Uncertainties Associated With Finite Element Modelling Of Coriolis Mass Flow Meters

R. Cheesewright[1] and Simon Shaw[1]
June 27, 2006

ABSTRACT

The problem of computing accurate Coriolis distortion modes in mass flow meters is discussed. This is illustrated by several numerical results, and it is tentatively suggested that the problem is due mainly to computer rounding error rather than any fundamental weakness in the finite element method or the eigensolvers. An empirically evaluated successful method, using a shifted inverse iteration, for computing high quality results is given.

1 Introduction

The use of FE models to predict the performance of Coriolis mass flow meters has become established practice since the pioneering work of Stack, Garnett and Pawlas, [1]. This approach has been used both by meter manufacturers in the design of meters and by researchers examining the response of meters to a range of external stimuli. Much of the work by manufacturers has not been published for reasons of commercial confidentiality but it is believed that it follows the approach described in [1] and utilises one of the commercially available FE packages such as MSC/NASTRAN, ANSYS, etc. No comprehensive assessment of the uncertainties associated with the modelling has been published and in [1] the agreement between predicted and experimentally measured characteristics, for a range of different shapes of meter tube is described as being at the 5% level.

The most comprehensive description of a FE model used by researchers is that by Belhadj, Cheesewright and Clark [2]. This follows the approach used in [1], representing the fluid flowing through the meter tube as a moving string, exerting only transverse forces on the tube as it is forced to move with the tube. The tube is discretised as a series of straight elements which are modelled as Timoshenko beam elements, using 4 degrees of freedom per element and the solution is obtained using the ANSYS package. A limited assessment of uncertainties is presented via a comparison with the theoretical results reported by Cheesewright and Clark [3] for the case of a straight tube meter, with the Timoshenko beam treatment replaced by an Euler-Bernoulli treatment. The main element of the comparison was the meter sensitivity which is defined as the phase difference between the signals from displacement detectors mounted on the tube, per mass flow rate through the tube. For the Euler-Bernoulli model the agreement between the theoretically and FE predicted meter sensitivity is quoted as being approximately 1% and, also, the difference between the predicted (by FE) meter sensitivities for the Euler-Bernoulli model and for the Timoshenko model is quoted as approximately 5%.

There have been a number of other published reports of the use of FE models of Coriolis mass flow meters including [4], [5], [6], [7] and [8] which derive from the work in [2] and none of them add anything to the knowledge of the uncertainties associated with the FE model. A different FE model is reported by Kutin and Bajsic [9] in which the flow tube is treated as a shell, but again there is no significant assessment of uncertainties.

In the course of work on the use of the Belhadj FE model, [2], to predict the dynamic response characteristics of meters it became desirable to have accurate values of displacements and velocities (in steady flow) for use as starting conditions for transient simulations. It was discovered that although the FE model

[1]School of Engineering and Design (Cheesewright) and Brunel Institute of Computational Mathematics (Shaw), Brunel University, Uxbridge UB8 3PH, England. ({robert.cheesewright,simon.shaw}@brunel.ac.uk).

gave reasonable predictions of the meter sensitivity, the predictions of the distribution of fluid-induced distortion along the tube did not have 180° rotational symmetry about the beam's midpoint. Since the simple straight tube meter is symmetric about the midpoint, this distortion cannot be physically realistic.

In assessing the implications of this observation it must be appreciated that in the general use of FE models in the design of meters an uncertainty of the order of 5% in predictions of frequencies and meter sensitivities would be acceptable. However, when it is desired to extend the use of the FE models to such things as the design of micro-machined meters, where it may be necessary to use distributed sensors rather than point sensors, and to the prediction of the manufacturing tolerances required for the positioning of sensors, the distribution of the fluid induced distortions along the tube must be more accurate. A detailed investigation of the uncertainties associated with the FE modelling was therefore undertaken and some of the results form the basis of this paper.

It was decided to use the modelling of a simple straight tube meter as the basis for the initial stages of this investigation although it was always appreciated that such a meter constitutes a special case, in the sense that for such a meter the only aspect ratio (length/tube diameter) which is relevant is the overall one for the whole meter. For more common (more complex) curved tube meters modelled as a series of straight elements, which constitute the majority of applications of FE modelling, the relevant aspect ratio is that for one element. It is usually stated that the aspect ratio should be greater than 6:1 for applicability of the Euler-Bernoulli equations and greater than 1:1 for the Timoshenko equations. A consequence of this is that while a straight tube meter can be represented by any number of elements, the number of elements used to represent a curved tube is limited by the element aspect ratio and the use of the Timoshenko model, rather than the simpler Euler-Bernoulli model, is usually justified because it allows more elements.

The layout of this article is as follows. In the next section we give an approximate analytical solution to the Euler-Bernoulli model. This is important in that we have no other 'benchmark' against which to judge our numerical computations. Then, in Section 3 we outline the finite element discretisation, derive the eigenvalue problem and propose four techniques for its solution. The numerical results are then given in Section 4, and we finish with a discussion of these results in Section 5 and a brief summary of our conclusions in Section 6.

The results that are of most interest are the least-in-modulus eigenvalue, and the real and imaginary parts of the corresponding eigenvector. The imaginary part is the Coriolis distortion mode and is the one that seems difficult to compute accurately.

2 An approximate analytical solution

Before starting a detailed investigation it was decided to seek an approximate analytical solution for the shape of the distortion modes, in order to confirm the implications of symmetry. For a simple straight tube meter the Euler-Bernoulli model can be used and an equation describing the transverse motion of the tube has been given by a number of authors. We will use that published by Cheesewright and Clark [3], which, for steady flow, has the form

$$(m_p + m_f)\frac{\partial^2 u}{\partial t^2} + EI\frac{\partial^4 u}{\partial x^4} + m_f\left[2V\frac{\partial^2 u}{\partial x \partial t} + V^2\frac{\partial^2 u}{\partial x^2}\right] = 0, \quad (1)$$

where $u = u(x,t)$ is the transverse displacement, V is the velocity of the fluid through the meter, m_f and m_p are respectively the mass of the fluid and the mass of the pipe, per length and E and I are respectively the Young's Modulus and the second moment of area of the pipe cross-section. For a meter of length L the boundary conditions with respect to x are,

$$u(0,t) = u(L,t) = 0, \quad (2)$$

$$\frac{\partial u}{\partial x}(0,t) = \frac{\partial u}{\partial x}(L,t) = 0. \quad (3)$$

Equation (1) describes a complex eigenvalue problem and it should be further noted that it is a 'gyroscopic conservative system' so that the eigen-values will be wholly real. It is known from [3] that, in the absence of effects due to starting conditions, the flow induced distortions will correspond to the same eigenvalue as the driven motion of the meter. Thus we can express the solution to Equation (1), for a given eigen-pair (λ, W_λ), as

$$\begin{aligned} u(x,t) &= \mathrm{Re}\, W_\lambda(x)\cos(\lambda t) \\ &\quad + \mathrm{Im}\, W_\lambda(x)\sin(\lambda t), \end{aligned} \tag{4}$$

where $\mathrm{Re}\, W_\lambda$ is the real part of W_λ and $\mathrm{Im}\, W_\lambda$ is the imaginary part.

If we substitute Equation (4) into Equation (1) it is possible to derive a pair of coupled fourth order ordinary differential equations for $\mathrm{Re}\, W_\lambda$ and $\mathrm{Im}\, W_\lambda$. However, we know that these two quantities are more than two orders of magnitude different and so rather than attempting to find $\mathrm{Re}\, W_\lambda$ and $\mathrm{Im}\, W_\lambda$, it is more convenient to utilise the fact that we know from standard textbooks on beam vibrations that $\mathring{W}_\lambda$ (corresponding the no-flow case, $V = 0$), is wholly real and is given by,

$$\begin{aligned} \mathring{W}_\lambda = C_0\Big(&\sinh(\beta_n x) - \sin(\beta_n x) \\ &+ \alpha_n\big(\cosh(\beta_n x) - \cos(\beta_n x)\big)\Big), \end{aligned} \tag{5}$$

where C_0 is a constant which is proportional to the amplitude of the driven transverse motion of the meter, $\beta_n L$ is the n-th root of $\cos(\beta_n L)\cosh(\beta_n L) = 1$ and,

$$\alpha_n = \frac{\sinh(\beta_n L) - \sin(\beta_n L)}{\cos(\beta_n L) - \cosh(\beta_n L)}.$$

The natural frequency, ω_n, of the nth mode of vibration of the tube for zero flow is given by $\omega_n = (\beta_n L)^2[EI(m_p + m_f)/L^4]^{1/2}$. The difference between this frequency and the frequency of the corresponding mode with flow is less than one part in 10^4 (see [3]).

Writing $\mathrm{Re}\, W_\lambda = \mathring{W}_\lambda + \mathrm{Re}\, W_\lambda$ in (4) and then substituting into (1) results in an equation in which each term contains either $\sin(\omega_n t)$, $\sin(\lambda_n t)$ or $\cos(\lambda_n t)$. Neglecting the difference between ω_n and λ_n this equation can only be true for all t if the respective sums of the $\sin(\lambda_n t)$ terms and of the $\cos(\lambda_n t)$ terms are both zero. If, for simplicity, we write $\mathrm{Re}\, W_\lambda = Wr_\lambda$ and $\mathrm{Im}\, W_\lambda = Wi_\lambda$ then, after some simplification, we obtain,

$$\begin{aligned} &Wr_\lambda - \frac{EI}{\lambda_n^2(m_p+m_f)}\frac{d^4 Wr_\lambda}{dx^4} \\ &\quad + \frac{2m_f V}{(m_p+m_f)\lambda_n}\frac{dWi_\lambda}{dx} \\ &\quad - \frac{m_f V^2}{\lambda_n^2(m_p+m_f)}\times \\ &\quad \left(\frac{d^2 Wr_\lambda}{dx^2} + \frac{d^2\mathring{W}_\lambda}{dx^2}\right) = 0 \end{aligned} \tag{6}$$

$$\begin{aligned} &Wi_\lambda - \frac{EI}{\lambda_n^2(m_p+m_f)}\frac{d^4 Wi_\lambda}{dx^4} \\ &+ \frac{2m_f V}{(m_p+m_f)\lambda_n}\left(\frac{d\mathring{W}_\lambda}{dx} + \frac{dWr_\lambda}{dx}\right) \\ &\quad - \frac{m_f V^2}{\lambda_n^2(m_p+m_f)}\frac{d^2 Wi_\lambda}{dx^2} = 0. \end{aligned} \tag{7}$$

Equations (5), (6) and (7) subject to boundary conditions,

$$\begin{aligned} &Wr_\lambda(0) = Wr_\lambda(L) \\ &\quad = \frac{dWr_\lambda}{dx}(0) = \frac{dWr_\lambda}{dx}(L) = 0, \end{aligned}$$

and

$$\begin{aligned} &Wi_\lambda(0) = Wi_\lambda(L) \\ &\quad = \frac{dWi_\lambda}{dx}(0) = \frac{dWi_\lambda}{dx}(L) = 0 \end{aligned}$$

define Wr_λ and Wi_λ and it should be noted that Wr_λ and Wi_λ are of closely similar orders of magnitude.

No analytical solution to Equations (6) and (7) could be found but when numerical values of the meter properties were inserted, a numerical solution was obtained. For a meter driven in the fundamental mode, with an amplitude of $\sim 1.6\,\mathrm{mm}$, Figure 1 shows the distribution of Wr and Figure 2 shows the distribution of Wi. These distributions are of the shapes that considerations of meter symmetry would suggest.

There are a number of different possible explanations for the physically unrealistic shape of the fluid distortion mode discussed

earlier. Firstly, a four degree of freedom finite element discretisation of a Timoshenko beam corresponds to using continuous piecewise linear shape functions for u and the rotation, θ. This is well known to be flawed in that the discrete beam equations can 'lock', see [10, Chap. 6§3], and when this happens there is no convergence to the exact solution as the number of elements is increased without limit.

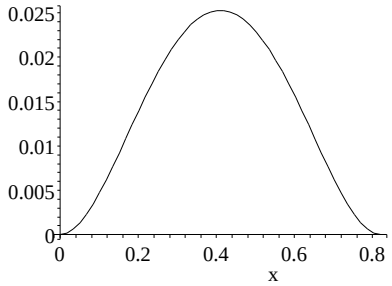

Figure 1: Distortion (in millimetres) due to centrifugal forces as a function of distance (in meters) along the tube, as predicted by analysis.

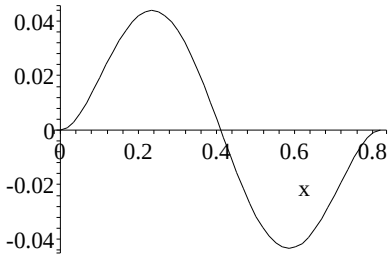

Figure 2: Distortion (in millimetres) due to Coriolis forces as a function of distance (in meters) along the tube, as predicted by analysis.

However, when the standard (and convergent) four degree of freedom representation of the Euler-Bernoulli beam (based on C^1 piecewise cubics) was inserted in ANSYS the predicted fluid distortion was still physically unrealistic. Similarly, the alternative methods of solution of the FE matrix equations which are available in ANSYS did not significantly improve the solution.

It is clear that there is an element of unreliability associated with the finite element computation of the Coriolis distortion mode, and we have not been able to find any treatment of this issue in the existing literature. However, one of the authors of [1] has reported the observation of similar effects and has suggested that the difficulty can be overcome by the use of an inverse iteration rather than a Lanczos solver, [11]. Therefore, in this article we attempt to build on this and to present a clearer picture of how this unreliability manifests itself, as well as to produce an 'algorithm' that appears to robustly calculate accurate mode shapes. To prepare for this, in the next section we derive and discuss the eigenvalue problem.

Lastly in this section, and for later reference, we mention that the Timoshenko model is the system,

$$(m_p + m_f)\frac{\partial^2 u}{\partial t^2} + m_f\left[2V\frac{\partial^2 u}{\partial x \partial t} + V^2\frac{\partial^2 u}{\partial x^2}\right] - \kappa G A_p\left(\frac{\partial^2 u}{\partial x^2} - \frac{\partial \theta}{\partial x}\right) = 0, \tag{8}$$

$$(\varrho_p I_p + \varrho_f I_f)\frac{\partial^2 \theta}{\partial t^2} - EI_p\frac{\partial^2 \theta}{\partial x^2} - \kappa G A_p\left(\frac{\partial u}{\partial x} - \theta\right) = 0, \tag{9}$$

where, as mentioned earlier, θ is the beam's rotation and the other notation is explained below in Section 4. Also, (2) continues to hold but (3) is replaced by,

$$\theta(0,t) = \theta(L,t) = 0. \tag{10}$$

3 The eigenvalue discretisation

In the absence of loads, and after finite element discretisation, the Euler-Bernoulli problem, (1), and the Timoshenko problem, (8) and (9), can each be written as a system of ordinary differential equations:

$$\boldsymbol{M}\frac{\partial^2 \boldsymbol{U}}{\partial t^2} + \boldsymbol{E}\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{A}\boldsymbol{U} = \boldsymbol{0}, \tag{11}$$

where $\boldsymbol{M}$ is the mass matrix, $\boldsymbol{A}$ is the stiffness matrix, and $\boldsymbol{E}$ is the velocity-damping matrix.

The mass matrix is always positive definite and, after imposing the boundary condi-

tions, (2) with either (3) or (10), the stiffness matrix will be also (if $m_f V^2$ is small enough).

The finite element formulation used here is quite standard. For the Euler-Bernoulli beam we have used two-noded C^1 cubic elements with degrees of freedom u and u_x at each end, while for the Timoshenko beam we have used three-noded C^0 quadratic elements with degrees of freedom u and θ at each node. As we have already mentioned, the linear four degree-of-freedom element cannot usually be used for the Timoshenko beam due to its tendency to 'locking'—see for example [10, Chap. 6§3].

We now derive the eigen-problem associated with the Coriolis meter. For this, assume the form $\boldsymbol{U} = \boldsymbol{V}e^{i\omega t}$ so that $\boldsymbol{U}_t = i\omega \boldsymbol{V}e^{i\omega t}$ and $\boldsymbol{U}_{tt} = -\omega^2 \boldsymbol{V}e^{i\omega t}$, and substitute in to (11) to get,

$$(\boldsymbol{A} + i\omega \boldsymbol{E} - \omega^2 \boldsymbol{M})\boldsymbol{V} = \boldsymbol{0}. \tag{12}$$

We employ several techniques to 'solve' this quadratic eigenvalue problem. Each uses MATLAB, and they are described below in the following subsections. Before we get to them we can deduce some properties of the eigensystem (12).

Let $\phi_1, \phi_2 \ldots$ be the shape functions associated with the finite element discretisation, and observe that the entries of the matrix $\boldsymbol{E}$, up to the coefficient $2m_f V$, are given by,

$$E_{ij} = \int_0^L \frac{d\phi_i}{dx}\phi_j \, dx.$$

Integrating by parts and using the zero boundary conditions then gives,

$$E_{ij} = \int_0^L \frac{d\phi_i}{dx}\phi_j \, dx = -\int_0^L \frac{d\phi_j}{dx}\phi_i \, dx = -E_{ji}.$$

Hence, $\boldsymbol{E} = -\boldsymbol{E}^T$.

Now, because $\boldsymbol{M}$ is symmetric and positive definite we always have $\boldsymbol{V} \cdot \boldsymbol{MV} > 0$ for all real vectors $\boldsymbol{V}$. If $\boldsymbol{V}$ is complex then we can also claim that $\overline{\boldsymbol{V}} \cdot \boldsymbol{MV} > 0$ for, if $\boldsymbol{V} = \boldsymbol{X} + i\boldsymbol{Y}$, then,

$$\overline{\boldsymbol{V}} \cdot \boldsymbol{MV} = \boldsymbol{X} \cdot \boldsymbol{MX} + \boldsymbol{Y} \cdot \boldsymbol{MY} > 0.$$

Clearly, $\overline{\boldsymbol{V}} \cdot \boldsymbol{AV} > 0$ also and so taking the scalar product of (12) with $\overline{\boldsymbol{V}}$ we get,

$$\omega^2 \overline{\boldsymbol{V}} \cdot \boldsymbol{MV} - i\omega \overline{\boldsymbol{V}} \cdot \boldsymbol{EV} - \overline{\boldsymbol{V}} \cdot \boldsymbol{AV} = 0$$

which is a quadratic equation for ω.

Notice that,

$$\overline{\overline{\boldsymbol{V}} \cdot \boldsymbol{EV}} = \boldsymbol{V} \cdot \boldsymbol{E}\overline{\boldsymbol{V}} = \overline{\boldsymbol{V}} \cdot \boldsymbol{E}^T\boldsymbol{V} = -\overline{\boldsymbol{V}} \cdot \boldsymbol{EV},$$

and so $\overline{\boldsymbol{V}} \cdot \boldsymbol{EV}$ equals the negative of its conjugate and therefore $\overline{\boldsymbol{V}} \cdot \boldsymbol{EV} = bi$ for some real b.

Hence, with $a = \overline{\boldsymbol{V}} \cdot \boldsymbol{MV}$ and $c = \overline{\boldsymbol{V}} \cdot \boldsymbol{AV}$ the quadratic for ω is,

$$aw^2 + b\omega - c = 0.$$

Since b is real and a and c are positive, it follows that each eigenvalue, ω, is real.

Now suppose that $(\omega, \boldsymbol{V})$ is an eigen-pair for (12) and take the complex conjugate of that equation to get,

$$\begin{aligned} \boldsymbol{0} &= \boldsymbol{A}\overline{\boldsymbol{V}} - i\omega \boldsymbol{E}\overline{\boldsymbol{V}} - \omega^2 \boldsymbol{M}\overline{\boldsymbol{V}}, \\ &= \boldsymbol{A}\overline{\boldsymbol{V}} + i(-\omega)\boldsymbol{E}\overline{\boldsymbol{V}} - (-\omega)^2 \boldsymbol{M}\overline{\boldsymbol{V}}, \end{aligned}$$

which demonstrates that if $(\omega, \boldsymbol{V})$ is an eigenpair for (12) then so too is $(-\omega, \overline{\boldsymbol{V}})$.

In summary, for (12): $\boldsymbol{M}$ and $\boldsymbol{A}$ are symmetric and positive definite; $\boldsymbol{E}$ satisfies the anti-symmetry $\boldsymbol{E}^T = -\boldsymbol{E}$; the eigenvalues, ω, are real; and, if $(\omega, \boldsymbol{V})$ is an eigenpair then so too is $(-\omega, \overline{\boldsymbol{V}})$.

We now describe the techniques that we will use to solve (12).

3.1 Technique 1: quadratic eigenvalue problem

The MATLAB fragment

```
[X, e] = polyeig(A, i*E, -M);
```

solves the quadratic eigenvalue problem in terms of a column of eigenvalues, `e`, and a matrix of eigenvectors, `X`. (Recall that $\boldsymbol{A}$ and $\boldsymbol{M}$ are invertible.)

3.2 Technique 2: linear eigenvalue problem

Setting $\boldsymbol{W} = \omega \boldsymbol{V}$ so that $-\omega^2 \boldsymbol{M}\boldsymbol{V} = -\omega \boldsymbol{M}\boldsymbol{W}$ and recalling that $\boldsymbol{M}$ is invertible we can write the quadratic eigenvalue problem as a standard eigenvalue problem of block-system type:

$$\begin{pmatrix} \boldsymbol{0} & \boldsymbol{I} \\ \boldsymbol{M}^{-1}\boldsymbol{A} & i\boldsymbol{M}^{-1}\boldsymbol{E} \end{pmatrix} \begin{pmatrix} \boldsymbol{V} \\ \boldsymbol{W} \end{pmatrix} = \omega \begin{pmatrix} \boldsymbol{V} \\ \boldsymbol{W} \end{pmatrix}.$$

This can be abbreviated to $\boldsymbol{B}\boldsymbol{X} = \boldsymbol{X}\boldsymbol{L}$, where $\boldsymbol{B}$ is the block-matrix, $\boldsymbol{L}$ is a diagonal matrix of eigenvalues and $\boldsymbol{X}$ is a matrix of eigenvectors, and can be solved in MATLAB via the fragment,

```
B = [ zeros(N,N) eye(N) ;
          M\A     i*M\E ];
[X L] = eig(B);
```

for the case where $\boldsymbol{A}$, $\boldsymbol{E}$ and $\boldsymbol{M}$ are $N \times N$ matrices.

3.3 Technique 3: Inverse iteration

If $\boldsymbol{D}$ is a non-singular matrix then, given an initial guess $\boldsymbol{x}_0$, the inverse power iteration,

$$\boldsymbol{z}_{n+1} = \boldsymbol{D}^{-1}\boldsymbol{x}_n, \qquad \boldsymbol{x}_{n+1} = \boldsymbol{z}_{n+1}\|\boldsymbol{z}_{n+1}\|_\infty^{-1}, \quad \text{for } n = 0, 1, 2, \ldots$$

is well defined. The second step, where

$$\|\boldsymbol{z}\|_\infty := \max\{|z_i| \colon 1 \leqslant i \leqslant N\}$$

for $\boldsymbol{z} = (z_1, z_2, \ldots, z_N)^T$, is a normalisation and is important to ensure that the iteration is convergent.

If $\boldsymbol{D}$ has a single eigenvalue of least modulus then this iteration converges to the eigenvector for that eigenvalue. In that case we can write $\boldsymbol{x}_n \to \boldsymbol{x}$, and then obtain the eigenvalue, λ say, from the Rayleigh quotient,

$$\lambda = \frac{\boldsymbol{x} \cdot \boldsymbol{D}\boldsymbol{x}}{\boldsymbol{x} \cdot \boldsymbol{x}}.$$

If there is more than one eigenvalue of minimal modulus the situation is less clear. The iterates may cycle over each of the associated eigenvectors, or even over, or converge to, a linear combination of the eigenvectors.

If these minimal modulus eigenvalues are identical then it is unclear how to proceed but, if they are algebraically distinct (e.g. $a+ib$ and $-a+ib$) then we can force the inverse iteration to converge by employing a 'shift'.

Suppose that $(\lambda, \boldsymbol{x})$ is an eigenpair of $\boldsymbol{D}$ then $(\lambda - p, \boldsymbol{x})$ is an eigenpair of $\boldsymbol{D} - p\boldsymbol{I}$ because,

$$\begin{aligned} \boldsymbol{0} &= (\boldsymbol{D} - \lambda\boldsymbol{I})\boldsymbol{x} \\ &= (\boldsymbol{D} - \lambda\boldsymbol{I})\boldsymbol{x} - p\boldsymbol{I}\boldsymbol{x} + p\boldsymbol{I}\boldsymbol{x} \\ &= ((\boldsymbol{D} - p\boldsymbol{I}) - (\lambda - p)\boldsymbol{I})\boldsymbol{x}. \end{aligned}$$

Here p is the 'shift' and if $\boldsymbol{D}$ has two minimal modulus eigenvalues, say,

$$a + bi \qquad \text{and} \qquad -a + bi,$$

then $\boldsymbol{D} - p\boldsymbol{I}$ has (for real p),

$$(a - p) + bi \qquad \text{and} \qquad -(a + p) + bi.$$

An informed choice for p therefore reduces the modulus of one eigenvalue while increasing the modulus of the other. The inverse iteration then becomes convergent.

This can be used for the Coriolis beam problem, and in this context the two main issues are the choice of initial guess, $\boldsymbol{x}_0$, and the value of the shift, p. We return to these points later.

3.4 Technique 4: shifting the quadratic eigenvalue problem

In technique 1, Subsection 3.1, we saw that the MATLAB fragment,

```
[X, e] = polyeig(A, i*E, -M);
```

can be used to solve the quadratic eigenvalue problem. It is relevant here to see how a shift can be introduced into this formulation of the problem.

The shifted version of the linear eigenvalue problem given in Method 2 is,

$$\begin{pmatrix} -p\boldsymbol{I} & \boldsymbol{I} \\ \boldsymbol{M}^{-1}\boldsymbol{A} & i\boldsymbol{M}^{-1}\boldsymbol{E} - p\boldsymbol{I} \end{pmatrix} \begin{pmatrix} \boldsymbol{V} \\ \boldsymbol{W} \end{pmatrix} = \omega \begin{pmatrix} \boldsymbol{V} \\ \boldsymbol{W} \end{pmatrix}.$$

From the top-row equation we derive,

$$\boldsymbol{W} = (\omega + p)\boldsymbol{V},$$

and substituting this into the bottom-row equation results (fairly obviously) in,

$$\boldsymbol{A}\boldsymbol{V} + i(\omega + p)\boldsymbol{E}\boldsymbol{V} - (p + \omega)^2\boldsymbol{M}\boldsymbol{V} = \boldsymbol{0}.$$

Collecting terms in like powers of ω results in another quadratic eigenvalue problem.

$$(\hat{\boldsymbol{A}} + i\omega\hat{\boldsymbol{E}} - \omega^2\boldsymbol{M})\boldsymbol{V} = \boldsymbol{0},$$

where $\hat{\boldsymbol{A}} = \boldsymbol{A} + ip\boldsymbol{E} - p^2\boldsymbol{M}$ and $\hat{\boldsymbol{E}} = \boldsymbol{E} + 2ip\boldsymbol{M}$.

Therefore, if `polyeig` is giving poor results due to two algebraically different but equal-in-minimal-modulus eigenvalues then, conceivably, this shifted system may improve matters. We investigate this below.

Of course, there exist many other algorithms and techniques for computing the solution to eigen-problems of the type described above but, as explained in the introduction, commonly used commercial packages often cannot accurately compute the Coriolis distortion mode. The four techniques just given exemplify this difficulty as well as containing an apparent remedy. This claim will be clearer once we have given some numerical results.

4 Numerical experiments

This section contains the outcome of several computations for both beam models. The coefficients in the partial differential equations, (1), (8) and (9), have the following meanings and (realistic) values, and correspond to a pipe with external tube diameter $0.0254\mathrm{m}$:

- κ: Timoshenko's shear correction factor (taken as unity).
- E: Young's modulus ($1.158 \times 10^{11}\mathrm{N/m^2}$).
- G: shear modulus ($4.3209 \times 10^{10}\mathrm{N/m^2}$).
- V: fluid velocity ($6\mathrm{m/s}$).
- A_p: cross-sectional area of the pipe ($7.001 \times 10^{-5}\mathrm{m^2}$).
- A_f: cross-sectional area of the fluid ($4.367 \times 10^{-4}\mathrm{m^2}$).
- m_p: mass per unit length of pipe ($0.31555\mathrm{kg/m}$).
- m_f: mass per unit length of fluid ($0.4367\mathrm{kg/m}$).
- ϱ_p: volume density of the pipe ($4507\mathrm{kg/m^3}$).
- ϱ_f: volume density of the fluid ($1000\mathrm{kg/m^3}$).
- I_p: second area moment of the pipe ($5.256 \times 10^{-9}\mathrm{m^4}$).
- I_f: polar second area moment of the pipe ($1.0512 \times 10^{-8}\mathrm{m^4}$).
- L: length of the pipe ($0.82\mathrm{m}$).

To obtain numerical results we have used a bespoke C++ code to generate the finite element matrices and then fed these in to the computing environment MATLAB (Release 14). We have used MATLAB because it is both simple and effective, and well suited to 'rapid prototyping'. We note also that, if desired, the entire numerical operation could be moved to MATLAB using, for example, the codes in [12] as a starting point.

For each of the techniques described in Subsections 3.1—3.4 the computed least-in-modulus value of $|\omega|$, for various numbers, N_e, of equal width finite elements, are shown for the Euler-Bernoulli beam in Table 1, with no shift, and Table 2, with shift $p = 900\,\mathrm{rad/sec}$. This choice for p was motivated by *a priori* knowledge that the required eigenvalue was approximately $940\,\mathrm{rad/sec}$ in modulus.

The analogous results for the Timoshenko beam are shown in Tables 3 and 4. (Note that

some results for larger values of N_e are missing for the Timoshenko beam due to the PC we were using having insufficient memory to deal with the large matrices.)

In each case ten inverse iterations were carried out, and the initial eigenvector for the inverse iteration was,

$$\begin{pmatrix} 1+10^{-5}i \\ 1+10^{-5}i \\ \vdots \\ 1+10^{-5}i \end{pmatrix}.$$

This choice was arrived at by trial and error, but motivated by the fact that the imaginary part is known to be several orders of magnitude less than the real part.

	$\lvert\omega\rvert_{\min}$ by technique...		
N_e	1	2	3
16	946.1995	946.1995	$\approx 1 \leftrightarrow 13000$
32	946.1948	946.1948	$\approx 1 \leftrightarrow 13000$
64	946.1945	946.1945	$\approx 1 \leftrightarrow 13000$
128	946.1941	946.1944	$\approx 1 \leftrightarrow 13000$
256	946.1941	946.1939	$\approx 1 \leftrightarrow 13000$
512	945.6362	946.1943	$\approx 1 \leftrightarrow 13000$

Table 1: Computed $|\omega|_{\min}$ for the Euler-Bernoulli beam (cubic elements) with no shift, $p = 0$.

	$\lvert\omega\rvert_{\min}$ by technique...		
N_e	1	2	3
16	946.1995	946.1995	946.1995
32	946.1948	946.1948	946.1948
64	946.1945	946.1945	946.1945
128	946.1945	946.1944	946.1943
256	946.1942	946.1939	946.1720
512	946.1923	946.1943	946.6341

Table 2: Computed $|\omega|_{\min}$ for the Euler-Bernoulli beam (cubic elements) with shift $p = 900$.

For a meter of length L, the meter sensitivity is defined as $(q_1 - q_2)/\dot{m}$ where $\dot{m}$ is the mass flow rate of the fluid, $q_1 = \arg V(L/4)$ and $q_2 = \arg V(3L/4)$. However, since $\dot{m}$ is the same for all cases studied, it is convenient

	$\lvert\omega\rvert_{\min}$ by technique...		
N_e	1	2	3
16	940.1753	940.1753	$\approx 1 \leftrightarrow 13000$
32	938.9266	938.9266	$\approx 1 \leftrightarrow 13000$
64	938.8359	938.8359	$\approx 1 \leftrightarrow 13000$
128	938.8298	938.8300	$\approx 1 \leftrightarrow 13000$
256	938.8296	938.8296	$\approx 1 \leftrightarrow 13000$

Table 3: Computed $|\omega|_{\min}$ for the Timoshenko beam (quadratic elements) with no shift, $p = 0$.

	$\lvert\omega\rvert_{\min}$ by technique...		
N_e	1	2	3
16	940.1753	940.1753	940.1753
32	938.9266	938.9266	938.9266
64	938.8359	938.8359	938.8359
128	938.8300	938.8300	938.8300
256	938.8296	938.8296	938.8296

Table 4: Computed $|\omega|_{\min}$ for the Timoshenko beam (quadratic elements) with shift $p = 900$.

to consider only the variations in $q_1 - q_2$. These values are shown in Tables 5 and 6 for the Euler-Bernoulli beam, and Tables 7 and 8 for the Timoshenko beam.

Figure 3 shows some graphical results for the Euler-Bernoulli eigen-system when 32 elements are employed and no shift is used, while Figure 4 shows the corresponding results when the system is shifted.

A similar set of results are shown for the Timoshenko beam in Figures 5 and 6.

5 Discussion

We can draw several conclusions in the context of the results just presented. Firstly for the 'pure' un-shifted eigensystem,

- MATLAB's `eig` and `polyeig` give high quality approximations to the eigenvalue, but the inverse iteration cycles between two incorrect iterates. See Tables 1 and 3.
- On the other hand, none of the meth-

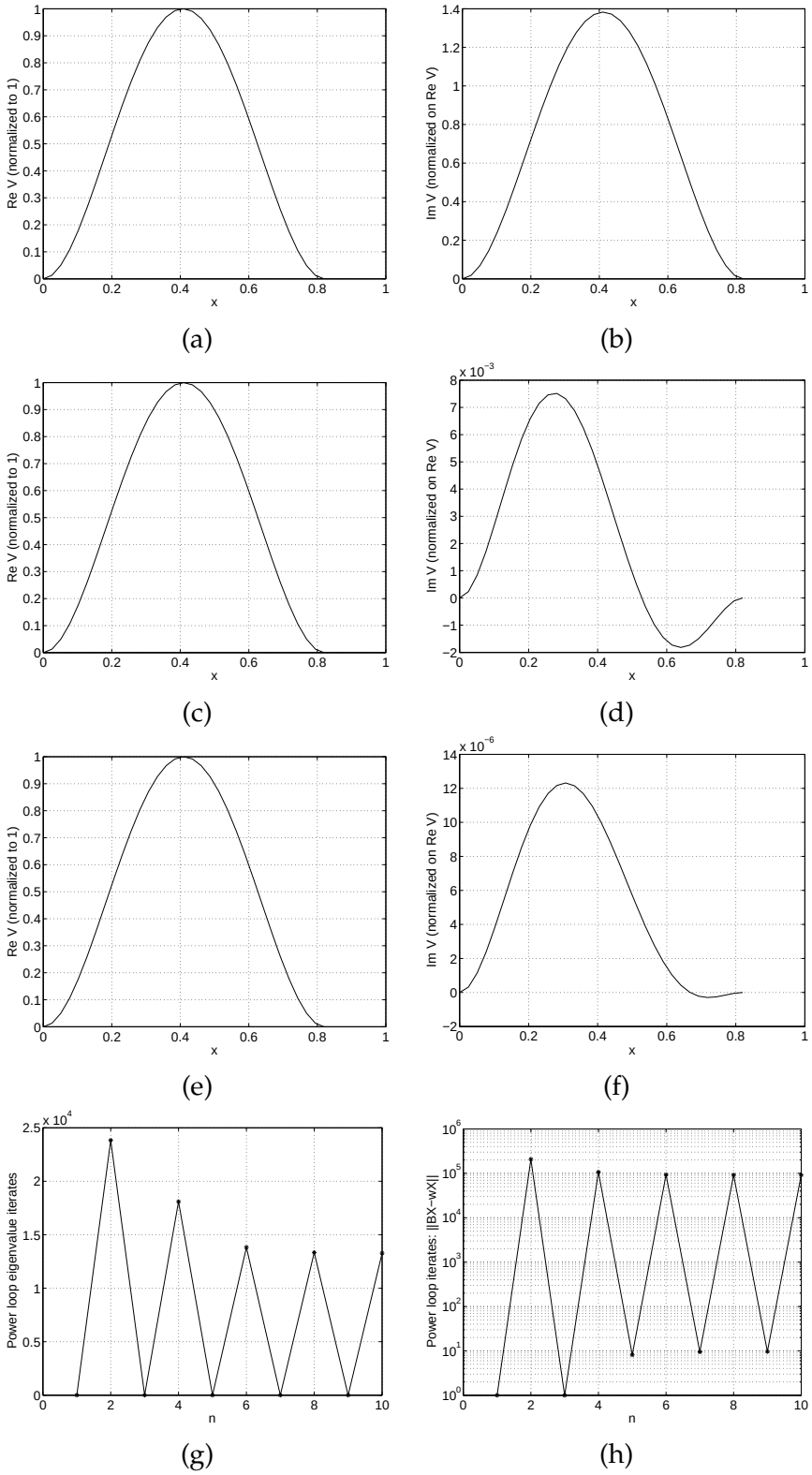

Figure 3: Computed eigen-results, without shift ($p = 0$), for the Euler-Bernoulli beam with 32 cubic elements: (a) and (b) show $\mathrm{Re}\, V$ and $\mathrm{Im}\, V$ from MATLAB's `polyeig` routine; (c) and (d) show the same but from MATLAB's `eig` routine; (e) and (f) again show the same but from the inverse iteration; (g) shows the inverse iteration's eigenvalue iterates; and, (h) shows how the norm of the residual, $\|\boldsymbol{BX} - \omega \boldsymbol{X}\|_\infty$, behaves during the iteration.

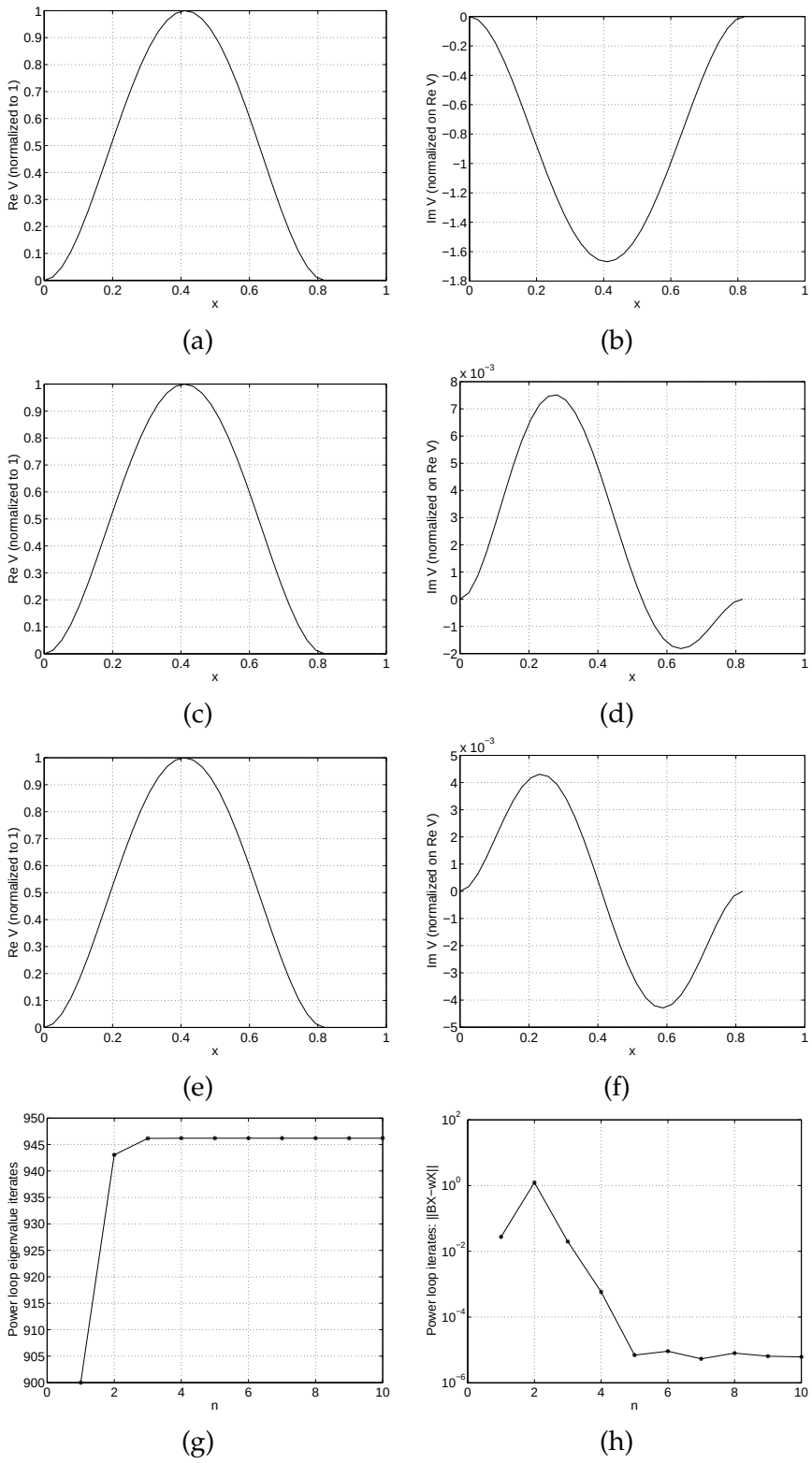

Figure 4: Computed eigen-results, with shift $p = 900$, for the Euler-Bernoulli beam with 32 cubic elements: (a) and (b) show $\mathrm{Re}\, V$ and $\mathrm{Im}\, V$ from MATLAB's `polyeig` routine; (c) and (d) show the same but from MATLAB's `eig` routine; (e) and (f) again show the same but from the inverse iteration; (g) shows the inverse iteration's eigenvalue iterates; and, (h) shows how the norm of the residual, $\|\boldsymbol{BX} - \omega \boldsymbol{X}\|_\infty$, behaves during the iteration.

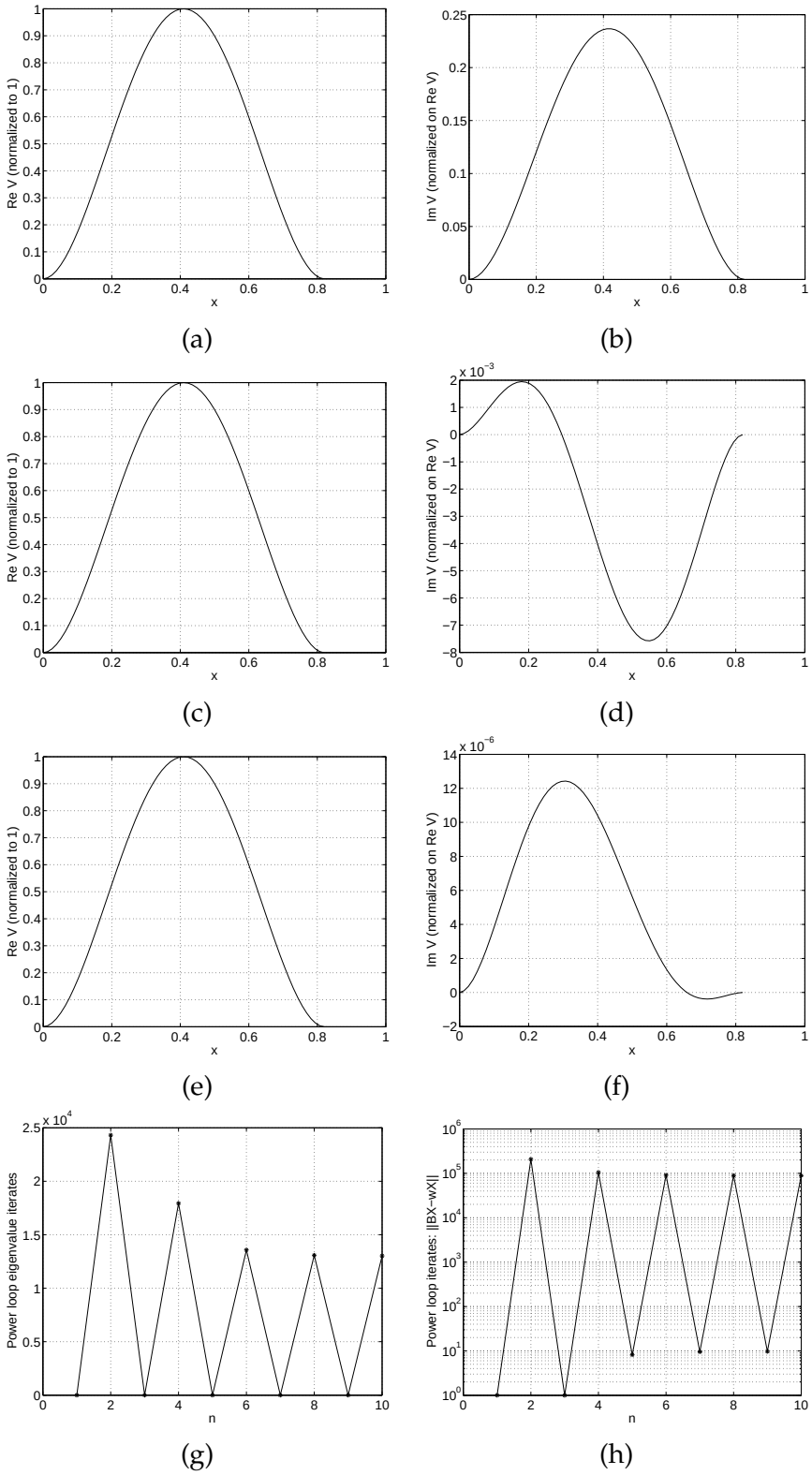

Figure 5: Computed eigen-results, without shift ($p = 0$), for the Timoshenko beam with 32 quadratic elements: (a) and (b) show $\mathrm{Re}\, V$ and $\mathrm{Im}\, V$ from MATLAB's `polyeig` routine; (c) and (d) show the same but from MATLAB's `eig` routine; (e) and (f) again show the same but from the inverse iteration; (g) shows the inverse iteration's eigenvalue iterates; and, (h) shows how the norm of the residual, $\|\boldsymbol{BX} - \omega \boldsymbol{X}\|_\infty$, behaves during the iteration.

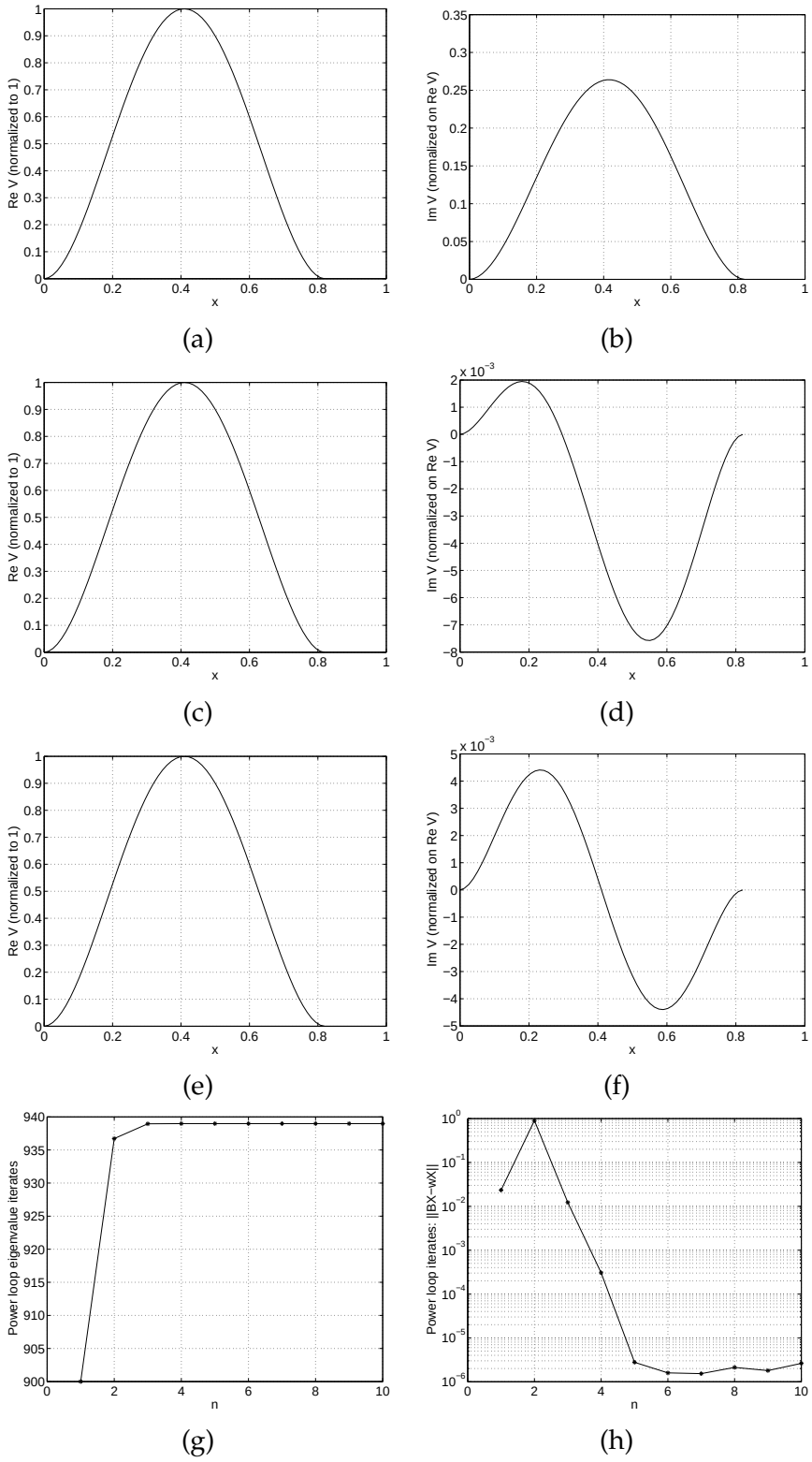

Figure 6: Computed eigen-results, with shift $p = 900$, for the Timoshenko beam with 32 quadratic elements: (a) and (b) show $\mathrm{Re}\,V$ and $\mathrm{Im}\,V$ from MATLAB's `polyeig` routine; (c) and (d) show the same but from MATLAB's `eig` routine; (e) and (f) again show the same but from the inverse iteration; (g) shows the inverse iteration's eigenvalue iterates; and, (h) shows how the norm of the residual, $\|\boldsymbol{BX} - \omega \boldsymbol{X}\|_\infty$, behaves during the iteration.

ods give a useful approximation to the Coriolis distortion mode: (b), (d), (f) in Figures 3 and 5. In particular, `polyeig` seems incapable of even picking up the inflection at the midpoint of the beam.

- Tables 5 and 7 show that `polyeig` and `eig` give a consistent approximation of $q_1 - q_2$ as around $\pm 0.88^\circ$. On the other hand, the inverse iteration is consistently in gross error.

However, when we introduce a shift of $900\,\mathrm{rad/sec}$:

- The performance of both `eig` and `polyeig` neither improves nor deteriorates in any noticeable way in terms of the eigenvalue approximations—see Tables 2 and 4.
- The inverse iteration now provides both high quality eigenvalues, see Tables 2 and 4, and Coriolis distortion modes: (f) in Figures 4 and 6.
- Tables 6 and 8 show that in terms of meter sensitivities the behaviour of both `polyeig` and `eig` is largely unchanged by the shift, but the inverse iteration provides a consistent approximation in all cases.

It seems clear that a shift is necessary in order to obtain useful mode shapes. For these results we were able to choose a 'good' value for this shift because we knew in advance the approximate value of $|\omega|_{\min}$. The question then arises as to how to choose this shift for a completely new problem in which no approximate values are known in advance.

We can suggest an answer to this as follows: use either `eig` or `polyeig` to compute $|\omega|_{\min}$, our results suggest that a reliable value will be generated even though the mode shapes are unreliable. Use this value to derive a shift for the inverse iteration which, our results suggest, should also be able to estimate reliable mode shapes and meter sensitivities.

A more extensive record of our computations is contained in a technical report, [13]. In that report we have considered finite element discretisations of both beam models using 16, 64, 128, 256, ... elements, as well as the 32-element case reported on here. And, for the Timoshenko beam, we have also used piecewise linear and piecewise cubic elements. Our reason for computing over such a range of discretisations is because we want to illustrate two particular points. Firstly, that inaccurate mode shapes are not just a feature of one beam model, or one type of discretisation and, secondly, correct mode shapes can be obtained with both Euler-Bernoulli and Timoshenko models with, in the latter case, any of the 'standard' finite elements. These more extensive results show no profound difference to the subset reported on here.

Again drawing on the material from [13], we can state another observation relating to how the norm, $\|\boldsymbol{BX} - \omega \boldsymbol{X}\|_\infty$, of the residual behaves during the inverse iteration (plot (h) in Figures 3 to 6).

For the Euler-Bernoulli beam this quantity (which should be zero) seems to bottom-out at a questionably large value when more than 64 elements are used (and for 512 elements it doesn't seem to even become consistently 'small'). On the other hand, for the Timoshenko beam the converged value seems to be consistently and reassuringly small across all the results. It would seem that this is due to a matrix conditioning issue: since the Timoshenko equations, (8) and (9), are of second order while the Euler-Bernoulli equation, (1), is of fourth order, it is not unreasonable to expect that the Timoshenko matrices are better conditioned, although further investigation is certainly warranted.

The results presented above along with those in [13] convincingly demonstrate that shifted inverse iteration gives consistently high quality estimates of the eigenvalue, the mode shapes and the meter sensitivities in all the cases under study, and also that the results are consistent with the approximate analytical results given earlier in Section 2. Hence, in the absence of any other analytical solution, we are forced to accept that these values are effectively 'exact'.

With that in mind, and given that `polyeig` gives poor mode shapes, it is surprising that the meter sensitivities computed from the `polyeig` output in Tables 5 to 8 agree so well with the 'exact' values computed from the shifted inverse iteration.

We are not able to give a definitive answer as to why `polyeig` and `eig` (and, presumably, other software products) are not able to produce accurate Coriolis distortion modes, but our computational experience strongly suggests that it is due to computer rounding error. If this is the case then the estimation of meter sensitivities in the absence of a high quality mode shape should not be regarded as reliable. Notice that for very small radian angles $\tan x = \sin x / \cos x \approx x$, we have,

$$\begin{aligned} q_1 - q_2 &= \tan^{-1} \frac{\operatorname{Im} V(L/4)}{\operatorname{Re} V(L/4)} \\ &\quad - \tan^{-1} \frac{\operatorname{Im} V(3L/4)}{\operatorname{Re} V(3L/4)} \\ &\approx \frac{\operatorname{Im} V(L/4) - \operatorname{Im} V(3L/4)}{\operatorname{Re} V(L/4)}, \end{aligned}$$

because, from plot (a) in Figures 3 to 6, we can see that $\operatorname{Re} V(L/4) \approx \operatorname{Re} V(3L/4)$. Hence, in order to give a reliable sensitivity, all that is required of the computed imaginary mode shape is that the vertical distance between the quarter point values be accurate. In our `eig` results it is (but not for the `polyeig` results), but if the shape inaccuracy is in fact due to rounding error, then we cannot possibly expect this always to be the case across all software platforms, programming languages and eigen-algorithms.

Lastly in this section, we note from [13] that no 'locking' can be observed for the Timoshenko beam with linear elements. It is not clear whether the fluid damping removes the danger, or whether (for these data) it will not appear until a larger number of elements are used.

6 Conclusions

The incorrect shapes of the Coriolis distortion mode predicted by finite element simulations using ANSYS or other similar commercial packages are not due to errors in the modelling.

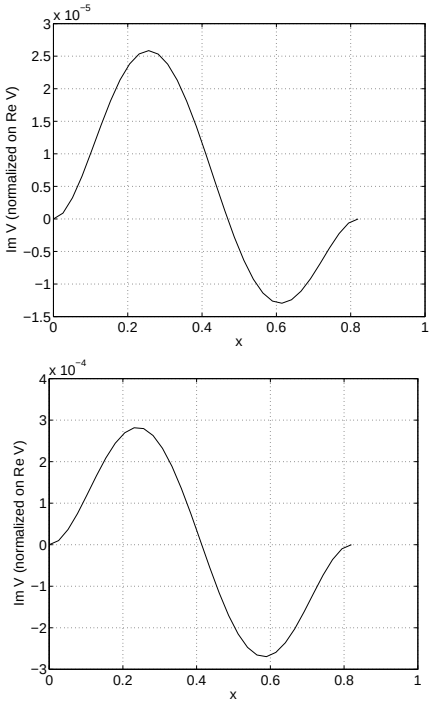

Figure 7: Versions of plot (f) in Figure 3 for velocities $V = 25\mathrm{m/s}$ (top) and $V = 250\mathrm{m/s}$ (bottom).

It seems most probable that the errors arise from computer rounding errors in the eigen-solvers. Although we cannot be certain of this we illustrate the effect of varying values of V in Figure 7. These are versions of plot (f) in Figure 3 but for velocities $V = 25\mathrm{m/s}$ and $V = 250\mathrm{m/s}$. We observe that the higher value of V gives a higher quality mode shape and we speculate that this is due to the centrifugal and Coriolis effects being of similar orders of magnitude and, therefore, the effect of rounding errors 'cancelling out'. This is far from conclusive and a deeper investigation, perhaps by using a higher precision computing environment, is needed.

The errors do not significantly contribute to the uncertainties associated with the predicted eigenvalues, which are generally less than 0.3%, although it must be accepted that differences of the order of 1% will be observed between results for the Euler-Bernoulli beam

models and Timoshenko beam models.

References

[1] C.P. Stack, R.B. Garnett, and G.E. Pawlas. A finite element for the vibration analysis of a fluid-conveying Timoshenko beam. Technical Report AIAA-93-1552-CP, 1993. AIAA Technical Paper, AIAA/ASME Structures, Structural Dynamics and Materials Conference.

[2] A. Belhadj, R. Cheesewright, and C. Clark. The simulation of Coriolis meter response to pulsating flow using a general purpose FE code. *Journal of Fluids and Structures*, 14:613—634, 2000.

[3] R. Cheesewright and C. Clark. The effect of flow pulsations on Coriolis mass flow meters. *Journal of Fluids and Structures*, 12:1025—1039, 1998.

[4] T. Wang, R.C. Baker, and Y. Hussain. An advanced numerical model for single straight tube Coriolis flowmeters. Technical report, 2004. Proc. FLOMEKO 2004, Guilin, China.

[5] T. Wang and R.C. Baker. Manufacturing variation of the measuring tube in a Coriolis flowmeter. *IEE Proc. Sci. Meas. Technol.*, 151(3):201—204, 2004.

[6] R. Cheesewright, A. Belhadj, and C. Clark. Effect of mechanical vibrations on Coriolis mass flow meters. *Journal of Dynamic Systems, Measurement, and Control*, 125:103—113, 2003.

[7] R. Cheesewright, C. Clark, A. Belhadj, and Y.Y. Hou. The dynamic response of Coriolis mass flow meters. *Journal of Fluids and Structures*, 18:165—178, 2003.

[8] S. Wang, C. Clark, and R. Cheesewright. The virtual Coriolis meter: A tool for simulation and design. *Proc. I. Mech. E., Part C, J. Mechanical Engineering Science*, In press: 2006.

[9] J. Kutin and I. Bajsic. Characteristics of shell-type Coriolis flowmeters. *Journal of Sound and Vibration*, 228:227—242, 1999.

[10] Dietrich Braess. *Finite Elements: theory, fast solvers and applications in solid mechanics*. Cambridge, 1997.

[11] R.B. Garnett. Private communication (Oct. 2005).

[12] Jochen Alberty, Carsten Carstensen, and Stefan A. Funken. Remarks around 50 lines of matlab: short finite element implementation. *Numerical Algorithms*, 20:117—137, 1999.

[13] Robert C Cheesewright and Simon Shaw. Computing the fundamental distortion mode in Coriolis mass flow meters. Technical report, BICOM, Brunel University, 2006. Technical Report 06/2, `www.brunel.ac.uk/bicom`.

N_e		sensitivities (degrees) by technique… 1	2	3
16	q_1	-4.844533×10^{1}	2.942431×10^{-1}	1.036817×10^{-3}
	q_2	-4.932315×10^{1}	-5.835774×10^{-1}	1.091165×10^{-4}
	$q_1 - q_2$	-8.778205×10^{-1}	-8.778205×10^{-1}	-9.277003×10^{-4}
32	q_1	5.368917×10^{1}	6.956238×10^{-1}	1.036820×10^{-3}
	q_2	5.456701×10^{1}	-1.822097×10^{-1}	1.090967×10^{-4}
	$q_1 - q_2$	8.778334×10^{-1}	-8.778335×10^{-1}	-9.277233×10^{-4}
64	q_1	3.547197×10^{1}	6.956246×10^{-1}	1.036821×10^{-3}
	q_2	3.459413×10^{1}	-1.822098×10^{-1}	1.090950×10^{-4}
	$q_1 - q_2$	-8.778397×10^{-1}	-8.778344×10^{-1}	-9.277257×10^{-4}
128	q_1	-4.777672×10^{1}	-6.688155×10^{-1}	1.036821×10^{-3}
	q_2	-4.865456×10^{1}	2.090189×10^{-1}	1.090948×10^{-4}
	$q_1 - q_2$	-8.778383×10^{-1}	8.778345×10^{-1}	-9.277261×10^{-4}
256	q_1	5.011923	-6.823000×10^{-1}	1.036821×10^{-3}
	q_2	5.889219	1.955341×10^{-1}	1.090948×10^{-4}
	$q_1 - q_2$	8.772961×10^{-1}	8.778341×10^{-1}	-9.277262×10^{-4}
512	q_1	-6.543358×10^{1}	2.023207×10^{-1}	1.036821×10^{-3}
	q_2	-6.455142×10^{1}	-6.757378×10^{-1}	1.090946×10^{-4}
	$q_1 - q_2$	8.821615×10^{-1}	-8.780585×10^{-1}	-9.277263×10^{-4}

Table 5: Computed meter sensitivities for the Euler-Bernoulli beam without shift ($p = 0$).

N_e		sensitivities (degrees) by technique… 1	2	3
16	q_1	-8.326008	2.942431×10^{-1}	4.398344×10^{-1}
	q_2	-7.448187	-5.835774×10^{-1}	-4.379861×10^{-1}
	$q_1 - q_2$	8.778205×10^{-1}	-8.778205×10^{-1}	-8.778205×10^{-1}
32	q_1	-5.950667×10^{1}	6.956238×10^{-1}	4.395348×10^{-1}
	q_2	-5.862883×10^{1}	-1.822097×10^{-1}	-4.382988×10^{-1}
	$q_1 - q_2$	8.778335×10^{-1}	-8.778335×10^{-1}	-8.778335×10^{-1}
64	q_1	-8.799232	6.956246×10^{-1}	4.394958×10^{-1}
	q_2	-7.921398	-1.822098×10^{-1}	-4.383385×10^{-1}
	$q_1 - q_2$	8.778341×10^{-1}	-8.778344×10^{-1}	-8.778343×10^{-1}
128	q_1	-5.142200×10^{1}	-6.688155×10^{-1}	4.394908×10^{-1}
	q_2	-5.054417×10^{1}	2.090189×10^{-1}	-4.383436×10^{-1}
	$q_1 - q_2$	8.778373×10^{-1}	8.778345×10^{-1}	-8.778344×10^{-1}
256	q_1	5.233954×10^{1}	-6.823000×10^{-1}	4.394915×10^{-1}
	q_2	5.321735×10^{1}	1.955341×10^{-1}	-4.383430×10^{-1}
	$q_1 - q_2$	8.778170×10^{-1}	8.778341×10^{-1}	-8.778345×10^{-1}
512	q_1	7.111125×10^{1}	2.023207×10^{-1}	4.394881×10^{-1}
	q_2	7.198967×10^{1}	-6.757378×10^{-1}	-4.383452×10^{-1}
	$q_1 - q_2$	8.784199×10^{-1}	-8.780585×10^{-1}	-8.778333×10^{-1}

Table 6: Computed meter sensitivities for the Euler-Bernoulli beam with shift $p = 900$.

N_e		sensitivities (degrees) by technique... 1	2	3
16	q_1	8.456385×10^{1}	-1.940224×10^{-1}	1.048495×10^{-3}
	q_2	8.366959×10^{1}	7.002301×10^{-1}	9.742365×10^{-5}
	$q_1 - q_2$	-8.942525×10^{-1}	8.942525×10^{-1}	-9.510714×10^{-4}
32	q_1	1.286257×10^{1}	1.952803×10^{-1}	1.051124×10^{-3}
	q_2	1.376057×10^{1}	-7.027220×10^{-1}	9.479458×10^{-5}
	$q_1 - q_2$	8.980022×10^{-1}	-8.980023×10^{-1}	-9.563294×10^{-4}
64	q_1	-6.430425×10^{1}	-6.757193×10^{-1}	1.051315×10^{-3}
	q_2	-6.340597×10^{1}	2.225550×10^{-1}	9.460349×10^{-5}
	$q_1 - q_2$	8.982804×10^{-1}	8.982743×10^{-1}	-9.567116×10^{-4}
128	q_1	-6.143330×10^{1}	-6.893710×10^{-1}	1.051328×10^{-3}
	q_2	-6.053498×10^{1}	2.089209×10^{-1}	9.459103×10^{-5}
	$q_1 - q_2$	8.983137×10^{-1}	8.982919×10^{-1}	-9.567365×10^{-4}
256	q_1	-6.455767×10^{1}	-2.157181×10^{-1}	1.051328×10^{-3}
	q_2	-6.545599×10^{1}	6.825750×10^{-1}	9.459023×10^{-5}
	$q_1 - q_2$	-8.983209×10^{-1}	8.982931×10^{-1}	-9.567381×10^{-4}

Table 7: Computed meter sensitivities for the Timoshenko beam (quadratic elements) without shift ($p = 0$).

N_e		sensitivities (degrees) by technique... 1	2	3
16	q_1	3.530840×10^{1}	-1.940224×10^{-1}	4.477007×10^{-1}
	q_2	3.620266×10^{1}	7.002301×10^{-1}	-4.465518×10^{-1}
	$q_1 - q_2$	8.942525×10^{-1}	8.942525×10^{-1}	-8.942525×10^{-1}
32	q_1	1.433462×10^{1}	1.952803×10^{-1}	4.495755×10^{-1}
	q_2	1.523262×10^{1}	-7.027220×10^{-1}	-4.484268×10^{-1}
	$q_1 - q_2$	8.980028×10^{-1}	-8.980023×10^{-1}	-8.980023×10^{-1}
64	q_1	3.689830×10^{1}	-6.757193×10^{-1}	4.497115×10^{-1}
	q_2	3.779658×10^{1}	2.225550×10^{-1}	-4.485628×10^{-1}
	$q_1 - q_2$	8.982745×10^{-1}	8.982743×10^{-1}	-8.982742×10^{-1}
128	q_1	-6.880750×10^{1}	-6.893710×10^{-1}	4.497203×10^{-1}
	q_2	-6.790921×10^{1}	2.089209×10^{-1}	-4.485716×10^{-1}
	$q_1 - q_2$	8.982929×10^{-1}	8.982919×10^{-1}	-8.982919×10^{-1}
256	q_1	4.069199×10^{1}	-2.157181×10^{-1}	4.497209×10^{-1}
	q_2	4.159029×10^{1}	6.825750×10^{-1}	-4.485722×10^{-1}
	$q_1 - q_2$	8.983055×10^{-1}	8.982931×10^{-1}	-8.982931×10^{-1}

Table 8: Computed meter sensitivities for the Timoshenko beam (quadratic elements) with shift $p = 900$.