarser mesh or estimated from unsteady simulations. New neutral stability solutions of the system are also obtained which are beyond the limit of previously used methods.

Keywords: Arnoldi method; stability analysis; collapsible channel flows; generalized eigenvalue problem; frontal solver; finite element method

1. Introduction

Eigenvalue problems occur frequently in problems arising in many branches of science, such as computational fluid mechanics [Cliffe *et al.*, 1994], statistics [Rapcsak, 2004], engineering [Roger *et al.*, 1986; Bathe and Wilson, 1973; Andy and Nair, 2005; Auckenthaler *et al.*, 2011; Misrikhanov and Ryabchenko, 2006], quantum physics [Scott *et al.*, 1990], and meteorologic modelling [Cullum and Willoughby, 1986]. Such problems are usually solved using numerical methods. Often the discretization of the systems leads to large asymmetric matrices that require efficient algorithms to manipulate and store. Seeking such algorithms has been a central focus over the last 50 years. Although various advances have been made, there is no single algorithm which is effective and efficient for different engineering problems. A traditionally used eigensolver for a generalized eigenvalue problem,

$$\mathbf{AX} = \lambda \mathbf{X},$$

is the generalized Schur decomposition method, which factorises both matrices as $\mathbf{A} = \mathbf{QSZ}^H$ and $\mathbf{B} = \mathbf{QTZ}^H$, where $\mathbf{Q}$ and $\mathbf{Z}$ are unitary, ^{H} denotes a conjugate-transpose, and $\mathbf{S}$ and $\mathbf{T}$ are upper triangular matrices [Moler and Stewart, 1973]. The generalised Schur decomposition is also known as the QZ method. This is based on the QR decomposition of a matrix $\mathbf{A}$ into a product $\mathbf{A} = \mathbf{QR}$ of an orthogonal matrix **Q** and an upper triangular matrix **R**. The QZ method solves for all the eigenmodes of the system. As such, it is only suitable for smaller sized eigenvalue problems (matrix size in the order of hundreds) and is impractical when the dimensions of the problem become large (e.g. matrix size $> 5000 \times 5000$). To overcome this problem, various projection methods have been developed since the 1950s [Wu and Simon, 2000; Parlett *et al.*, 1985; Freund *et al.*, 1993; Parlett and cott, 1979; Bathe, 1971, 2013; Morgan, 2000].

Among these, the Arnoldi-type method [Morgan, 2000] is the one that can be used to solve large sparse asymmetric (non-Hermitian) eigenvalue problems. Such a problem commonly occurs in the stability analysis of complex systems, such as flows in a collapsible channel [Luo *et al.*, 2008; Stewart *et al.*, 2010b; Xu *et al.*, 2013, 2014]. Flows in collapsible tubes, or flows in collapsible channels when simplified in two dimensions, have grasped researchers' attention over the last 30 years [Shapiro, 1977; Elad *et al.*, 1987; Kamm and Shapiro, 1979; Cancelli and Pedley, 1985; Jensen, 1990; Luo and Pedley, 1996; Stewart *et al.*, 2010a], because it has provided insight for many physiological applications such as flow through vocal folds [Cisonni *et al.*, 2010], collapsed intramyocardial coronary blood arteries during heart contraction in systole [Guiot *et al.*, 1990], branchial arteries compressed by a sphygmomanometry cuff [Bertram and Ribreau, 1989], and flows in giraffe jugular veins [Brook and Pedley, 2002]. One characteristic of such systems is that they can be dynamically unstable due to fluid-structure interaction. Stability analysis has been widely used for studying the various oscillation mechanisms [Luo and Pedley, 1998]. Often such an analysis leads to a generalised eigenvalue problem with asymmetric large sparse matrices [Cai and Luo, 2003]. Recent work by Liu *et al.* [2012], using the QZ solver, showed that the stability structure in collapsible channel flows can be quite different in the flow- and pressure-driven systems (where the driving force is either the flow rate or the pressure drop along the channel). However, further investigation on these stability structures is prohibited by the extensive memory and CPU requirements of the QZ solver they used. Using an orthogonal projection method such as the Arnoldi iteration enables us to solve a reduced eigenproblem containing only the first few eigenpairs. However, even with such a model reduction, the memory requirement of the Arnoldi iteration can still be huge as it requires the full assembly of the global matrices. In order to solve the stability problem for collapsible channel flows with a non-trivial basic state, one needs first to solve the nonlinear fluid-structure interaction equations numerically [Luo, 2015]. To avoid dealing with the large sparse matrices from the discretization of the finite element (FE) governing equations, a frontal scheme developed by Irons [Irons, 1970] was successfully used in the full numerical simulations [Rast, 1994; Luo and Pedley, 1998; Cai and Luo, 2003; Luo *et al.*, 2008; Liu *et al.*, 2012]. The frontal solver is CPU efficient as it assembles the large sparse matrix and eliminates equations only on the "front", i.e., a subset of elements, at a time. The front represents the transition region between the active and inactive element entries of the global matrices. In this work, we develop a combined Arnoldi-frontal approach and use it to solve the eigenvalue problem of the collapsible channel flows for the first time. Similar ideas, albeit for simpler problems, have been explored by Meerbergen et al. [Meerbergen and Roose, 1996, 1997], and Lehoucq et al. [Lehoucq *et al.*, 1997, 1998]. For example, Lehoucq *et al.* [1997] used the implicitly restarted Arnoldi methods with matrix transformation to compute the eigenvalues for discretised Navier-Stokes equations. They concluded that with careful implementation, implicitly restarted Arnoldi methods are reliable for linear stability analysis. This group also applied the Arnoldi iteration driven by a novel implementation of the Cayley transformation to the stability analysis of three dimensional steady flows on parallel computers [Lehoucq and Salinger, 2001]. We extend the previous studies by applying the analysis to a strongly coupled fluid-structure interaction problem, and show that by combining the implicitly restarted Arnoldi Method [Lehoucq *et al.*, 1997] with the frontal solver for our general asymmetric eigenvalue problem, we not only resolve the memory issue but also significantly reduce the computational time. Thus, the new solver enables us to validate the previous results and obtain additional neutral points that are beyond the reach of the QZ solver.

2. Model of Flows in a Collapsible Channel

2.1. *The problem description*

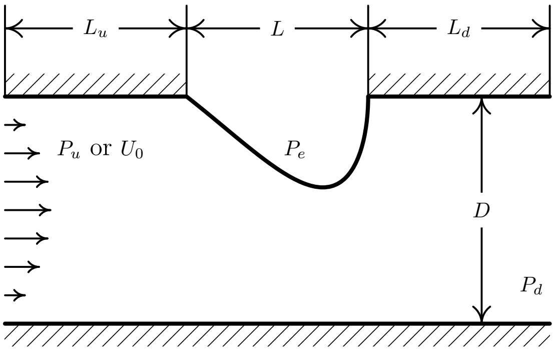


Fig. 1. The model configuration (not to scale), where the upper middle wall is replaced with an elastic beam of (undeformed) length L. The lengths of the upstream and downstream sections are denoted as L_u and L_d, respectively, the channel height is D, the external and downstream pressures are indicated as P_e,and P_d, respectively. Depending on the flow-driven or pressure-driven systems, either a steady parabolic entry flow with an average velocity U_0, or a steady plug flow with a pressure P_u, is specified at the upstream end.

The model configuration of the collapsible channel flow is shown in Fig. 1. The rigid channel has a width D, with a part of the upper wall being replaced by an elastic beam, which is subjected to an external pressure P_e. The flow is assumed incompressible and laminar, with the fluid density ρ and viscosity μ. We assume that the beam is a linear elastic material but allows geometrical nonlinearity. The extensional and bending stiffness of the beam are EA and EJ, respectively, where E is the Young' modulus, A is the cross-sectional area of the beam, and J is the second moment of cross-sectional area of the beam. The initial tension and the inertia contribution of the beam are assumed to be zero in this paper, the impact of these parameters has been studied in our earlier work [Luo and Pedley, 1998, 2000]. Damping and rotational inertia of the beam are both neglected.

2.2. *The Dimensionless Governing Equations*

The governing equations for the fully coupled fluid-structure interaction system are [Cai and Luo, 2003]:

$$\frac{\partial u_i}{\partial t} + u_j u_{i,j} = -p_{,i} + \frac{1}{Re} u_{i,jj}, \quad u_{i,i} = 0, \ i, j = 1, 2, \tag{1}$$

$$c_\kappa \kappa \kappa' + c_\lambda \lambda' + \lambda \tau_n = 0, \tag{2}$$

$$c_\kappa \left(\frac{1}{\lambda} \kappa' \right)' - c_\lambda \lambda \kappa (\lambda - 1) - \lambda \sigma_n + \lambda p_e = 0, \tag{3}$$

$$x_1' = \lambda \cos\theta, x_2' = \lambda \sin\theta, \tag{4}$$

$$\lambda \kappa = \theta', \tag{5}$$

where (1) are the Navier-Stokes equations for the flow, (2–3) are the momentum balance equations for the elastic beam, and (4–5) are auxiliary equations which are introduced to simplify the computations. All the quantities are non-dimensionlized as:

$$u_i = \frac{\bar{u}_i}{U_0}, \ \sigma_{ij} = \frac{\bar{\sigma}_{ij}}{\rho U_0^2}, p = \frac{\bar{p}}{\rho U_0^2}, \ t = \frac{\bar{t} U_0}{D}, \ l = \frac{\bar{l}}{D}, \ x_i = \frac{\bar{x}}{D}$$

$$\kappa^* = \kappa D, c_\lambda = \frac{EA}{\rho U_0^2 D}, c_\kappa = \frac{EJ}{\rho U_0^2 D^3}, Re = \frac{U_0 D \rho}{\mu} \tag{6}$$

where p is the fluid pressure, x_i and u_i are the coordinates and velocity components $(i = 1, 2)$, σ_{ij} are fluid stress components $(i, j = 1, 2)$, and σ_n, τ_n denote the fluid normal and shear stresses acting on the beam, respectively. t is time, l is the material coordinate of the beam in the undeformed configuration, κ and λ are curvature and stretch of the beam, respectively, θ is the angle of the beam with respect to the x-axis, and prime means derivative with respect to l. Re is the Reynolds number, and c_λ and c_κ represent the extensional and bending stiffness of the beam.

2.3. *Boundary conditions*

The boundary conditions are set up such that at the outlet it is stress free, which essentially sets the reference pressure to be zero ($P_d \simeq 0$); along all the walls, no-slip boundary condition is applied, with an external pressure P_e, applied along the beam. At the two ends of the beam, clamped support is employed. Specifically,

rigid wall :	$u = v = 0$,	at $y = 0, 0 \leqslant x \leqslant L_0$, $L_0 = L_u + L + L_d$ at $y = 1, 0 \leqslant x \leqslant L_u$ and $L_u + L \leqslant x \leqslant L_0$
beam:	$u = u_w$, $v = v_w$, $P_e = constant$	at $L_u < x < L_u + L$ and $y = 1$
beam ends :	$\theta = 0$,	at $x = L_u$, $y = 1$ and $x = L_u + L$, $y = 1$
outlet flow :	$\sigma_n = \sigma_t = 0$,	at $x = L_0$ and $0 \leqslant y \leqslant 1$

where $x = x_1, y = x_2$ are the system coordinates, with the origin at the bottom left corner of the channel, $u = u_1, v = u_2$, are the velocities of the fluid, and u_w, v_w are the velocities of the beam, σ_n, σ_t are the normal and tangential fluid stresses, respectively.

Unlike flows through a rigid tube, here more combinations of control parameters are possible. For example, one may specify the flow rate Q, or pressure drop P_{ud}, while keeping downstream transmural pressure $P_e - P_d$ constant. These are referred to as "flow-driven system"(also known as flux-driven), or "pressure-driven system", respectively [Liu *et al.*, 2012]. Each of these settings determines a specific system with its own unique characteristics. The commonly observed and "pressure-drop limitation" [Bertram and Castles, 1999], and "flow limitation" [Gavriely *et al.*, 1989], are interesting phenomena associated to these systems. Experimentally, these can be achieved by providing a hydraulic head upstream (pressure-driven), or a suction downstream (flow-driven).

The boundary conditions for these systems are, at the inlet,

flowrate driven:	$u = 6y(y-1)$, v=0,	at $x = 0$, $0 \leqslant y \leqslant 1$
pressure driven:	$P_{ud} = constant$,	at $x = 0$, $0 \leqslant y \leqslant 1$

2.4. *The stability analysis*

A Petrov-Galerkin method is used to discretize the system equations (1)–(5). The element type for flow is six-node triangular, with the second-order shape function N_i used for u and v, and the linear shape function L_i used for p. For the elastic beam, the three-node beam elements with second-order shape function are used for all the variables (x, y, θ, λ, and κ). The discretized finite-element equations of the coupled system can be written in a matrix form as

$$\mathbf{M}(\mathbf{U})\frac{d\mathbf{U}}{dt} + \mathbf{K}(\mathbf{U})\mathbf{U} - \mathbf{F} = \mathbf{0}, \tag{7}$$

where $\mathbf{U} = (u_j, v_j, p_j, x_j, y_j, \theta_j, \lambda_j, \kappa_j)$ is the global vector of unknowns, and $j = 1, \ldots, Nod$, Nod is the total nodal number. $\mathbf{M}$, $\mathbf{K}$ are the $n \times n$ mass and stiffness matrices, respectively, with $n \approx 8 \times Nod$, and $\mathbf{F}$ is a force like vector with dimension n. An arbitrary Lagrangian Eulerian (ALE) solver [Cai and Luo, 2003], which is shown to satisfy the geometrical conservation law [Liu *et al.*, 2012], is used to solve (7).

To study the stability of the system, we denote $\bar{\mathbf{U}}$ as a steady solution of (7), so that

$$\mathbf{K}(\bar{\mathbf{U}})\bar{\mathbf{U}} - \mathbf{F} = \mathbf{0}, \tag{8}$$

and apply an infinitesimal perturbation $\Delta\mathbf{U} = e^{\omega t}\tilde{\mathbf{U}}$, to get a perturbed solution, $\mathbf{U} = \bar{\mathbf{U}} + \Delta\mathbf{U}$, of (7). Here ω $(= \omega_R + i\omega_I)$ and $\tilde{\mathbf{U}}$ are the complex eigenvalues and eigenvectors, respectively. Specifically,

$$\begin{aligned}(\Delta\mathbf{U})_k &= Real\left[e^{(\omega_R+i\omega_I)t}\left((\tilde{\mathbf{U}}_R)_k + i(\tilde{\mathbf{U}}_I)_k\right)\right]\\ &= e^{\omega_R t}\left[(\tilde{\mathbf{U}}_R)_k \cos(\omega_I t) - (\tilde{\mathbf{U}}_I)_k \sin(\omega_I t)\right]\\ &= e^{\omega_R t}\left[\|\tilde{\mathbf{U}}_k\| \cos(\omega_I t + \phi_k)\right], \quad k = 1, \ldots, 8 \times Nod \end{aligned} \tag{9}$$

where $\|\tilde{\mathbf{U}}_k\| = \sqrt{(\tilde{\mathbf{U}}_R)^2_k + (\tilde{\mathbf{U}}_I)^2_k}$, is the eigen-amplitude, and $\phi_k = \arctan(\tilde{\mathbf{U}}_I)_k/(\tilde{\mathbf{U}}_R)_k$ is the phase angle at $t = 0$. It is clear that for a positive ω_R, the system is unstable, and for a negative one, the system is stable. $\omega_R = 0$ indicates a neutral stability which is associated with sustained self-excited oscillations when $\omega_I \neq 0$.

Substituting $\mathbf{U} = \bar{\mathbf{U}} + e^{\omega t}\tilde{\mathbf{U}}$ into (7), making use of the Taylor expansion and (8), we obtain a generalised eigenvalue problem [Luo *et al.*, 2008]:

$$\overline{\mathbf{K}}\tilde{\mathbf{U}} = \omega\overline{\mathbf{M}}\tilde{\mathbf{U}}, \tag{10}$$

where $\overline{\mathbf{M}} = \mathbf{M}(\overline{\mathbf{U}})$, and $\overline{\mathbf{K}} = \mathbf{K}(\overline{\mathbf{U}}) + \nabla_{\mathbf{U}}\mathbf{K}(\mathbf{U})|_{\overline{\mathbf{U}}}\overline{\mathbf{U}}$. Both $\overline{\mathbf{K}}$ and $\overline{\mathbf{M}}$ are sparse, and asymmetric matrices. $\overline{\mathbf{K}}$ is also positive definite. However, $\overline{\mathbf{M}}$ is necessarily singular since the continuity equation $(1)_2$ does not contribute to the mass matrix. For simplicity, henceforth we drop the overbar and tilda in (10).

3. The Eigensolvers

Three different algorithms are employed to solve the generalized eigenvalue problem (10). These are, the QZ Algorithm, the Arnoldi method with global matrices (AR-G), and the Arnoldi method with a frontal solver (AR-F). In developing these eigensolvers we have made use of the ARPACK software (`http://www.caam.rice.edu/software/ARPACK`).

3.1. *The QZ Algorithm*

The QZ solver adopts the following steps [Bai *et al.*, 1987]:

1. $\mathbf{K}$ is first reduced to an upper Hessenberg form and $\mathbf{M}$ is reduced to an upper triangular form (Schur form).
2. The effect of a shifted QR decomposition on $\mathbf{K}^{-1}\mathbf{M}$ (without forming the matrix product) is simulated by unitary equivalence transformations $\mathbf{Q}$ and $\mathbf{Z}$ on the matrix pair $\mathbf{K}$ and $\mathbf{M}$. This is done iteratively until $\mathbf{K}$ is reduced to triangular or quasi-triangular form, while preserving the triangular structure of $\mathbf{M}$.
3. Compute the eigenvalues and eigenvectors from the triangular matrix problem, and then transform back to get the original eigenpairs.

The QZ algorithm solves for all the eigenvalues and, optionally, all the eigenvectors. It requires $O(n^3)$ floating point operations and $O(n^2)$ memory locations, where $n \times n$ is the size of $\mathbf{K}$ and $\mathbf{M}$. Therefore the demand on the computer memory is prohibitively high and is extremely inefficient for problems of large matrices.

3.2. *The Arnoldi iterations with global matrices (AR-G)*

In the AR-G approach, we first transform the generalised eigenproblem (10) into a standard eigenproblem:

$$\mathbf{CU} = \Theta\mathbf{U}, \tag{11}$$

where $\Theta = 1/\omega$, and $\mathbf{C} = \mathbf{K}^{-1}\mathbf{M}$. The idea of the Arnoldi approach is that for a given $n \times n$ matrix $\mathbf{C}$, information on its largest eigenvalue can be sought by repeated application of $\mathbf{C}$ to a random vector, $\mathbf{v}_1$, to form the so-called Krylov subspace $\mathcal{K}_m(\mathbf{v}_1, ...\mathbf{v}_m)$[Arnoldi, 1951; Saad, 1996], where $\mathbf{v}_i = \mathbf{C}^{i-1}\mathbf{v}_1$ for $i = 1, ..., m$. Hence, $\mathbf{C}$ can be projected in the subspace:

$$\mathbf{CV}_m = \mathbf{V}_m\mathbf{H}_m + \mathbf{r}_m\mathbf{e}_m^T, \tag{12}$$

where $\mathbf{V}_m = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m)$ is a $n \times m$ matrix whose columns form the set of the Arnoldi vectors, with the normalisation $\mathbf{V}_m^H\mathbf{V}_m = \mathbf{I}_m$. The matrix $\mathbf{H}_m = \mathbf{V}_m^H\mathbf{C}\mathbf{V}_m$, is an $m \times m$ upper Hessenberg projection matrix, $\mathbf{r}_m = (\mathbf{I}_n - \mathbf{V}_m\mathbf{V}_m^H)\mathbf{C}\mathbf{v}_{m+1}$ is the residual vector, and $\mathbf{e}_m$ is the m-th standard basis vector of dimension m. Hence, we only need to solve a much smaller eigenvalue problem:

$$\mathbf{H}_m\mathbf{y} = \tilde{\Theta}\mathbf{y}, \tag{13}$$

where $\mathbf{V}_m\mathbf{y} \simeq \mathbf{U}$ in (11), and $\tilde{\Theta}$ is an approximation of Θ. The details are shown in the Algorithm 1.

We remark that while using the Arnoldi iterations reduces the computational time, it still requires the inversion of the stiffness matrix $\mathbf{K}$, which is expensive. In addition, the formation of $\mathbf{C}$ destroys the sparse structure of the original matrices.

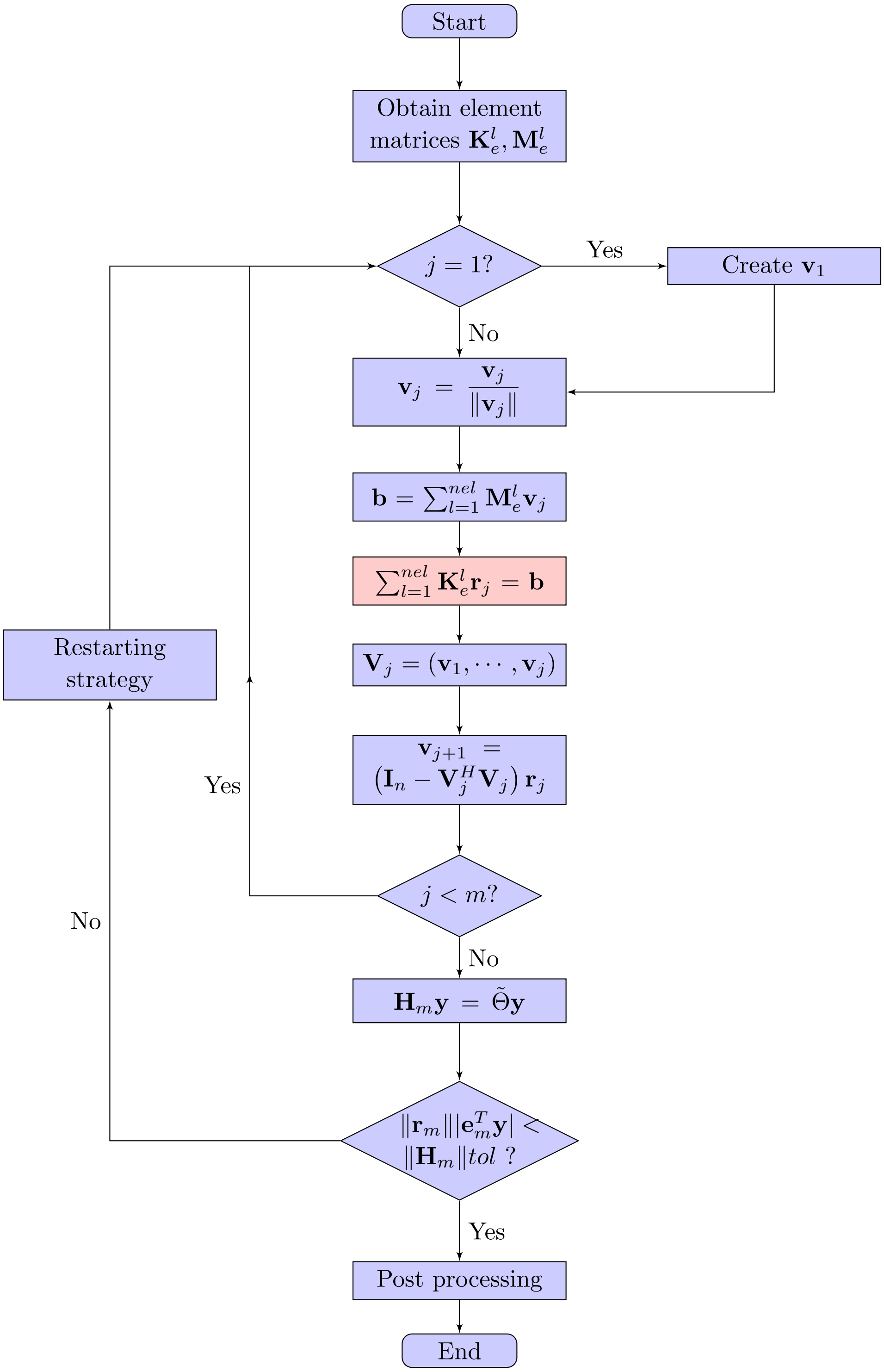
Algorithm 1 The AR-G solver

```
Input: Specify dimension of the subspace $m$ ($m \ll n$), number of required eigenvalues ($k$), error tolerance ($tol$), and the random starting vector ($\mathbf{v}_1$)
Output: Eigenpairs.
 1: Calculate $\mathbf{C} = \mathbf{K}^{-1}\mathbf{M}$
 2: Iteration:
 3: for $j = 1$ to $m$ do
 4:     $\mathbf{r}_j = \mathbf{C}\mathbf{v}_j, \mathbf{V}_j = (\mathbf{v}_1, \ldots, \mathbf{v}_j)$;
 5:     $\mathbf{r}_j = (\mathbf{I} - \mathbf{V}_j\mathbf{V}_j^H)\mathbf{r}_j$;
 6:     $\mathbf{v}_{j+1} = \mathbf{r}_j/\|\mathbf{r}_j\|$
 7: end for
 8: Solving a low-dimensional eigenproblem $\mathbf{H}_m\mathbf{y} = \tilde{\Theta}\mathbf{y}$ using the QZ algorithm; the approximate eigenpairs of (11).
 9: If $\|\mathbf{r}_j\||\mathbf{e}_m^T\mathbf{y}| < \|\mathbf{H}_m\|tol$, stop. Otherwise, go to step 3.
10: Restart: go to 3 with a new $\mathbf{v}_1$.
```

3.3. *The Arnoldi iterations with a frontal solver (AR-F)*

The AR-F approach is developed to avoid the explicit formation of the two sparse matrices **K** and **M** as in §3.2, by making use of the frontal solver [Irons, 1970; Hood, 1976]. The frontal approach is a very efficient method for solving the finite element global matrix equations. Building on a LU or Cholesky decomposition, it assembles the global matrix and eliminates the equations only on a subset of elements at a time. The subset is so-called the "front" and it is essentially the transitional region between the part of the system already finished and the part untouched. During the whole process, the fully sparse matrix is never assembled. Only the parts of the matrix are assembled as they enter the front. Processing the front involves dense matrix operations, which uses the CPU efficiently.

In essence, for the AR-F solver, instead of generating the large matrix **C**, we make use of the frontal solver by storing the element matrices together with a steering matrix which gives the location of the frontal element entries in the global matrices. The flow chart of the AR-F algorithm is shown in Fig. 2.



Start
Obtain element matrices $\mathbf{K}_e^l, \mathbf{M}_e^l$
$j = 1?$
Yes
Create $\mathbf{v}_1$
No
$\mathbf{v}_j = \frac{\mathbf{v}_j}{\|\mathbf{v}_j\|}$
$\mathbf{b} = \sum_{l=1}^{nel} \mathbf{M}_e^l \mathbf{v}_j$
$\sum_{l=1}^{nel} \mathbf{K}_e^l \mathbf{r}_j = \mathbf{b}$
Restarting strategy
$\mathbf{V}_j = (\mathbf{v}_1, \cdots, \mathbf{v}_j)$
$\mathbf{v}_{j+1} = \left(\mathbf{I}_n - \mathbf{V}_j^H \mathbf{V}_j\right) \mathbf{r}_j$
Yes
$j < m?$
No
No
$\mathbf{H}_m \mathbf{y} = \tilde{\Theta} \mathbf{y}$
$\|\mathbf{r}_m\| |\mathbf{e}_m^T \mathbf{y}| < \|\mathbf{H}_m\| tol$?
Yes
Post processing
End

Fig. 2. The flowchart of AR-F, where the the element matrices $\mathbf{K}_e^l$, $\mathbf{M}_e^l$ are used via the frontal method, and *nel* is the total number of the finite elements.

4. Numerical Results

4.1. *Choice of parameters for the eigenvalue problem*

We choose the dimensional parameters to be $\mu = 10^{-3}\ Pas^{-1}$, $\rho = 10^{3}\ kgm^{-3}$, $D = 10^{-2}\ m$, thus the non-dimensional parameters are $L_u = L = 5$, $Ld = 30$, $Re = 1 - 600$, $T = 0$, $\rho_m = 0$, $P_e - P_d$=1.95, $P_d = 0$, $c_\lambda = 1 - 2500$, and $c_\kappa/c_\lambda = h^2/12D^2 \approx 10^{-5}$ ($h/D = 0.01$). The mid-point of the elastic beam will be $x = 7.5$. The values are chosen to be in the range of parameters that have been used in previous studies [Cai and Luo, 2003; Luo *et al.*, 2008; Liu *et al.*, 2012].

4.2. *Choice of parameters for the Arnoldi solvers*

The Arnoldi algorithms require the selection of several parameters, namely, the number of eigenpairs required (`NEV`/2), the rank of the subspace (m=`NCV`), and the maximum number of iterations (`MAXITR`). The number of converged eigenpairs is denoted as `NO`/2, which is usually smaller or equal to `NEV`/2. In principle, the greater the rank of the subspace and number of iterations allowed, the more likely the eigensolution in question can be located in the Krylov subspace, but the longer the computation. To strike a balance between the computational time and the solution approximation, an "optimal" group of parameters of `NEV`, `NCV`, and `MAXITR`, in the ranges of 2 to 8 (`NEV`), 20 to 200 (`NCV`), and 50-300 (`MAXITR`), are carefully selected after extensive computational tests. The results of the tests are shown in Table 1.

4.3. *Computational verifications*

Since the eigensolvers require one to solve for steady solutions iteratively, these are performed using a Python script, which allows us to automatically search for the neutral solutions in a systematic way. In all the computations, the tolerance for the Arnoldi iteration is set to be 1×10^{-16}, which is the default value used in the ARPACK.

To validate our eigensolvers, we first test the results with the solutions from the eigs subroutine in Matlab for a much smaller matrix size. This has led to a good agreement. We then increase the grid points to test the grid independence. The corresponding dimensions of the finite element (FE) matrices are $n \times n$, for n=933, 2063, 2629, 3942, 5117, 6152, 7325, and 55329, respectively. All three solvers yield the same eigenpairs at the same physical parameters, for the same grid, except when N=55329, which is too big for either the QZ or the AR-G solver to cope. All computations were run on the Linux Workstations (2×Hexa-core HT Intel(R), Xeon(R), CPU E5650, 2.65 GHz) at the School of Mathematics and Statistics, the University of Glasgow.

Table 1: The optimal sets of parameters, and lapsed time(in seconds) of different meshes for the same input parameters (c_λ, Re or P_{ud})

Matrix	QZ	AR-G					AR-F				
n	time (s)	NEV	NCV	MAXIT	NO.	TIME(s)	NEV	NCV	MAXIT	NO.	TIME(s)
933	18	2	20	50	2	15	2	40	50	2	7
2063	261	2	20	100	2	151	2	20	100	2	23
		2	40	50	2	144	2	60	50	2	19
2629	577	2	60	50	2	336	2	60	50	2	28
		4	40	150	2	610	4	40	150	4	169
		6	40	150	6	726	6	40	150	6	164
		6	40	200	6	726	6	40	200	6	164
		6	40	250	4	392	6	60	250	6	164
3942	2554	2	40	50	2	1196	2	40	50	2	50
		2	60	50	2	1156	2	60	50	2	55
5117	5116	2	40	50	2	2687	2	40	50	2	78
		2	60	50	2	2670	2	60	50	2	70
6152	9158	2	40	50	2	4879	2	40	50	2	93
		2	40	100	2	4822	2	40	100	2	92
7325	16816	2	80	250	2	8053	2	40	50	2	115
		2	80	300	2	8052	2	40	100	2	116
55392	N/A	N/A					2	60	100	2	3426
							2	90	100	2	4584

4.4. *Comparison of the three eigensolvers*

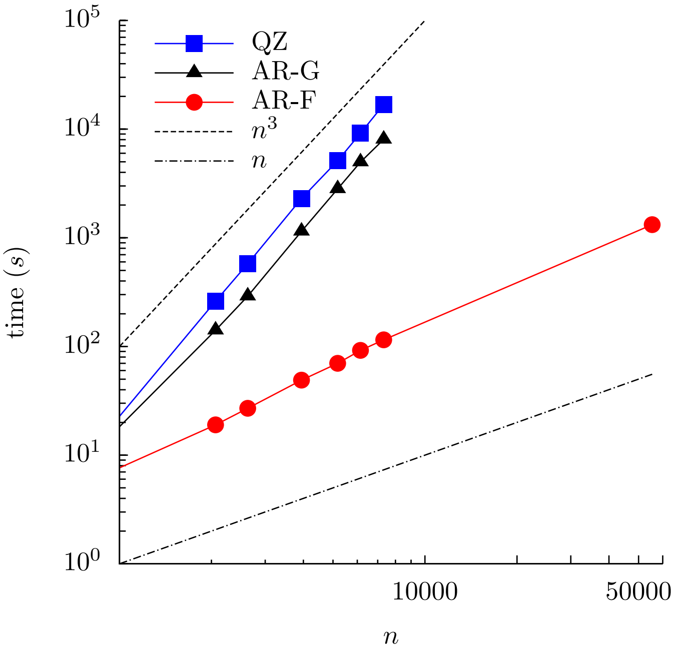


Fig. 3. Comparison of the computational time (seconds) against the matrix size indicator (n) using the QZ, AR-G & AR-F solvers. The AR-F method costs scales between $O(n)$ and $O(n^2)$, while the QZ and AR-G solvers scale to $O(n^3)$, although the AR-G solver is slightly more efficient than the QZ solver. Note the shortest times required by each solver, using combinations of the parameters as shown in Table 1, are used to plot this figure.

Figure 3 shows the log-plot of the CPU time versus n of the matrix size $n \times n$ for these three algorithms. For comparison purposes, n^2 and n^3 are also shown as dotted and dash-dot lines. We can see that the QZ algorithm converges in the order of $O(n^3)$. Whereas AR-G, though in general requires less time, converges with the rate only slightly less than the QZ solver. The AR-F algorithm, on the other hand, converges approximately in the order between $O(n)$ and $O(n^2)$, hence is by far the most efficient one. It is also clear that the AR-F solver can solve much large eigenvalue problems. However, we must mention that the QZ solver obtains all the eigensolutions by the end of the computations, while the Arnoldi solvers can only locate the first or first couple of eigenpairs.

4.5. *Neutral stability of the system*

We now apply the AR-F algorithm in the stability analysis of collapsible channel flows. The eigenmodes are classified according to the number of wavelengths in the oscillation of the elastic section, mode-i means there are $i/2$ full wavelengths.

For the flow driven-system, Luo *et al.* [2008] revealed a cascade structure, in particular, they obtained a mode-2 neutral stability curve in the c_λ-Re space using the QZ algorithm for a relatively coarse grid of n=6152.

For the pressure-driven system, Liu *et al.* [2012] identified a mode-1 neutral curve. Mathematically, the pressure-driven system is a harder to be solved numerically as it presents a very thin boundary layer upstream of the collapsed section, it requires a much-refined mesh to resolve the flow there. Liu *et al.* [2012] were unable to perform the eigenvalue analysis for all the dimensions required using the QZ solver, so they resorted to testing the stability by laboriously running the unsteady FE solver combined with a bisection search.

In the following, we use the AR-F solver for the stability analysis of both systems.

4.5.1. *Flow-driven system*

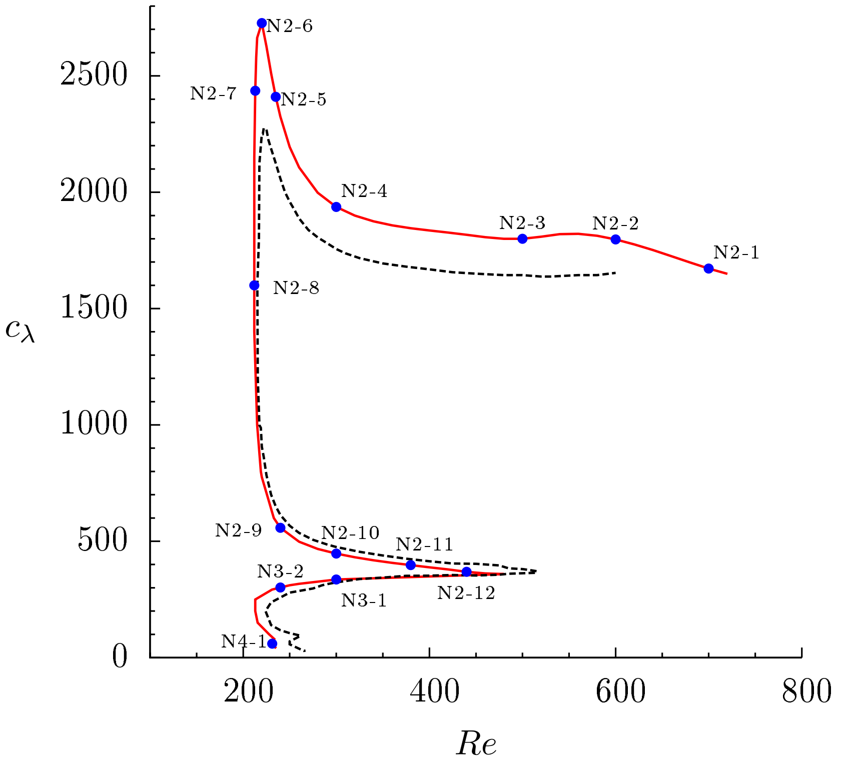


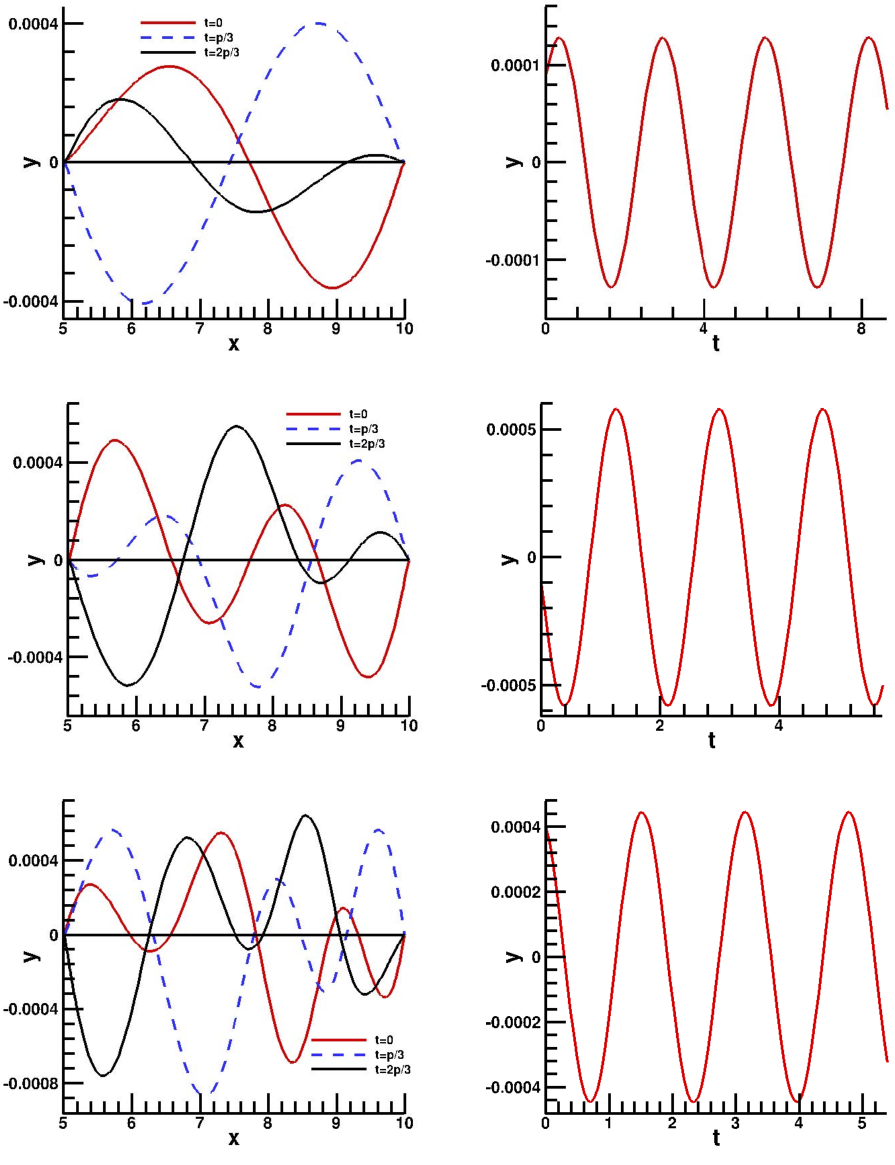
Fig. 4. The mode-2 neutral curve in the Re - c_λ space obtained using the AR-F solver for a very dense mesh where n=55392 (red solid line), compared to the one obtained by Luo *et al.* [2008] using a coarse mesh n = 6152 (dotted line). The parameters are listed in Table 2.

For the flow-driven system, we revisit the eigensolutions at the neutral stability points using the AR-F solver for a very dense mesh ($n = 55392$). The results are plotted in the Re - c_λ space (where c_λ indicates how stiff the elastic beam is) and are compared with those obtained by Luo *et al.* [2008] in Fig. 4 where they used $n = 6152$. The neutral curves consists of two branches, $N2$ branch represents the solution of a mode-2 neutral curve, and the $N3$ and $N4$ branches are the mode-3 and mode-4 neutral solutions, respectively. Details of the neutral behaviour of the system were discussed in [Luo *et al.*, 2008].

Figure 4 suggests that the cascade structure found by Luo *et al.* [2008] using a much coarser mesh is reasonable but with some discrepancies; the upper curve is shifted upwards, and the lower branch moves slightly rightwards. However, the qualitative behaviour remains the same. Notice that although many more points along the curve are computed, only a selection of the solutions (marked by blue dots and named as N2-1, N2-2, etc.) are listed in Table 2. Three eigen-solutions, one at each branch (i.e. N2-1, N3-2, and N4-1) - computed using the AR-F solver, are also plotted in Fig. 5. We have tested that, when converged, all the three solvers give the same eigensolutions.

Table 2 Neutral points of the flow-driven system using the AR-F solver for n=55392.

Point	Re	c_λ	ω_R	ω_I	$Mode$
N2-1	700	1672	-3.30023×10^{-5}	2.40366	Mode-2
N2-2	600	1797.5	1.14649×10^{-6}	2.42381	Mode-2
N2-3	500	1800	6.13079×10^{-6}	2.35544	Mode-2
N2-4	300	1937	-1.76015×10^{-6}	2.03575	Mode-2
N2-5	235	2410	-1.93529×10^{-7}	1.72480	Mode-2
N2-6	220	2727	-1.22905×10^{-6}	1.57726	Mode-2
N2-7	213	2436	-6.90523×10^{-7}	1.43017	Mode-2
N2-8	212	1600	-2.39754×10^{-4}	1.30612	Mode-2
N2-9	240	580	2.62525×10^{-6}	1.24025	Mode-2
N2-10	300	447	-3.63519×10^{-5}	1.27266	Mode-2
N2-11	380	397.5	3.62141×10^{-6}	1.24308	Mode-2
N2-12	440	369	-9.00410×10^{-6}	1.20026	Mode-2
N3-1	300	335.61	5.67991×10^{-6}	4.07781	Mode-3
N3-2	250	311	-7.02530×10^{-5}	3.61682	Mode-3
N4-1	231.15	60	-9.69742×10^{-6}	3.85157	Mode-4



(a) y(x,t)

(b) $y(x, t)$ at $x = 7.5$

Fig. 5. Neutral solutions for the flow-driven system. (a) The elastic beam shape $y(x, t)$ for $x = 5-10$ at $t = 0$ (red solid), $\pi/3$ (blue dotted), and $2\pi/3$ (black solid), and (b) history of the middle point of the beam $y(x, t)$ at $x = 7.5$. These are plotted for points N2-1 (top), N3-2 (middle), and N4-1 (bottom) (see Fig. 4 and Table 2). Notice the absolute values of these solutions are arbitrary.

4.5.2. *Pressure-driven system*

To study the mode-1 neutral curve in the pressure-driven system, we use the same grids as used by Liu *et al.* [2012] ($N = 55416$). The neutral curve in the pressure-driven system can be more conveniently shown in the $c_\lambda - P_{ud}$ space, since P_{ud} is now the control parameter, and the flow rate (or U_0), on which the Reynolds number is defined, becomes an output. The AR-F solver can reproduce all the neutral points obtained previously [Liu *et al.*, 2012], as shown in Table 3 and Fig. 6. Again, many more points along the curve are computed, but only a selection of the solutions are listed in Table 3. In particular, we have obtained the neutral solutions between N1-2 and N1-1, which were estimated by Liu *et al.* [2012] from unsteady simulations since the size of the matrix is too big for the QZ solver. Using an unsteady solver to identify a neutral point is a very lengthy process; it takes a week to pinpoint a neutral point, but requires only 30 minutes using the AR-F solver.
Table 3 shows that the AR-F solver reproduced the solutions obtained by Liu *et al.* [2012], with a percentage difference less than 3%. This is because both solvers used the same (and very fine) mesh. However, with the AR-F solver, we can also computer new points above the point N1-1 (when $c_\lambda \geq 2.0 \times 10^7$) – the upper limit that Liu *et al.* [2012] could reach with the unsteady solver. In addition, the AR-F solver gives the eigen-frequencies of the neutral points, which was difficult to estimate from the unsteady simulations. The corresponding Reynolds numbers are also listed in Table 3, which are in the similar range to the (most of) neutral points obtained from the flow-driven system. Three selected eigensolutions using the AR-F solver are shown in Fig. 7.

Table 3: The neutral points of the pressure-driven system using the AR-F solver. The top five points were compared against the computed solutions by Liu *et al.* [2012] (in brackets, with percentage difference) for the matrix size ($n = 55416 \times 55416$). The bottom (unnamed) points are the new points obtained in this study.

Points	P_{ud}	c_λ	Re	ω_R	ω_I
N1-5	1.0	308.40	115.61	9.003×10^{-6}	0.519
(Liu *et al.* [2012])	(1.0, 0%)	(307.85, 0.2%)	(115.60, 0%)	(N/A)	(0.519, 0.02%)
N1-4	0.703	927	107.89	1.607×10^{-5}	0.726
(Liu *et al.* [2012])	(0.7,0.45%)	(927, 0%)	(107.40, 0.5%)	(N/A)	(0.726, 0.0%)
N1-3	0.687	2000	113.83	3.317×10^{-5}	0.847
(Liu *et al.* [2012])	(0.68, 1%)	(2000, 0%)	(112.66, 1%)	(N/A)	(0.848, 0.18%)
(Liu *et al.* [2012])	0.708	5500	122.16	-4.591×10^{-6}	1.014
N1-2	(0.7, 1.2%)	(5500,0%)	(121.90, 0.21%)	(N/A)	(1.016, 0.16%)
N1-1	1.21	2.0×10^7	226.62	-7.085×10^{-6}	3.619
(Liu *et al.* [2012])	(1.2, 0.83%)	(2.0×10^7, 0%)	(219.00, 3.4%)	(N/A)	(3.623, 0.11%)
	1.263	3.0×10^7	237.02	2.177×10^{-5}	3.967
	1.290	3.5×10^7	241.64	-3.590×10^{-6}	4.144
	1.315	4.0×10^7	253.02	2.735×10^{-6}	4.325
new	1.413	6.0×10^7	264.91	5.713×10^{-6}	5.037
points	1.499	8.0×10^7	281.09	-6.634×10^{-6}	5.713
above N1-1	1.608	1.1×10^8	301.79	2.088×10^{-6}	6.637
	1.728	1.5×10^8	322.68	-4.890×10^{-7}	7.724
	1.849	2.0×10^8	347.15	-4.533×10^{-6}	8.913

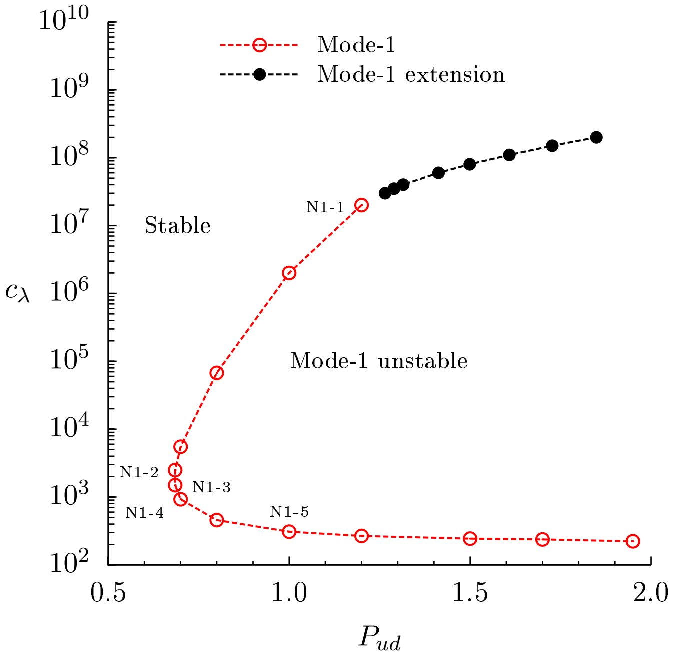
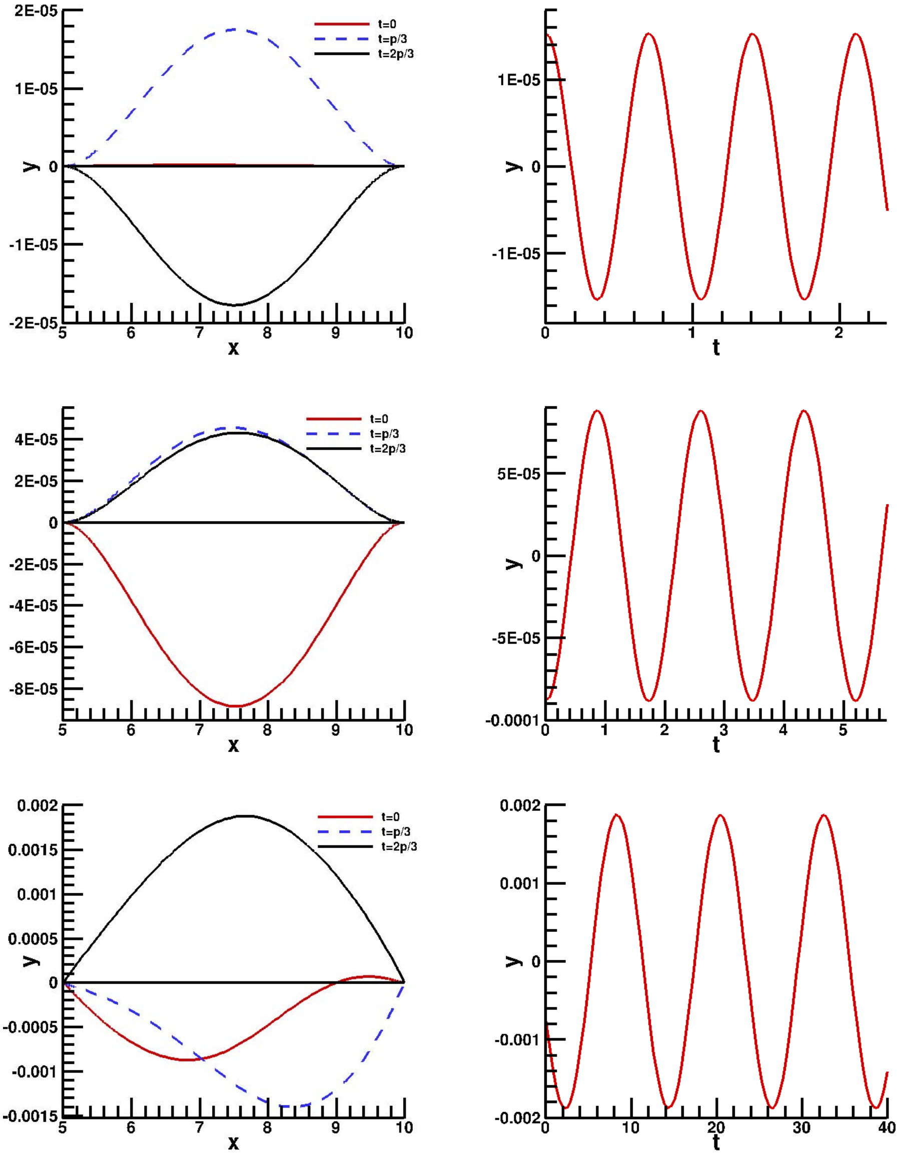


Fig. 6. The mode-1 neutral curve in the c_λ-P_{ud} space of the pressure-driven system obtained using the AR-F solver. The neutral points below N1-1 agree with those obtained by Liu *et al.* [2012], see details in Table 3. The points above N1-1 (in black) are the new solutions.



(a) $y(x,t)$

(b) $y(x,t)$ at $x = 7.5$

Fig. 7. Neutral solutions for the pressure-driven system. (a) The elastic beam shape $y(x,t)$ for $x = 5 - 10$ at $t = 0$ (red solid), $\pi/3$ (blue dotted), and $2\pi/3$ (black solid), and (b) history of the middle point of the beam $y(x,t)$ at $x = 7.5$. These are plotted for the new point ($P_{ud} = 1.849$, $c_\lambda = 2.0 \times 10^8$)(top), N1-1 (middle), and N1-5 (bottom) (see Fig. 6 and Table 3). Notice the absolute values of these solutions are arbitrary.

5. Discussion

Stability analysis of collapsible channel flows has been a great challenge numerically due to the large matrix size and the asymmetric structure of the eigenvalue problems. Previous studies used the QZ method to solve for all the eigensolutions, which requires extremely large memory and CPU time and is impractical in many applications. The pressure-driven system of the flows in a collapsible channel, in particular, can generate very thin boundary layers upstream the elastic section. If the flow details of these boundary layers are not resolved, the eigensolutions computed are either inaccurate or the solvers fail to converge. Therefore, the advantages of the AR-F solver is particularly useful for systems like this.

In this paper, we are able to produce neutral points faster and identify new solutions which we cannot obtain using the traditional eigensolvers. The flow-driven system of the flows in a collapsible channel, on the other hand, does not require such a refined mesh, since it does not have a very thin boundary layers upstream owing to the parabolic entry flow. The neutral curve identified by Luo *et al.* [2008] using a relatively coarse mesh seems to provide a reasonable approximation with small discrepancies compared to the new results based on a much finer mesh. We must mention that although we have presented some new results, we have not devoted our effort to identifying new neutral curves for a different set of parameters, in particular, we only studied the cases when $P_e - P_d = 1.95$, $T = 0$, and $\rho_m = 0$. The neutral curves will change if different parameter regions are considered.

The AR-F solver converges much faster than the QZ, or the AR-G methods since the latter still requires solving the inverse of the global FE matrix. The disadvantage of the Arnoldi-type solvers, however, is that we have to select optimal Krylov subspaces in order to locate the first few eigenvalues of the system. Since the initial vector is generated randomly, it can only guarantee that the most unstable eigenpair (the largest eigenvalue) is secured. With the current approach, it is difficult to find the second or third unstable eigenpairs consistently. Even if we introduce shift and orthogonalization to systematically filter out the lower order of the complex eigenvectors, which is non-trivial, we cannot ensure that the Krylov subspace always contains the next unstable modes. In this sense, the QZ solver is still the most reliable one in terms of determining all the required eigenpairs in the right order and hence is recommended for small sized eigenvalue problems.

We remark that although all the computations are performed in serial, the efficiency of the new approach could equally benefit from parallel simulations. In all the computations, the tolerance for the Arnoldi iteration is set to be 1×10^{-16}, which is the default value used in ARPACK. While this may be unnecessary for the coarse meshes used, we kept this unchanged in order to make fair comparisons of the computational times used by all the solvers. Finally, we must acknowledge that linear stability analysis is not applicable to certain fluid dynamics problems, and different approaches may need to be considered [Trefethen *et al.*, 1993].

6. Conclusion

In this paper, we have developed a combined Arnoldi-frontal approach for solving large and complex general eigenvalue problems from the finite element simulations. We show that this approach not only overcomes the memory limitation for large sparse matrices but also significantly reduces the computational time. Using our new solver the rate of the CPU time for a matrix size $n \times n$ is reduced from $O(n^3)$, which is required by the QZ or other traditional Arnoldi solvers, to almost $O(n)$. The memory saving is also huge; instead of storing the full global matrix, only the front involving a subset of elements is in memory. When the new solver is applied to the stability analysis of flows in a collapsible channel, both for the flow-driven where the inlet flow is specified, or the pressure-driven system, where inlet the pressure is specified, it is able to locate neutral stability points previously unattainable on a single workstation. A greater advantage of such an approach will be its application in the three dimensional stability analysis, which is currently underway.

Acknowledgements

This work is supported by the UK Engineering and Physical Sciences Research Council grants (No. EP/I029990/1 and EP/N014642/1).

References

Andy, J. and Nair, P. [2005] *Computational Approaches for Aerospace Design The Pursuit of Excellence* (John Wiley and Sons. Ltd.).

Arnoldi, W. E. [1951] The princilple of minimized iteration in the solution of the matrix eigenvalue problem, *Quart. J. Applied Mathematrics* **9**, 17–29.

Auckenthaler, T., Blum, V., Bungartz, H., Huckle, T., Johanni, R., Krmer, L., Lang, B., Lederer, H. and Willems, P. [2011] Parallel solution of partial symmetric eigenvalue problems from electronic structure calculations, *Parallel Computing* **37**(12), 783.

Bai, Z., Demmel, J., Dongarra, J., Ruhe, A. and Van Der Vorst, H. [1987] *Templates for the solution of algebraic eigenvalue problems: a practical guide* (Society for Industrial and Applied Mathematics).

Bathe, K. [1971] *Solution methods for large generalized eigenvalue problems in structural engineering* (National Technical Information Service, US Department of Commerce).

Bathe, K. and Wilson, E. [1973] Solution methods for eigenvalue problems in structural mechanics, *International Journal for Numerical Methods in Engineering* **6**, 213–226.

Bathe, K.-J. [2013] The subspace iteration method–revisited, *Computers & Structures* **126**, 177–183.

Bertram, C. and Ribreau, C. [1989] Cross-sectional area measurement in collapsed tubes using the transformer principle, *Medical and Biological Engineering and Computing* **27**(4), 357–364.

Bertram, C. D. and Castles, R. J. [1999] Flow limitation in uniform thick-walled collapsible tubes, *Journal of Fluids and Structures* **13**(3), 399–418.

Brook, B. S. and Pedley, T. J. [2002] A model for time-dependent flow in (giraffe jugular) veins: uniform tube properties, *Journal of biomechanics* **35**(1), 95–107.

Cai, Z. and Luo, X. [2003] A fluid-beam model for flow in a collapsible channel, *Journal of Fluid and Structures* **17**, 125–146.

Cancelli, C. and Pedley, T. [1985] A separated-flow model for collapsible-tube oscillations, *Journal of Fluid Mechanics* **157**(AUG), 375–404.

Cisonni, J., Van Hirtum, A., Luo, X. and Pelorson, X. [2010] Experimental validation of quasione-dimensional and two-dimensional steady glottal flow models, *Med Biol Eng Comput* **48**, 903–910.

Cliffe, K., Garratt, T. and Spence, A. [1994] Eigenvalues of block matrices arising from problems in fluid mechanics, *Siam J. Matrix Anal. Appl.* **15**, 1310–1318.

Cullum, J. and Willoughby, R. [1986] *Large Scale Eigenvalue Problems* (Elsevier Science.).

Elad, D., Kamm, R. and Shapiro, A. [1987] Choking phenomena in a lung-like model, *Journal of Biomechanical Engineering-Transactions of the ASME* **109**(1), 1–9.

Freund, R., Guthnecht, M. and Nachtigal, N. [1993] An implementation of the look-ahead lanczos algorithm for non-Hermitan matrices, *SIAM J. Sci. Comp.* **14**, 137–158.

Gavriely, N., Shee, T. R., Cugell, D. W. and Grotberg, J. B. [1989] Flutter in flow-limited collapsible tubes: a mechanism for generation of wheezes, *Journal of Applied Physiology* **66**(5), 2251.

Guiot, C., Pianta, P., Cancelli, C. and Pedley, T. [1990] Prediction of coronary blood flow with a numerical model based on collapsible tube dynamics, *American Journal of Physiology-Heart and Circulatory Physiology* **258**(5), H1606–H1614.

Hood, P. [1976] Frontal solution program for unsymmetric matrices, *International Journal For Numerical Methods In Engineering* **10**, 379–399.

Irons, B. M. [1970] A frontal solution scheme for finite element analysis. *nt. J. Numer. Methods Eng.* **2**, 5–32.

Jensen, O. [1990] Instabilities of flow in a collapsed tube, *Journal of Fluid Mechanics* **220**, 623–659.

Kamm, R. and Shapiro, A. [1979] Unsteady-flow in a collapsible tube subjected to external-pressure or body forces, *Journal of Fluid Mechanics* **95**(NOV), 1–78.

Lehoucq, R., Meerbergen, K. *et al.* [1998] Using generalized cayley transformations within an inexact rational krylov sequence method, *SIAM Journal on matrix analysis and applications* **20**, 131–148.

Lehoucq, R. and Salinger, A. [2001] Large-scale eigenvalue calculations for stability analysis of steady flows on massively parallel computers, *International Journal for Numerical Methods in Fluids* **36**(3), 309–327.

Lehoucq, R. B., Sorensen, D. C. and Yang, C. [1997] *ARPACK user's guide: solution of large scale eigenvalue problems with implicitly restarted Arnoldi methods.*

Liu, H., Luo, X. and Cai, Z. [2012] Stability and energy budget of pressure-driven collapsible channel flows, *Journal of Fluid Mechanics* **705**, 348–370.

Luo, X. [2015] Modelling flows in collapsible tubes, *Biomechanics, Doblar'e M and Merodio, J (eds), Encyclopedia of Life Support Systems (EOLSS)* , 213–240.

Luo, X., Cai, Z., Li, W. and Pedley, T. [2008] The cascade structure of linear instability in collapsible channel flows, *Journal of Fluid Mechanics* **600**, 45–76.

Luo, X. and Pedley, T. [1996] A numerical simulation of unsteady flow in a 2-d collapsible channel, *J. Fluid Mech* **314**, 191–225.

Luo, X. and Pedley, T. [1998] The effects of wall inertia on flow in a two-dimensional collapsible channel, *Journal of Fluid Mechanics* **363**, 253–280.

Luo, X. and Pedley, T. [2000] Multiple solutions and flow limitation in collapsible channel flows, *Journal of Fluid Mechanics* **420**, 301–324.

Meerbergen, K. and Roose, D. [1996] Matrix transformations for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems, *IMA Journal of Nu-*

merical Analysis **16**(3), 297–346.

Meerbergen, K. and Roose, D. [1997] The restarted Arnoldi method applied to iterative linear system solvers for the computation of rightmost eigenvalues, *SIAM J. Matrix. Anal. Appl.* **18**, 1–20.

Misrikhanov, M. and Ryabchenko, V. [2006] The quadratic eigenvalue problem in elctric power systems, *Journal of Automation and Remote Control.* **67**, 24–47.

Moler, C. B. and Stewart, G. W. [1973] An algorithm for generalized matrix eigenvalue problems, *SIAM Journal on Numerical Analysis* **10**(2), 241–256.

Morgan, R. [2000] Implicitly restarted GMRES and Arnoldi methods for nonsymmetric systems of equations, *SIAM J. Matrix Anal. Appl.* **21**, 1112–1135.

Parlett, B., Taylor, D. and Liu, Z. [1985] A look-ahead Lanczos algorithm for unsymmetric matrices, *Math. Comp.* **44**, 105–124.

Parlett, R. and cott, D. [1979] The Lanczos algorithm with selective orthogonalization, *Math. Comp.* **33**, 217–238.

Rapcsak, T. [2004] Some optimization problems in multivariate statistic, *Journal of Global Optimization* **28**, 217–228.

Rast, M. [1994] Simultaneous solution of the navier-stokes and elastic membrane equations by a finite element method, *International journal for numerical methods in fluids* **19**(12), 1115–1135.

Roger, G., Grimes, J., Lewis and Horst, D. [1986] Eigenvalue problems and algorithms in structural engineering, *North-Holland Mathematics Studies* **127**, 81–93.

Saad, Y. [1996] *Numerical Method for Large Eigenvalue Problems* (Manchester University Press Series in Algorithms and Architectures for Advanced Scientific Computing).

Scott, T., Moore, R., Monagan, M., Fee, G. and Vrscay, E. [1990] Perturbative solutions of quantum mechanical problems by symbolic computation, *Journal of Computational Physics* **87**(2), 366–395.

Shapiro, A. [1977] Steady flow in collapsible tubes, *Journal of Biomechanical engineering-Transactions of the ASME* **99**, 126.

Stewart, H. M., PS, Waters, S. and Jensen, O. [2010a] Sloshing and slamming oscillations in collapsible channel flow, *Journal of Fluid Mechanics* **662**, 288–319.

Stewart, P. S., Heil, M., Waters, S. L. and Jensen, O. E. [2010b] Sloshing and slamming oscillations in a collapsible channel flow, *Journal of Fluid Mechanics* **662**, 288–319.

Trefethen, L., Trefethen, A., Reddy, S., Driscoll, T. and Others [1993] Hydrodynamic stability without eigenvalues, *Science* **261**(5121), 578–584.

Wu, K. and Simon, H. [2000] Thick-restart Lanczos method for large symmetric eigenvalue problems, *SIAM J. Matrix Anal. Appl.* **22**, 602–616.

Xu, F., Billingham, J. and Jensen, O. E. [2013] Divergence-driven oscillations in a flexible-channel flow with fixed upstream flux, *Journal of Fluid Mechanics* **723**, 706–733.

Xu, F., Billingham, J. and Jensen, O. E. [2014] Resonance-driven oscillations in a flexible-channel flow with fixed upstream flux and a long downstream rigid segment, *Journal of Fluid Mechanics* **746**, 368–404.